T.C.
MARMARA ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
İŞLETME ANABİLİM DALI
SAYISAL YÖNTEMLER BİLİM DALI

VERİ MADENCİLİĞİ TEKNİKLERİ İLE İSTENMEYEN TÜRKÇE E-POSTALARIN ÖNLENMESİ ÜZERİNE BİR UYGULAMA

Yüksek Lisans Tezi

Sefa SAYLAN

İstanbul, 2018

T.C.
MARMARA ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
İŞLETME ANABİLİM DALI
SAYISAL YÖNTEMLER BİLİM DALI

VERİ MADENCİLİĞİ TEKNİKLERİ İLE İSTENMEYEN TÜRKÇE E-POSTALARIN ÖNLENMESİ ÜZERİNE BİR UYGULAMA

Yüksek Lisans Tezi

Sefa SAYLAN

Danışman: Doç. Dr. Özgür ÇAKIR

İstanbul, 2018

T.C.
MARMARA ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ MÜDÜRLÜĞÜ

TEZ ONAY BELGESİ

İŞLETME Anabilim Dalı SAYISAL YÖNTEMLER Bilim Dalı TEZLİ YÜKSEK LİSANS öğrencisi SEFA SAYLAN'nın VERİ MADENCİLİĞİ TEKNİKLERİ İLE İSTENMEYEN TÜRKÇE E-POSTALARIN ÖNLENMESİ ÜZERİNE BİR UYGULAMA adlı tez çalışması, Enstitümüz Yönetim Kurulunun 19.07.2018 tarih ve 2018-20/18 sayılı kararıyla oluşturulan jüri tarafından oy birliği / ~~oy çokluğu~~ ile Yüksek Lisans Tezi olarak kabul edilmiştir.

Tez Savunma Tarihi 30/07/2018

Öğretim Üyesi Adı Soyadı			**İmzası**
1.	Tez Danışmanı	Doç. Dr. ÖZGÜR ÇAKIR	
2.	Jüri Üyesi	Prof. Dr. ÖMER ÖNALAN	
3.	Jüri Üyesi	Doç. Dr. NİHAN YILDIRIM	

ÖZET

İstenmeyen e-postalara maruz kalmak işletmelerin iş süreçlerinde aksamalara, zaman kayıplarına ve hatta maddi kayıplarına sebep olduğundan günümüzün önemli sorunlarından biri olarak görülmektedir. İstenmeyen e-postaların engellenmesi için öncelikle tespit edilmeleri gerekmektedir. Bu çalışmada, gelen e-postaların sınıflandırılması ve istenmeyen Türkçe e-postaların tespiti için Naive Bayes algoritmaları **(**iki terimli ve çok terimli**)** ve Destek Vektör Makinesi algoritmaları (doğrusal ve RBF çekirdek fonksiyonlu) kullanılmıştır. Çalışmada, öğrenme kümesinin Türkçede kullanılan etkisiz kelimelerden arındırılması ve arındırılmaması durumunda TF-IDF yöntemi ile oluşturulan farklı boyutlardaki özellik vektörlerinin sınıflandırma başarısına etkisi 72 farklı model oluşturularak incelenmiştir. Öğrenme kümesinden etkisiz kelimelerin arındırılmaması durumunda oluşturulan modellerin çoğunlukla daha yüksek başarı ile sınıflandırma işlemini gerçekleştirdiği sonucuna ulaşılmıştır. En yüksek başarıyı elde eden sınıflandırma algoritmasının çok terimli naive bayes algoritması olduğu gözlemlenmiştir.

Anahtar Sözcükler: İstenmeyen Türkçe E-postalar, Sınıflandırma, Naive Bayes, Bernoulli Naive Bayes, Multinomial Naive Bayes, Destek Vektör Makinesi, Lineer Çekirdek Fonksiyonu, RBF Çekirdek Fonksiyonu

ABSTRACT

Nowadays, spam (Junk) mails might be considered as an important issue since they causes disruptions of business processes, a waste of time and also financial losses. The first step to prevent spam mails have to be detecting them. In this study, Naïve Bayes (Bernoulli and Multinomial) and Support Vector Machin**e (**Linear and RBF Kernel Functions) algorithms are applied to a data set in order to classify incoming mails and prevent unwanted ones. Besides, in 72 different models, it is examined how different size TF-IDF feature vectors affect the accuracy of classification in learning data set with or without stop-words used in Turkish. In case of not removed stop-words used in Turkish success of classification in learning data has been observed to increase. In this study using Multinomial Naive Bayes classification algorithm achieved the best result.

Keywords: Spam/Junk mails, Classification, Naive Bayes, Bernoulli Naive Bayes, Multinomial Naive Bayes, Support Vector Machine, Linear Functions, RBF Kernel

ÖNSÖZ

Yüksek lisans tez sürecimde veri madenciliği alanında çalışma olanağı sağlayan, bana yol gösteren ve beni yönlendiren sayın hocam Doç. Dr. Özgür ÇAKIR'a bana gösterdiği sabır ve öğrettiği her şey için teşekkürü borç bilirim.

Lisans eğitimimden bu yana desteklerini ve ilgilerini benden esirgemeyen Dr. Murat GEZER, Dr. Emre AKADAL, Dr. Serra ÇELİK ve Prof. Dr. Sevinç GÜLSEÇEN hocalarıma eğitim kariyerimi yönlendirmemdeki katkıları için teşekkür ederim.

Her konuda yanımda olan aileme desteklerinden dolayı teşekkür ederim.

İÇİNDEKİLER

SAYFA NO

ÖZET i

ABSTRACT ii

ÖNSÖZ iii

TABLOLAR vi

ŞEKİLLER ix

KISALTMALAR x

GİRİŞ 1

1. İSTENMEYEN E-POSTA VE ENGELLEME YÖNTEMLERİ 2

1.1 İSTENMEYEN E-POSTA VE ENGELLENMESİ 2

1.2 İSTENMEYEN E-POSTALARI ENGELLEME YÖNTEMLERİ 4

1.2.1 Klasik Yöntemler 4

1.2.1.1 Yasaklı Kelimelerle Engelleme Yöntemi 5

1.2.1.2 Kural Tabanlı Engelleme Yöntemi 5

1.2.1.3 Gerçek Zamanlı Kara Listeler ile Engelleme Yöntemi 6

1.2.1.4 SPF Kaydı ve Ters DNS (PTR) İle Engelleme Yöntemi 7

1.2.1.5 Honeypots Yöntemi 8

1.2.1.6 Fingerprinting (Hashing) ile Engelleme Yöntemi 9

1.2.1.7 Challenge / Response Sistemleri 10

1.2.2 Makine Öğrenmesi Algoritmaları 10

1.2.2.1 Özellik Vektörü Oluşturma (TF-IDF) 12

1.3 NAIVE BAYES 13

1.3.1 Thomas Bayes ve Teoremi 13

1.3.2 Bir Sınıflandırma Algoritması Olarak Naive Bayes 14

1.3.3 Naive Bayes ile Metin Sınıflandırma 14

1.3.3.1 Bernoulli Naive Bayes 15

1.3.3.2 Multinomial Naive Bayes 18

1.3.4 Naive Bayesle Gerçekleştirilmiş Çalışmalar 19

1.4 DESTEK VEKTÖR MAKİNELERİ 24

1.4.1 Destek Vektör Makineleri Çalışma Prensibi 25

1.4.2 Destek Vektör Makineleri ile Metin Sınıflandırma 31

1.4.3 Destek Vektör Makineleriyle Gerçekleştirilmiş Çalışmalar 32

1.5 MODEL PERFORMANS DEĞERLENDİRME ÖLÇÜTLERİ 37

2. TÜRKÇE İSTENMEYEN E-POSTALARIN ENGELLEME YÖNTEMLERİ ÜZERİNE BİR UYGULAMA 41

2.1 AMAÇ VE KAPSAM 41

2.2 VERİ KÜMESİ VE ÖZELLİKLERİ 42

2.3 YÖNTEM VE UYGULAMA 42

2.4 BULGULAR 53
2.4.1 Etkisiz Kelimelerin Öğrenme Kümesinden Arındırılmadığı Modeller 54
2.4.1.1 RBF Destek Vektör Makinesi Algoritmasıyla Sınıflandırma Sonuçları 54
2.4.1.2 Lineer Destek Vektör Makinesi Algoritmasıyla Sınıflandırma Sonuçları 59
2.4.1.3 Bernoulli Naive Bayes Algoritmasıyla Sınıflandırma Sonuçları 65
2.4.1.4 Multinomial Naive Bayes Algoritmasıyla Sınıflandırma Sonuçları 70
2.4.2 Etkisiz Kelimelerin Öğrenme Kümesinden Arındırılmadığı Modellerde Farklı Boyutlardaki Özellik Vektörlerinin Sınıflandırma Başarısına Etkisi 75
2.4.3 Etkisiz Kelimelerin Öğrenme Kümesinden Arındırıldığı Modeller 77
2.4.3.1 RBF Destek Vektör Makinesi Algoritmasıyla Sınıflandırma Sonuçları 77
2.4.3.2 Lineer Destek Vektör Makinesi Algoritmasıyla Sınıflandırma Sonuçları 82
2.4.3.3 Bernoulli Naive Bayes Algoritmasıyla Sınıflandırma Sonuçları 87
2.4.3.4 Multinomial Naive Bayes Algoritmasıyla Sınıflandırma Sonuçları 93
2.4.4 Etkisiz Kelimelerin Öğrenme Kümesinden Arındırıldığı Modellerde Farklı Boyutlardaki Özellik Vektörlerinin Sınıflandırma Başarısına Etkisi 98
2.4.5 Model 0 ile Model 1'nin Karşılaştırılması 100
2.4.6 İstenmeyen E-postaları En İyi Sınıflandıran Model 101
2.4.7 İstenen E-postaları En İyi Sınıflandıran Modeller 102
2.4.8 Etkisiz Kelimelerin Eğitim Kümesinden Arındırılıp Arındırılmama Durumlarının Modellerdeki Sınıflandırma Başarılarına Etkisi 102
2.4.9 F-Ölçütü ile Modellerin Karşılaştırılması 102

SONUÇ 105

EKLER 106
Ek 1 : Etkisiz Kelimelerin Öğrenme Kümesinde Bulunduğu Modellerin Farklı Özellik Vektörleri ile Analizi 106
Ek 2 : Etkisiz Kelimelerin Öğrenme Kümesinde Bulunmadığı Modellerin Farklı Özellik Vektörleri ile Analizi 107
Ek 3 : Multinomial NB Model 0.4 Sınıflandırma Sonucu 108
Ek 4 : Bernoulli NB Model 0.3 Sınıflandırma Sonucu 110
Ek 5 : Bernoulli NB Model 0.3.10000 112
Ek 6 : Etkisiz kelimelerin eğitim kümesinden arındırılıp arındırılmama durumlarının sınıflandırma başarısına etkisi 114

KAYNAKÇA 122

TABLOLAR

SAYFA NO

Tablo 1 : Metinlerin Matrise Dönüştürülmesi 32
Tablo 2 : Örnek Sınıflandırma Tablosu 38
Tablo 3 : Türkçe Etkisiz Kelimeler Tablosu 45
Tablo 4 : Model 0.1 Etkisiz Kelimeler Dahil (RBF Destek Vektör Makineleri) 55
Tablo 5 : Model 0.1.100 Etkisiz Kelimeler Dahil (RBF Destek Vektör Makineleri) 55
Tablo 6 : Model 0.1.200 Etkisiz Kelimeler Dahil (RBF Destek Vektör Makineleri) 56
Tablo 7 : Model 0.1.300 Etkisiz Kelimeler Dahil (RBF Destek Vektör Makineleri) 56
Tablo 8 : Model 0.1.400 Etkisiz Kelimeler Dahil (RBF Destek Vektör Makineleri) 57
Tablo 9 : Model 0.1.500 Etkisiz Kelimeler Dahil (RBF Destek Vektör Makineleri) 57
Tablo 10 : Model 0.1.1000 Etkisiz Kelimeler Dahil (RBF Destek Vektör Makineleri) 58
Tablo 11 : Model 0.1.5000 Etkisiz Kelimeler Dahil (RBF Destek Vektör Makineleri) 58
Tablo 12 : Model 0.1.10000 Etkisiz Kelimeler Dahil (RBF Destek Vektör Makineleri) 59
Tablo 13 : RBF Destek Vektör Makinesi Algoritmasıyla Oluşturulan, Etkisiz Kelimelerin Öğrenmeye Dahil Olduğu Modellerin Model Performans Ölçütleri 59
Tablo 14 : Model 0.2 Etkisiz Kelimeler Dahil (Lineer Destek Vektör Makineleri) 60
Tablo 15 : Model 0.2.100 Etkisiz Kelimeler Dahil (Lineer Destek Vektör Makineleri) 61
Tablo 16 : Model 0.2.200 Etkisiz Kelimeler Dahil (Lineer Destek Vektör Makineleri) 61
Tablo 17 : Model 0.2.300 Etkisiz Kelimeler Dahil (Lineer Destek Vektör Makineleri) 62
Tablo 18 : Model 0.2.400 Etkisiz Kelimeler Dahil (Lineer Destek Vektör Makineleri) 62
Tablo 19 : Model 0.2.500 Etkisiz Kelimeler Dahil (Lineer Destek Vektör Makineleri) 63
Tablo 20 : Model 0.2.1000 Etkisiz Kelimeler Dahil (Lineer Destek Vektör Makineleri) 63
Tablo 21 : Model 0.2.5000 Etkisiz Kelimeler Dahil (Lineer Destek Vektör Makineleri) 64
Tablo 22 : Model 0.2.10000 Etkisiz Kelimeler Dahil (Lineer Destek Vektör Makineleri) 64
Tablo 23 : Lineer Destek Vektör Makinesi Algoritmasıyla Oluşturulan, Etkisiz Kelimelerin Öğrenmeye Dahil Olduğu Modellerin Model Performans Ölçütleri 65
Tablo 24 : Model 0.3 Etkisiz Kelimeler Dahil (Bernoulli Naive Bayes) 65
Tablo 25 : Model 0.3.100 Etkisiz Kelimeler Dahil (Bernoulli Naive Bayes) 66
Tablo 26 : Model 0.3.200 Etkisiz Kelimeler Dahil (Bernoulli Naive Bayes) 66
Tablo 27: Model 0.3.300 Etkisiz Kelimeler Dahil (Bernoulli Naive Bayes) 67
Tablo 28 : Model 0.3.400 Etkisiz Kelimeler Dahil (Bernoulli Naive Bayes) 67
Tablo 29 : Model 0.3.500 Etkisiz Kelimeler Dahil (Bernoulli Naive Bayes) 68
Tablo 30 : Model 0.3.1000 Etkisiz Kelimeler Dahil (Bernoulli Naive Bayes) 68
Tablo 31 : Model 0.3.5000 Etkisiz Kelimeler Dahil (Bernoulli Naive Bayes) 69
Tablo 32 : Model 0.3.10000 Etkisiz Kelimeler Dahil (Bernoulli Naive Bayes) 69
Tablo 33 : Bernoulli Naive Bayes Algoritmasıyla Oluşturulan, Etkisiz Kelimelerin Öğrenmeye Dahil Olduğu Modellerin Model Performans Ölçütleri 70
Tablo 34 : Model 0.4 Etkisiz Kelimeler Dahil (Multinomial Naive Bayes) 70
Tablo 35 : Model 0.4.100 Etkisiz Kelimeler Dahil (Multinomial Naive Bayes) 71
Tablo 36 : Model 0.4.200 Etkisiz Kelimeler Dahil (Multinomial Naive Bayes) 71

Tablo 37 : Model 0.4.300 Etkisiz Kelimeler Dahil (Multinomial Naive Bayes)..... 72
Tablo 38 : Model 0.4.400 Etkisiz Kelimeler Dahil (Multinomial Naive Bayes)..... 72
Tablo 39 : Model 0.4.500 Etkisiz Kelimeler Dahil (Multinomial Naive Bayes)..... 73
Tablo 40 : Model 0.4.1000 Etkisiz Kelimeler Dahil (Multinomial Naive Bayes)..... 73
Tablo 41 : Model 0.4.5000 Etkisiz Kelimeler Dahil (Multinomial Naive Bayes)..... 74
Tablo 42 : Model 0.4.10000 Etkisiz Kelimeler Dahil (Multinomial Naive Bayes)..... 74
Tablo 43 : Multinomial Naive Bayes Algoritmasıyla Oluşturulan, Etkisiz Kelimelerinde Öğrenmeye Dahil Olduğu Modellerin Model Performans Ölçütleri 75
Tablo 44 : Etkisiz Kelimelerden Arındırılmadan Oluşturulan Modellerin Pozitifleri Farklı Özellik Vektörleri ile Sınıflandırma Başarısı..... 75
Tablo 45 : Etkisiz Kelimelerden Arındırılmadan Oluşturulan Modellerin Negatifleri Farklı Özellik Vektörleri ile Sınıflandırma Başarısı..... 76
Tablo 46 : Model 1.1 Etkisiz Kelimeler Hariç (RBF Destek Vektör Makineleri)..... 78
Tablo 47 : Model 1.1.100 Etkisiz Kelimeler Hariç (RBF Destek Vektör Makineleri)..... 78
Tablo 48 : Model 1.1.200 Etkisiz Kelimeler Hariç (RBF Destek Vektör Makineleri)..... 79
Tablo 49 : Model 1.1.300 Etkisiz Kelimeler Hariç (RBF Destek Vektör Makineleri)..... 79
Tablo 50 : Model 1.1.400 Etkisiz Kelimeler Hariç (RBF Destek Vektör Makineleri)..... 80
Tablo 51 : Model 1.1.500 Etkisiz Kelimeler Hariç (RBF Destek Vektör Makineleri)..... 80
Tablo 52 : Model 1.1.1000 Etkisiz Kelimeler Hariç (RBF Destek Vektör Makineleri)..... 81
Tablo 53 : Model 1.1.5000 Etkisiz Kelimeler Hariç (RBF Destek Vektör Makineleri)..... 81
Tablo 54 : Model 1.1.10000 Etkisiz Kelimeler Hariç (RBF Destek Vektör Makineleri)..... 82
Tablo 55 : RBF Destek Vektör Makinesi Algoritmasıyla Oluşturulan, Etkisiz Kelimelerin Öğrenmeye Dahil Olmadığı Modellerin Model Performans Ölçütleri 82
Tablo 56 : Model 1.2 Etkisiz Kelimeler Hariç(Lineer Destek Vektör Makineleri)..... 83
Tablo 57 : Model 1.2.100 Etkisiz Kelimeler Hariç(Lineer Destek Vektör Makineleri)..... 83
Tablo 58 : Model 1.2.200 Etkisiz Kelimeler Hariç(Lineer Destek Vektör Makineleri)..... 84
Tablo 59 : Model 1.2.300 Etkisiz Kelimeler Hariç(Lineer Destek Vektör Makineleri)..... 84
Tablo 60 : Model 1.2.400 Etkisiz Kelimeler Hariç(Lineer Destek Vektör Makineleri)..... 85
Tablo 61 : Model 1.2.500 Etkisiz Kelimeler Hariç(Lineer Destek Vektör Makineleri)..... 85
Tablo 62 : Model 1.2.1000 Etkisiz Kelimeler Hariç(Lineer Destek Vektör Makineleri)..... 86
Tablo 63 : Model 1.2.5000 Etkisiz Kelimeler Hariç(Lineer Destek Vektör Makineleri)..... 86
Tablo 64 : Model 1.2.10000 Etkisiz Kelimeler Hariç (Lineer Destek Vektör Makineleri)..... 87
Tablo 65 : Lineer Destek Vektör Makinesi Algoritmasıyla Oluşturulan, Etkisiz Kelimelerin Öğrenmeye Dahil Olmadığı Modellerin Model Performans Ölçütleri 87
Tablo 66 : Model 1.3 Etkisiz Kelimeler Hariç(Bernoulli Naive Bayes)..... 88
Tablo 67 : Model 1.3.100 Etkisiz Kelimeler Hariç(Bernoulli Naive Bayes)..... 88
Tablo 68 : Model 1.3.200 Etkisiz Kelimeler Hariç (Bernoulli Naive Bayes)..... 89
Tablo 69 : Model 1.3.300 Etkisiz Kelimeler Hariç (Bernoulli Naive Bayes)..... 89
Tablo 70 : Model 1.3.400 Etkisiz Kelimeler Hariç (Bernoulli Naive Bayes)..... 90
Tablo 71 : Model 1.3.500 Etkisiz Kelimeler Hariç(Bernoulli Naive Bayes)..... 90
Tablo 72 : Model 1.3.1000 Etkisiz Kelimeler Hariç(Bernoulli Naive Bayes)..... 91
Tablo 73 : Model 1.3.5000 Etkisiz Kelimeler Hariç(Bernoulli Naive Bayes)..... 91
Tablo 74 : Model 1.3.10000 Etkisiz Kelimeler Hariç (Bernoulli Naive Bayes)..... 92
Tablo 75 : Bernoulli Naive Bayes Algoritmasıyla Oluşturulan, Etkisiz Kelimelerin Öğrenmeye Dahil Olmadığı Modellerin Model Performans Ölçütleri..... 92

Tablo 76 : Model 1.4 Etkisiz Kelimeler Hariç (Multinomial Naive Bayes)....93
Tablo 77 : Model 1.4.100 Etkisiz Kelimeler Hariç (Multinomial Naive Bayes)....94
Tablo 78 : Model 1.4.200 Etkisiz Kelimeler Hariç(Multinomial Naive Bayes)....94
Tablo 79 : Model 1.4.300 Etkisiz Kelimeler Hariç(Multinomial Naive Bayes)....95
Tablo 80 : Model 1.4.400 Etkisiz Kelimeler Hariç(Multinomial Naive Bayes)....95
Tablo 81 : Model 1.4.500 Etkisiz Kelimeler Hariç (Multinomial Naive Bayes)....96
Tablo 82 : Model 1.4.1000 Etkisiz Kelimeler Hariç(Multinomial Naive Bayes)....96
Tablo 83 : Model 1.4.5000 Etkisiz Kelimeler Hariç(Multinomial Naive Bayes)....97
Tablo 84 : Model 1.4.10000 Etkisiz Kelimeler Hariç(Multinomial Naive Bayes)....97
Tablo 85 : Multinomial Naive Bayes Algoritmasıyla Oluşturulan, Öğrenmeye Etkisiz Kelimelerin Dahil Olmadığı Modellerin Model Performans Ölçütleri....98
Tablo 86 : Etkisiz Kelimelerden Arındırılarak Oluşturulan Modellerin Pozitifleri Farklı Özellik Vektörleriyle Sınıflandırma Başarısı....98
Tablo 87 : Etkisiz Kelimelerden Arındırılarak Oluşturulan Modellerin Negatifleri Farklı Özellik Vektörleriyle Sınıflandırma Başarısı....99
Tablo 88 : Etkisiz Kelimelerin Mevcut Olduğu Modellerin F-Ölçütleri....103
Tablo 89 : Etkisiz Kelimelerin Mevcut Olmadığı Modellerin F-Ölçütleri....103
Tablo 90 : F-Ölçütleri ile Modellerin Karşılaştırılması....104

ŞEKİLLER

SAYFA NO

Şekil 1: Makine Öğrenmesi Algoritmaları 11
Şekil 2: Bernoulli Naive Bayes Algoritmasının Sözde Kodu 16
Şekil 3: Multinomial Naive Bayes Algoritmasının Sözde Kodu 18
Şekil 4: Destek Vektör Makineleri 26
Şekil 5: Doğrusal olmayan sınıflandırma problemi 29
Şekil 6: Şekil 5'in Denklem 11 ile dönüşümün grafiği 30
Şekil 7 : Vektör Uzayı Modeli Grafiği 31
Şekil 8 : Uygulamanın akış diyagramı 53
Şekil 9 : Etkisiz kelimelerden arındırılmadan oluşturulan modellerin pozitifleri farklı özellik vektörleri ile sınıflandırma başarısı 76
Şekil 10 : Etkisiz kelimelerden arındırılmadan oluşturulan modellerin negatifleri farklı 77
özellik vektörleri ile sınıflandırma başarısı 77
Şekil 11 : Etkisiz kelimelerden arındırılarak oluşturulan modellerin pozitifleri farklı özellik vektörleriyle sınıflandırma başarısı 99
Şekil 12 : Etkisiz kelimelerden arındırılarak oluşturulan modellerin negatifleri farklı özellik vektörleriyle sınıflandırma başarısı 100
Şekil 13 : Özellik vektörü kısıtlanmamış modellerin pozitifleri sınıflandırma başarısı 100
Şekil 14 : Özellik vektörü kısıtlanmamış modellerin negatifleri sınıflandırma başarısı 101
Şekil 15 : İstenilen e-postaların Model 100 ile sınıflandırma başarısı 114
Şekil 16 : İstenmeyen e-postaların Model 100 ile sınıflandırma başarısı 114
Şekil 17 : İstenilen e-postaların Model 200 ile sınıflandırma başarısı 115
Şekil 18 : İstenmeyen e-postaların Model 200 ile sınıflandırma başarısı 115
Şekil 19 : İstenilen e-postaların model 300 ile sınıflandırılma başarısı 116
Şekil 20 : İstenmeyen e-postaların model 300 ile sınıflandırılma başarısı 116
Şekil 21 : İstenilen e-postaların model 400 ile sınıflandırılma başarısı 117
Şekil 22 : İstenmeyen e-postaların model 400 ile sınıflandırılma başarısı 117
Şekil 23 : İstenilen e-postaların model 500 ile sınıflandırılma başarısı 118
Şekil 24 : İstenmeyen e-postaların model 500 ile sınıflandırılma başarısı 118
Şekil 25 : İstenilen e-postaların model 1000 ile sınıflandırılma başarısı 119
Şekil 26 : İstenmeyen e-postaların model 10000 ile sınıflandırılma başarısı 119
Şekil 27 : İstenen e-postaların model 5.000 ile sınıflandırılma başarısı 120
Şekil 28 : İstenmeyen e-postaların model 5.000 ile sınıflandırılma başarısı 120
Şekil 29 : İstenen e-postaların model 10.000 ile sınıflandırılma başarısı 121
Şekil 30 : İstenmeyen e-postaların model 10.000 ile sınıflandırılma başarısı 121

KISALTMALAR

NB : Naive Bayes

DVM : Destek Vektör Makinesi

RBF (Radial Based Function) : Radyal Tabanlı Fonksiyon

Pozitifler : İstenen E-Postalar

Negatifler : İstenmeyen E-Postalar

ARPANET (Advanced Research Projects Agency Network) : Gelişmiş Araştırma Projeleri Dairesi Ağı

SPF (Sender Policy Framework) : Gönderen Politikaları Çerçevesi

DNS (Domain Name System) : Alan Adı Sistemi

PTR (Reverse Domain Name System) : Ters Alan Adı Sistemi

IP : İnternet Protokol

DoS (Denial of Service) : Bilişim alanında bir hizmetin servis dışı kalma durumu.

TF (Term Frequency) : Terim Frekansı

IDF (Inverse Document Frequency) : Ters Doküman Frekansı

NKÜ : Namık Kemal Üniversitesi

GİRİŞ

İstenmeyen e-postalar konusu ele alındığında sınırlı kaynak olarak karşımıza "zaman" kavramı çıkmaktadır. Gün içerisinde birçok e-posta alışverişi gerçekleştiren bir işletmede çalışanların hesaplarına gelen postaların cevaplanması bir zaman maliyetine neden olmaktadır. İstenmeyen e-postalara maruz kalmak işletmelerin iş süreçlerinde aksamalara ve hatta maddi kayıplara sebep olduğundan günümüzün önemli sorunlarından biri olarak görülmektedir.

Çalışmada Naive Bayes (iki terimli, çok terimli) ve Destek Vektör Makinesi algoritmaları (doğrusal ve RBF çekirdek fonksiyonlu) kullanılarak istenmeyen e-postaların sınıflandırılması hedeflenmiştir. Bu doğrultuda farklı boyutta özellik vektörleri oluşturularak sınıflandırma başarısına etkileri incelenmiştir. Ayrıca etkisiz kelimelerin öğrenme kümesinden arındırılıp/arındırılmama durumlarının da sınıflandırma başarısındaki etkisi incelenmiştir.

İstenmeyen e-postaların engellenmesi ile zaman kayıplarının önüne geçmek, işletmelerde verimliliği ve kârı arttırmak, veri depolama maliyetini azaltmak, veri kayıplarının önüne geçmek hedeflenmektedir. Ayrıca veri madenciliği alanında sıklıkla kullan sınıflandırma algoritmalarından olan destek vektör makinesi (rbf, lineer) ve naive bayes (multinomial, bernoulli) algoritmalarını karşılaştırmak hedeflenmiştir. Gerçekleştirdiğimiz bu çalışmanın metin madenciliği alanında doküman sınıflandırma problemini ele alan araştırmacılar için yol göstermesi amaçlanmıştır.

Tez çalışması iki bölümden oluşmaktadır. Birinci bölümde istenmeyen e-postaların engellenme yöntemlerine, neden engellenmesi gerektiğine değinilerek uygulama aşamasında kullanılan yöntemlerin teknik detayları sunulmuştur. Oluşturulacak modellerin değerlendirilmesi için kullanılan model performans değerlendirme ölçütleri konusunda bilgiler verilmiştir. İkinci bölümde istenmeyen e-postaları sınıflandırmadaki amacımıza değinilip daha sonra uygulamada kullandığımız veri kümesi tanıtılmıştır. Gerçekleştirilmiş uygulamada kullanılan yöntemlere değinildikten sonra elde edilen bulgular paylaşılmış ve sonuçları tartışılmıştır.

1. İSTENMEYEN E-POSTA VE ENGELLEME YÖNTEMLERİ

Bu bölümde, istenmeyen e-postaların tarihçesine ve istenmeme nedenlerine değinilmiştir. Sonrasında istenmeyen e-postaları engelleme yöntemleri sunulmuştur. Bunların içerisinden sıklıkla kullanılan yöntemlerden olan naive bayes ve destek vektör makineleri yöntemleri ayrı başlıklar altında incelenmiş ve en sonunda veri madenciliği alanında oluşturulan modellerin değerlendirilebilmesi için geliştirilen model performans değerlendirme ölçütleri konusunda bilgiler verilmiştir.

1.1 İSTENMEYEN E-POSTA VE ENGELLENMESİ

Spam, ABD'de kurulan Hormel Foods Corporation tarafından 1937 yılında üretilen konserve gıda ürünüdür[1]. İçerisinde saf domuz eti ve jambon bulunmaktadır[2]. İkinci Dünya Savaşı yıllarında (1939-1942) müttefik güçler için haftada on beş milyon spam konservesi üretilmiştir. Savaş sırasında buzdolabına ihtiyaç duymayan ve soğuk olarak tüketilebilen gıda ürünü askerlerin beslenmesinde önemli rol oynamıştır[3,4].

İkinci Dünya Savaşı Britanya'da kıtlığa yol açmış ve halka karne ile erzak dağıtılmaya başlanmıştır. Bu yıllarda vatandaşlar et ürünlerine konulan kotalardan dolayı konserve ürünlere yönelmiştir. Ucuz ve pratik olarak kullanılan spam konserveleri savaş yılları sonrasında da marketlerde ve kafelerde yerini almıştır. Britanya'da ucuz üretimin sağlanması açısından birçok et ürününde spam kullanılmıştır. Britanyalı komedi grubu Monty Python'un 1970 yılında spam içeren et ürünlerinin istenmediği fikrini konu alan skeci yayınlanmıştır[5].

Hayatımıza bir gıda ürünü olarak giren spam kelimesi yayınlanan skecin toplumda oluşturmuş olduğu algıyla istenmeyen bir ürünü ifade ederek farklı bir anlam kazanmıştır.

[1] (FAQs | Hormel Foods, 1998)
[2] (SPAM® Family Of Products, 1998)
[3] (spam.com, 2012)
[4] (WW II Encyclopedia -- SPAM, 2002)
[5] (Monty Python - Spam, 2007)

Spam kelimesi Sivanadyan tarafından; *talep edilmemiş ya da istenmemiş ürün veya web sayfası reklamı, hızlı para kazanma yöntemi, zincirleme gönderiler ya da pornografik içerikler barındıran e-postalar* olarak tanımlanmıştır[6].

Guinness rekorlar kitabındaki verilere göre ilk istenmeyen e-posta 3 Mayıs 1978 yılında DigitalEquipment Corp (DEC) firmasında çalışan Gray Thuerk tarafından, ABD Savunma Bakanlığına bağlı olan ARPANET'de bulunan 397 kişiye gönderilmiştir. Gray alıcılarını DEC tarafından üretilen DECSYSTEM-20 ve aynı aileden olan farklı bilgisayarların da tanıtıldığı ürün tanıtım günlerine davet etmiştir[7].

İstenmeyen e-postalar konusu ele alındığında sınırlı kaynak olarak karşımıza "zaman" kavramı çıkmaktadır. Gün içerisinde birçok e-posta alışverişi gerçekleştiren bir işletmede çalışanların hesaplarına gelen postaların cevaplanması bir zaman maliyetine neden olmaktadır.

Sahami ve arkadaşlarının çalışmasında kullanıcıların hesaplarına gönderilen istenmeyen e-postaların maddi zararları olduğundan bahsedilmektedir. Çalışmada bu tür e-postaların yalnızca zaman kayıplarına neden olmayıp binlerce kullanıcının bulunduğu e-posta sunucusunda depolama alanının hızlı bir şekilde tükenmesine neden olacağından da bahsedilmiştir[8].

Teknolojinin veri depolama alanında gerçekleştirmiş olduğu büyük gelişmeler sayesinde daha az bir maliyetle veri depolamanın mümkün olduğu bilinmektedir[9]. Ancak istenmeyen e-postaların göz ardı edilerek e-posta sunucunda yer işgal etmeleri depolama maliyetini beraberinde getirecektir.

İstenmeyen e-postalar sınıfına giren "Aldatıcı e-postalar" birçok işletmede ve akademik kurumda çalışan kullanıcıların kullanıcı ve şifre bilgilerini ele geçirerek

[6] (Sivanadyan, 2003)
[7] (guinnessworldrecords.com, 2016)
[8] (Sahami, 1998)
[9] (matt komorowski - a history of storage cost, 2014)

kişilere veya işletmelere ait olan özel verilerin üçüncü şahısların eline geçmesine neden olabilmektedir[10].

Akademik kurumlar, kamu ve özel sektörde yer alan bir çok işletmenin bilgisayarlarına bulaşan virüslerin kaynağı olarak aklımıza istenmeyen e-postalar gelebilir. Günümüzde e-postalar aracılığıyla bilgisayarlara bulaşan ve doküman dosyalarını (*.doc, *.pdf, *.xls, *.docx, *.xlsx uzantılı) hedef alan bu virüsler ilgili dosyaların şifrelenerek kullanılamaz duruma gelmesine neden olmaktadır[11].

Bu durum kurumlarda ve firmalarda veri kayıplarını beraberinde getirmektedir. Dosyalara yeniden ulaşılabilmesi için fidye talep eden bu virüsler işletmeler açısından maddi kayıplara da neden olabilmektedir.

1.2 İSTENMEYEN E-POSTALARI ENGELLEME YÖNTEMLERİ

Geçmişten günümüze gerek günlük hayatta gerekse işletmelerde karşılaşılan problemlere birçok farklı yaklaşımla çözümler aranmıştır. İstenmeyen e-postalara bir problem olarak yaklaşıldığında bu problem istenen ve istenmeyen e-postaları sınıflandırma problemi olarak düşünülebilir. Geçmişte kullanılan klasik yaklaşımlar ve teorisi yeni olmasa da bu alanda kullanımı yeni olan makine öğrenmesi algoritmaları kullanılarak istenmeyen e-postaları sınıflandırma problemine çözümler aranmıştır. Tez boyunca istenilen e-posta pozitif istenmeyen e-posta ise negatif olarak adlandırılacaktır.

1.2.1 Klasik Yöntemler

Bu alanda kullanılan klasik yöntemler; yasaklı kelimelerle engelleme yöntemi, kural tabanlı engelleme yöntemi, gerçek zamanlı kara listelerle engelleme yöntemi, spf kaydı ve ters dns (ptr) ile engelleme yöntemi, honeypots ile engelleme yöntemi, fingerprinting (hashing) ile engelleme yöntemi ve challenge / response sistemleri başlıkları altında incelenecektir.

[10] (Aldatıcı E-Posta Mesajları Hk. - Bilgi İşlem Daire Başkanlığı - Marmara Üniversitesi, 2015)
[11] (Düşgör 2016)

1.2.1.1 Yasaklı Kelimelerle Engelleme Yöntemi

Bu yöntemde istenmeyen kelimelerin belirlenmesiyle oluşturulan bir kelime listesi kullanılmaktadır. E-postanın içerisindeki kelimelerin oluşturulmuş olan kelime listesindeki bir kelimeyle eşleşmesi durumunda ilgili e-posta, negatif olarak sınıflandırılacaktır. Herhangi bir kelime ile eşleşmemesi durumunda ise e-posta pozitif olarak sınıflandırılacaktır[12] .

Bu yöntemde uzun bir pozitif içerisinde, istenmeyen kelimeler listesinde belirlediğimiz bir kelimeye rastlama olasılığımız çok yüksektir. Ayrıca istenmeyen kelimeler listesinde bulunan bir kelimenin eş anlamlı karşılığının kullanılması veya listede bulunan kelimenin bilinerek yanlış yazılıp kelime listesinde eşleşememesi durumu da sınıflandırma işlemini başarısız kılacaktır. Bu detaylar göz önünde alındığında yöntemin zafiyetleri görülmektedir.

1.2.1.2 Kural Tabanlı Engelleme Yöntemi

Alınan e-postanın konu ve içerik bilgisine bakılarak içerik tabanlı bir kural oluşturmanın mümkün olduğu engelleme yöntemdir. Yasaklı kelimeler yönteminden farklı olarak birden fazla kelime mantıksal ifadelerle bir araya getirilerek kural oluşturulabilir. E-posta metninin içerisinde bedava, kolay para kazanma gibi kelimeler bulunuyorsa bu e-posta negatif olarak sınıflandırılabilir. Bu yöntem kelime kuralları oluşturabilmek için kelime listesine ihtiyaç duymaktadır. Yasaklı kelimelerle engelleme yönteminde karşımıza çıkan zafiyetler bu yöntem için de geçerlidir. Kuralların manuel olarak belirlenmesi ve belirlenen bu kuralların zamana bağlı olarak farklılıklar göstermesi yine bu yöntemin zafiyetlerindendir[13].

Ayrıca bu yöntemde bir başka yaklaşım olarak sınıflandırılma işlemini gerçekleştirebilmek için e-postanın başlık bilgisinden yararlanılmaktadır. Başlık bilgisi içerisinden alıcı ve göndericiye ait bir takım bilgiler elde edildikten sonra bu bilgilerle oluşturulan kurallar aracılığıyla negatifler belirlenebilir. Elde edilen bilgiler; e-postanın

[12] (Allman, 2003)
[13] (Cohen, 1996)

gönderici bilgisi, göndericiye geri dönüş yolu, alıcıya geri dönüş yolu, kimden kime ulaştığı, alınan kaynak bilgisi, konu bilgisi, gönderilme zamanı gibi bilgilerdir[14].

Wu çalışmasında çalışanların gün içerisinde saat 06:30 ile 23:30 arasında e-posta alışverişi gerçekleştirdikleri bilgisini vermiştir. Buradaki bilgiden yola çıkılarak 06:30 ile 23:30 saatleri dışında gelen e-postaların kabul edilmeyeceğini belirten zamana bağlı bir kuralla negatifler engellenebilir. Ancak bu kural gece yarısı gelen bir pozitifin negatif olarak algılanmasına neden olacaktır.

1.2.1.3 Gerçek Zamanlı Kara Listeler ile Engelleme Yöntemi

1997 yılında Dave Rand ve Paul Vixie'in oluşturduğu engelleme yöntemdir. Dave ve Paul negatif olarak belirledikleri ya da zararlı olduğunu düşündükleri e-postaların gönderildiği sunucuların IP adreslerini elde ederek bir liste oluşturmuştur. Bu liste yardımıyla aynı göndericiden gelen sonraki e-postalar negatif olarak sınıflandırılmıştır. Bu yöntemde oluşturulan liste gerçek zamanlı kara liste olarak adlandırılmaktadır. Günümüzde birçok kurum tarafından oluşturulan kara listeler ağ yöneticilerinin kullanımına sunulmuştur[15].

Bu yöntem sonucunda gelen bir e-posta belirlenen bir klasöre otomatik olarak yönlendirilebilir ya da sunucu tarafından yöneticinin oluşturacağı bir filtre yardımıyla sunucuya ulaştıktan sonra kullanıcılara gösterilmeden silinebilir. Ancak yanlış sınıflandırmadan kaynaklı olarak önemli e-postaların silinebilme ihtimali göz önüne alındığında bu kullanım veri kayıplarına neden olabilmektedir[16].

İnternet servis sağlayıcıları aktif olarak e-posta gönderen tüm IP'leri ve alan adlarını izlerler. Tüm internet servis sağlayıcıları e-posta gönderen IP adresini değerlendirirken başvurulabilecek bir yol haritası oluştururlar. Normalde kara listede bulunmayan bir IP adresinden artan bir şekilde e-posta göndermesi bu IP adresinin kara listeye girmesine neden olabilir. IP adresine ait çok fazla istenmeyen içerik gönderdiğine dair şikayet alınırsa yine IP adresi kara listeye girebilir. Bu yöntemde de içerikte bedava,

[14] (Wu, 2009)
[15] (Levine, 2010)
[16] (Segal, 2004)

kolay para kazanın gibi ifadeler yer alıyorsa bu durum IP adresinin kara listeye girmesinde neden olabilir. Kara listeler yönteminde kullanılan kurallardan dolayı hatalı sınıflandırma yapılması çok sık rastlanılan bir durumdur[17].

E-posta servis sağlayıcılarının herhangi bir kullanıcısı servisi kötü amaçlı olarak kullanabilir. Bu durum sağlayıcının IP adresinin kara listeye girmesine neden olacaktır. Dolayısıyla aynı servis sağlayıcısının kullanıcıları negatif gönderimde bulunmasalar dahi filtrenin IP tabanlı bir engelleme gerçekleştirmesinden dolayı mağdur duruma düşeceklerdir.

Hem çok sık miktarda yanlış sınıflandırmada bulunması hem de IP tabanlı filtrelemeden kaynaklı olarak e-posta servis sağlayıcısının kullanıcılarının mağdur olması bu yöntemin zafiyetlerindendir.

1.2.1.4 SPF Kaydı ve Ters DNS (PTR) İle Engelleme Yöntemi

Kendi e-posta hesap uzantılarına sahip olan kişi veya kuruluşlar tarafından gerçekleştirilen SPF (Sender Policy Framework) kaydı Türkçe karşılığı ile "Gönderen Politikaları Çerçevesi" anlamına gelmektedir. Hangi DNS altında hangi e-posta adreslerinin e-posta gönderme yetkisinin bulunduğunu tanımlamak için kullanılır. Günümüzde istenmeyen e-postalarla mücadele amacı ile kullanımı mevcuttur. SPF kaydını kontrol eden sunucularda ilgili doğrulama işleminin gerçekleşmemesi durumunda gelen e-posta negatif olarak sınıflandırılır[18].

PTR (ters DNS) yöntemi, göndericinin IP adresini kullanılarak ters DNS işlemi uygular. Bu sayede alan adı belirlenir. Belirlenen alan adına ait ip adresinin gelen e-postadaki IP adresi ile eşleşmemesi durumunda e-posta negatif olarak sınıflandırılır[19,20]

[17] (totalsend.com, 2014)
[18] (openspf.org, 2018)
[19] (Zink, 2016)
[20] (McMichael, 2018)

PTR ve SPF kaydı gibi yöntemlerin çalışma prensibi alıcıların göndericiyi doğrulaması için gönderici tarafından oluşturulmuş olan kayıtların kontrol edebilmesine dayanmaktadır.

Bu yöntemlerde kötü amaçlı kişiler tarafından kullanılabilecek birçok farklı protokol ve DNS saldırısı yöntemi bulunmaktadır. Bu saldırıları önlemek için işlem limitleri uygulanmaktadır. Fakat kötü amaçlı bir kişi bir mağdurun alan adını kullanarak SPF kayıtları oluşturabilir ve bu kayıtlarla birçok kişiye e-posta gönderebilir. Alıcılar gelen e-postanın SPF kaydını doğrulamak için kayıtta belirtilen sunucuya doğrulama talebinden bulunurlar. Birden fazla kişiden gelen bu talep nedeniyle bir DoS saldırısı gerçekleştirilmiş olur. Bu sayede saldırganın gerçek kaynağı da gizlenmiş olacaktır. İşlemler sırasında DNS kontrolü yapılırken kötü amaçlı kişiler tarafından işlem limitleri aşılarak boş yere işlemci ve hafıza kullanımı gibi problemle de karşılaşılmaktadır[21].

1.2.1.5 Honeypots Yöntemi

Eric Cole ve Stephen Northcutt'un yazmış olduğu "Honeypots: A Security Manager's Guide to Honeypots" yazısında bilgi güvenliğinin ana fikrini "Bir saldırıyı önlemek idealdir fakat tespit etmek zorunluluktur" sözleriyle açıklamaktadır. Ayrıca yazıda kişisel veya kurumsal e-posta hesaplarının negatiflere maruz kalma durumu bir saldırı olduğu anlamında yorumlanmıştır.

Bu yazıda honeypots sistemlerinin kurumlar veya kişiler tarafından negatifleri gönderenlerin saldırısına açık yapılar olarak kurulduğundan bahsedilmektedir. Saldırganlar için tuzak niteliğinde olan honeypots sistemlerinin yöneticisi saldırılar dahilinde gelen negatiflerin içerisinden gönderene ait tüm bilgileri elde etmiş olacaktır. Bir sonraki aşama eldeki bilgilere göre ilgili adreslerden gelecek olan saldırıların önüne geçmektir. Bu yöntemin zafiyeti henüz gerçekleşmemiş olan saldırılara karşı korumasız olmasıdır [22].

[21] (Schlitt, 2006)
[22] (sans.edu, -)

1.2.1.6 Fingerprinting (Hashing) ile Engelleme Yöntemi

Günlük hayatımızda sıklıkla kullanmış olduğumuz web uygulamalarının şifrelerini, online bankacılık işlemlerinde kullanılan şifrelerimizi ya da bir firma içi uygulamanın kullanıcı şifrelerini hedef alan kötü niyetli kişiler bu bilgileri ele geçirebilmek için sahte web formları oluşturarak e-posta üzerinden alıcılarıyla iletişime geçmektedirler.

İlgili web sayfası şifreyi yenilemek için sizden mevcut olan şifreyi ve yenilenecek olan şifreyi istemektedir. Bu bilgiler verildiği takdirde e-posta hesabının mevcut şifresi kötü niyetli kişilerin eline geçmiş olacaktır. Bu türdeki e-postaları son kullanıcıya ulaşmadan önce engellemek veri güvenliği ve kişisel verilerin korunması için oldukça önem taşımaktadır.

Bu yöntemde ilk olarak önemli kuruluşların listesi ve oturum açma sayfaları elde edilir. Her sayfanın hash bilgisi veri tabanında saklanır. Daha sonra şüpheli web sayfaları veri tabanında var olan bilgilerle karşılaştırılır.

Jonathan makalesinde hasing algoritmasının çalışma prensibini şu şekilde açıklamaktadır; kötü niyetli kişiler tarafından oluşturulduğu düşünülen html iletişim formu üzerinde hashing işlemi gerçekleştirilir. Orijinal web sitesi tarafından oluşturulan ve gerçekten şifreyi yenileme sayfası olan sayfanın da hashing bilgisi veri tabanında mevcuttur. İki web sayfasında da aynı Hashing algoritması kullanılmalıdır. Bir sonraki aşamada iki farklı veriye ait hash değerleri karşılaştırılır.

Hashing fonksiyonu *; uzun bir girdiyi alarak daha kısa bir alanda göstermektir. Amaç girende bir değişiklik olduğunda bunun çıkışa da yansımasıdır. Buna göre hashing fonksiyonları ya veri güvenliğinde, verinin farklı olup olmadığını kontrol etmeye yarar ya da verileri sınıflandırmak için kullanılır* [23] .

[23] (Şeker, 2008)

Bu yöntemde oturum açma formu ya da şifre değiştirme formunun veri tabanında bulunmaması durumu söz konusu olabilir veya web sayfasının ilgili formu değişikliğe uğramış olabilir. Bu durumda filtre yanlış sınıflandırma yapmamıza neden olacaktır. Ayrıca bizim kullandığımız benzerlik algoritmaları ve hashing karşılaştırma yöntemleri saldırganlar tarafından da bilinen yöntemlerdir. Böylelikle aynı filtreleri kendilerinin de uygulayarak çok düşük bir farklılıkla taklit e-posta oluşturma ihtimalleri kuvvetli olabilmektedir [24].

1.2.1.7 Challenge / Response Sistemleri

Bu yöntemle kaynağı bilinmeyen bir e-postanın alıcıya ulaşmadan önce göndereni doğrulama işlemini gerçekleştirmek amaçlanmaktadır. Gönderene e-posta ile bağlantı linki üzerinden doğrulama sorusu gönderilerek yanıt istenmektedir. Bu doğrulama sorusu kullanıcıdan bir görsel üzerindeki harfleri ya da ses dosyası içerisinden duyulan harfleri isteyen bir soru olabilir. Doğrulama işlemi gerçekleştirilmezse e-posta alıcıya ulaşamayacaktır [25].

Bu yöntem gün içerisinde birçok e-posta alışverişi gerçekleştiren işletmeler/kişiler için zaman kaybı problemine neden olacaktır.

Yukarıda söz edilen klasik yöntemlerin dışında negatifleri sınıflandırmak için anti virüs yazılımları gibi yöntemlerde yine klasik yöntemler içinde yer almaktadır. Ayrıca negatifleri engellemek için makine öğrenmesi algoritmaları da kullanılabilmektedir.

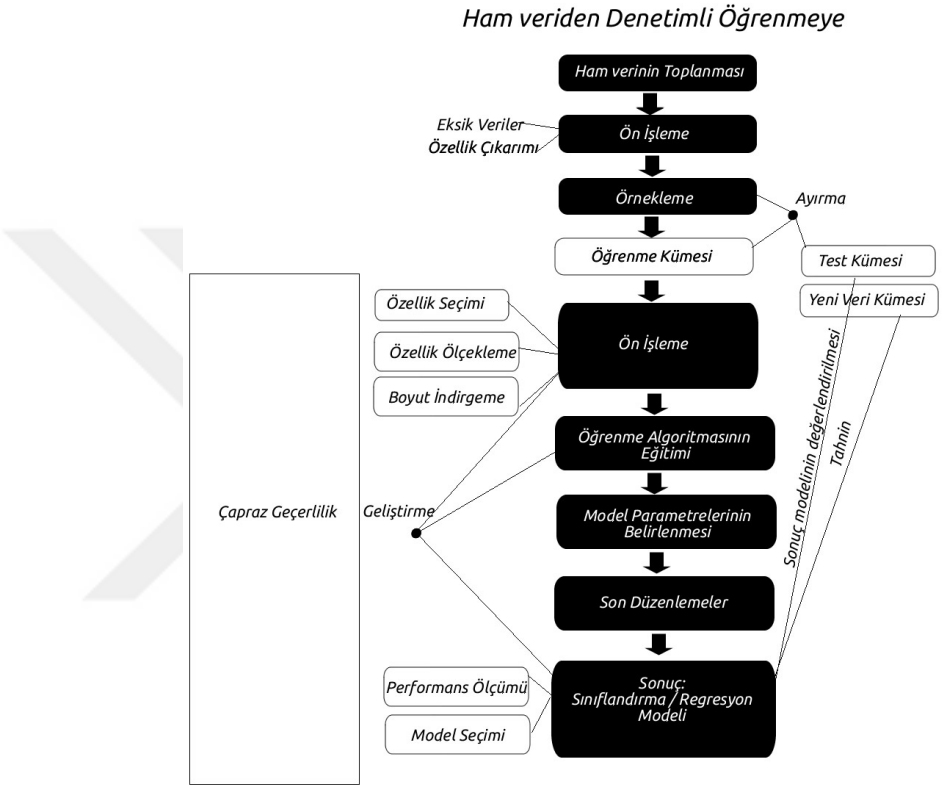
1.2.2 Makine Öğrenmesi Algoritmaları

İnsanlar karar verme süreçlerinde tecrübelerinden yararlanırlar. İlk defa karşılaşılan bir probleme nazaran daha önceden çözülmüş bir problemle karşılaşıldığında kolay bir karar verme süreci gerçekleşebilir.

Yukarıdaki örnekten yola çıkarak makine öğrenmesi algoritmalarını insanların tecrübeleriyle açıklayabiliriz. Biz insanların daha önceden edindiği tecrübeyi öğrenme

[24] (Jonathan Zdziarski, 2006)
[25] (Challenge/Response Authentication Services - SpamHelp)

veri kümesi olarak tanımlayabiliriz. Makine öğrenmesi algoritmaları çözülmesi istenilen problemle ilgili bu veri kümesinden tecrübe kazanır ve daha sonra bu tecrübelerini başka problemleri çözmek için kullanırlar. Makinelerin karar verme sürecinde daha iyi bir konuma gelebilmesi için daha çok tecrübeye yani daha büyük hacimli bir veri kümesine ihtiyaç duyulmaktadır.



Şekil 1: Makine Öğrenmesi Algoritmaları

Kaynak: Mühendislikte Yapay Zeka ve Uygulamaları, Makine Öğrenmesi Kütüphaneleri, Gezer, Saylan Sayfa 57

Makine öğrenmesi algoritmalarının bir kısmı öğrenme süreçlerinde veri kümesi içerisinden veriye özgü özelliklerin çıkarımını kendisi gerçekleştirirken bir kısmı ise bu özellikleri algoritmayı kullanan kullanıcıdan istemektedir. Veriye özgü özellikler çıkarılarak bir vektöre dönüştürülür. Bu vektör "özellik vektörü" olarak adlandırılır. Özellik vektörü oluşturmak için; bilgi kazanımı, olasılık oranı , log olasılık oranı, ki kare, terim frekansı ve ters doküman frekansı gibi birçok farklı yöntem mevcuttur[26].

[26] (Nicolosi, 2018)

Tez çalışmamızda bu alanda sıklıkla kullanılan terim frekansı ve ters doküman frekansı yönteminden yararlanılmıştır.

Bilinen birçok farklı makine öğrenme algoritması mevcuttur. Bu farklılık algoritmaların problemi değişik bakış açıları ile ele almasından kaynaklanmaktadır. Makine öğrenmesi algoritmaları özellik vektöründen edinmiş olduğu bilgileri kullanarak modeller üretir. Üretilen bu modeller daha sonra sınama verisi ile sınanır. Elde edilen sonuçlardan modellerin performansı değerlendirilir ve problemlerin çözümünde en iyi olan model belirlenir. Belirlenen bu model ilgili problemin çözümünde kullanılır.

Metin madenciliği alanından kullanılan makine öğrenmesi algoritmaları; naive bayes, destek vektör makineleri, yapay sinir ağları, genetik algoritmalar, k-en yakın komşuluk, karar ağaçları ve derin öğrenme algoritmaları olarak karşımıza çıkmaktadır. İncelenen çalışmalarda diğer algoritmalara göre metin madenciliği alanında daha sık kullanılan ve daha iyi sonuçlar vermiş olan naive bayes ve destek vektör makineleri yaklaşımları ayrı başlıklar altında incelenecektir.

1.2.2.1 Özellik Vektörü Oluşturma (TF-IDF)

Terim frekansı (TF) ve ters doküman frekansı (IDF) değerlerinin hesaplanarak birbirleri ile çarpılması sonucu TF-IDF değeri elde edilir. Bu çarpma işlemiyle ilgili terim ağırlıklandırılır. Elde edilen bu değerden yararlanılarak özellik vektörü oluşturulur. TF-IDF yöntemi kullanılarak gerçekleştirilen birçok farklı çalışma gözlemlemiştir.

TF-IDF yöntemi ile oluşturulan özellik vektörünün bize sağlamış olduğu avantaj sınıflandırma başarısını daha yüksek tutacak kelimeleri dokümanlar içerisinden seçmesidir. Elimizde 300 adet e-postanın bulunduğunu ve bütün e-postaların içerisinde "tl" kelimesinin gözlemlendiğini varsayalım bütün dokümanlarda gözlemlenen bu ifadenin sınıflandırmaya etkisi çok düşük olacaktır. Bu ifadenin IDF değeri hesaplandığında "0" olarak bulunacaktır[27.] Bu yöntem yardımıyla sınıflandırma başarısını daha yüksek kılacak olan kelimeler özellik vektörünü oluşturacaktır[28].

[27] (Raschka, 2014)
[28] (Christopher D. Manning, 2008)

$$(\text{Terim Frekans}), \mathrm{TF(t,d)} = \frac{\text{Terimin dökümanda geçme sıklığı}}{\text{Dokümandaki toplam terim sayısı (tekrarlı)}} \quad (1.1)$$

$$(\text{Ters Doküman Frekansı}) , \mathrm{IDF} = \log\left(\frac{\text{Toplam doküman sayısı}}{\text{Terimin geçtiği doküman sayıs}}\right) \quad (1.2)$$

1.3 NAIVE BAYES

Bu bölümde istatistikte ve metin madenciliğinde sıklıkla kullanmış olduğumuz formüllerden birisi olan bayes formülüne ve onun yazarı olan Thomas Bayes'in hayatına kısaca yer verilmiştir. Daha sonra bir sınıflandırma algoritması olarak naive bayes formülü ele alınmıştır. Sınıflandırma işleminin metin madenciliğinde nasıl kullanıldığı ayrıca bir başlık altında incelenmiştir. Metin madenciliği alanında sıklıkla kullanılan iki bayes yaklaşımı olan Multinomial ve Bernoulli Naive Bayes yaklaşımlarının sözde kodları üzerinden çalışma prensipleri açıklanmıştır. Metin madenciliği alanında naive bayes yaklaşımıyla gerçekleştirilen çalışmalar incelenerek sunulmuştur.

1.3.1 Thomas Bayes ve Teoremi

Doğum tarihi tam olarak bilinmemekle birlikte kaynaklarda 1701 yılında dünyaya geldiği belirtilen Thomas Bayes İngiliz bir matematikçidir. Olasılık problemlerimizi çözmek için kullandığımız teoremlerden birisi olan bayes teoremi Thomas Bayes tarafından oluşturulmuştur. Çalışmasını yayımlayamadan 7 Nisan 1761 tarihinde vefat eden Thomas Bayes'in bu çalışması vefatından iki yıl sonra 1763 yılında arkadaşı Richard Price tarafından yayınlanmıştır[29].

Bayes'in teoremi; B durumunun A koşulu ile gerçekleşmesi olasılığının A durumunun B koşulu ile gerçekleşmesi olasılığından farklı olduğunu belirtmekte ve bu farklılıktaki ilişkiyi ortaya çıkartmaktadır. Denklem 2'de bayes formülü görülmektedir[30].

$$\mathrm{P(A|B)} = \frac{\mathrm{P(B|A)} * \mathrm{P(A)}}{\sum \mathrm{P(B)}} \quad (2)$$

[29] (Bellhouse, 2004)
[30] (Jurafsky, 2016)

1.3.2 Bir Sınıflandırma Algoritması Olarak Naive Bayes

Denetimli bir sınıflandırma algoritması olan naive bayes yaklaşımı sınıflandırma işlemini gerçekleştirebilmek için eğitim kümesine ihtiyaç duymaktadır. Eğitim verilerinden öncelikle özellik çıkarımı yapılır ve bu özelliklerin gözlenme sıklıkları hesaplanarak bir frekans tablosu oluşturulur. Bu tablo aracılığıyla olasılıklar elde edilir ve bu olasılıklar üzerinden bir olayın hangi sınıfa ait olduğu bilgisi naive bayes sınıflandırma algoritması kullanılarak hesaplanır.

1.3.3 Naive Bayes ile Metin Sınıflandırma

Denetimli öğrenme algoritmalarından birisi olan naive bayes algoritması sınıflandırma probleminin çözümü için eğitim kümesine ihtiyaç duymaktadır. Eğitim kümesi içerisinde negatif ve pozitif olarak tanımlanmış verilerden kategorilerini en iyi temsil edecek kelimeler çıkarılarak bu kelimelerle bir vektör oluşturulur.

Bu vektör özellik vektörü olarak adlandırılır. Eğitim kümesi kullanılarak oluşturulan özellik vektörü kullanılarak model oluşturulur. Model aracılığıyla hangi kategoriye ait olduğu bilinmeyen bir e-postanın kategorisi tahmin edilebilir[28].

Denklem 2'de bayes formülü incelendiğinde formüldeki P(A|B) ifadesinde A; indirim, fırsat, kaçırma, bedava gibi özellik vektörünün kelimelerine, B ise negatif ve pozitif kategorisine karşılık gelmektedir. Her bir kelimenin hem negatiflerde hem de pozitiflerde bulunma olasılığı hesaplanır. Özellik vektöründe yer alan her bir kelimenin istenilen ve istenmeyen sınıfına ait 1 yumuşatılmış laplace yumuşatması değeri hesaplanır.

Laplace yumuşatmasının kullanılmasının nedeni hiç gözlenmemiş olan bir terim denkleme girdi olarak verildiğinde olasılık sonucunun sıfır olarak hesaplanmasıdır. Bu durum "sıfır problemi" olarak adlandırılır. Problemin çözümü için Laplace yumuşatması yöntemi kullanılmaktadır[31].

İstenmeyen e-postalar için 1 yumuşatılmış laplace olasılığı denklem 3'de verilmiştir[32]

[31] (stanford.edu, Naive Bayes text classification, 2009)
[32] (Jurafsky, 2016)

$$P(\text{aranan} \mid \text{istenmeyen}) = \frac{K_{\text{istenmeyen}} + 1}{\sum_1^n L_{\text{istenmeyen}} + 1} \quad (3)$$

$K_{istenmeyen}$, aranan ifadenin negatiflerde gözlenme sıklığını ifade etmektedir. $L_{istenmeyen}$ ise özellik vektöründeki ifadelerin negatiflerde gözlenme sıklığını ifade etmektedir.

İstenilen e-postalar için 1 yumuşatılmış laplace olasılığı denklem 4'de verilmiştir[32].

$$P(\text{aranan} \mid \text{istenilen}) = \frac{K_{\text{istenilen}} + 1}{\sum_1^n L_{\text{istenilen}} + 1} \quad (4)$$

$K_{istenilen}$, aranan ifadenin pozitiflerde gözlenme sıklığını ifade etmektedir. $L_{istenilen}$ ise özellik vektöründeki ifadelerin pozitiflerde gözlenme sıklığını ifade etmektedir.

X naive bayes tarafından sınıflandırılacak bir e-posta olsun. X e-postasının içerisindeki kelimelerden özellik vektörünün elemanı olanlar belirlenir. Belirlenen kelimeler yardımıyla kelimelerin istenilen ve istenmeyen sınıfına ait olasılıklar toplanarak P (X, istenen) ve P(X, istenmeyen) adlı iki olasılık değerine ulaşılır, elde edilen olasılık değerleri karşılaştırılarak X e-postasının kategorisi belirlenir[32].

Metin madenciliği alanında metin sınıflandırmak için sıklıkla kullanılan iki farklı naive bayes yaklaşımı olan bernoulli naive bayes ve Multinomial naive bayes yaklaşımlarının algoritmalarına ve bu algoritmaların açıklamalarına aşağıda yer verilmiştir.

1.3.3.1 Bernoulli Naive Bayes

Aşağıdaki şekilde (***Şekil 2***) sözde kodu verilmiş olan Bernoulli Naive Bayes algoritması iki farklı fonksiyondan meydana gelmektedir. Bunlardan ilki öğrenme fonksiyonu ikincisiyse uygulama fonksiyonudur. Öncelikle öğrenme fonksiyonu sonrasında ise uygulama fonksiyonu algoritmasının her bir satırının açıklaması aşağıda sunulmuştur.

```
TRAINBERNOULLINB(ℂ, 𝔻)
1  V ← EXTRACTVOCABULARY(𝔻)
2  N ← COUNTDOCS(𝔻)
3  for each c ∈ ℂ
4  do N_c ← COUNTDOCSINCLASS(𝔻, c)
5     prior[c] ← N_c/N
6     for each t ∈ V
7     do N_ct ← COUNTDOCSINCLASSCONTAININGTERM(𝔻, c, t)
8        condprob[t][c] ← (N_ct + 1)/(N_c + 2)
9  return V, prior, condprob

APPLYBERNOULLINB(ℂ, V, prior, condprob, d)
1  V_d ← EXTRACTTERMSFROMDOC(V, d)
2  for each c ∈ ℂ
3  do score[c] ← log prior[c]
4     for each t ∈ V
5     do if t ∈ V_d
6           then score[c] += log condprob[t][c]
7           else score[c] += log(1 − condprob[t][c])
8  return arg max_{c∈ℂ} score[c]
```

Şekil 2: Bernoulli Naive Bayes Algoritmasının Sözde Kodu

Kaynak: stanford.edu. (2009, 4 7). *The Bernoulli model*. 6 8, 2017 tarihinde stanford.edu: https://nlp.stanford.edu/IR-book/html/htmledition/the-bernoulli-model-1.html adresinden alındı

Öğrenme fonksiyonu için başlangıç değeri olarak tüm sınıfların ve tüm öğrenme dokümanlarının bilgisi girdi olarak verilir. Öncelikle dokümanların içerisinden elde edilen tüm kelimeler "V" (vocabulary) ile ifade edilir.

"N" ifadesi öğrenme kümesine girdi olarak verilen toplam doküman sayısını ifade eder. Her bir sınıf için döngü oluşturulur. Bizim problemimizde negatifler ve pozitifler olmak üzere iki sınıf mevcuttur. Her bir sınıf için doküman sayısı hesaplanır. Bir sınıf içerisindeki doküman sayısının tüm dokümanlara oranından yararlanılarak her bir sınıf için öncelikli olasılık hesaplanır $\left(\frac{N_c}{N}\right)$. Buradaki prior[c] çok boyutlu bir diziyi ifade eder. Bizim problemimizde istenen ve istenmeyen sınıfına ait e-postalar için öncelikli olasılık değerini barındırır. V ifadesinin içinde yer alan tüm kelimelerin her biriyle işlem yapabilmek için döngü oluşturulur, Sınıf içerisinde ilgili kelimeyi içeren doküman sayısı hesaplanır (N_{ct}).

Her bir kelimenin her bir sınıf için koşullu olasılık değeri hesaplanır**,** condprob ifadesi koşullu olasılığı ifade eden (conditional probability) çok boyutlu bir dizidir. Daha açık bir ifadeyle $t \in V$, $c \in (istenilen, istenmeyen)$ olduğu bilinen bu algoritmada condprob dizisi öğrenme dokümanında yer alan her bir kelimenin her bir sınıf için koşullu

olasılık değerini içerir ve bu koşullu olasılık değeri denklem 5'de sunulan formülle hesaplanmaktadır.

$$\frac{(N_{ct}+1)}{(N_c+2)} \tag{5}$$

N_{ct}: t kelimesini içeren c sınıfındaki doküman sayısı , N_c: c sınıfındaki döküman sayısı

Öğrenme fonksiyonu; tüm dokümanlardaki kelimeleri, öncelikli olasılık ve koşullu olasılık değerlerini sonuç olarak döndürmektedir.

Uygulama fonksiyonunun kullanımı için başlangıç değeri olarak tüm sınıfların bilgisi, öğrenme veri kümesinde kullanılan dokümanların içerisindeki tüm terimler, öncelikli olasılık bilgisi, koşullu olasılık bilgisi ve sınıflandırma işlemine konu olan doküman bilgisi girdi olarak verilir. Sınıflandırılacak olan dokümandaki tüm terimler "V_d" ile temsil edilir. Her bir sınıf için döngü oluşturulur.

Her bir sınıf için puan değeri sınıfa ait öncelikli olasılık değerinin logaritması alınarak hesaplanır (score[c] = log(prior[c])). Score[c] çok boyutlu bir dizidir ve tüm sınıfların puan durumunu içinde barındırır.

Öğrenme verisinin çıktısı olan V ifadesi (tüm dokümanlardaki kelimeler) için bir döngü oluşturulur. Döngüdeki kelimenin sınıflandırılacak olan dokümandaki tüm terimlerin içerisinde var olup olmadığı kontrol edilir. Eğer kelime varsa koşullu olasılık değerinin logaritması alınarak sınıfa ait skor değerine eklenir. Aksi durumda kelimenin koşullu olasılığı 1 değerinden çıkarılır ve elde edilen sonucun logaritması alınarak sınıfa ait skor değerine eklenir. Uygulama fonksiyonumuz son olarak içerisinde bizim problemimiz için pozitiflere ve negatiflere ait olasılıkları barındıran score[c] dizisine $argmax$ fonksiyonunu uygulayarak (Bu fonksiyon dizi içerisindeki en büyük elemanın değerini döndürür) en yüksek değere sahip olan sınıfı belirler. Belirlenen bu sınıf bizim

dokümanımızın sınıfı olarak tanımlanır. Böylece sınıflandırma işlemi tamamlanmış olur[33,34].

1.3.3.2 Multinomial Naive Bayes

Aşağıdaki şekilde (***Şekil 3***) sözde kodu verilmiş olan Multinomial Naive Bayes algoritması Bernoulli Naive Bayes algoritması gibi öğrenme ve uygulama fonksiyonundan meydana gelmektedir.

```
TRAINMULTINOMIALNB(ℂ, 𝔻)
 1  V ← EXTRACTVOCABULARY(𝔻)
 2  N ← COUNTDOCS(𝔻)
 3  for each c ∈ ℂ
 4  do N_c ← COUNTDOCSINCLASS(𝔻, c)
 5     prior[c] ← N_c / N
 6     text_c ← CONCATENATETEXTOFALLDOCSINCLASS(𝔻, c)
 7     for each t ∈ V
 8     do T_ct ← COUNTTOKENSOFTERM(text_c, t)
 9     for each t ∈ V
10     do condprob[t][c] ← (T_ct+1) / Σ_t'(T_ct'+1)
11  return V, prior, condprob

APPLYMULTINOMIALNB(ℂ, V, prior, condprob, d)
1  W ← EXTRACTTOKENSFROMDOC(V, d)
2  for each c ∈ ℂ
3  do score[c] ← log prior[c]
4     for each t ∈ W
5     do score[c] += log condprob[t][c]
6  return arg max_c∈ℂ score[c]
```

Şekil 3: Multinomial Naive Bayes Algoritmasının Sözde Kodu

Kaynak: *Naive Bayes text classification*. 6 6, 2017 tarihinde stanford.edu: https://nlp.stanford.edu/IR-book/html/htmledition/naive-bayes-text-classification-1.html adresinden alındı.

Bernoulli Naive Bayes algoritmasından farklı olarak öğrenme algoritmasında her sınıftaki dokümanların birleştirilmiş hali $text_c$ ile ifade edilir. Bu durumda işlemler sınıf içerisindeki doküman sayısı ile değil, sınıf içerisindeki dokümanların kelimelerinin birleştirilmesi ile elde edilen $text_c$ ifadesi üzerinden gerçekleştirilmektedir. V ifadesinin içinde yer alan tüm kelimelerin her biriyle (Her bir kelime = t) işlem yapabilmek için döngü oluşturulur. Bu döngüde her bir sınıfa ait $text_c$ içerisinde t teriminin gözlenme sıklığı hesaplanır ve bu değer T_{ct} olarak ifade edilir. V ifadesi içinde yer alan kelimeler

[33] (stanford.edu, The Bernoulli)
[34] (scipy.org)

için tekrar bir döngü oluşturulur. Kelimelerin koşullu olasılık değeri denklem 6'da verilen denklem ile hesaplanır.

$$\hat{P}(t|c) = \left(\frac{T_{ct}+1}{\sum_{t' \epsilon V}(T_{ct'}+1)}\right) = \left(\frac{T_{ct}+1}{\sum_{t' \in V} T_{ct'}+\sum_{t' \in V}}\right) \quad (6)$$

T_{ct}: Sınıfa ait $text_c$ içerisinde t kelimesinin gözlenme sıklığıdır.

$\sum_{t' \in V} T_{ct'}$: V içerisinde bulunan her bir kelimenin (tekrarsız) $text_c$ içerisinde gözlenme sıklığının toplamı.

$\sum_{t' \in V}$: V içerisinde bulunan her bir (tekrarsız) kelime adedinin toplamı. +1 ifadesi Laplace yumuşatmasından dolayı denkleme dahil olmuştur. Öğrenme fonksiyonu; tüm dokümanlardaki kelimeleri, öncelikli olasılık ve koşullu olasılık değerlerini sonuç olarak döndürmektedir.

Uygulama fonksiyonunda Bernoulli Naive Bayes yaklaşımından farklı olarak sınıfı belirlenmek istenen dokümanın içerisindeki tüm kelimeler W ifadesi ile ifade edilmektedir. Her bir sınıf için döngü oluşturulur. Her bir sınıf için öncelik olasılık değerinin logaritması alınarak sınıfın skoruna eklenir. Daha sonra doküman içerisindeki tüm terimler için koşullu olasılık değerinin logaritması alınarak sınıfın skoruna dahil olur.

Yine Bernoulli naive bayes algoritması gibi argmax fonksiyonu yardımıyla ilgili dokümanın hangi sınıfa ait olduğu bilgisi elde edilir. Böylelikle sınıflandırma işlemi tamamlanmış olur[35].

1.3.4 Naive Bayesle Gerçekleştirilmiş Çalışmalar

Sınıflandırma problemlerini çözmek için kullanılan denetimli öğrenme algoritmalarından birisi olan naive bayes yaklaşımı metin madenciliği alanında sıklıkla kullanılmaktadır. Uygulanmasının basit, diğer sınıflandırma algoritmalarıyla karşılaştırıldığında başarısının metin madenciliği alanında yüksek olması nedeniyle bu alanda sıklıkla tercih edilmektedir.

[35] (stanford.edu, Naive Bayes text classification)

Naive bayes sınıflandırma algoritması; bir metnin yazarının belirlenmesinde, bir haber metninin hangi kategoriye ait olduğunun bulunmasında, herhangi bir makalenin yazarının yaşının ve cinsiyetinin tahmin edilmesinde, bir kullanıcı tarafından gönderilen twitter metninin hangi duygularla yazıldığının belirlenmesinde, film veya dizi sektöründe izleyicilerinin yorumlarına bakılarak herhangi film veya dizinin izleyenler tarafından beğenilip beğenilmediğinin araştırılmasında , twitter kullanıcıları tarafından yapılan yorumlardan herhangi bir ürüne karşı tutumun belirlenmesinde, kullanıcının telefonuna gelen SMS'in istenilen veya istenmeyen bir SMS olmasının belirlenmesinde, İngiliz alfabesi kullanılarak yazılmış Türkçe metinlerin Türk alfabesine göre yeniden oluşturulması aşamasında, e-posta hesaplarımıza gelen e-postaların negatif veya pozitif olarak sınıflandırılması gibi daha sayamadığımız birçok sınıflandırma probleminin çözümünde kullanılmaktadır.

Sahami ve arkadaşları *A Bayesian Apporach to Filtering Junk E-Mail* adlı çalışmalarında negatifleri sınıflandırmak için naive bayes yaklaşımını kullanmışlardır. Bunun yanında bir e-postanın konu başlığına bakarak alfa numerik karakterlerin alfa numerik olmayan karakterlere oranı hesaplanmış ve bu oranın %5'den büyük olması durumunda ilgili e-posta negatif olarak sınıflandırmışlardır.

Bu çalışmada iki adet veri kümesi kullanılmıştır. İlki 1789 adet ikincisi 1183 adet e-posta verisi içermektedir. Birinci veri kümesi daha çok yetişkin içerikli web sitelerden gelen negatiflerden oluşurken ikinci veri kümesi ise kolay yoldan para kazanma konusunda gelen negatiflerden oluşmaktadır.

Çalışmada kelimeler, kelime grupları ve alana özgü kelimeler kullanılarak üç farklı özellik vektörü çıkarılmıştır, birinci veri kümesi için en iyi sonuç bu üç özellik vektörü kullanılarak %100 kesinlikle ve %98.3 hassasiyetle negatifler belirlenirken %96.2 kesinlik ve %100 hassasiyetle pozitifler belirlenmiştir. Çalışmanın devamında negatifler kendi içerisinde yetişkin içerikli negatifler ve diğer negatifler olarak ayrıca iki ayrı sınıfa ayrılmıştır.

Bu çalışmanın en önemli özelliği naive bayes yaklaşımını e-postaların sınıflandırmasında kullanan ilk çalışma olmasıdır. Aynı zamanda bu çalışma destek

vektör makinelerinin öğrenirken parametrelerin varyansını kontrol ettiği için daha iyi sonuç vereceğini söylemektedir[36].

An Evaluation of Naive Bayesian Anti-Spam Filtering adlı çalışmada herkesin kullanımına sunulacak Ling-Spam adında negatifleri ve pozitifleri barındıran derlem oluşturulmuştur. Derlem 2893 adet e-postadan meydana gelmektedir ve bu e-postaların %16.6'sı negatiflerden oluşmaktadır. Bu derlem içerisinde; e-posta listeleri, iş ilanları, haber grubu listeleri hakkında duyurular içermektedir.

Bu çalışmada Sahami ve arkadaşlarının çalışması[28] model alınarak ortak bilgi fonksiyonundan yararlanılmış 700 boyutlu bir özellik vektörü oluşturulmuş ve 50'şer 50'şer adımlarla bu özellik vektörü kullanılarak sınıflandırma işlemi gerçekleştirilmiştir.

Sınıflandırmada sınıflandırma başarısına etkisi olmayacak olan en sık kullanılan kelimeleri bir listede toplayarak (etkisiz kelimeler) ilgili e-postalardan ön işleme aşamasında çıkartıp eğitim modelleri oluşturulmuş ve daha sonrasında sınıflandırma adımına geçilmiştir. Çalışmada λ=999 t=0.999 olarak seçildiği durumda 300 elemanlı özellik vektörüyle e-postalar %100 kesinlik ve %63.05 hassasiyetle sınıflandırılmıştır[37].

An experimental comparison of naive Bayesian and keyword-based anti-spam filtering with personal e-mail Messages adlı çalışma An Evaluation of Naive Bayesian Anti-Spam Filtering adlı makalede sunulan çalışmanın kişisel e-postalara uygulanmasıyla ortaya çıkan bir çalışmadır. Ek olarak bu çalışmada kişisel bir e-posta derlemi oluşturulmuştur (PU1).

1099 adet e-posta içeren bu derlem 22 aylık bir periyotta meydana getirilmiş ve içerisinde 481 adet negatif barındırır. Çalışmada kullanılan veri kümesinin %43.8'i istenmeyen e-postadır.

Yapılan bu çalışmada t=0.999, λ=999 olarak seçildiği durumda 600 elemanlı özellik vektörü ile %49.45 kesinlik ve %98.31 hassasiyetle e-postalar sınıflandırılmıştır[38].

---

[36] (Sahami, 1998)
[37] (Androutsopoulos I. K., 2000, An Evaluation...)
[38] (Androutsopoulos I. K., 2000, An experimental...)

Spam Filtering with Naive Bayes – Which Naive Bayes? adlı çalışmada hangi Naive bayes türünün metin madenciliğinde daha iyi sonuç verdiği ile ilgili bir çalışma yapılmıştır. Bu çalışmada 5 farklı naive bayes yaklaşımı (Mutlivariate Bernoulli Naive Bayes , Multinomial NB, TF attributes - Multinomial NB, Boolean attributes - Multivariate Gauss NB , Flexible Bayes) karşılaştırılmıştır. Çalışmada "Enron-Spam" veri tabanı kullanılmış içerisinde 6 farklı veri kümesi içeren veri kümesi kullanılarak elde edilen sonuçlarda; ortalama olarak en iyi sonuç Multinomial NB ile elde edilmiştir. Negatifler %97.53 ve pozitifler %97.26 hassasiyetle sınıflandırılmıştır. Negatiflerin sınıflandırılmasında bernoulli naive bayes %96.52 başarı ile en iyi üçüncü algoritma olurken pozitiflerin sınıflandırılmasında %89.05 başarıyla en düşük sınıflandırma başarısına sahip algoritma olmuştur[39].

Reklam içerikli e-postaların metin madenciliği yöntemleri ile otomatik tespiti adlı çalışmada 400'ü pozitif 400'ü negatif 800 adet Türkçe e-posta kullanılmıştır. Destek vektör makineleri, K-en yakın komşuluk ve Naive bayes algoritmalarıyla gerçekleştirilen çalışmada en iyi sonuç K-en yakın komşuluk algoritması ile k=14 ve 4263 elemanlı binary özellik vektörü ile %96,5 hassasiyetle sınıflandırma işlemi gerçekleştirilmiştir.

Ayrıca çalışmada Naive bayes ve k-en yakın komşuluk algoritmasının ortalama olarak büyük ölçüde başarı sağladığına değinilmektedir[40].

Farklı özellik vektörleri ile Türkçe dokümanların yazarlarının belirlenmesi adlı çalışmada Hürriyet, Sabah ve Vatan gazetelerinin web sayfalarında politika, güncel, spor ve magazin konularında köşe yazısı yazan 18 farklı yazarın 35 köşe yazısı toplanmıştır. Sınıflandırma için 10 farklı özellik vektörü kullanılmıştır. Sınıflandırma algoritması olarak Naive Bayes, SVM, C 4.5 ve RF algoritmaları kullanılmıştır. Sınıflandırma başarılarının ortalamasına bakıldığında en iyi sonucu Naive Bayes algoritmasıyla ve ikinci en iyi sonuç ise SVM algoritmasıyla elde edilmiştir, ortalama olarak en iyi sonuç veren özellik vektörünün en yüksek başarısı SVM algoritmasıyla ve ikinci en yüksek başarısı ise Naive Bayes algoritmasıyla elde edilmiştir[41] .

---

[39] (Metsis, 2006)
[40] (Çalış, 2013)
[41] (M. Fatih Amasyalı, 2006)

Türkçe dilbilgisi özelliklerini kullanarak web tabanlı haber metinlerinin sınıflandırılması adlı çalışmada iki ayrı veri kümesi kullanılmıştır birinci veri kümesinde NKÜ tarafından hazırlanan Hürriyet, Milliyet, Vatan Gazetesi, Cumhuriyet ve Haber Türk adlı haber kuruluşlarının web sayfalarından ekonomi, sağlık, magazin, spor ve siyaset kategorilerinden iki farklı veri toplanmıştır. Her bir kategoriden 100 adet eğitim 50 adet test verisi alınarak veri kümesi oluşturulmuştur. YTÜ veri kümesi Amasyalı ve Beken tarafından oluşturulan 1150 adet haber metninden meydana gelmektedir. Çalışmada dil bilgisi işlemleri gerçekleştirebilmek için zemberek adlı doğal dil işleme kütüphanesinden yararlanılmış ve haber metni içerisindeki cümlelerde yer alan kelimeler isim, fiil, sıfat, bağlaç ve zarf olarak belirlenmiştir.

Yukarıda belirlenen kelimelerin kombinasyonları ile özellik vektörleri oluşturulmuş ve Naive Bayes, SVM, C 4.5 ve RF sınıflandırma algoritmaları kullanılarak sınıflandırma işlemi gerçekleştirilmiştir. Her iki veri kümesinden de en yüksek sonuç Naive Bayes algoritmasıyla elde edilmiştir[42].

İngiliz alfabesi kullanılarak yazılmış Türkçe metinlerin Türk alfabesine göre yeniden oluşturulması adlı çalışmada İngilizce harflerle yazılan Türkçe metinlerin Türk alfabesine göre yeniden oluşturmayı amaçlamaktadır. Sözlük anlamı belirleme (SAB) , cümle içerisinde İngilizce alfabe ile yazılan bir kelimenin Türk alfabesine dönüştürülmesi için anlam bazlı çıkarımlar makine öğrenmesi algoritmaları ile sağlanmıştır. Naive bayes, n-Gram, Ada Boost, J48 algoritmaları kullanılarak sonuçları birbirileri ile karşılaştırılmış ve en iyi sonuç Naive Bayes algoritması ile elde edilmiştir[43].

Turkish anti-spam filtering using binary and probabilistic models adlı çalışmada ilk defa akademik olarak kullanılabilecek bir Türkçe e-postalar için istenen ve istenmeyen e-postaları içeren veri kümesi oluşturulmuştur. Bu veri kümesi içerisinde 300 adet istenen 300 adet istenmeyen e-postalardan oluşan 600 adet öğrenme e-posta verisi barındırmaktadır. 100 adet istenen 100 adet istenmeyen e postanın da bulunduğu ayrı bir sınama verisini de barındıran bu veri kümesi toplamda 800 adet veriden oluşmaktadır .

[42] (Tüfekci, 2012)
[43] (Okur, 2013)

Çalışmada ortak bilgi fonksiyonundan yararlanarak 50 ve 75 elemanlı özellik vektörü oluşturulmuştur ve sınıflandırma için binary model ve olasılık modeli kullanılmıştır. Sınıflandırma işlemine geçmeden önce metinlerden kelime köklerini çıkarmak yerine n-gram 5 olarak seçilmiş ve işlemler bu koşul altında gerçekleştirilmiştir. Binary modelin uygulanması sonucunda en iyi sonuç; katsayısı 10 olan durumda 75 elemanlı özellik vektörü kullanılarak istenmeyen e-postaları %88 ve istenilen e-postaları %99 doğrulukla sınıflandırmıştır. Olasılık modellinde katsayısı 10 olan durumda 75 elemanlı özellik vektörü kullanılarak istenmeyen e-postaları %86 ve istenilen e-postaları %90 doğrulukla sınıflandırılmıştır. Ortak bilgi fonksiyonu kullanılarak kelimelerin istenen ve istenmeyen e-postalarda gözlenme sıklıklarına bakılarak her bir kelimenin ortak bilgi puanı oluşturulmuştur. Çalışmada en iyi sonuç %93'lük bir başarıyla binary modelde elde edilmiştir[44].

1.4 DESTEK VEKTÖR MAKİNELERİ

Bu bölümde Destek Vektör Makinelerinin tarihçesine değinilmiş ve sonrasında Destek Vektör Makinelerinin çalışma prensibi anlatılmıştır. Farklı Destek Vektör Makineleri yaklaşımlarının hangi durumlar altında kullanılacağına değinilmiştir. Destek Vektör Makinelerinin metin madenciliği alanından nasıl kullanıldığı ile ilgili bilgiler verildikten sonra Destek Vektör Makineleri ile gerçekleştirilen çalışmalar sunulmuştur.

Destek vektör makinelerinin temelleri 1992 yılında Bernhard E. Boser, Isabelle M. Guyon ve Vladimir N. Vapnik'in ortak çalışması sonucunda "A Training Algorithm for Optimal Margin Classifers[45]" adlı makaleyle ortaya atılmıştır.

Leon Bottou ve diğerlerinin 1994 yılında yayınlanan "Comparison of Classifer Methods : A Case Study in Handwritten Digit Recognition[46]" , adlı çalışmasında diğer sınıflandırma algoritmaları ile Optimal Margin Classifers algoritması karşılaştırılarak sonuçlar tartışılmıştır. OMC'nin çok iyi bir doğrulukla sınıflandırma yaptığına değinilmiş

[44] (Ergin S., 2012)
[45] (Boser, 1992)
[46] (Bottou, 1994)

ve görüntünün pikselleri şifrelenmiş olsa bile (örneğin rasgele permütasyon ile şifrelenmiş olsun) yine de bu algoritma yardımıyla sınıflandırılabileceği düşünülmüştür.

Cortes ve Vapnik tarafından 1995 yılında yayınlanan "Support-Vector Networks [47]" adlı çalışmada ilk defa Destek Vektör kavramına değinilmiştir. Bu çalışmanın amacı elle yazılan rakamları doğru olarak bilgisayar tarafından sınıflandırabilmektir. Çalışmada diğer sınıflandırma algoritmaları ile destek vektör makinesi algoritması karşılaştırılmış ve hata oranları üzerinden en az hatayla en iyi sonuçlardan birisi destek vektör makinesi algoritması ile elde edilmiştir.

Çalışmada iki ayrı veri kümesi kullanılmıştır. Bunlardan birincisi ABD posta ofisinin rakam kütüphanesi (7300 adet öğrenme ve 2000 adet test verisi) ikincisi ise Ulusal Standartlar ve Teknoloji Enstitüsünün verileridir (60.000 adet öğrenme ve 10.000 adet sınama verisi) . İkinci veri kümesi ile yapılan çalışmada SVN ve LeNet1 (Yapay Sinir Ağları Algoritması) %1.1 hata göstererek en iyi sınıflandırmayı gerçekleştirmiştir.

Destek vektör makineleri elle yazılan rakamları tanıma yani günümüzde bilgisayarlar tarafından karakter tanıma (OCR, optical character recognition) olarak adlandırılan problemdeki başarısı sayesinde popüler bir algoritma olarak veri madenciliği alanında çalışan kişiler tarafından tanınmaktadır.

Vapnik'in riskleri minimize etmek için kullandığı algoritması daha sonralarda regresyon problemleri, tahmin problemleri, örüntü tanımlama alanlarında da kullanılarak iyi sonuçlar elde edilmiştir. Yine destek vektör makineleri günümüzde metin madenciliği alanında da sıklıkla kullanılmaktadır [48,49].

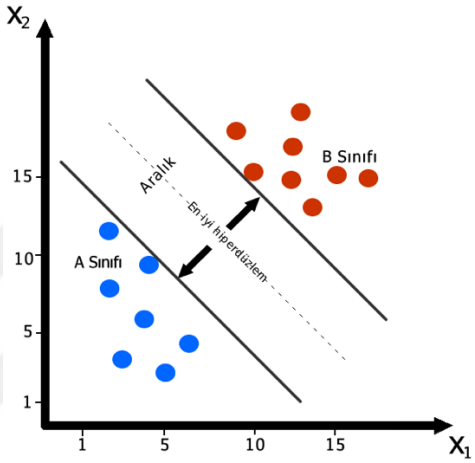
1.4.1 Destek Vektör Makineleri Çalışma Prensibi

Destek vektör makineleri iki sınıfı birbirinden ayırmak için kullanılan denetimli öğrenme algoritmasıdır. İki sınıfı birbirinden ayırabilmek için sınıflar arasındaki en uzak mesafeyi optimum şekilde hesaplamamız gerekmektedir. Optimum mesafe bulunduktan

[47] (Cortes, 1995)
[48] (clrc.rhul.ac.uk, 2012)
[49] (columbia.edu, 2015)

sonra buraya bir hiperdüzlem çizilir ve artık bu düzlem sayesinde iki sınıfı birbirinden ayırılarak sınıflandırma işlemi geçekleştirilmiş olur.

Destek vektör makineleri iki sınıfı birbirinden ayırdığı gibi daha sonraki çalışmalarda çoklu destek vektör makinesi sınıflandırıcısı da kullanılmıştır. Çoklu destek vektör makinelerini elimizde var olan tüm ikililerin destek vektör makineleri ile değerlendirilmesi sonucunda elde edilmiş olacaktır[50,51,52].



Şekil 4: Destek Vektör Makineleri

Kaynak: listendata.com. (2017, 1). *Support Vector Machine Simplified using R.* 6 8, 2017 tarihinde listendata.com: http://www.listendata.com/2017/01/support-vector-machine-in-r-tutorial.html adresinden alındı

Yukarıda (Şekil 4) görüldüğü üzere destek vektör makineleri A ve B gibi iki farklı sınıfı birbirinden ayırmak için kullanılmaktadır. Buradaki en iyi uzaklığı bulabilmek için nasıl bir algoritma yapısına sahip olduğunu inceleyelim.

Elimizdeki aralık vektörünün boyutunun mutlak değeri alındığı zaman iki sınıf arasında hesaplanan uzaklık elde edilmiş olur. Sağ ve sol oklarla gösterilen aralığın iki eşit parçaya bölündüğünü kabul ettiğimizde elimizdeki aralığın uzunluğunun ifadesi Denklem 7'deki formülle ifade edilebilir.

[50] (Ng, 2015)
[51] (Law, 2011)
[52] (Korting, 2015)

$$\frac{1}{||\vec{w}||} + \frac{1}{||\vec{w}||} = \frac{2}{||\vec{w}||} \quad (7)$$

Bizim buradaki amacımız en büyük aralık için en iyi hiperdüzlemi çizmekti. Bunun için $(\vec{w})$ ne kadar küçük olursa elimizdeki aralık o kadar büyük olacaktır.

Bu ifadenin minimizasyonunu gerçekleştirmek için doğrusal olmayan optimizasyon işlemi uygulanır ve çözüm için Karush Kuhn Tucker'ın yaklaşımı kullanılmıştır[53]. 8.1 ve 8.2 denklemlerinden yararlanarak doğrusal olmayan optimizasyon işlemi gerçekleştirilir. λi KKT ile optimizasyonu gerçekleştirmek için kullanılan Lagrange çarpanıdır.

Karush Kuhn Tucker'ın optimizasyon algoritması kısıtlar altında karar vermek için oluşturulmuştur. Bizim problemimizde kısıt olarak optimum uzaklığı belirlerken öğrenme veri kümesindeki elemanların çizilen sınırın içinde kalmaları olarak tanımlayabiliriz. Bu tür problemlerin çözümünde Karush Kuhn Tucker'ın kuadratik programlama tekniğinden yararlanılmaktadır[54].

$$\vec{\mathrm{w}} = \sum_{\mathrm{i}=0}^{\mathrm{N}} \lambda_i y_i \vec{\mathrm{x}} \quad (8.1)$$

$$\sum_{\mathrm{i}=0}^{\mathrm{N}} \lambda_i y_i = 0 \quad (8.2)$$

Şekil 4'deki kesikli çizgiler hiperdüzlemi temsil etmektedir. Bu düzlem en iyi uzaklığın bulunduğu varsayımı altında çizilmiştir.

$$(\text{Düzlem Denklemi}): g(\vec{x}) = \vec{w}^T * \vec{x} + \overrightarrow{w_o} \quad (9)$$

Şekil 4'de çizmiş olduğumuz hiperdüzlemi bir formülle ifade etmemiz gerekirse Denklem 9'da gösterilen düzlem denklemi ile ifade edebiliriz.

[53] (listendata.com, 2017)
[54] (Taha, 2000)

"Destek vektör makineleri ile sınıflandırmada genellikle {-1,+1} şeklinde sınıf etiketleri ile gösterilen iki sınıfa ait örneklerin, eğitim verisi ile elde edilen bir karar fonksiyonu yardımıyla birbirinden ayrılması amaçlanır. Söz konusu karar fonksiyonu kullanılarak eğitim verisini en uygun şekilde ayırabilecek hiperdüzlem bulunur"[55].

Şekil 4'de sunulan grafikte kırmızı noktaların etiket değeri +1 , mavi noktaların etiket değeri -1 olarak tanımlanırsa ;

Kırmızı noktaların sınır çizgisi için;

$$\text{Her } y_i(\text{Etiket değeri}) = +1 \text{ için } \vec{w}^T * \vec{x} + \overrightarrow{w_o} = +1 \quad [53] \qquad (9.1)$$

Mavi noktaların sınır çizgisi için ;

$$\text{Her } y_i(\text{Etiket değeri}) = -1 \text{ için } \vec{w}^T * \vec{x} + \overrightarrow{w_o} = -1 \quad [53] \qquad (9.2)$$

Denklemler elde edilir. Bu denklemler sınır noktaların üzerinden oluşan hiperdüzlemin denklemleridir.

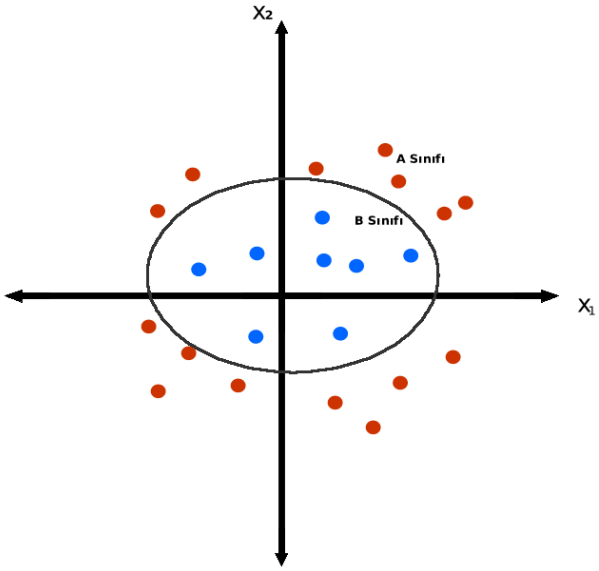
Bu denklemler yardımıyla aşağıdaki denklemler oluşturulur.

$$g(\vec{x}) = \vec{w}^T * \vec{x} + \overrightarrow{w_o} \leq -1 \, , x \in A \; sınıfı \qquad (10.1)$$

$$g(\vec{x}) = \vec{w}^T * \vec{x} + \overrightarrow{w_o} \geq +1 \, , x \in B \; sınıfı \qquad (10.2)$$

Böylelikle sınıflandırma işlemi gerçekleştirilir.

[55] (Taşkın KAVZOĞLU, 2010)

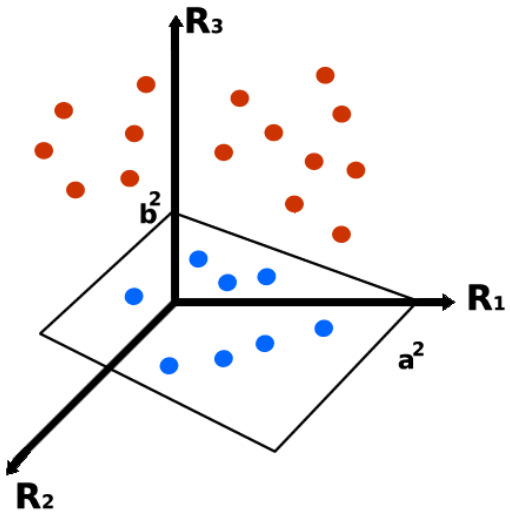


Şekil 5: Doğrusal olmayan sınıflandırma problemi

Kaynak: listendata.com. (2017, 1). *Support Vector Machine Simplified using R.* 6 8, 2017 tarihinde listendata.com: http://www.listendata.com/2017/01/support-vector-machine-in-r-tutorial.html adresinden alındı.

Sınıflandırma problemlerimiz her zaman doğrusal olarak karşımıza çıkmayabilir. Bu örnekte görüldüğü gibi doğrusal olmayan problemlerin çözümü için çekirdek fonksiyonumuzu değiştirmemiz gerekmektedir.

Şekil 5'de bulunan sınıflandırma problemimiz doğrusal bir çekirdek fonksiyonu ile çözülmek istendiğinde daha düşük bir doğrulukla sınıflandırma işlemi gerçekleştirilecektir. Doğrusal yöntemlerle yüksek doğrulukla sınıflandırılamayan problemler için doğrusal olmayan çekirdek fonksiyonlarını kullanarak, sınıflandırma problemi daha yüksek boyutlu uzaylara taşınır, bu uzaylarda düzlemler yardımıyla destek vektörleri oluşturulur ve bu vektörler sayesinde sınıflandırma işlemi gerçekleştirilir.



Şekil 6: Şekil 5'in Denklem 11 ile dönüşümün grafiği

Kaynak: listendata.com. (2017, 1). *Support Vector Machine Simplified using R*. 6 8, 2017 tarihinde listendata.com: http://www.listendata.com/2017/01/support-vector-machine-in-r-tutorial.html adresinden alındı.

$$\phi : (x_1, x_2) \rightarrow \left(x_1^2, \sqrt{2}x_1x_2, x_2^2\right) \qquad (11)$$

Şekil 5'deki grafikle ifade edilen örnek problemin Denklem 11'de verilen çekirdek fonksiyonu ile daha yüksek bir uzaya taşındığının temsili görseli Şekil 6'da verilmiştir. Bu dönüşüm için polinomal çekirdek fonksiyonu kullanılmıştır.

Polinomal çekirdek fonksiyonları doğal dil işleme uygulamaları, sepet analizi gibi problemlerde kullanılabilir. Doğal dil işleme alanında gerçekleştirilen çalışmalarda polinomal bir çekirdek fonksiyonu kullanıldığında polinomun derecesinin 2 olması önerilmektedir, daha yüksek boyutlu polinomlarda ise verinin ezberlenmesi sorunu ile karşılaşılabilir[56,57]. Destek vektör makineleri ile sınıflandırma işlemini gerçekleştirmek için kullanılan çekirdek fonksiyonları; doğrusal çekirdek fonksiyonu, polinomal çekirdek fonksiyonu, radyal temelli çekirdek fonksiyonu ve sigmoid çekirdek fonksiyonudur.

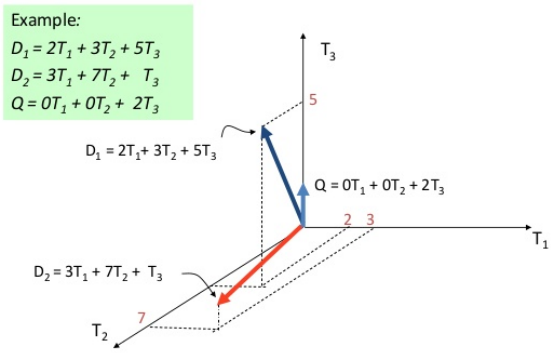
[56] (Goldberg, 2008)
[57] (Broomhead, 1998)

Radyal temelli çekirdek fonksiyonları genellikle sinyal ve görüntü işleme alanlarında kullanılmaktadır. Bu fonksiyon gaussien çekirdek olarak da adlandırılmaktadır. Rbf çekirdek fonksiyonu gauss fonksiyonunun özel bir hali olarak tanımlanmıştır[58]. Veri madenciliği alanında radyal tabanlı fonksiyon ağı adıyla farklı bir kullanımı mevcuttur. radyal tabanlı fonksiyon ağı, üç katmanlı bir yapay sinir ağıdır. Bir adet giriş bir adet gizli katman ve son olarak bir çıktı katmanı içermektedir[59].

Sigmoid çekirdek fonksiyonları örüntü tanıma, kümeleme, optimizasyon, kestirim ve tahmin problemleri için kullanılır. Sigmoid çekirdek fonksiyonu yapay sinir ağlarında da kullanılan bir çekirdek fonksiyondur[60].

1.4.2 Destek Vektör Makineleri ile Metin Sınıflandırma

Yukarıdaki açıklamalarda faklı destek vektör makinesi çekirdeklerinin kullanılması ile farklı problemlerin nasıl çözümlendiği konusuna değinilmiştir. Bu bölümde diğerlerinden farklı olarak destek vektör makineleri algoritmasının metin madenciliğinde kullanımı özetlenmektedir. Bunun için öncelikle öğrenme kümemizde var olan metinlerimizin nasıl vektörlerle ifade edileceğini inceleyelim. Aşağıdaki görselde üç adet elemana sahip olan özellik vektörü kullanılarak üç dokümanın (d1,d2,q) vektörlerle ifadesi gösterilmektedir.



Şekil 7 : Vektör Uzayı Modeli Grafiği
Kaynak: http://web.eecs.umich.edu/~mihalcea/498IR/Lectures/VectorSpaceModel.pdf

[58] (Bernstein, 2017)
[59] (McCormick, 2013)
[60] (Korting, 2015)

Yukarıdaki görselde D1,D2,q dokümanları temsil etmektedir. T1,T2,T3 ifadeleri ise özellik vektörümüzdeki üç adet kelimeye karşılık gelmektedir.
Eğer özellik vektörümüzde daha fazla eleman varsa problemimiz özellik vektörünün sahip olduğu eleman sayısının boyutunda bir uzaya taşınacaktır. Aşağıda görselin matris olarak nasıl temsil edilebileceği Tablo 1'de gösterilmektedir. Özellik vektöründe bulunan ifadelerin dokümanlarda (d1,d2,d3, ...) gözlenme sıklığından yararlanarak aşağıdaki matris elde edilmiştir.

Tablo 1 : Metinlerin Matrise Dönüştürülmesi

Özellik Vektörümüzdeki Kelimeler	**D1**	**D2**	**D3**	**...**
tl	1	1	0	...
alışveriş	2	1	0	...
indirim	0	0	1	...
.....	.....	.....	.....	...

Kaynak: Learning from Data Lecture 4: Vector Space Models K-Nearest Neighbor (+ Support Vector Machines), ESSLLI, 25 August 2016

Elde edilen bu matris kullanılarak destek vektör makineleri algoritmasıyla öğrenme işlemi gerçekleştirilir. Öğrenme işlemi gerçekleştirildiğinde destek vektörleri kullanılarak oluşturulmuş bir model elde edilir. Bu model kullanılarak X bir dokümanın sınıfı belirlenebilir[61,62] .

1.4.3 Destek Vektör Makineleriyle Gerçekleştirilmiş Çalışmalar

Using SVMs for text categorization adlı çalışmada Reuters gazetesinin metin madenciliği çalışmaları için oluşturmuş olduğu veri kümesi kullanılmıştır. 12,902 adet haberden oluşan bu veri kümesinin %75'i öğrenme %25'i test verisidir. Çalışmada haber metinlerini doğru olarak sınıflandırmak hedeflenmektedir. Özellik vektörü çıkarımında ortak bilgi fonksiyonundan yararlanılmıştır.

[61] (Mihalcea, 2015)
[62] (Malvina Nissim, 2016).

Findsim, Naive Bayes, BayesNets, Trees, Lineer SVM sınıflandırma algoritmaları kullanılarak gerçekleştirilen bu çalışmada toplamda en yüksek başarı destek vektör makineleri ile elde edilmiştir[63].

Support vector machines for spam categorization adlı çalışmada negatiflerin sınıflandırılması için 4 sınıflandırma algoritması (Ripper, Rocchio, and boosting desicion tree, DVM) iki farklı veri kümesinde test edilmiştir. Çalışma birden fazla özellik vektörü kullanılarak (TF, TF-IDF, Binary) gerçekleştirilmiştir. Veri kümesi olarak AT&T firmasının çalışanlarının 850 negatif ve 2150 pozitif verisi kullanılmıştır. En iyi 1000 özellik ortak bilgi fonksiyonu ile hesaplanmıştır.

Bu veri kümesi kullanılarak 6 farklı veri kümesi elde edilmiştir. E-postaların içerisindeki etkisiz kelimelerin arındırılması / arındırılmaması durumunda 2 farklı veri kümesi, e-postaların konu başlıklarının etkisiz kelimelerden arındırılması / arındırılmaması durumunda 2 farklı veri kümesi ve son olarak hem e-posta içeriğinin hem de e-posta konusundan etkisiz kelimeler arındırılması / arındırılmaması durumunda 2 farklı veri kümesi elde edilmiştir. İkinci veri kümesi AT&T teknik servis çalışanlarının adreslerinden toplanmıştır. Veri kümesi 314 negatif ve 303 pozitif kullanılarak oluşturulmuştur.

Veri kümesinin mevcut haliyle (büyük ve küçük harf içeren) ilk veri kümesi oluşturulmuştur, bütün kelimelerin küçük harfe çevrilmesiyle ayrıca bir veri kümesi daha elde edilmiştir. Tüm kelimeler küçük harfe çevrildiğinde hem e-posta içeriğinin hem de e-posta konusunun etkisiz kelimeler arındırılmadığı durumda sınıflandırma algoritması olarak boosting algoritması ve özelik vektörü çıkarımında terim frekansı yönteminden yararlanarak en iyi sonuca ulaşılmıştır.

Boosting algoritması en küçük hata oranına sahipken destek vektör makineleri binary özellik vektörü çıkarımı yöntemini kullanarak hataların dağılmasında en iyi sonuca sahiptir.

[63] (Dumais, 1998)

Çalışmada boosting karar ağacı algoritması için öğrenme süresinin aşırı uzun olduğuna değinilmektedir. Ayrıca etkisiz kelimelerin modelden arındırılmadığı durumda başarının daha yüksek olduğu da öneriler arasındadır[64].

Performance analysis of naive bayes classification, support vector machines and Neural Networks for Spam Categorization adlı çalışmada naive bayes, destek vektör makineleri ve yapay sinir ağları algoritmalarının karşılaştırmalı başarısı ortak bir derlem belirlenerek ölçülmüştür. Çalışmada (Androutsopoulos ve Diğerlerinin geliştirdiği 2000) Lingspam derlemi (2412 pozitif ve 481 negatif) kullanılmıştır. Özellik vektörü çıkarımında ortak bilgi fonksiyonundan yararlanılmıştır. Karşılaştırma işleminde birçok farklı eşik değer için $\lambda \epsilon (1,9,999)$ sınanmıştır.

Sonuç olarak 10 kat çapraz doğrulamalı geçerlilik ile sınanan sınıflandırma algoritmalarında en yüksek başarı yapay sinir ağları ile bulunmuş eşik değer çok yüksekken $\lambda = 99$ naive bayes destek vektör makinesinden daha iyi sonuç vermektedir, diğer durumlarda ise destek vektör makinesi naive bayes algoritmasından daha iyi sonuç vermektedir[65].

Konuşma dili kullanılarak demografik bilgilerin sınıflandırılması adlı çalışma İstanbul Üniversitesi psikoloji bölümü tarafından toplanan "Yaşanan Ahlaki Çatışma Örnekleri üzerinden Türkiye'de gündelik Ahlak Anlayışı" adlı konuda toplam 484 kişiye sorulan "Yaşadığınız bir ahlaki çatışmaya örnek verir misiniz?" sorusuna verilen yanıtlar demografik bilgileri de alınarak kaydedilmiş ve bu veriler kullanılarak çalışma gerçekleştirilmiştir.

Soruya cevap veren kişilerin yanıtları kullanılarak cinsiyet (kadın, erkek), eğitim (hiç okumamış/ortaokul mezunu, lise/üniversite öğrencisi , üniversite/ yüksek lisans/ doktora mezunu), yaş (10-20 yaş arası, 21-35 yaş arası, 36-66 yaş arası) ve medeni durumu (evli, bekar) tahmini için naive bayes, destek vektör makinesi, k-en küçük komşuluk algoritmaları kullanılmıştır.

[64] (Drucker, 1999)
[65] (Tantuğ, 2006)

Bu işlemi gerçekleştirmek için özellik vektörü olarak kelime köklerinin ters doküman frekansları, kök, tip, 2-gram bilgileri kullanılmıştır. Cinsiyet tahmininde en yüksek başarı k-en yakın komşuluk(%78.1), medeni hal sınıflandırmada en yüksek başarı destek vektör makineleri (%86.1), eğitim derecesinin sınıflandırmasında en yüksek başarı naive bayes algoritması ile (%95.2) ve yaş için en yüksek başarı sınıflandırma başarısı destek vektör makineleriyle (%87.3) elde edilmiştir[66].

Bileşik modellemeli ve modellemesiz metin bağımsız konuşmacı tanıma için SVM adlı çalışmada gaussian karışım modeli, destek vektör makineleri, k-en yakın komşuluk ve naive bayes sınıflandırıcısı kullanılmıştır. Çalışmada Kuzeydoğu Amerika eyaletleri lehçesinde 18 kadın ve 31 erkek konuşmacının her birinin kurduğu 10 cümlelik veri kümesi kullanılmıştır ve bu veri kümesinin ilk 8 cümlesi öğrenme verisi son 2 cümlesi ise sınama verisi olarak kullanılmıştır. Destek vektör makineleri , k- en yakın komşuluk ve naive bayes yaklaşımlarıyla sınıflandırma işlemi gerçekleştirilmiş ve en iyi sonuç K-en yakın komşuluk algoritmasında %37'lik bir başarı oranı ile bulunmuştur. Bir sonraki aşamada ise bu sınıflandırıcılarla birlikte gaussian karışım modelini birleştirerek hibrit modeller oluşturulmuş ve %96 başarı oranı ile destek vektör makinelerinde elde edilen hibrit model en iyi sonucu vermiştir. Ayrıca tüm hibrit modeller birleştirildiğinde başarı oranını %100 olarak bulunmuştur[67].

EEG İşaretlerinin Sınıflandırma ve Kümeleme Teknikleri İle Analizi adlı çalışmada 100 adet normal 100 adet epilepsi hastası verisi EEG(elektroensefalogram) ile elde edilmiştir. Bu verilerde epilepsi hastası olan ya da epilepsi hastası olmayan sağlıklı insanların sınıflandırılması hedeflenmektedir.

Özellik çıkarımında ortalama mutlak değer özbağlanımlı kovaryans yöntemi, yule-walker AR yöntemi, kümeleme işlemini gerçekleştirmek için K Ortalama, Bulanık C Ortalama algoritmaları, sınıflandırma için lineer diskriminant analiz ve destek vektör makineleri kullanılmıştır.

[66] (Türkmen, 2011)
[67] (Nahayo, 2016)

Destek vektör makineleri ve lineer diskriminant analizi ile en yüksek başarı %99.5 olarak elde edilmiştir[68].

Medline veri tabanı üzerinde bulunan tıbbi dokümanların kanser türlerine göre otomatik sınıflandırılması adlı çalışmada 5600 adet tıbbi dergi ve 21 milyonundan fazla makale barındıran Medline veri tabanında sorgu gerçekleştiren ve tıp araştırmacıları tarafından kullanılan Pubmed arama motoru üzerinden bir crawler yardımı ile kanser türleri hakkında yazılan 25962 adet makale özeti verisi elde edilmiş. Bu veri üzerinden öğrenme ve sınama verileri oluşturularak kanser ile ilişkili olan tıbbi dokümanının hangi kanser türü ile ilişkili olduğunun bulunması hedeflenmiştir.

Çalışma kendi içinde iki farklı şekilde gerçekleştirilmiştir. Dokümanların sınıflandırılması için özellik seçme aşamasında hem ki-kare hem de bilgi kazancıyla iki farklı özellik çıkarımı yapılmıştır. Çalışmada destek vektör makineleri ve naive bayes algoritmaları kullanılarak bu özellik vektörlerinin başarısına değinilmiştir. Ayrıca elde edilen makale özetlerinden anahtar kelimeler çıkarılarak (kanser türü ile en fazla ilişkili olan kelimeler Örneğin, Tiroit Kanseri - thyroid) ilgili yöntemler tekrar sınanmıştır. Çalışmanın sonucunda en yüksek başarı destek vektör makinesiyle polinomal çekirdek fonksiyonu kullanılarak elde edilmiştir[69].

Yazar tanımada köşe yazısı ve tweetlerin çapraz kullanımı adlı çalışmada Türkiye'de en çok takip edilen Twitter kullanıcıları arasında yer alan 10 köşe yazarının 200'er tweeti ve 50'şer makalesi kullanılmıştır. Twitter metininden makalelerin yazarlarının tahmini ve makalelerden de twitter metinlerini gönderen yazarların tahmini yapılmıştır. Bunun için tweet metinler hem tek bir metin olarak hem de gruplanarak ele alınıp farklı çalışmalar gerçekleştirilmiştir.

Çalışmada naive bayes, destek vektör makineleri, karar ağaçları (J48), bagging sınıflandırıcıları kullanılmış ve en iyi sonuçlar destek vektör makineleri algoritması ile elde edilmiştir.

[68] (Tekin, 2013)
[69] (Haltaş, 2016)

Kelimeler, kökler ve 3-gramlar ile sınıflandırma işlemi gerçekleştirilen çalışmada en iyi sonuç grup haline getirilen twitter metinleri kullanılarak10 kez çapraz geçerlilikle sınanan destek vektör makineleri algoritmasıyla %100'lük bir başarı elde edilmiştir[70].

Application of support vector machine algorithm in e-mail spam filtering adlı çalışmada halka açık olarak sunulan SpamAssassin veri kümesi kullanılarak gerçekleştirilmiştir. Özellik vektörü olarak TF-IDF kullanılmıştır. Negatifleri ve pozitifleri barındıran veri kümesi ayarlanmamıştır, pozitiflerin negatiflere oranı 2/3 olarak gözlenmektedir. Verilerin %20'si eğitim ve %80'i sınama verisi olarak kullanılmış, destek vektör makinesi , Naive Bayes, ZeroR algoritmalarıyla 5 kat geçerli çapraz doğrulama kullanılarak sınıflandırma işlemi gerçekleştirilmiştir.

En iyi sınıflandırma başarısı destek vektör makineleriyle c (yanlış sınıflandırma maliyeti) 1 olarak seçildiğinde %98.59 olarak elde edilmiştir[71].

1.5 MODEL PERFORMANS DEĞERLENDİRME ÖLÇÜTLERİ

Veri madenciliği alanında kullanılan makine öğrenmesi algoritmalarının değerlendirilmesinde ve birbirleri ile karşılaştırılmasında ihtiyaç duyduğumuz ölçütler model performans değerlendirme ölçütleridir. Sınıflandırma problemine konu olan bir veri kümesinde makine öğrenmesi algoritmaları kullanılarak verinin hangi sınıfa ait olduğuna dair tahminde bulunulur. Bu tahmin değerleri ve mevcut olan gözlem değerleri kullanılarak hata matrisi oluşturulur ve bu matris değerlendirilerek modelin başarısı hesaplanır.

Model performans değerlendirme ölçülerinin formülleri verilmeden önce formüllerde kullanılan sembolleri şu şekilde açıklayabiliriz; "p" ifadesi pozitifleri temsil ederken "n" ifadesi negatifleri temsil etmektedir. Yanlış tahmin "y" doğru tahmin "d" ile ifade edilmektedir. Pozitiflerin doğru tahmin edilmesi "dp" negatiflerin doğru tahmin edilmesi "dn" ile ifade edilir. Pozitiflerin yanlış tahmin edilmesi "yn" negatiflerin yanlış tahmin edilmesi "yp" ile temsil edilir. Toplam veri adedi (doğru tahmin edilen negatifler (dn) + doğru tahmin edilen pozitifler (dp) + yanlış tahmin edilen pozitifler(yn) + yanlış

[70] (Mayda, 2016)
[71] (Bluszcz, 2016)

tahmin edilen negatifler(yp)) = m ile temsil edilir. Aşağıda verilen formüllerde (Balaban E.M., 2015) çalışmasından yararlanılmıştır. Tablo 2'de örnek bir sınıflandırma tablosu üzerinden temsili olarak sembollerinin kullanımı gösterilmektedir.

Tablo 2 : Örnek Sınıflandırma Tablosu

Gerçek / Öngörü	**Pozitif**	**Negatif**	
Pozitif	DP	FP	Toplam Pozitif Öngörülenler
Negatif	YN	DN	Toplam Negatif Öngörülenler
	Toplam Pozitif Gerçekler	Toplam Negatif Gerçekler	Tüm Veri Adedi

Doğruluk; doğru olarak tahmin edilen verilerin tüm verilere oranıdır. Tüm verilerin doğru olarak sınıflandırılması sonucunda modelin doğruluğu maksimum (1) değerini alır. Hiçbir veri doğru olarak sınıflandırılmazsa modelin doğruluğu minimum (0) değerini alır.

$$\text{Doğruluk (ACC)} = \frac{\text{dp} + \text{dn}}{\text{m}} \qquad (12.1)$$

Hata oranı; yanlış olarak tahmin edilen değerlerin tüm verilere oranıdır. Tüm verilerin yanlış olarak sınıflandırılması durumunda modelin hata oranı maksimum (1) değerini alır. Hiçbir veri yanlış olarak sınıflandırılmazsa modelin hata oranı minimum (0) değerini alır.

$$\text{Hata Oranı (ERR)} = 1 - \text{ACC} \qquad (12.2)$$

Duyarlılık; modelin pozitif sınıfı doğru olarak sınıflandırma başarısıdır. Doğru sınıflandırılan pozitiflerin tüm pozitiflere olan oranıdır. Eğer pozitiflerin tamamı doğru olarak sınıflandırılırsa modelin duyarlılığı maksimum (1) değerini alacaktır. Hiçbir pozitif değer doğru sınıflandırılmazsa modelin duyarlılığı minimum (0) değerini alacaktır.

$$\text{Duyarlılık, Hassasiyet (TPR)} = \frac{\text{dp}}{\text{dp} + \text{yn}} \qquad (12.3)$$

Belirleyicilik; modelin negatifleri doğru olarak sınıflandırma başarısıdır. Doğru sınıflandırılan negatiflerin tüm negatiflere oranıdır. Eğer tüm negatifler doğru olarak sınıflandırılırsa modelin belirleyiciliği maksimum (1) değerini alacaktır. Hiçbir negatif doğru olarak sınıflandırılmazsa modelin belirleyiciliği minimum (0) değerini alacaktır.

$$\text{Belirleyicilik (SPC)} = (\text{TNR}) = \frac{\text{dn}}{\text{dn} + \text{yp}} \qquad (12.4)$$

Yanlış pozitif oranı; modelin pozitif olarak sınıflandırdığı ancak gerçekte negatif sınıfta bulunan verilerin tüm negatiflere oranıdır. Tüm negatiflerin hatalı sınıflandırılması durumunda modelin yanlış pozitif oranı maksimum (1) değerini alacaktır.

Eğer hiçbir negatif yanlış olarak sınıflandırılmazsa modelin yanlış pozitif oranı minimum (0) değerini alacaktır.

$$\text{Yanlış pozitif oranı (FPR)} = 1 - \text{SPC} = \frac{\text{yp}}{\text{yp} + \text{dn}} \qquad (12.5)$$

Yanlış negatif oranı; modelin negatif olarak sınıflandırdığı ancak gerçekte pozitif sınıfta bulunan verilerin tüm pozitif verilere oranıdır. Tüm pozitiflerin hatalı sınıflandırılması durumunda modelin yanlış negatif oranı maksimum (1) değerini alacaktır. Eğer hiçbir pozitif yanlış olarak sınıflandırılmazsa modelin yanlış negatif oranı minimum (0) değerini alacaktır.

$$\text{Yanlış negatif oranı (FNR)} = 1 - \text{TPR} = \frac{\text{yn}}{\text{yn} + \text{dp}} \qquad (12.6)$$

Kesinlik (Pozitif öngörü değeri); model tarafından doğru sınıflandırılan pozitif örneklerin tüm pozitif olarak sınıflandırılan örneklere oranıdır. Bu ölçütte maksimum kesinliğe ulaşmak için tüm pozitif örneklerin doğru olarak sınıflandırılması ve hiçbir negatifin hatalı sınıflandırılmaması gerekmektedir.

$$\text{Kesinlik} - \text{ Pozitif öngörü değeri (PPV)} = \frac{\text{dp}}{\text{dp} + \text{yp}} \qquad (12.7)$$

Negatif öngörü değeri; Model tarafından doğru sınıflandırılan negatiflerin tüm negatif olarak sınıflandırılan örneklere oranıdır. Bu ölçütte maksimum değere ulaşmak için tüm negatif örneklerin doğru olarak sınıflandırılması ve hiçbir negatifin hatalı sınıflandırılmaması gerekmektedir.

$$\text{Negatif öngörü değeri (NPV)} = \frac{\text{dn}}{\text{dn} + \text{yn}} \qquad (12.8)$$

F Ölçüsü; Kesinlik ve duyarlılık değerlerinin ölçülerinin harmonik ortalamasıdır ve her iki ölçütü birlikte ele alma imkanı vermektedir.

$$\text{F} - \text{ölçüsü (NPV)} = \frac{2 * \text{PPV} * \text{TPR}}{\text{PPV} + \text{TPR}} \qquad (12.9\ )$$

2. TÜRKÇE İSTENMEYEN E-POSTALARIN ENGELLEME YÖNTEMLERİ ÜZERİNE BİR UYGULAMA

Bu bölümde istenmeyen e-postaları sınıflandırmadaki amacımıza değinilip daha sonra uygulamada kullandığımız veri kümesi tanıtılmıştır. Gerçekleştirmiş olduğumuz uygulamada kullanmış olduğumuz yöntemlere değinildikten sonra elde edilen bulgular paylaşılmış ve sonuçları tartışılmıştır.

2.1 AMAÇ VE KAPSAM

İstenmeyen e-postalar konusu ele alındığında sınırlı kaynak olarak karşımıza "zaman" kavramı çıkmaktadır. Gün içerisinde birçok e-posta alışverişi gerçekleştiren bir işletmede çalışanların hesaplarına gelen postaların cevaplanması bir zaman maliyetine neden olmaktadır.

Sahami ve arkadaşlarının çalışmasında kullanıcıların hesaplarına gönderilen istenmeyen e-postaların maddi zararları olduğundan bahsedilmektedir. Çalışmada bu tür e-postaların yalnızca zaman kayıplarına neden olmayıp binlerce kullanıcının bulunduğu e-posta sunucusunda depolama alanının hızlı bir şekilde tükenmesine neden olacağından da bahsedilmiştir [72].

Teknolojinin veri depolama alanında gerçekleştirmiş olduğu büyük gelişmeler sayesinde daha az bir maliyetle veri depolamanın mümkün olduğu bilinmektedir[73]. Ancak istenmeyen e-postaların göz ardı edilerek e-posta sunucusunda yer işgal etmeleri depolama maliyetini beraberinde getirecektir.

Bir işletmenin e-posta hesabına gönderilen ve içerisinde fidye virüsü barındıran istenmeyen e-postanın işletme çalışanlarından biri tarafından açılarak virüsün aktif hale getirildiği varsayımında bulunursak; bu durum işletmelerde dosya kaybı, para ve zaman kaybı gibi problemlere neden olacaktır.

[72] (Sahami, 1998)
[73] (matt komorowski - a history of storage cost, 2014)

Bu çerçevede istenmeyen e-postaların engellenmesi ile zaman kayıplarının önüne geçmek, işletmelerde verimliliği ve kârı arttırmak, veri depolama maliyetini azaltmak, virüsler tarafından gerçekleşecek olan veri kayıplarının önüne geçmek hedeflenmektedir.

Ayrıca veri madenciliği alanında sıklıkla kullan sınıflandırma algoritmalarından olan destek vektör makinesi (rbf, lineer) ve naive bayes (multinomial, bernoulli) algoritmalarını karşılaştırmak hedeflenmiştir. Gerçekleştirdiğimiz bu çalışmanın metin madenciliği alanında doküman sınıflandırma problemini ele alan araştırmacılar için yol göstermesi amaçlanmıştır.

2.2 VERİ KÜMESİ VE ÖZELLİKLERİ

Bu tez çalışmasında veri kümesi olarak S. Ergin ve diğerlerinin 2013 yılında yayınlanan makalesindeki verilerden yararlanılmıştır. İçerisinde 300 adet pozitif ve 300 adet negatiften oluşan 600 adet öğrenme verisi, 100 adet pozitif ve 100 adet negatiften oluşan sınama verisi barındıran bu veri kümesi toplamda 800 adet veriden oluşmaktadır[74]. Çalışmada kullanılan veriler önceden hazırlandığından dolayı ayrıca bir veri hazırlama (öğrenme ve sınama verisini bölme) işlemi gerçekleştirilmemiştir.

2.3 YÖNTEM VE UYGULAMA

Sahami "*A Bayesian Apporach to Filtering Junk E-Mail"* adlı çalışmasında destek vektör makinesi algoritmalarının naive bayes sınıflandırıcısına göre daha iyi bir başarı sağlayacağını söylemektedir. Ion Androutsopoulos ve arkadaşlarının yapmış olduğu "*Spam Filtering with Naive Bayes – Which Naive Bayes*" adlı çalışmadaysa hem negatifler hem de pozitifler için multinomial naive bayes sınıflandırıcısıyla en yüksek sınıflandırma başarısı elde edilmiştir, aynı çalışmada bernoulli naive bayes algoritması negatifleri sınıflandırmada en iyi üçüncü algoritma olma başarısını gösterirken pozitifleri sınıflandırmada başarısı en düşük algoritma olmuştur. Drucker ve arkadaşlarının "Support vector machines for spam categorization*"* adlı çalışmasında ise etkisiz kelimelerin öğrenme kümesinden arındırılması ve arındırılmaması durumu incelenmiştir. Elde edilen sınıflandırma sonuçlarından etkisiz kelimelerin öğrenme kümesinden

[74] (S. Ergin, 2011)

arındırılmaması önerilmektedir. Yine Androutsopoulos ve arkadaşlarının "*An Evaluation of Naive Bayesian Anti-Spam Filtering"* adlı çalışmasında farklı boyutlardaki özellik vektörleri kullanılarak sınıflandırma işlemi gerçekleştirilmiştir.

Oğuz Kaynar ve arkadaşlarının Ergin ve diğerleri tarafından[74] yayınlamış olduğu veri kümesi kullanarak gerçekleştirdikleri "Oto Kodlayıcı Tabanlı Derin Öğrenme Makinaları ile Spam Tespiti"[75] adlı çalışmalarında doğruluk değeri: %98 belirleyicilik: %97, hassasiyet: %98,9 olarak bulunmuştur.

Tüm bu çalışmalar göz önüne alındığında destek vektör makineleri algoritmaları (lineer, rbf), multinomial naive bayes algoritması ve bernoulli naive bayes algoritması kullanarak eğitim kümesinin etkisiz kelimelerden arındırılması ve arındırılmaması, özellik vektörünün farklı eleman sayısına sahip olması (100, 200, 300, 400, 500, 1000, 5000, 10000) ve özellik vektöründe bulunan ifadelerin en az iki dokümanda bulunması koşulu altında istenen ve istenmeyen e-postalar için "***64****"* farklı model oluşturulmuştur. Ayrıca özellik vektörü kısıtlanmadan etkisiz kelimelerin arındırılması ve arındırılmaması durumlarında "***8***" farklı model oluşturulmuştur. Toplamda "***72***" farklı model ile uygulama gerçekleştirilmiştir. Oluşturulan modeller "Model Performans Değerlendirme Ölçütleri" adlı başlık altında verilen formüller yardımıyla analiz edilmiştir. Çalışmamızda yapay sinir ağları ve derin öğrenme algoritmaları gibi sezgisel optimizasyonun çok yoğun olarak kullanıldığı sınıflandırma algoritmaları kullanılmamıştır.

Kullanılan sınıflandırma algoritmaları ve metin ön işleme işlemlerinin gerçekleştirilebilmesi için Intel(R) Core (TM) i5-4210U 1.70 GHz işlemcili, 8 gb RAM donanım özellikli, işletim sistemi olarak Linux - Ubuntu'nun 16.04.2 LTS sürümü mevcut olan bir bilgisayar kullanılmıştır. Programlama aşamaları ise Python dilinin 2.7 sürümü kullanılarak gerçekleştirilmiştir.

[75] (Oguz Kaynar, 2016)

Sınıflandırma işlemlerini gerçekleştirmek için python programlama dilinde hazır bir kütüphane olarak sunulan ve veri madenciliği algoritmalarını içerisinde barındıran scikit-learn[76] kütüphanesinden yararlanılmıştır.

Kullanılan veri kümesi metin dosyalarından oluşmaktadır (*.txt). Dosyaların analize hazır hale getirilebilmesi için metin ön işleme aşamaları gerçekleştirilmiştir. Öğrenme veri kümesinde bulunan metinlerin tüm kelimeleri küçük harfe dönüştürülmüş, html etiketleri, noktalama işaretleri, içerisinde bulunan e-posta hesapları, konu, gönderen ve alıcı bilgilerinden arındırılmıştır. Sınama veri kümesindeyse sadece kelimeler küçük harfe dönüştürülmüştür.

Veriyi ön işleme aşaması tamamlandıktan sonra öğrenme veri kümesinde hem pozitif hem de negatif e-postalarda karşılaştığımız Türkçe etkisiz kelimeler çıkarılarak ayrıca bir öğrenme veri kümesi elde edilmiştir.

Bunun için 2008 yılında Fazlı Can ve arkadaşları tarafından yayınlanan "Information Retrieval on Turkish Texts" adlı makalede Türkçe dili için bir tablo olarak sunulan 147 adet kelime içeren etkisiz kelimeler listeden yararlanılmıştır[77].

Fazlı Can ve arkadaşlarının çalışmasından elde edilen bu listenin 2014 yılında Kevin Bougé tarafından genişletilmiş hali mevcuttur. Ayrıca Google tarafından etkisiz kelimelerin paylaşıldığı bir web sayfası da mevcuttur[78].Google tarafından proje geliştiricileri için sunulan dosya indirme platformunun 2014 yılında hizmetini durdurduğundan dolayı Kevin Bougé'nin çalışmasına farklı web sayfaları aracılığı ile erişmek mümkündür. Kevin Bougé'nin listesinde 209 adet etkisiz kelime mevcuttur[79].

[76] (Pedregosa, 2011)
[77] (Can, 2008)
[78] (Google)
[79] (Bougé, 2015).

Tablo 3 : Türkçe Etkisiz Kelimeler Tablosu

ama, ancak, arada, ayrıca , bana , bazı, belki, ben, beni, benim, beri, bile, bir, birçok, biri, birkaç, biz, bize , bizi, bizim, böyle , böylece, bu , buna , bundan , bunlar, bunları, bunları, bunu, bunun, burada, çok, çünkü, da , daha, de, değil, diğer, diye, dolayı, dolayısıyla, edecek, eden , ederek , edilecek, ediliyor , edilmesi, eğer , etmesi, etti, ettiği, ettiğini, gibi , göre, halen, hangi, hatta, hem, henüz, her , herhangi, herkesin , hiç , hiçbir , için , ile, ilgili , ise , işte, itibaren, itibariyle , kadar, karşın, kendi, kendilerine, kendini, kendisi, kendisine, kendisini, ki , kim , kimse , mi , mı, mü, mu , ediyor , nasıl , üzere, olur, ne , neden, nedenle, o , olan , olarak, oldu, olduğu, olduğunu, olduklarını, olmadı, olmadığı, olmak , olması, olmayan , olmaz , olsa , olsun , olup, olursa , oluyor, ona, onlar, onları, onların, onu, onun, öyle , oysa , pek, rağmen, sadece , şey, siz , şöyle , şu, şunları, tarafından, var , vardı, ve , veya , ya , yani , yapacak, yapılan , yapılması, yapıyor , yapmak , yaptı, yaptığı, yaptığını, yaptıkları, yerine , yine , yoksa, zaten

Kaynak: Can, F. K. (2008). Information retrieval on Turkish texts. Journal of the American Society for Information Science and Technology , 59 (4), s. 407-421.

Çalışmada kullanılan makine öğrenmesi algoritmalarının mevcut parametrelerinde hiçbir değişiklik yapılmadan (varsayılan parametre değerleriyle) sınıflandırma işlemi gerçekleştirilmiştir.

Naive Bayes yaklaşımlarında kullanılan iki (Bernoulli ve Multinomial) yaklaşımın varsayılan parametreleri aşağıda sunulmuştur. *Bernoulli Naive Bayes* algoritması kullanılırken varsayılan parametreler; (alpha=1.0, binarize=0.0, fit_prior=True, class_prior=None) [80] *Multinomial Naive Bayes* algoritması kullanılırken varsayılan parametreler; (alpha=1.0, fit_prior=True, class_prior=None) [81]

Alpha parametresi : Laplace yumuşatmasının derecesini belirten parametredir. Varsayılan olarak "1" değeri tanımlanmıştır. Herhangi bir yumuşatma işleminin yapılmaması istenildiğinde bu değer "0" olarak tanımlanmalıdır.

[80] (scikit-learn.org, sklearn.naive_bayes.BernoulliNB - scikit-learn 0.18.1 documentation, 2017)
[81] (scikit-learn.org, sklearn.naive_bayes.MultinomialNB - scikit-learn 0.18.1 documentation)

Fit prior parametresi : Sınıfın öncelikle olasılık değerinin olup olmadığını belirtmek için kullanılır, Mantıksal değerler alır (True, False). Aynı zamanda bu parametre ile öğrenme kümesinin dağlımı da belirtilmektedir. Eğer False olarak belirlenirse dağılımın düzgün dağılım olduğu kabul edilecektir. Bu parametrenin varsayılan değeri True olarak belirtilmiştir.

Class prior parametresi : Sınıfa ait öncelikli olasılık değerini belirten parametredir. Öncelikli olasılığın olmadığı durum varsayılan olarak tanımlanmıştır. Bu parametrenin değeri dizi ile tanımlanır. Dizinin boyutu sınıf adedi kadar olmak zorundadır.

Yukarıdaki parametreler bernoulli naive bayes ve Multinomial naive bayes algoritmaları için ortak kullanılan parametrelerdir. Binarize parametresi yalnızca bernoulli naive bayes algoritması tarafından kullanılmaktadır.

Binarize parametresi : Özellik vektörünü ikili özellik vektörüne dönüştürmek için gerekli olan eşik değerini temsil eder. Kullanılan verinin ikili özellik vektörü ile tanımlandığı varsayımında bulunularak "0.0" değeri varsayılan olarak tanımlanmıştır. Float sayı türünde değer alır.

Lineer Destek Vektör Makineleri algoritmasında kullanılan varsayılan parametrelerin açıklamaları aşağıda sunulmuştur.

Destek Vektör Makineleri (Lineer) algoritmaları kullanılırken varsayılan parametreler; (penalty='l2',loss='squared_hinge',dual=True,tol=0.0001, C=1.0, multi_class='ovr', fit_intercept=True, intercept_scaling=1, class_weight=None, verbose=0, random_state=None, max_iter=1000) [82]

Penalty parametresi : Hatalı sınıflandırmada uygulanacak olan cezayı belirtir. L1 düzeltmesi ve L2 düzeltmesi olarak iki farklı kullanımı mevcuttur. Varsayılan olarak L2 düzeltmesi tanımlanmıştır.

[82] (scikit-learn.org, sklearn.svm.LinearSVC — scikit-learn 0.18.1 documentation)

L1 düzeltmesi: Gerçek değer ile tahmin edilen değer arasındaki tüm mutlak farkların toplamını minimize etmek için kullanılır[83].

$$\text{L1 (Loss fonsiyonu)} = \sum_{i=1}^{n} |y_{doğru} - y_{tahmin}| \qquad (13.1)$$

Kaynak: https://www.letslearnai.com/2018/03/10/what-are-l1-and-l2-loss-functions.html (What Are L1 and L2 Loss Functions?)

L2 düzeltmesi gerçek değer ile tahmin edilen değer arasındaki tüm farkların karesinin toplamını minimize etmek için kullanılır[83].

$$\text{L2(Loss fonsiyonu)} = \sum_{i=1}^{n} (y_{doğru} - y_{tahmin})^2 \qquad (13.2)$$

Kaynak: https://www.letslearnai.com/2018/03/10/what-are-l1-and-l2-loss-functions.html (What Are L1 and L2 Loss Functions ?)

Bu parametrenin varsayılan değeri L2 olarak tanımlanmıştır.

Loss parametresi: Destek vektör makinelerinde maksimum aralığı belirlemek için loss parametresi kullanılmaktadır. Loss parametresi ise varsayılan olarak Hinge loss fonksiyonun karesini alınarak oluşturulmuş olan 'square hinge' fonksiyonunu kullanmaktadır. Bu parametreyi anlamak için öncelikle Hinge loss fonksiyona değinmemiz gerekmektedir. Hinge fonksiyonu destek vektör makinelerinin standartlaşmış hata fonksiyonudur. Hinge fonksiyonundaki t değeri ($t = \pm 1$) istenilen etiketin değerine, y ise sınıflandırma skoruna karşılık gelmektedir. Lineer destek vektör makinelerinde $y = w.x + b$ olduğu bilinmektedir. W, b hiperdüzlem parametreleridir. X ise sınıflandırılacak noktayı temsil etmektedir. Eğer t ve y aynı işaretli ise doğru tahmin gerçekleştirilmiştir. Eğer $|y| \geq 1$ ise hinge loss "0" olur. T ve y değerleri zıt işaretli olduklarında hata y ile doğrusal olarak artar[84,85] .

[83] (Shekhar, 2018)
[84] (Duan, 2005)
[85] (Rosasco, 2014)

$$\text{Hinge Loss Fonskiyonu} = \text{l(y)} = \max\,(0\,, 1 - \ max_{t!=y}(w_t \text{x} - w_t \text{y})) \quad (14)$$

Dual parametresi: Bu parametre konveks optimizasyon problemlerinde karşımıza çıkmaktadır. Herhangi bir optimizasyon probleminde ilkel değişkenleri kullanarak çözüme ait alt sınırı belirlemeyi sağlayan yöntemdir. True veya False değerlerini alır. Varsayılan olarak True değeri tanımlanmıştır. Örnek sayısının özellik sayısından büyük olduğu durumlarda False olarak tanımlanması önerilmektedir [86,87].

Tol parametresi : İterasyonu durdurmak için tanımlanan tolerans parametresidir. Varsayılan olarak 10^-4 değeri tanımlanmıştır. Eğer hata değeri bu değerin altında olursa iterasyon durdurulur. Aksi durumda iterasyon işlemi max iter parametresinde tanımlanan ifade kadar devam eder[87].

C parametresi : Eğitim verileri üzerinde yanlış sınıflandırma maliyetidir (cost). Yüksek bir C değeri düşük bir bias (yanlılık) ve yüksek variyete(değişkenlik) anlamına gelmektedir. Yanlış sınıflandırma maliyeti çok fazla olduğundan dolayı bias düşük olacaktır. Yüksek C değerinde yanlış sınıflandırma maliyeti yüksektir. Böylece birbirine yakın olan verilerde veriyi ezberleme durumu söz konusu olabilir. Ufak C değeri yüksek bias ve düşük değişkenlik anlamına gelir. Ufak C değeri yanlış sınıflandırma maliyetini düşürür. Böylece yanlış sınıflandırma daha geniş bir alana yayılabilir. C değeri ne çok keskin ne de çok serbest bir değer olarak tanımlanmalıdır. Çapraz geçerlilik, yeniden ölçeklendirme gibi yöntemlerle en iyi C değerini bulmak için çalışmalar gerçekleştirilebilir. C parametresinin varsayılan değeri "1.0" olarak tanımlanmıştır[87,88].

Multi class parametresi : Eğer sınıflandırma problemimiz ikiden fazla sınıf içeriyorsa bu parametre ovr " one vs rest " olarak tanımlanır. Tüm sınıflar üzerinden ortak bir hedefi optimize etmek isteyen problemler için multi class parametresi "crammer single" olarak tanımlanabilir. Crammer single yöntemi hesaplanması daha maliyetli bir yöntemdir. Bundan dolayı pratikte kullanımı nadir olarak gözlemlenmektedir. Eğer "crammer

[86] (Stephen Boyd, 2004)
[87] (scikit-learn.org, sklearn.svm.LinearSVC — scikit-learn 0.18.1 documentation, 2016)
[88] (listendata.com, 2017)

single" yöntemi kullanılırsa loss ve penalty parametreleri göz ardı edilir. Bu parametrenin varsayılan değeri "ovr" olarak tamamlanmıştır[87].

Fit intercept parametresi : Doğrusal olarak gerçekleştirilen sınıflandırma işleminde Y eksininin kesildiği noktaları hesaplamak için kullanılmaktadır. İsteğe bağlı bir parametredir. Eğer "False" olarak tanımlanırsa herhangi kesim noktasını hesaplamaz ve tüm verinin ortalanmış olarak verildiği varsayımında bulunur[87,89,90].

Intercept scaling parametresi : Fit intercept parametresi "true" olarak tanımlandığı zaman karşımıza çıkan bir parametredir. Düzeltme işlemlerinin (L1,L2) gerçekleşmesinden kaynaklanan etkilerin azaltılması için kullanılmaktadır[87].

Random state parametresi : Veriyi karıştırmak için kullanılan rasgele sayıyı üreten parametredir. Instance ya da None değerlerini alabilir. Eğer instance değerini alırsa "RandomState" sayı üreticiden yararlanılır. Instance değeri ile " seed " ifadesi de etkin olur. Bu sayede öğrenme işlemi tekrarlandığında aynı rasgele değerler üreterek aynı verileri kullanılacaktır. Böylece sonuçlar herkes tarafından aynı şekilde bulunabilecektir. Eğer none olarak belirlenirse "np.random" fonksiyonunu kullanarak rasgele sayı üretir. Seed ifadesi aktif olmadığı için öğrenmek için kullanılan veriler her seferinde değişecektir. Bu parametrenin varsayılan değeri "None" olarak tanımlanmıştır[87].

Class weight : Öğrenme aşamasında sınıfların ağırlıklarının belirlendiği parametredir. Eğer bir sınıflandırma ağırlığı tanımlanmışsa ağırlık değerinin belirlenen C parametresi ile çarpılması sonucu elde edilen değerlerle öğrenme işlemi gerçekleştirilir. Eğer class weight parametresi "None" olarak tanımlanırsa bu durum tüm verilerin dengelenmiş (balanced) olduğu anlamına gelir. Böylece tüm sınıfların ağırlığı "1" olarak kabul edilecektir. Class weight parametresinin varsayılan değeri "None" olarak tanımlanmıştır[87].

[89] (Press, 2009)
[90] (Bayramlı, 2015)

Verbose parametresi: Öğrenme işlemi ile ilgili ayrıntılı çıktı almak için kullanılan parametredir. Varsayılan olarak "0 = False" değeri tanımlanmıştır. (liblinear ile ilgili bilgiler verir)[87]. Sıfır (False) veya bir (True) değeri alabilir.

Max iter parametresi : Maksimum iterasyon sayısıdır varsayılan olarak 1000 iterasyon tanımlanmıştır[86].

RBF Destek Vektör Makineleri algoritmasında kullanılan varsayılan parametrelerin açıklamaları aşağıda sunulmuştur.

Destek Vektör Makineleri (RBF) algoritmaları kullanılırken varsayılan parametreler; (*C*=1.0, kernel='rbf', degree=3, gamma='auto', coef0=0.0, shrinking=True, probability=False, tol=0.001, cache_size=200, *class_weight*=None, *verbose*=False, max_iter=-1, decision_function_shape=None, *random_state*=None) [91]

C, *class weight, random state,* parametreleri; Lineer Destek vektör makineleri parametreleri ile aynı değerlere ve açıklamalara sahiptir.

Kernel parametresi : Sınıflandırma aşamasında kullanılacak olan algoritma tipini belirlemek için kullanılan parametredir. Lineer, polinomal, rbf, sigmoid olarak tanımlanabilir. Eğer hiçbir değer belirtilmezse rbf olarak tanımlanacaktır[90].

$$K_{RBF}(x, x') = e^{-\gamma||x-x'||^2}$$

(RBF Çekirdek Fonksiyonu)

Kaynak: http://pages.cs.wisc.edu/~matthewb/pages/notes/pdf/svms/RBFKernel.pdf

$$K(u, v) = (\gamma * u'v + coef0)^d$$

(Polinom Çekirdek Fonksiyonu) [92]

Degree parametresi : Bu parametre polinomal çekirdek fonksiyonu kullanıldığı zaman geçerlidir, polinomun derecesini belirtir. Diğer çekirdek fonksiyonlarında göz ardı edilen

[91] (scikit-learn.org, sklearn.svm.SVC - scikit-learn 0.18.1 documentation)
[92] (D. Jude Hemanth, 2018)

parametredir. Degree parametresinin varsayılan değeri "3" olarak tanımlanmıştır. Metin madenciliği alanında gerçekleştirilen çalışmalarda degree parametresinin "2" olması önerilmektedir[93,94].

Gamma parametresi: Rbf, polinomal ve sigmoid çekirdek fonksiyonları için kernel katsayısıdır. Eğer varsayılan olarak "auto" değeri seçilirse $\frac{1}{\text{Özellik vektöründeki eleman sayısı}}$ ifadesi gamma parametresi olarak belirlenir. Kernel parametresi içerisinde tanımlanan polinom kernel ve rbf kernel fonksiyonlarında gamma parametresi gösterilmektedir[94].

Coef0 parametresi: Kernel fonksiyonundan bağımsız bir parametredir. Polinomal ve sigmoid kernel fonksiyonlarıyla kullanılan bir katsayıdır. Varsayılan değeri "0.0" olarak tanımlanmıştır. Tanımlanması zorunlu bir parametre değildir. 1 ve 0 değerleri arasında değerler almaktadır. Polinomal kernel kullanılarak oluşturulan modellerin düşük dereceli polinomlara karşı yüksek dereceli polinomlardan ne kadar etkilendiğini kontrol eder[94,95].

Tol parametresi : İterasyonu durdurmak için tanımlanan tolerans parametresidir. Varsayılan olarak 10^-3 değeri tanımlanmıştır. Eğer hata değeri bu değerin altında olursa iterasyon durdurulur. Aksi durumda iterasyon işlemi max iter parametresinde tanımlanan ifade kadar devam eder[94].

Verbose parametresi: Öğrenme işlemi ile ilgili ayrıntılı çıktı almak için kullanılan parametredir. Varsayılan olarak "False" değeri tanımlanmıştır. (libsvm ile ilgili bilgiler verir)[94].

Max iter parametresi : Maksimum iterasyon sayısının belirtildiği parametredir. Varsayılan olarak "-1" değeri tanımlanmıştır. "-1" ifadesi herhangi bir limitin bulunmadığı anlamına gelmektedir[94].

Cache size parametresi: Büyük problemlerin çözümünde yüksek miktarda RAM'e ihtiyaç duyulmaktadır. Bu değer problemin çözümü için algoritmanın çalışma süresinin

[93] (Goldberg, 2008)
[94] (scikit-learn.org, sklearn.svm.SVC - scikit-learn 0.18.1 documentation, 2016)
[95] (Géron, 2017)

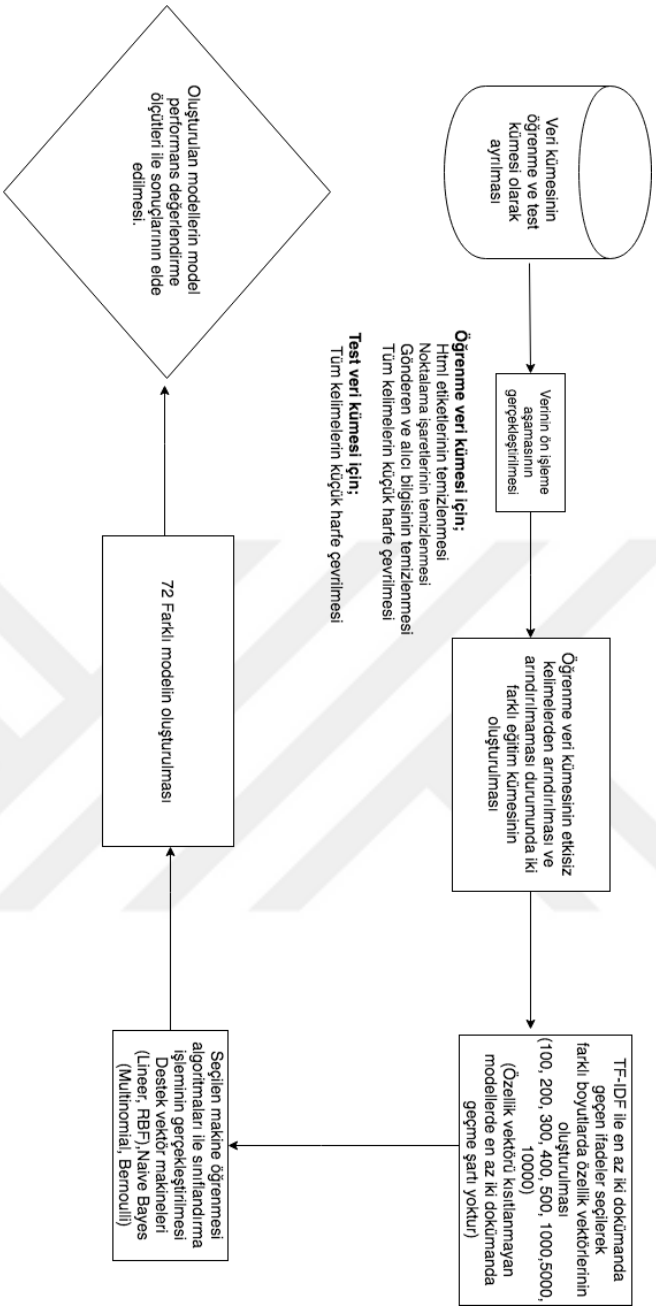
belirlemesinde oldukça kuvvetli bir etkiye sahiptir. Varsayılan olarak "200" MB değeri tanımlanmıştır. Eğer RAM miktarınız fazlaysa bu değeri 500 ya da 1000 MB olarak ayarlamanız önerilmektedir[94].

Probablity parametresi: Çok sınıflı (2 sınıftan daha fazla sınıfı) sınıflandırma işlemlerinde destek vektör makineleri ikili karşılaştırma prensibinden yararlanarak (her bir sınıfı ayrı ayrı ikili olarak karşılaştırarak) sınıflandırma işlemini gerçekleştirir. Bu durum sınıflandırma maliyetini beraberinde getirecektir. Probality parametresi sayesinde tüm ikili karşılaştırmaları birleştirerek çok sınıflı bir olasılık tahmini gerçekleştirmek hedeflenmektedir. Elde edilen sonuçların birleştirilmesi ve olasılıklar sonucunda sınıfın tahmininin gerçekleştirilmesi için bu parametrenin "True" olarak belirlenmesi gerekmektedir. Aksi durumda bu tür problemlerin çözümünde maliyet oldukça yüksek olacaktır. Bu parametrenin varsayılan değeri genellikle iki sınıflı sınıflandırma işlemleri gerçekleştirildiği için "False" olarak tanımlanmıştır[94,96].

Shrinking parametresi: Destek Vektör Makinelerinde hesaplama süresini etkileyen en önemli faktörlerden birisi KKT koşullarıdır. Bu koşulların daha hızlı bir şekilde kontrol edilmesi için sezgisel optimizasyon yaklaşımından yararlanmaktadır ("shrinking heuristic"). Bu parametre yardımıyla Karush Kuhn Tucker'ın optimizasyon algoritmasının getirmiş olduğu hesaplama maliyetini azaltmak hedeflenmektedir. Parametrenin varsayılan değeri " True" olarak tanımlanmıştır. İstenilirse shrinking parametresi False olarak seçilebilir. Bu durum hesaplama süresinin uzamasına neden olacaktır[94].

Desicion function shape parametresi: Bu parametre karar fonksiyonunun şeklini belirler. OVR "one-vs-rest" biri ya da kalanı olarak karşımıza çıkan karar fonksiyonudur. Sınıflandırmaya konu olan problemin sadece iki sınıfa sahip olması durumunda kullanılmaktadır. OVO "one of one" karar fonksiyonu ise bire bir karşılaştırma işlemini gerçekleştirir. Sınıflandırma problemi ikiden fazla sınıfa sahipse bu karar fonksiyonu kullanılmaktadır. OVR değeri varsayılan olarak tanımlanmıştır[94].

[96] (Ting-Fan Wu, 2004)



Şekil 8 : Uygulamanın akış diyagramı

2.4 BULGULAR

İstenmeyen e-postaları sınıflandırırken etkisiz kelimelerin öğrenme kümesinden arındırılıp/arındırılmama durumları ve farklı boyutlardaki özellik vektörlerinin sınıflandırma başarısına etkisi 72 farklı modelle incelenmiştir. Elde edilen bulgular bu bölümde paylaşılmıştır.

Paylaşılan sonuçlarda model isimlerinde kullanılan sayıların açıklamaları sırasıyla; etkisiz kelimelerden arındırılıp arındırılmama durumunu belirlemek için 0 ve 1 ifadeleri kullanılmıştır. 0: arındırılmamış, 1: arındırılmış modeli temsil etmektedir. Kullanılan sınıflandırma algoritmasını belirlemek için (1,2,3,4) ifadeleri kullanılmıştır. 1: RBF Destek Vektör Makineleri, 2: Lineer Destek Vektör Makineleri, 3: Bernoulli Naive Bayes, 4: Multinomial Naive Bayes algoritmalarına karşılık gelmektedir. Model isimlerinin adlandırılmasında kullanılan sayılardan en sonuncusu ise özellik vektörünün boyutunu temsil etmektedir.

2.4.1 Etkisiz Kelimelerin Öğrenme Kümesinden Arındırılmadığı Modeller

Öğrenme aşamasında etkisiz kelimelerinde bulunduğu veri kümesinde tekrarlı 216.461, tekrarsız 139.086 adet kelime bulunmaktadır. Bu veri kümesi kullanılarak herhangi bir kısıtlamada bulunulmadan özellik vektörü oluşturulursa 43.013 adet kelimeyle özellik vektörünü oluşturacaktır. Eğer minimum 2 dokümanda geçen kelimeler kullanılarak özellik vektörü oluşturulursa 14.358 adet kelimeyle özellik vektörü oluşturulacaktır. Bu veri kümesi kullanılarak oluşturulan modeller ve bu modellerden elde edilen sonuçlar aşağıda paylaşılmıştır.

2.4.1.1 RBF Destek Vektör Makinesi Algoritmasıyla Sınıflandırma Sonuçları

RBF destek vektör makinesi sınıflandırma algoritması kullanılarak oluşturulan modellerden elde edilen sonuçlar Model 0.1, Model 0.1.100, Model 0.1.200, Model 0.1.300, Model 0.1.400, Model 0.1.500, Model 0.1.1000, Model 0.1.5000 ve Model 0.1.10000 başlıkları altında sunulmuştur.

Model 0.1

Eğitim aşamasında özellik vektöründe herhangi bir kısıtlama gerçekleştirilmeyen bu modelde sınıflandırma algoritması olarak rbf destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 4***) sunulmuştur.

Tablo 4 : Model 0.1 Etkisiz Kelimeler Dahil (RBF Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	53	0	53
İstenilen E-postalar	47	100	147
Toplam	100	100	**200**

Model 0.1.100

Eğitim aşamasında özellik vektörü 100 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak rbf destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 5***) sunulmuştur.

Tablo 5 : Model 0.1.100 Etkisiz Kelimeler Dahil (RBF Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	84	0	84
İstenilen E-postalar	16	100	116
Toplam	100	100	**200**

Model 0.1.200

Eğitim aşamasında özellik vektörü 200 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak rbf destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 6***) sunulmuştur.

Tablo 6 : Model 0.1.200 Etkisiz Kelimeler Dahil (RBF Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 0.1.300

Eğitim aşamasında özellik vektörü 300 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak rbf destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 7***) sunulmuştur.

Tablo 7 : Model 0.1.300 Etkisiz Kelimeler Dahil (RBF Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 0.1.400

Eğitim aşamasında özellik vektörü 400 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak rbf destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 8***) sunulmuştur.

Tablo 8 : Model 0.1.400 Etkisiz Kelimeler Dahil (RBF Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 0.1.500

Eğitim aşamasında özellik vektörü 500 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak rbf destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 9***) sunulmuştur.

Tablo 9 : Model 0.1.500 Etkisiz Kelimeler Dahil (RBF Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 0.1.1000

Eğitim aşamasında özellik vektörü 1000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak rbf destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 10***) sunulmuştur.

Tablo 10 : Model 0.1.1000 Etkisiz Kelimeler Dahil (RBF Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	79	0	79
İstenilen E-postalar	21	100	121
Toplam	100	100	**200**

Model 0.1.5000

Eğitim aşamasında özellik vektörü 5000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak rbf destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 11***) sunulmuştur.

Tablo 11 : Model 0.1.5000 Etkisiz Kelimeler Dahil (RBF Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	79	0	79
İstenilen E-postalar	21	100	121
Toplam	100	100	**200**

Model 0.1.10000

Eğitim aşamasında özellik vektörü 10000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak rbf destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 12***) sunulmuştur.

Tablo 12 : Model 0.1.10000 Etkisiz Kelimeler Dahil (RBF Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	78	0	78
İstenilen E-postalar	22	100	122
Toplam	100	100	**200**

RBF destek vektör makinesi algoritmasıyla oluşturulan modellerden elde edilen sonuçların model performans ölçütleri ile analizinin sonuçları aşağıdaki tabloda (***Tablo 13***) paylaşılmıştır.

Tablo 13 : RBF Destek Vektör Makinesi Algoritmasıyla Oluşturulan, Etkisiz Kelimelerin Öğrenmeye Dahil Olduğu Modellerin Model Performans Ölçütleri

Yöntem (DVM destek vektör makineleri)	**dp**	**yn**	**dn**	**yp**	**Doğruluk**	**Hata Oranı**	**TPR**	**SPC**	**FPR**	**FNR**	**PPV**	**NPV**	**F-ölçütü**
Model 0.1	100	0	53	47	0.77	0.23	1	0.53	0.47	0	0.68	1	0.81
Model 0.1.100	100	0	84	16	0.92	0.08	1	0.84	0.16	0	0.86	1	0.925
Model 0.1.200	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 0.1.300	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 0.1.400	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 0.1.500	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 0.1.1000	100	0	79	21	0.9	0.1	1	0.79	0.21	0	0.83	1	0.907
Model 0.1.5000	100	0	79	21	0.9	0.1	1	0.79	0.21	0	0.83	1	0.907
Model 0.1.10000	100	0	78	22	0.89	0.11	1	0.78	0.22	0	0.82	1	0.901

2.4.1.2 Lineer Destek Vektör Makinesi Algoritmasıyla Sınıflandırma Sonuçları

Lineer destek vektör makinesi sınıflandırma algoritması kullanılarak oluşturulan modellerden elde edilen sonuçlar Model 0.2, Model 0.2.100, Model 0.2.200, Model 0.2.300, Model 0.2.400, Model 0.2.500, Model 0.2.1000, Model 0.2.5000 ve Model 0.2.10000 başlıkları altında sunulmuştur.

Model 0.2

Eğitim aşamasında özellik vektöründe herhangi bir kısıtlama gerçekleştirilmeyen bu modelde sınıflandırma algoritması olarak lineer destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 14***) sunulmuştur.

Tablo 14 : Model 0.2 Etkisiz Kelimeler Dahil (Lineer Destek Vektör Makineleri)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 0.2.100

Eğitim aşamasında özellik vektörü 100 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak lineer destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 15***) sunulmuştur.

Tablo 15 : Model 0.2.100 Etkisiz Kelimeler Dahil (Lineer Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	91	24	115
İstenilen E-postalar	9	76	85
Toplam	100	100	**200**

Model 0.2.200

Eğitim aşamasında özellik vektörü 200 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak lineer destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 16***) sunulmuştur.

Tablo 16 : Model 0.2.200 Etkisiz Kelimeler Dahil (Lineer Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	97	19	116
İstenilen E-postalar	3	81	84
Toplam	100	100	**200**

Model 0.2.300

Eğitim aşamasında özellik vektörü 300 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak lineer destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 17***) sunulmuştur.

Tablo 17 : Model 0.2.300 Etkisiz Kelimeler Dahil (Lineer Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	98	16	114
İstenilen E-postalar	2	84	86
Toplam	100	100	**200**

Model 0.2.400

Eğitim aşamasında özellik vektörü 400 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak lineer destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 18***) sunulmuştur.

Tablo 18 : Model 0.2.400 Etkisiz Kelimeler Dahil (Lineer Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	93	14	107
İstenilen E-postalar	7	86	93
Toplam	100	100	**200**

Model 0.2.500

Eğitim aşamasında özellik vektörü 500 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak lineer destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 19***) sunulmuştur.

Tablo 19 : Model 0.2.500 Etkisiz Kelimeler Dahil (Lineer Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	87	6	93
İstenilen E-postalar	13	94	107
Toplam	100	100	**200**

Model 0.2.1000

Eğitim aşamasında özellik vektörü 1000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak lineer destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 20***) sunulmuştur.

Tablo 20 : Model 0.2.1000 Etkisiz Kelimeler Dahil (Lineer Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	86	1	87
İstenilen E-postalar	14	99	113
Toplam	100	100	**200**

Model 0.2.5000

Eğitim aşamasında özellik vektörü 5000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak lineer destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 21***) sunulmuştur.

Tablo 21 : Model 0.2.5000 Etkisiz Kelimeler Dahil (Lineer Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	85	0	85
İstenilen E-postalar	15	100	115
Toplam	100	100	**200**

Model 0.2.10000

Eğitim aşamasında özellik vektörü 10000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak lineer destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 22***) sunulmuştur.

Tablo 22 : Model 0.2.10000 Etkisiz Kelimeler Dahil (Lineer Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	85	0	85
İstenilen E-postalar	15	100	115
Toplam	100	100	**200**

Lineer destek vektör makinesi algoritmasıyla oluşturulan modellerden elde edilen sonuçların model performans ölçütleri ile analizinin sonuçları aşağıdaki tabloda (***Tablo 23***) paylaşılmıştır.

Tablo 23 : Lineer Destek Vektör Makinesi Algoritmasıyla Oluşturulan, Etkisiz Kelimelerin Öğrenmeye Dahil Olduğu Modellerin Model Performans Ölçütleri

Yöntem (Lineer DVM)	dp	yn	dn	yp	Doğruluk	Hata Oranı	TPR	SPC	FPR	FNR	PPV	NPV	F-ölçütü
Model 0.2	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 0.2.100	76	24	91	9	0.84	0.16	0.76	0.91	0.09	0.24	0.89	0.79	0.82
Model 0.2.200	81	19	97	3	0.89	0.11	0.81	0.97	0.03	0.19	0.96	0.84	0.879
Model 0.2.300	84	16	98	2	0.91	0.09	0.84	0.98	0.02	0.16	0.98	0.86	0.905
Model 0.2.400	86	14	93	7	0.9	0.1	0.86	0.93	0.07	0.14	0.92	0.87	0.889
Model 0.2.500	94	6	87	13	0.91	0.09	0.94	0.87	0.13	0.06	0.88	0.94	0.909
Model 0.2.1000	99	1	86	14	0.93	0.07	0.99	0.86	0.14	0.01	0.88	0.99	0.932
Model 0.2.5000	100	0	85	15	0.93	0.07	1	0.85	0.15	0	0.87	1	0.93
Model 0.2.10000	100	0	85	15	0.93	0.07	1	0.85	0.15	0	0.87	1	0.93

2.4.1.3 Bernoulli Naive Bayes Algoritmasıyla Sınıflandırma Sonuçları

Bernoulli naive bayes sınıflandırma algoritması kullanılarak oluşturulan modellerden elde edilen sonuçlar Model 0.3, Model 0.3.100, Model 0.3.200, Model 0.3.300, Model 0.3.400, Model 0.3.500, Model 0.3.1000, Model 0.3.5000 ve Model 0.3.10000 başlıkları altında sunulmuştur.

Model 0.3

Eğitim aşamasında özellik vektöründe herhangi bir kısıtlama gerçekleştirilmeyen bu modelde sınıflandırma algoritması olarak bernoulli naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 24***) sunulmuştur.

Tablo 24 : Model 0.3 Etkisiz Kelimeler Dahil (Bernoulli Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	99	58	157
İstenilen E-postalar	1	42	43
Toplam	100	100	**200**

Model 0.3.100

Eğitim aşamasında özellik vektörü 100 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak bernoulli naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 25***) sunulmuştur.

Tablo 25 : Model 0.3.100 Etkisiz Kelimeler Dahil (Bernoulli Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	84	0	84
İstenilen E-postalar	16	100	116
Toplam	100	100	**200**

Model 0.3.200

Eğitim aşamasında özellik vektörü 200 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak bernoulli naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 26***) sunulmuştur.

Tablo 26 : Model 0.3.200 Etkisiz Kelimeler Dahil (Bernoulli Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 0.3.300

Eğitim aşamasında özellik vektörü 300 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak bernoulli naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 27***) sunulmuştur.

Tablo 27: Model 0.3.300 Etkisiz Kelimeler Dahil (Bernoulli Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 0.3.400

Eğitim aşamasında özellik vektörü 400 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak bernoulli naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 28***) sunulmuştur.

Tablo 28 : Model 0.3.400 Etkisiz Kelimeler Dahil (Bernoulli Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 0.3.500

Eğitim aşamasında özellik vektörü 500 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak bernoulli naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 29***) sunulmuştur.

Tablo 29 : Model 0.3.500 Etkisiz Kelimeler Dahil (Bernoulli Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 0.3.1000

Eğitim aşamasında özellik vektörü 1000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak bernoulli naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 30***) sunulmuştur.

Tablo 30 : Model 0.3.1000 Etkisiz Kelimeler Dahil (Bernoulli Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 0.3.5000

Eğitim aşamasında özellik vektörü 5000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak bernoulli naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 31***) sunulmuştur.

Tablo 31 : Model 0.3.5000 Etkisiz Kelimeler Dahil (Bernoulli Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 0.3.10000

Eğitim aşamasında özellik vektörü 10000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak bernoulli naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 32***) sunulmuştur.

Tablo 32 : Model 0.3.10000 Etkisiz Kelimeler Dahil (Bernoulli Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	88	0	88
İstenilen E-postalar	12	100	112
Toplam	100	100	**200**

Bernoulli naive bayes algoritmasıyla oluşturulan modellerden elde edilen sonuçların model performans ölçütleri ile analizinin sonuçları aşağıdaki tabloda (***Tablo 33***) paylaşılmıştır.

Tablo 33 : Bernoulli Naive Bayes Algoritmasıyla Oluşturulan, Etkisiz Kelimelerin Öğrenmeye Dahil Olduğu Modellerin Model Performans Ölçütleri

Yöntem (Bernoulli Naive Bayes)	**dp**	**yn**	**dn**	**yp**	**Doğruluk**	**Hata Oranı**	**TPR**	**SPC**	**FPR**	**FNR**	**PPV**	**NPV**	**F-ölçütü**
Model 0.3	42	58	99	1	0.71	0.29	0.42	0.99	0.01	0.58	0.98	0.63	0.588
Model 0.3.100	100	0	84	16	0.92	0.08	1	0.84	0.16	0	0.86	1	0.925
Model 0.3.200	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 0.3.300	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 0.3.400	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 0.3.500	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 0.3.1000	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 0.3.5000	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 0.3.10000	100	0	88	12	0.94	0.06	1	0.88	0.12	0	0.89	1	0.942

2.4.1.4 Multinomial Naive Bayes Algoritmasıyla Sınıflandırma Sonuçları

Multinomial naive bayes sınıflandırma algoritması kullanılarak oluşturulan modellerden elde edilen sonuçlar Model 0.4, Model 0.4.100, Model 0.4.200, Model 0.4.300, Model 0.4.400, Model 0.4.500, Model 0.4.1000, Model 0.4.5000 ve Model 0.4.10000 başlıkları altında sunulmuştur.

Model 0.4

Eğitim aşamasında özellik vektöründe herhangi bir kısıtlama gerçekleştirilmeyen bu modelde sınıflandırma algoritması olarak multinomial naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 34***) sunulmuştur.

Tablo 34 : Model 0.4 Etkisiz Kelimeler Dahil (Multinomial Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	91	0	91
İstenilen E-postalar	9	100	109
Toplam	100	100	**200**

Model 0.4.100

Eğitim aşamasında özellik vektörü 100 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak multinomial naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 35***) sunulmuştur.

Tablo 35 : Model 0.4.100 Etkisiz Kelimeler Dahil (Multinomial Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	88	9	97
İstenilen E-postalar	12	91	103
Toplam	100	100	**200**

Model 0.4.200

Eğitim aşamasında özellik vektörü 200 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak multinomial naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 36***) sunulmuştur.

Tablo 36 : Model 0.4.200 Etkisiz Kelimeler Dahil (Multinomial Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	84	7	91
İstenilen E-postalar	16	93	109
Toplam	100	100	**200**

Model 0.4.300

Eğitim aşamasında özellik vektörü 300 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak multinomial naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 37***) sunulmuştur.

Tablo 37 : Model 0.4.300 Etkisiz Kelimeler Dahil (Multinomial Naive Bayes)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	84	5	89
İstenilen E-postalar	16	95	111
Toplam	100	100	**200**

Model 0.4.400

Eğitim aşamasında özellik vektörü 400 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak Multinomial naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 38***) sunulmuştur.

Tablo 38 : Model 0.4.400 Etkisiz Kelimeler Dahil (Multinomial Naive Bayes)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	84	5	89
İstenilen E-postalar	16	95	111
Toplam	100	100	**200**

Model 0.4.500

Eğitim aşamasında özellik vektörü 500 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak multinomial naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 39***) sunulmuştur.

Tablo 39 : Model 0.4.500 Etkisiz Kelimeler Dahil (Multinomial Naive Bayes)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	84	3	87
İstenilen E-postalar	16	97	113
Toplam	100	100	**200**

Model 0.4.1000

Eğitim aşamasında özellik vektörü 1000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak multinomial naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 40***) sunulmuştur.

Tablo 40 : Model 0.4.1000 Etkisiz Kelimeler Dahil (Multinomial Naive Bayes)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	85	0	85
İstenilen E-postalar	15	100	115
Toplam	100	100	**200**

Model 0.4.5000

Eğitim aşamasında özellik vektörü 5000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak multinomial naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 41***) sunulmuştur.

Tablo 41 : Model 0.4.5000 Etkisiz Kelimeler Dahil (Multinomial Naive Bayes)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	85	0	85
İstenilen E-postalar	15	100	115
Toplam	100	100	**200**

Model 0.4.10000

Eğitim aşamasında özellik vektörü 10000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak multinomial naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcuttur. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 42***) sunulmuştur.

Tablo 42 : Model 0.4.10000 Etkisiz Kelimeler Dahil (Multinomial Naive Bayes)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	86	0	86
İstenilen E-postalar	14	100	114
Toplam	100	100	**200**

Lineer destek vektör makinesi algoritmasıyla oluşturulan modellerden elde edilen sonuçların model performans ölçütleri ile analizinin sonuçları aşağıdaki tabloda (***Tablo 43***) paylaşılmıştır

Tablo 43 : Multinomial Naive Bayes Algoritmasıyla Oluşturulan, Etkisiz Kelimelerinde Öğrenmeye Dahil Olduğu Modellerin Model Performans Ölçütleri

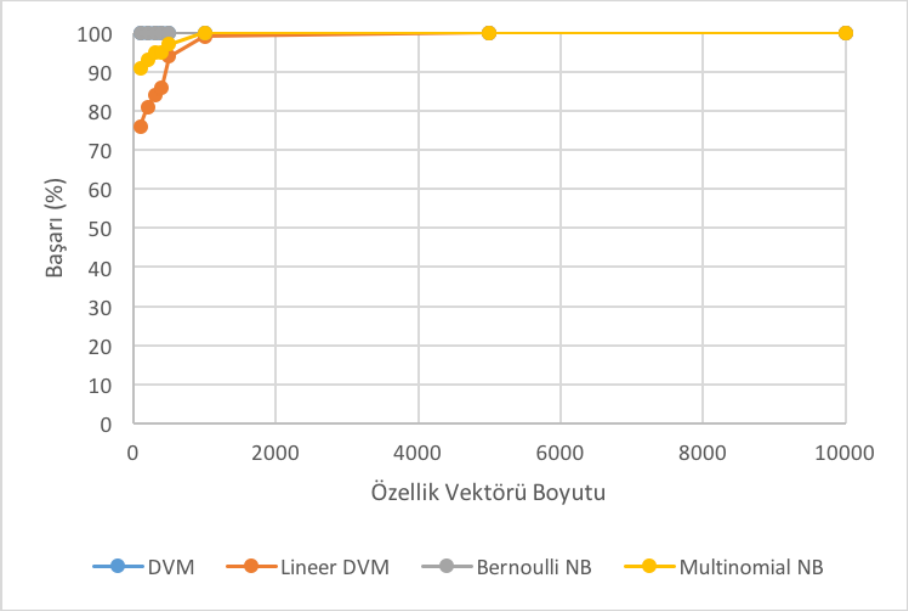
Yöntem (Multinomial Naive Bayes)	dp	yn	dn	yp	Doğruluk	Hata Oranı	TPR	SPC	FPR	FNR	PPV	NPV	F-ölçütü
Model 0.4	100	0	91	9	0.96	0.04	1	0.91	0.09	0	0.92	1	0.958
Model 0.4.100	91	9	88	12	0.9	0.1	0.91	0.88	0.12	0.09	0.88	0.91	0.895
Model 0.4.200	93	7	84	16	0.89	0.11	0.93	0.84	0.16	0.07	0.85	0.92	0.888
Model 0.4.300	95	5	84	16	0.9	0.1	0.95	0.84	0.16	0.05	0.86	0.94	0.903
Model 0.4.400	95	5	84	16	0.9	0.1	0.95	0.84	0.16	0.05	0.86	0.94	0.903
Model 0.4.500	97	3	84	16	0.91	0.09	0.97	0.84	0.16	0.03	0.86	0.97	0.912
Model 0.4.1000	100	0	85	15	0.93	0.07	1	0.85	0.15	0	0.87	1	0.93
Model 0.4.5000	100	0	85	15	0.93	0.07	1	0.85	0.15	0	0.87	1	0.93
Model 0.4.10000	100	0	86	14	0.93	0.07	1	0.86	0.14	0	0.88	1	0.936

2.4.2 Etkisiz Kelimelerin Öğrenme Kümesinden Arındırılmadığı Modellerde Farklı Boyutlardaki Özellik Vektörlerinin Sınıflandırma Başarısına Etkisi

Etkisiz kelimelerden arındırılmadan oluşturulan modellerin farklı boyutlardaki özellik vektörleriyle pozitifleri ve negatifleri sınıflandırma sonuçları aşağıdaki tablolarda ve grafiklerde sunulmuştur. Etkisiz kelimelerden arındırılmadan oluşturulan modellerin pozitifleri farklı özellik vektörleriyle sınıflandırma başarısı aşağıdaki tabloda (***Tablo 44***) ve şekilde (***Şekil 9***) sunulmuştur.

Tablo 44 : Etkisiz Kelimelerden Arındırılmadan Oluşturulan Modellerin Pozitifleri Farklı Özellik Vektörleri ile Sınıflandırma Başarısı

Özellik Vektörü / Yöntem	100	200	300	400	500	1000	5000	10000
DVM	100	100	100	100	100	100	100	100
Lineer DVM	76	81	84	86	94	99	100	100
Bernoulli NB	100	100	100	100	100	100	100	100
Multinomial NB	91	93	95	95	97	100	100	100

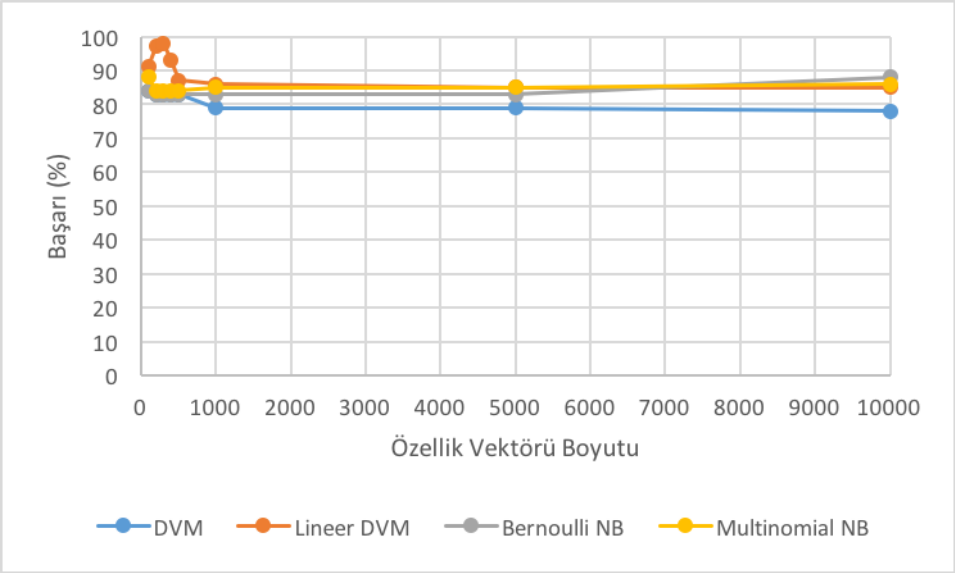


Şekil 9 : Etkisiz kelimelerden arındırılmadan oluşturulan modellerin pozitifleri farklı özellik vektörleri ile sınıflandırma başarısı

Etkisiz kelimelerden arındırılarak oluşturulan modellerin negatifleri farklı özellik vektörleriyle sınıflandırma başarısı aşağıdaki tabloda (***Tablo 45***) ve şekilde (***Şekil 10***) sunulmuştur.

Tablo 45 : Etkisiz Kelimelerden Arındırılmadan Oluşturulan Modellerin Negatifleri Farklı Özellik Vektörleri ile Sınıflandırma Başarısı

Özellik Vektörü / Yöntem	100	200	300	400	500	1000	5000	10000
DVM	84	83	83	83	83	79	79	78
Lineer DVM	91	97	98	93	87	86	85	85
Bernoulli NB	84	83	83	83	83	83	83	88
Multinomial NB	88	84	84	84	84	85	85	86



Şekil 10 : Etkisiz kelimelerden arındırılmadan oluşturulan modellerin negatifleri farklı özellik vektörleri ile sınıflandırma başarısı

2.4.3 Etkisiz Kelimelerin Öğrenme Kümesinden Arındırıldığı Modeller

Öğrenme aşamasında etkisiz kelimelerin bulunmadığı veri kümesinde tekrarlı 175.747, tekrarsız 126,609 adet kelime bulunmaktadır. Bu veri kümesi kullanılarak herhangi bir kısıtlamada bulunulmadan özellik vektörü oluşturulursa 42.869 adet kelimeyle özellik vektörünü oluşturacaktır. Eğer minimum 2 dokümanda geçen kelimeler kullanılarak özellik vektörü oluşturulursa 14.214 adet kelimeyle özellik vektörü oluşturulacaktır. Bu özellik vektörü kullanılarak elde edilen sınıflandırma sonuçları aşağıda sunulmuştur.

2.4.3.1 RBF Destek Vektör Makinesi Algoritmasıyla Sınıflandırma Sonuçları

RBF destek vektör makinesi sınıflandırma algoritması kullanılarak oluşturulan modellerden elde edilen sonuçlar Model 1.1, Model 1.1.100, Model 1.1.200, Model 1.1.300, Model 1.1.400, Model 1.1.500, Model 1.1.1000, Model 1.1.5000 ve Model 1.1.10000 başlıkları altında sunulmuştur.

Model 1.1

Eğitim aşamasında özellik vektöründe herhangi bir kısıtlama gerçekleştirilmeyen bu modelde sınıflandırma algoritması olarak rbf destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 46***) sunulmuştur.

Tablo 46 : Model 1.1 Etkisiz Kelimeler Hariç (RBF Destek Vektör Makineleri)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	58	0	58
İstenilen E-postalar	42	100	142
Toplam	100	100	**200**

Model 1.1.100

Eğitim aşamasında özellik vektörü 100 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak rbf destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 47***) sunulmuştur.

Tablo 47 : Model 1.1.100 Etkisiz Kelimeler Hariç (RBF Destek Vektör Makineleri)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 1.1.200

Eğitim aşamasında özellik vektörü 200 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak rbf destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada

kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 48***) sunulmuştur.

Tablo 48 : Model 1.1.200 Etkisiz Kelimeler Hariç (RBF Destek Vektör Makineleri)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	82	0	82
İstenilen E-postalar	18	100	118
Toplam	100	100	**200**

Model 1.1.300

Eğitim aşamasında özellik vektörü 300 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak rbf destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 49***) sunulmuştur.

Tablo 49 : Model 1.1.300 Etkisiz Kelimeler Hariç (RBF Destek Vektör Makineleri)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 1.1.400

Eğitim aşamasında özellik vektörü 400 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak rbf destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 50***) sunulmuştur.

Tablo 50 : Model 1.1.400 Etkisiz Kelimeler Hariç (RBF Destek Vektör Makineleri)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	81	0	81
İstenilen E-postalar	19	100	119
Toplam	100	100	**200**

Model 1.1.500

Eğitim aşamasında özellik vektörü 500 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak rbf destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 51***) sunulmuştur.

Tablo 51 : Model 1.1.500 Etkisiz Kelimeler Hariç (RBF Destek Vektör Makineleri)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	81	0	81
İstenilen E-postalar	19	100	119
Toplam	100	100	**200**

Model 1.1.1000

Eğitim aşamasında özellik vektörü 1000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak rbf destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 52***) sunulmuştur.

Tablo 52 : Model 1.1.1000 Etkisiz Kelimeler Hariç (RBF Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	75	0	75
İstenilen E-postalar	25	100	125
Toplam	100	100	**200**

Model 1.1.5000

Eğitim aşamasında özellik vektörü 5000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak rbf destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 53***) sunulmuştur.

Tablo 53 : Model 1.1.5000 Etkisiz Kelimeler Hariç (RBF Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	78	0	78
İstenilen E-postalar	22	100	122
Toplam	100	100	**200**

Model 1.1.10000

Eğitim aşamasında özellik vektörü 10.000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak rbf destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 54***) sunulmuştur.

Tablo 54 : Model 1.1.10000 Etkisiz Kelimeler Hariç (RBF Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	78	0	78
İstenilen E-postalar	22	100	122
Toplam	100	100	**200**

Tablo 55 : RBF Destek Vektör Makinesi Algoritmasıyla Oluşturulan, Etkisiz Kelimelerin Öğrenmeye Dahil Olmadığı Modellerin Model Performans Ölçütleri

Yöntem RBF SVM	dp	yn	dn	yp	Doğruluk	Hata Oranı	TPR	SPC	FPR	FNR	PPV	NPV	F-ölçütü
Model 1.1	100	0	58	42	0.79	0.21	1	0.58	0.42	0	0.7	1	0.824
Model 1.1.100	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 1.1.200	100	0	82	18	0.91	0.09	1	0.82	0.18	0	0.85	1	0.919
Model 1.1.300	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 1.1.400	100	0	81	19	0.91	0.09	1	0.81	0.19	0	0.84	1	0.913
Model 1.1.500	100	0	81	19	0.91	0.09	1	0.81	0.19	0	0.84	1	0.913
Model 1.1.1000	100	0	75	25	0.88	0.12	1	0.75	0.25	0	0.8	1	0.889
Model 1.1.5000	100	0	78	22	0.89	0.11	1	0.78	0.22	0	0.82	1	0.901
Model 1.1.10000	100	0	78	22	0.89	0.11	1	0.78	0.22	0	0.82	1	0.901

2.4.3.2 Lineer Destek Vektör Makinesi Algoritmasıyla Sınıflandırma Sonuçları

Lineer destek vektör makinesi sınıflandırma algoritması kullanılarak oluşturulan modellerden elde edilen sonuçlar Model 1.2, Model 1.2.100, Model 1.2.200, Model 1.2.300, Model 1.2.400, Model 1.2.500, Model 1.2.1000, Model 1.2.5000 ve Model 1.2.10000 başlıkları altında sunulmuştur.

Model 1.2

Eğitim aşamasında özellik vektöründe herhangi bir kısıtlama gerçekleştirilmeyen bu modelde sınıflandırma algoritması olarak lineer destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 56***) sunulmuştur.

Tablo 56 : Model 1.2 Etkisiz Kelimeler Hariç(Lineer Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 1.2.100

Eğitim aşamasında özellik vektörü 100 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak lineer destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 57***) sunulmuştur.

Tablo 57 : Model 1.2.100 Etkisiz Kelimeler Hariç(Lineer Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	93	40	133
İstenilen E-postalar	7	60	67
Toplam	100	100	**200**

Model 1.2.200

Eğitim aşamasında özellik vektörü 200 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak lineer destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 58***) sunulmuştur.

Tablo 58 : Model 1.2.200 Etkisiz Kelimeler Hariç(Lineer Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	94	35	129
İstenilen E-postalar	6	65	71
Toplam	100	100	**200**

Model 1.2.300

Eğitim aşamasında özellik vektörü 300 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak lineer destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 59***) sunulmuştur.

Tablo 59 : Model 1.2.300 Etkisiz Kelimeler Hariç(Lineer Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	96	29	125
İstenilen E-postalar	4	71	75
Toplam	100	100	**200**

Model 1.2.400

Eğitim aşamasında özellik vektörü 400 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak lineer destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 60***) sunulmuştur.

Tablo 60 : Model 1.2.400 Etkisiz Kelimeler Hariç(Lineer Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	88	14	102
İstenilen E-postalar	12	86	98
Toplam	100	100	**200**

Model 1.2.500

Eğitim aşamasında özellik vektörü 500 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak lineer destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 61***) sunulmuştur.

Tablo 61 : Model 1.2.500 Etkisiz Kelimeler Hariç(Lineer Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	86	9	95
İstenilen E-postalar	14	91	105
Toplam	100	100	**200**

Model 1.2.1000

Eğitim aşamasında özellik vektörü 1000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak lineer destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 62***) sunulmuştur.

Tablo 62 : Model 1.2.1000 Etkisiz Kelimeler Hariç(Lineer Destek Vektör Makineleri)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	86	1	87
İstenilen E-postalar	14	99	113
Toplam	100	100	**200**

Model 1.2.5000

Eğitim aşamasında özellik vektörü 5.000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak lineer destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 63***) sunulmuştur.

Tablo 63 : Model 1.2.5000 Etkisiz Kelimeler Hariç(Lineer Destek Vektör Makineleri)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	85	0	85
İstenilen E-postalar	15	100	115
Toplam	100	100	**200**

Model 1.2.10000

Eğitim aşamasında özellik vektörü 10.000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak lineer destek vektör makinesi yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 64***) sunulmuştur.

Tablo 64 : Model 1.2.10000 Etkisiz Kelimeler Hariç (Lineer Destek Vektör Makineleri)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	85	0	85
İstenilen E-postalar	15	100	115
Toplam	100	100	**200**

Lineer destek vektör makinesi algoritmasıyla oluşturulan modellerden elde edilen sonuçların model performans ölçütleri ile analizinin sonuçları aşağıdaki tabloda (***Tablo 65***) paylaşılmıştır.

Tablo 65 : Lineer Destek Vektör Makinesi Algoritmasıyla Oluşturulan, Etkisiz Kelimelerin Öğrenmeye Dahil Olmadığı Modellerin Model Performans Ölçütleri

Yöntem Lineer DVM	dp	yn	dn	yp	Doğruluk	Hata Oranı	TPR	SPC	FPR	FNR	PPV	NPV	F-ölçütü
Model 1.2	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 1.2.100	60	40	93	7	0.77	0.23	0.6	0.93	0.07	0.4	0.9	0.7	0.72
Model 1.2.200	65	35	94	6	0.8	0.2	0.65	0.94	0.06	0.35	0.92	0.73	0.762
Model 1.2.300	71	29	96	4	0.84	0.16	0.71	0.96	0.04	0.29	0.95	0.77	0.813
Model 1.2.400	86	14	88	12	0.87	0.13	0.86	0.88	0.12	0.14	0.88	0.86	0.87
Model 1.2.500	91	9	86	14	0.89	0.11	0.91	0.86	0.14	0.09	0.87	0.91	0.89
Model 1.2.1000	99	1	86	14	0.93	0.07	0.99	0.86	0.14	0.01	0.88	0.99	0.932
Model 1.2.5000	100	0	85	15	0.93	0.07	1	0.85	0.15	0	0.87	1	0.93
Model 1.2.10000	100	0	85	15	0.93	0.07	1	0.85	0.15	0	0.87	1	0.93

2.4.3.3 Bernoulli Naive Bayes Algoritmasıyla Sınıflandırma Sonuçları

Bernoulli naive bayes sınıflandırma algoritması kullanılarak oluşturulan modellerden elde edilen sonuçlar Model 1.3, Model 1.3.100, Model 1.3.200, Model 1.3.300, Model 1.3.400, Model 1.3.500, Model 1.3.1000, Model 1.3.5000 ve Model 1.3.10000 başlıkları altında sunulmuştur.

Model 1.3

Eğitim aşamasında özellik vektöründe herhangi bir kısıtlama gerçekleştirilmeyen bu modelde sınıflandırma algoritması olarak bernoulli naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 66***) sunulmuştur.

Tablo 66 : Model 1.3 Etkisiz Kelimeler Hariç(Bernoulli Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	99	61	160
İstenilen E-postalar	1	39	40
Toplam	100	100	**200**

Model 1.3.100

Eğitim aşamasında özellik vektörü 100 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak bernoulli naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 67***) sunulmuştur.

Tablo 67 : Model 1.3.100 Etkisiz Kelimeler Hariç(Bernoulli Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 1.3.200

Eğitim aşamasında özellik vektörü 200 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak bernoulli naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 68***) sunulmuştur.

Tablo 68 : Model 1.3.200 Etkisiz Kelimeler Hariç (Bernoulli Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 1.3.300

Eğitim aşamasında özellik vektörü 300 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak bernoulli naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 69***) sunulmuştur.

Tablo 69 : Model 1.3.300 Etkisiz Kelimeler Hariç (Bernoulli Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 1.3.400

Eğitim aşamasında özellik vektörü 400 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak bernoulli naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 70***) sunulmuştur.

Tablo 70 : Model 1.3.400 Etkisiz Kelimeler Hariç (Bernoulli Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 1.3.500

Eğitim aşamasında özellik vektörü 500 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak bernoulli naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 71***) sunulmuştur.

Tablo 71 : Model 1.3.500 Etkisiz Kelimeler Hariç(Bernoulli Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 1.3.1000

Eğitim aşamasında özellik vektörü 1000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak bernoulli naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 72***) sunulmuştur.

Tablo 72 : Model 1.3.1000 Etkisiz Kelimeler Hariç(Bernoulli Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 1.3.5000

Eğitim aşamasında özellik vektörü 5000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak bernoulli naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 73***) sunulmuştur.

Tablo 73 : Model 1.3.5000 Etkisiz Kelimeler Hariç(Bernoulli Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	83	0	83
İstenilen E-postalar	17	100	117
Toplam	100	100	**200**

Model 1.3.10000

Eğitim aşamasında özellik vektörü 10000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak bernoulli naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 74***) sunulmuştur.

Tablo 74 : Model 1.3.10000 Etkisiz Kelimeler Hariç (Bernoulli Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	85	0	85
İstenilen E-postalar	15	100	115
Toplam	100	100	**200**

Bernoulli naive bayes algoritmasıyla oluşturulan modellerden elde edilen sonuçların model performans ölçütleri ile analizinin sonuçları aşağıdaki tabloda (***Tablo 75***) paylaşılmıştır.

Tablo 75 : Bernoulli Naive Bayes Algoritmasıyla Oluşturulan, Etkisiz Kelimelerin Öğrenmeye Dahil Olmadığı Modellerin Model Performans Ölçütleri

Yöntem (Bernoulli Naive Bayes)	**dp**	**yn**	**dn**	**yp**	**Doğruluk**	**Hata Oranı**	**TPR**	**SPC**	**FPR**	**FNR**	**PPV**	**NPV**	**F-ölçütü**
Model 1.3	39	61	99	1	0.69	0.31	0.39	0.99	0.01	0.61	0.98	0.62	0.558
Model 1.3.100	100	0	83	17	0.92	0.08	1	0.84	0.16	0	0.86	1	0.925
Model 1.3.200	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 1.3.300	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 1.3.400	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 1.3.500	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 1.3.1000	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 1.3.5000	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 1.3.10000	100	0	85	15	0.93	0.07	1	0.85	0.15	0	0.87	1	0.93

2.4.3.4 Multinomial Naive Bayes Algoritmasıyla Sınıflandırma Sonuçları

Multinomial naive bayes sınıflandırma algoritması kullanılarak oluşturulan modellerden elde edilen sonuçlar Model 1.4, Model 1.4.100, Model 1.4.200, Model 1.4.300, Model 1.4.400, Model 1.4.500, Model 1.4.1000, Model 1.4.5000 ve Model 1.4.10.000 başlıkları altında sunulmuştur.

Model 1.4

Eğitim aşamasında özellik vektöründe herhangi bir kısıtlama gerçekleştirilmeyen bu modelde sınıflandırma algoritması olarak multinomial naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 76***) sunulmuştur.

Tablo 76 : Model 1.4 Etkisiz Kelimeler Hariç (Multinomial Naive Bayes)

Gerçek / **Öngörü**	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	91	3	94
İstenilen E-postalar	9	97	106
Toplam	100	100	**200**

Model 1.4.100

Eğitim aşamasında özellik vektörü 100 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak multinomial naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 77***) sunulmuştur.

Tablo 77 : Model 1.4.100 Etkisiz Kelimeler Hariç (Multinomial Naive Bayes)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	86	24	110
İstenilen E-postalar	14	76	90
Toplam	100	100	**200**

Model 1.4.200

Eğitim aşamasında özellik vektörü 200 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak multinomial naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 78***) sunulmuştur.

Tablo 78 : Model 1.4.200 Etkisiz Kelimeler Hariç(Multinomial Naive Bayes)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	84	14	98
İstenilen E-postalar	16	86	102
Toplam	100	100	**200**

Model 1.4.300

Eğitim aşamasında özellik vektörü 300 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak multinomial naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 79***) sunulmuştur.

Tablo 79 : Model 1.4.300 Etkisiz Kelimeler Hariç(Multinomial Naive Bayes)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	84	11	95
İstenilen E-postalar	16	89	105
Toplam	100	100	**200**

Model 1.4.400

Eğitim aşamasında özellik vektörü 400 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak multinomial naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 80***) sunulmuştur.

Tablo 80 : Model 1.4.400 Etkisiz Kelimeler Hariç(Multinomial Naive Bayes)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	84	5	89
İstenilen E-postalar	16	95	111
Toplam	100	100	**200**

Model 1.4.500

Eğitim aşamasında özellik vektörü 500 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak multinomial naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 81***) sunulmuştur.

Tablo 81 : Model 1.4.500 Etkisiz Kelimeler Hariç (Multinomial Naive Bayes)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	84	3	87
İstenilen E-postalar	16	97	123
Toplam	100	100	**200**

Model 1.4.1000

Eğitim aşamasında özellik vektörü 1000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak Multinomial naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 82***) sunulmuştur.

Tablo 82 : Model 1.4.1000 Etkisiz Kelimeler Hariç(Multinomial Naive Bayes)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	86	0	86
İstenilen E-postalar	14	100	114
Toplam	100	100	**200**

Model 1.4.5000

Eğitim aşamasında özellik vektörü 5000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak multinomial naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 83***) sunulmuştur.

Tablo 83 : Model 1.4.5000 Etkisiz Kelimeler Hariç(Multinomial Naive Bayes)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	86	0	86
İstenilen E-postalar	14	100	114
Toplam	100	100	**200**

Model 1.4.10000

Eğitim aşamasında özellik vektörü 10000 elemanla sınırlandırılan bu modelde sınıflandırma algoritması olarak multinomial naive bayes yaklaşımından yararlanılmıştır. Çalışmada kullanılan eğitim kümesinde etkisiz kelimeler mevcut değildir. Elde edilen sonuçlar aşağıdaki tabloda (***Tablo 84***) sunulmuştur.

Tablo 84 : Model 1.4.10000 Etkisiz Kelimeler Hariç(Multinomial Naive Bayes)

Gerçek / Öngörü	İstenmeyen E-Postalar	İstenilen E-postalar	**Toplam**
İstenmeyen E-Postalar	86	0	86
İstenilen E-postalar	14	100	114
Toplam	100	100	**200**

Lineer destek vektör makinesi algoritmasıyla oluşturulan modellerden elde edilen sonuçların model performans ölçütleri ile analizinin sonuçları aşağıdaki tabloda (***Tablo 85***) paylaşılmıştır.

Tablo 85 : Multinomial Naive Bayes Algoritmasıyla Oluşturulan, Öğrenmeye Etkisiz Kelimelerin Dahil Olmadığı Modellerin Model Performans Ölçütleri

Yöntem (Multinomial Naive Bayes)	**dp**	**yn**	**dn**	**yp**	**Doğruluk**	**Hata Oranı**	**TPR**	**SPC**	**FPR**	**FNR**	**PPV**	**NPV**	**F-ölçütü**
Model 1.4	97	3	91	9	0.94	0.06	0.97	0.91	0.09	0.03	0.92	0.97	0.944
Model 1.4.100	76	24	86	14	0.81	0.19	0.76	0.86	0.14	0.24	0.84	0.78	0.798
Model 1.4.200	86	14	84	16	0.85	0.15	0.86	0.84	0.16	0.14	0.84	0.86	0.85
Model 1.4.300	89	11	84	16	0.87	0.13	0.89	0.84	0.16	0.11	0.85	0.88	0.87
Model 1.4.400	95	5	84	16	0.9	0.1	0.95	0.84	0.16	0.05	0.86	0.94	0.903
Model 1.4.500	97	3	84	16	0.91	0.09	0.97	0.84	0.16	0.03	0.86	0.97	0.912
Model 1.4.1000	100	0	86	14	0.93	0.07	1	0.86	0.14	0	0.88	1	0.936
Model 1.4.5000	100	0	86	14	0.93	0.07	1	0.86	0.14	0	0.88	1	0.936
Model 1.4.10000	100	0	86	14	0.93	0.07	1	0.86	0.14	0	0.88	1	0.936

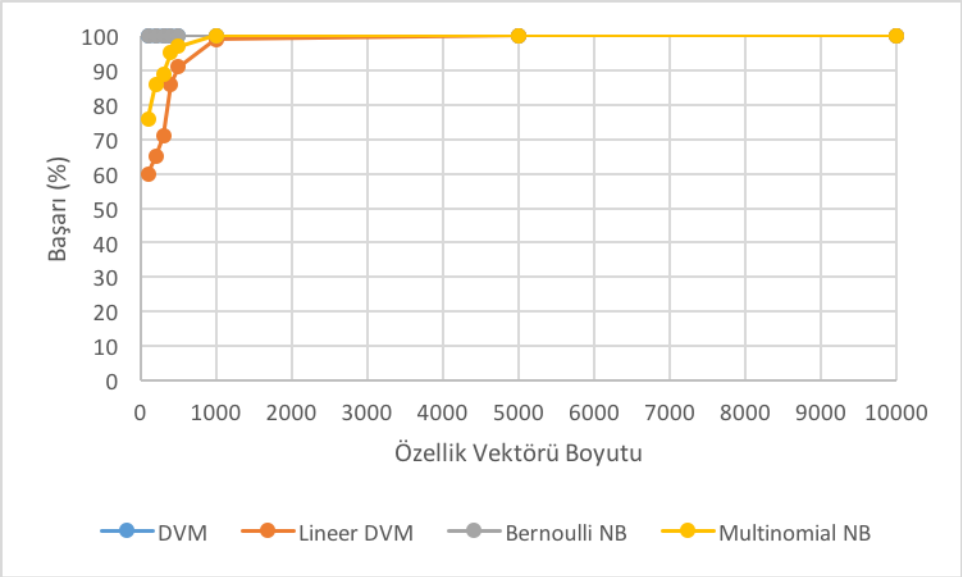
2.4.4 Etkisiz Kelimelerin Öğrenme Kümesinden Arındırıldığı Modellerde Farklı Boyutlardaki Özellik Vektörlerinin Sınıflandırma Başarısına Etkisi

Etkisiz kelimelerden arındırılarak oluşturulan modellerin farklı boyutlardaki özellik vektörleriyle pozitifleri ve negatifleri sınıflandırma sonuçları aşağıdaki tablolarda ve grafiklerde sunulmuştur.

Etkisiz kelimelerden arındırılarak oluşturulan modellerin pozitifleri farklı özellik vektörleriyle sınıflandırma başarısı aşağıdaki tabloda (***Tablo 86***) ve şekilde (***Şekil 11***) sunulmuştur.

Tablo 86 : Etkisiz Kelimelerden Arındırılarak Oluşturulan Modellerin Pozitifleri Farklı Özellik Vektörleriyle Sınıflandırma Başarısı

Özellik Vektörü / **Yöntem**	**100**	**200**	**300**	**400**	**500**	**1000**	**5000**	**10000**
DVM	100	100	100	100	100	100	100	100
Lineer DVM	60	65	71	86	91	99	100	100
Bernoulli NB	100	100	100	100	100	100	100	100
Multinomial NB	76	86	89	95	97	100	100	100

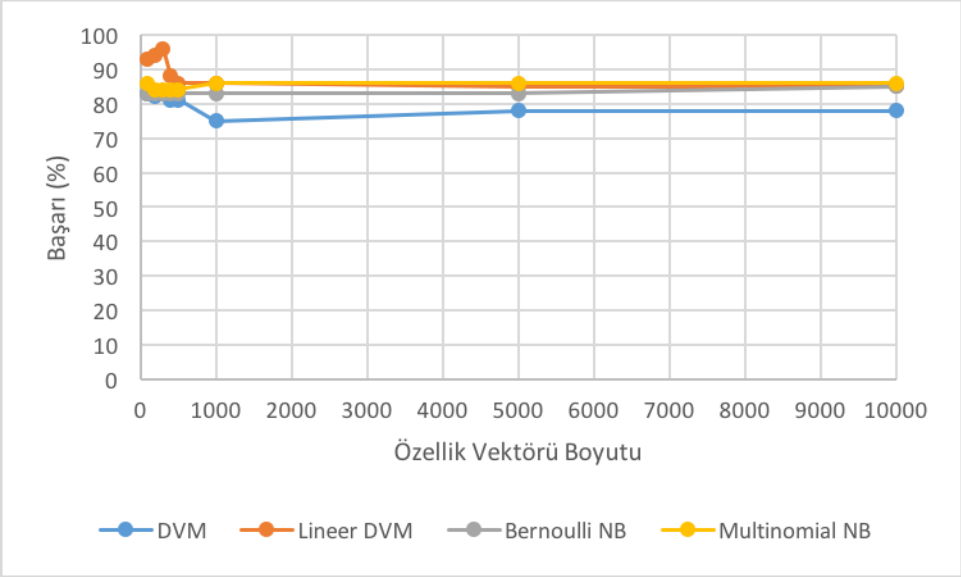


Şekil 11 : Etkisiz kelimelerden arındırılarak oluşturulan modellerin pozitifleri farklı özellik vektörleriyle sınıflandırma başarısı

Etkisiz kelimelerden arındırılarak oluşturulan modellerin negatifleri farklı özellik vektörleriyle sınıflandırma başarısı aşağıdaki tabloda (***Tablo 86***) ve şekilde (***Şekil 12***) sunulmuştur.

Tablo 87 : Etkisiz Kelimelerden Arındırılarak Oluşturulan Modellerin Negatifleri Farklı Özellik Vektörleriyle Sınıflandırma Başarısı

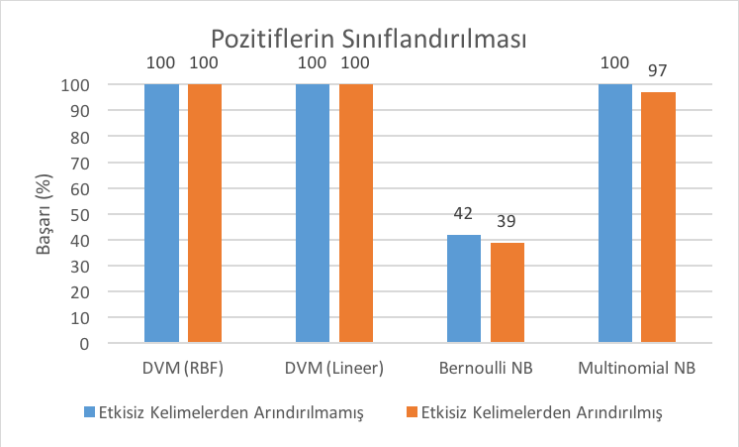
Özellik Vektörü / Yöntem	100	200	300	400	500	1000	5000	10000
DVM	83	82	83	81	81	75	78	78
Lineer DVM	93	94	96	88	86	86	85	85
Bernoulli NB	83	83	83	83	83	83	83	85
Multinomial NB	86	84	84	84	84	86	86	86



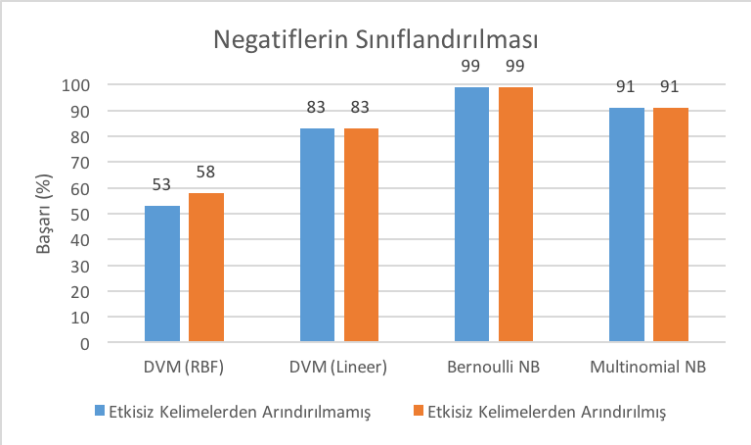
Şekil 12 : Etkisiz kelimelerden arındırılarak oluşturulan modellerin negatifleri farklı özellik vektörleriyle sınıflandırma başarısı

2.4.5 Model 0 ile Model 1'nin Karşılaştırılması

Destek vektör makineleri (RBF, Lineer) ve Naive Bayes (Multinomial , Bernoulli) algoritmaları kullanılarak gerçekleştirilen çalışmamızda özellik vektörü kısıtlanmamış modeller (Model 0, Model 1) karşılaştırılmıştır. Böylece etkisiz kelimelerin eğitim kümesinden arındırılıp arındırılmama durumunun sınıflandırma başarısına etkisi araştırılmıştır. Elde edilen bulgular aşağıdaki şekillerde (***Şekil 13 , Şekil 14***) paylaşılmıştır.



Şekil 13 : Özellik vektörü kısıtlanmamış modellerin pozitifleri sınıflandırma başarısı



Şekil 14 : Özellik vektörü kısıtlanmamış modellerin negatifleri sınıflandırma başarısı

Etkisiz kelimelerin modelden arındırılması durumunda; Bernoulli Naive Bayes ve Multinomial Naive Bayes algoritmalarının pozitiflerin sınıflandırılmasındaki başarısında %3'lük bir azalma gözlenmektedir. Negatiflerin sınıflandırılmasında ise etkisiz kelimelerden arındırılan Destek Vektör Makineleri algoritmasının %5'lik daha yüksek sınıflandırma başarına sahip olduğu gözlemlenmiştir.

2.4.6 İstenmeyen E-postaları En İyi Sınıflandıran Model

İstenmeyen e-postaları en yüksek doğru sınıflandırma başarısına sahip olan modeller; özellik vektörü kısıtlanmamış, etkisiz kelimelerin öğrenme kümesinde bulunup bulunmama durumunun fark etmediği, sınıflandırma algoritması olarak Bernoulli Naive Bayes algoritmasının kullanıldığı modellerdir. Sonuç olarak 100 istenmeyen e-postanın 99 adedini doğru olarak sınıflandırılmıştır. İstenmeyen e-postaları sınıflamada en yüksek başarıya sahip olan bu modellerin istenilen e-postaları sınıflandırma başarısı incelendiğinde diğer modellere göre çok düşük bir sınıflandırma başarısına sahip olduğu gözlemlenmiştir. F-Ölçütü değerine göre diğer sınıflandırma algoritmalarından daha düşük doğru sınıflandırma başarısına sahiptir. Ayrıca yine F-Ölçütüne göre bu çalışmada etkisiz kelimelerin arındırılmadığı model etkisiz kelimelerin arındırıldığı modelden daha yüksek bir sınıflandırma başarısına sahiptir. Detaylar için lütfen Ek 1 ve Ek 2'de paylaşılan sonuçlardan Bernoulli Naive Bayes sınıflandırmasıyla oluşturulan Model 0.3 ve Model 1.3 sonuçlarına bakınız.

2.4.7 İstenen E-postaları En İyi Sınıflandıran Modeller

Elde edilen tüm modellerde istenilen e-postaları doğru sınıflandırmadaki başarıları bakımından Bernoulli Naive Bayes ve Destek Vektör Makineleri (RBF) algoritmaları özellik vektörleri kısıtlanan modellerde %100 doğrulukla sınıflandırma işlemini gerçekleştirmiştir. Destek Vektör Makineleri (RBF) algoritması özellik vektörünün kısıtlanmadığı modelde de %100 doğrulukla sınıflandırma işlemini gerçekleştirmiştir. Ek 1 ve Ek 2'de sonuçlar detaylı olarak sunulmuştur.

2.4.8 Etkisiz Kelimelerin Eğitim Kümesinden Arındırılıp Arındırılmama Durumlarının Modellerdeki Sınıflandırma Başarılarına Etkisi

Elde edilen sonuçlar incelendiğinde Model 0.2.100 ile Model 1.2.100 lineer destek vektör makineleri ile gerçekleştirilen negatiflerin sınıflandırma sonucu, özellik vektörü kısıtlanmamış modellerde destek vektör makineleri (rbf) Model 0.1 ile Model 1.1 negatiflerin sınıflandırılmasında, Model 0.4.1000 Model 1.4.1000 ile gerçekleştirilen negatiflerin sınıflandırılması ve Model 0.4.5000 Model 1.4.5000 ile gerçekleştirilen negatiflerin sınıflandırılması ile ulaşılan sonuçlar dışında etkisiz kelimelerin eğitim kümesinden çıkarılması sonucunda birçok modelin sınıflandırma başarısında azalma gözlemlenmektedir. Etkisiz kelimelerin modelden arındırılıp arındırılmama durumlarında elde edilen F-Ölçütlerinin karşılaştırılması sonucunda etkisiz kelimelerden arındırılmayan modellerin daha yüksek sınıflandırma başarısına sahip olduğu gözlemlenmiştir.

2.4.9 F-Ölçütü ile Modellerin Karşılaştırılması

Aşağıdaki tablolarda ***(Tablo 88, Tablo 89***) farklı özellik vektörleri ile oluşturulan modellerin f-ölçütü değerine göre karşılaştırılma sonuçları paylaşılmıştır.

Tablo 88 : Etkisiz Kelimelerin Mevcut Olduğu Modellerin F-Ölçütleri

F – Ölçütleri / Modeller	Etkisiz Kelimelerin Mevcut Olduğu Modeller			
	DVM (RBF)	Lineer DVM	Bernoulli NB	Multinomial NB
	F-Ölçütü	F-Ölçütü	F-Ölçütü	F-Ölçütü
Model 0.(1/2/3/4)	***0.81***	***0.919***	***0.588***	***0.958***
Model 0.(1/2/3/4).100	**0.925**	0.82	**0.925**	0.895
Model 0.(1/2/3/4).200	**0.919**	0.879	**0.919**	0.888
Model 0.(1/2/3/4).300	**0.919**	0.905	**0.919**	0.903
Model 0.(1/2/3/4).400	**0.919**	0.889	**0.919**	0.903
Model 0.(1/2/3/4).500	**0.919**	0.909	**0.919**	0.912
Model 0.(1/2/3/4).1000	0.907	0.932	0.919	0.93
Model 0.(1/2/3/4).5000	0.907	0.93	0.919	0.93
Model 0.(1/2/3/4).10000	0.901	0.93	**0.942**	0.936

Tablo 89 : Etkisiz Kelimelerin Mevcut Olmadığı Modellerin F-Ölçütleri

F – Ölçütleri / Modeller	Etkisiz Kelimelerin Mevcut Olmadığı Modeller			
	DVM (RBF)	Lineer DVM	Bernoulli NB	Multinomial NB
	F-Ölçütü	F-Ölçütü	F-Ölçütü	F-Ölçütü
Model 1.(1/2/3/4)	***0.824***	***0.919***	***0.558***	***0.944***
Model 1.(1/2/3/4).100	0.919	0.72	**0.925**	0.798
Model 1.(1/2/3/4).200	**0.919**	0.762	**0.919**	0.85
Model 1.(1/2/3/4).300	**0.919**	0.813	**0.919**	0.87
Model 1.(1/2/3/4).400	0.913	0.87	**0.919**	0.903
Model 1.(1/2/3/4).500	0.913	0.89	**0.919**	0.912
Model 1.(1/2/3/4).1000	0.889	0.932	0.919	**0.936**
Model1.(1/2/3/4).5000	0.901	0.93	0.919	**0.936**
Model 1.(1/2/3/4).10000	***0.901***	***0.93***	***0.93***	***0.936***

Tablo 90 : F-Ölçütleri ile Modellerin Karşılaştırılması

En iyi Sonuç Veren Modeller	
Model 0, Model 1	Multinomial NB En iyi Model
Model (0/1).(1/2/3/4).100	Bernoulli NB DVM(RBF) En iyi Modeller
Model (0/1).(1/2/3/4).200	Bernoulli NB DVM(RBF) En iyi Modeller
Model (0/1).(1/2/3/4).300	Bernoulli NB DVM(RBF) En iyi Modeller
Model (0/1).(1/2/3/4).400	Bernoulli NB DVM(RBF) En iyi Modeller
Model (0/1).(1/2/3/4).500	Lineer (DVM) ,Bernoulli NB) En iyi Modeller
Model (0/1).(1/2/3/4).1000	Multinomial NB En iyi Model
Model (0/1).(1/2/3/4).5000	Multinomial NB En iyi Model
Model (0/1).(1/2/3/4).10000	Bernoulli NB DVM(RBF) En iyi Model

Elde edilen verilerden ulaşılan sonuçlar yukarıdaki tabloda (***Tablo 90***) paylaşılmıştır. Bernoulli Naive Bayes Algoritması farklı boyutlu özellik vektörü ile oluşturulan birçok farklı modelde en yüksek sınıflandırma başarısını göstermiştir. Özellik vektörü boyutunun 500'den az olduğu modellerde Bernoulli Naive Bayes, Destek Vektör Makineleri (RBF) sınıflandırma algoritmaları en yüksek sınıflandırma başarısına sahipken yüksek boyutlu özellik vektörlerinde Multinomial Naive Bayes algoritmasının daha iyi bir sınıflandırma başarısına sahip olduğu gözlemlenmiştir. Ayrıca bu sonuçlar incelendiğinde aynı boyutta özellik vektörleri ve aynı sınıflandırma algoritmaları kullanılarak oluşturulan modellerin F-Ölçütleri karşılaştırıldığında; eğitim kümesinde etkisiz kelimeleri barındıran modellerin eğitim kümesinde etkisiz kelimeleri barındırmayan modellere nazaran daha yüksek değere sahip olduğu görülmektedir.

SONUÇ

Bu çalışmayı gerçekleştirmek için Semih Ergin ve arkadaşlarının yayınlamış oldukları "Turkish Anti-Spam Filtering Using Binary and Probabilistic Models"[73] adlı makalenin veri kümesinden yararlanılmıştır. Ergin ve arkadaşlarının çalışmasında en iyi sonuca Binary Modelle gerçekleştirilen sınıflandırmayla %93 oranında ulaşılmıştır.

Çalışmamızda en yüksek sınıflandırma başarısına eğitim kümesinde etkisiz kelimelerin de bulunduğu, özellik vektörünün kısıtlanmadığı Multinomial Naive Bayes algoritması kullanılarak oluşturulan model ile ulaşılmıştır. F- Ölçütü %95,8, doğruluk %95,5, hassasiyet %100, belirleyicilik : %91 .

4 farklı sınıflandırma algoritması kullanılarak 72 farklı modelle gerçekleştirilen çalışmamızda en yüksek sınıflandırma sonuçları multinomial naive bayes algoritması ile elde edilmiştir.

Elde edilen modellerin tamamı incelendiğinde, Drucker ve arkadaşlarının makalesinde "etkisiz kelimelerin öğrenme kümesinden çıkarılmaması" önerisinin Türkçe e-postalar için de geçerli olduğu gözlemlenmiştir.

Etkisiz kelimelerin bulunmadığı durum göz önüne alındığında en yüksek sınıflandırma başarısına özellik vektörünün kısıtlanmadığı Multinomial Naive Bayes algoritmasıyla ulaşılmıştır.

Özellik vektörünün kısıtlandığı modellerde ise en yüksek sınıflandırma başarısına eğitim kümesine etkisiz kelimelerinde bulunduğu 10.000 elemanlı özellik vektörü ve Bernoulli Naive Bayes algoritması kullanılarak oluşturulan model ile ulaşılmıştır.

Özellik vektörü boyutunun 500'den az olduğu modellerde Bernoulli Naive Bayes ve Destek Vektör Makineleri (RBF) sınıflandırma algoritmaları en yüksek sınıflandırma başarısına sahiptir. Fakat yüksek boyutlu özellik vektörlerinde Multinomial Naive Bayes algoritmasının daha iyi bir sınıflandırma başarısına sahip olduğu gözlemlenmiştir.

Özellik vektörünün kısıtlanmadığı modeller ile özellik vektörünün kısıtlandığı modeller karşılaştırılmıştır. Bernoulli Naive Bayes algoritmasıyla gerçekleştirilen pozitiflerin sınıflandırılması işleminde özellik vektörü kısıtlanan modele göre özellik vektörü kısıtlanmayan model daha düşük bir başarıyla sınıflandırma işlemini gerçekleştirmiştir. Yine Destek Vektör Makineleri (Rbf) algoritmasının negatifleri sınıflandırma başarısında özellik vektörü kısıtlanmamış model özellik vektörü kısıtlanan modele göre daha düşük başarıyla sınıflandırma işlemini gerçekleştirmiştir.

Bu sonuçlar göz önüne alındığında daha az eleman sayısına sahip özellik vektörü kullanılarak daha yüksek sınıflandırma sonuçlarına ulaşabildiği gözlemlenmiştir. Böylelikle özellik vektörü seçiminin sınıflandırma başarısındaki önemi elde edilen bulgular üzerinden ispatlanmıştır.

Çalışmada sınama veri kümesinde bulunan istenilen e-postaları %100 başarı ile sınıflandırmak mümkün olurken, istenmeyen e-postaları sınıflandırma başarısı daha düşük olarak gerçekleştirilmiştir.

Ergin ve arkadaşlarının makalesinde de belirtildiği üzere birçok veri madenciliği problemi gibi istenmeyen e-postaların sınıflandırılmasında da veri miktarının fazla olması ulaşılacak başarının daha yüksek olmasına katkı sağlamaktadır.

Gerçekleştirilen çalışmada elde edilen bulgular incelendiğinde bazı algoritmaların negatifleri ve pozitifleri sınıflandırırken ya negatifleri ya da pozitifleri daha iyi sınıflandırma eğiliminde olduğu gözlemlenmiştir. Bu sonuçlardan yola çıkarak hibrit modeller üretilebilir. Üretilen bu modeller yardımıyla sınıflandırma başarısı arttırılabilir.

Çalışmamızda kullanılan derlem hem veri sayısı bakımından hem de verinin elde edilmesinde kullanılan farklı e-posta hesabı (kaynak) bakımından kısıtlı durumdadır. Kısıtlı kaynağın en önemli etkilerinden birisi kaynağın sürekli aynı türden e-postalara maruz kalmasıdır. Bu durum eğitim kümesinin sınırlarını belirlemektedir. Daha başarılı çalışmalar için kaynak çeşitliliğinin arttırılması faydalı olacaktır.

Çalışmamızda etkisiz kelimelerin sınıflandırma başarısına etkisi TF-IDF yöntemi ile elde edilen özellik vektörleriyle incelenmiştir. Daha sonraki çalışmalarda farklı özellik vektörü oluşturma yöntemleri kullanarak etkisiz kelimelerin sınıflandırma başarısına etkisi incelenebilir.

Veri madenciliği teknikleri ile gerçekleştirilen görsel sınıflandırma uygulamalarında veri kümesinin yetersiz olduğu durumlarda mevcut veriler kullanılarak suni veriler sentezlenmekte (görselin kırpılması, döndürülmesi) ve bu veriler kullanılarak eğitim kümesi zenginleştirilmektedir. Aynı uygulamanın metin madenciliği alanında da gerçekleştirilmesi sınıflandırma başarısını arttırabilir.

EKLER

Ek 1 : Etkisiz Kelimelerin Öğrenme Kümesinde Bulunduğu Modellerin Farklı Özellik Vektörleri ile Analizi

Yöntem (Rbf Destek Vektör Makineleri)	dp	yn	dn	Yp	Doğruluk	Hata Oranı	TPR	SPC	FPR	FNR	PPV	NPV	F-ölçütü
Model 0.1	100	0	53	47	0.765	0.235	1	0.53	0.47	0	0.68	1	0.81
Model 0.1.100	100	0	84	16	0.92	0.08	1	0.84	0.16	0	0.86	1	0.925
Model 0.1.200	100	0	83	17	0.915	0.085	1	0.83	0.17	0	0.85	1	0.919
Model 0.1.300	100	0	83	17	0.915	0.085	1	0.83	0.17	0	0.85	1	0.919
Model 0.1.400	100	0	83	17	0.915	0.085	1	0.83	0.17	0	0.85	1	0.919
Model 0.1.500	100	0	83	17	0.915	0.085	1	0.83	0.17	0	0.85	1	0.919
Model 0.1.1000	100	0	79	21	0.895	0.105	1	0.79	0.21	0	0.83	1	0.907
Model 0.1.5000	100	0	79	21	0.895	0.105	1	0.79	0.21	0	0.83	1	0.907
Model 0.1.10000	100	0	78	22	0.89	0.11	1	0.78	0.22	0	0.82	1	0.901
Yöntem (Lineer Destek Vektör Makineleri)	dp	yn	dn	Yp	Doğruluk	Hata Oranı	TPR	SPC	FPR	FNR	PPV	NPV	F-ölçütü
Model 0.2	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 0.2.100	76	24	91	9	0.84	0.16	0.76	0.91	0.09	0.24	0.89	0.79	0.82
Model 0.2.200	81	19	97	3	0.89	0.11	0.81	0.97	0.03	0.19	0.96	0.84	0.879
Model 0.2.300	84	16	98	2	0.91	0.09	0.84	0.98	0.02	0.16	0.98	0.86	0.905
Model 0.2.400	86	14	93	7	0.9	0.1	0.86	0.93	0.07	0.14	0.92	0.87	0.889
Model 0.2.500	94	6	87	13	0.91	0.09	0.94	0.87	0.13	0.06	0.88	0.94	0.909
Model 0.2.1000	99	1	86	14	0.93	0.07	0.99	0.86	0.14	0.01	0.88	0.99	0.932
Model 0.2. 5000	100	0	85	15	0.93	0.07	1	0.85	0.15	0	0.87	1	0.93
Model 0.2.10000	100	0	85	15	0.93	0.07	1	0.85	0.15	0	0.87	1	0.93
Yöntem (Bernoulli Naive Bayes)	dp	yn	dn	yp	Doğruluk	Hata Oranı	TPR	SPC	FPR	FNR	PPV	NPV	F-ölçütü
Model 0.3	***42***	***58***	***99***	***1***	***0.71***	***0.29***	***0.42***	***0.99***	***0.01***	***0.58***	***0.98***	***0.63***	***0.588***
Model 0.3.100	100	0	84	16	0.92	0.08	1	0.84	0.16	0	0.86	1	0.925
Model 0.3.200	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 0.3.300	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 0.3.400	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 0.3.500	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 0.3.1000	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 0.3.5000	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 0.3.10000	***100***	***0***	***88***	***12***	***0.94***	***0.06***	***1***	***0.88***	***0.12***	***0***	***0.89***	***1***	***0.942***
Yöntem (Multinomial Naive Bayes)	dp	yn	dn	yp	Doğruluk	Hata Oranı	TPR	SPC	FPR	FNR	PPV	NPV	F-ölçütü
Model 0.4	***100***	***0***	***91***	***9***	***0.955***	***0.045***	***1***	***0.91***	***0.09***	***0***	***0.92***	***1***	***0.958***
Model 0.4.100	91	9	88	12	0.9	0.1	0.91	0.88	0.12	0.09	0.88	0.91	0.895
Model 0.4.200	93	7	84	16	0.89	0.11	0.93	0.84	0.16	0.07	0.85	0.92	0.888
Model 0.4.300	95	5	84	16	0.9	0.1	0.95	0.84	0.16	0.05	0.86	0.94	0.903
Model 0.4.400	95	5	84	16	0.9	0.1	0.95	0.84	0.16	0.05	0.86	0.94	0.903
Model 0.4.500	97	3	84	16	0.91	0.09	0.97	0.84	0.16	0.03	0.86	0.97	0.912
Model 0.4.1000	100	0	85	15	0.93	0.07	1	0.85	0.15	0	0.87	1	0.93
Model 0.4.5000	100	0	85	15	0.93	0.07	1	0.85	0.15	0	0.87	1	0.93
Model 0.4.10000	100	0	86	14	0.93	0.07	1	0.86	0.14	0	0.88	1	0.936

Ek 2 : Etkisiz Kelimelerin Öğrenme Kümesinde Bulunmadığı Modellerin Farklı Özellik Vektörleri ile Analizi

Yöntem (Rbf Destek Vektör Makineleri)	dp	yn	dn	yp	Doğruluk	Hata Oranı	TPR	SPC	FPR	FNR	PPV	NPV	F-ölçütü
Model 1	100	0	58	42	0.79	0.21	1	0.58	0.42	0	0.7	1	0.824
Model 1.1.100	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 1.1.200	100	0	82	18	0.91	0.09	1	0.82	0.18	0	0.85	1	0.919
Model 1.1.300	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 1.1.400	100	0	81	19	0.91	0.09	1	0.81	0.19	0	0.84	1	0.913
Model 1.1.500	100	0	81	19	0.91	0.09	1	0.81	0.19	0	0.84	1	0.913
Model 1.1.1000	100	0	75	25	0.88	0.12	1	0.75	0.25	0	0.8	1	0.889
Model 1.1.5000	100	0	78	22	0.89	0.11	1	0.78	0.22	0	0.82	1	0.901
Model 1.1.10000	100	0	78	22	0.89	0.11	1	0.78	0.22	0	0.82	1	0.901
Yöntem (Lineer Destek Vektör Makineleri)	dp	yn	dn	yp	Doğruluk	Hata Oranı	TPR	SPC	FPR	FNR	PPV	NPV	F-ölçütü
Model 1.2	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 1.2.100	60	40	93	7	0.77	0.23	0.6	0.93	0.07	0.4	0.9	0.7	0.72
Model 1.2.200	65	35	94	6	0.8	0.2	0.65	0.94	0.06	0.35	0.92	0.73	0.762
Model 1.2.300	71	29	96	4	0.84	0.16	0.71	0.96	0.04	0.29	0.95	0.77	0.813
Model 1.2.400	86	14	88	12	0.87	0.13	0.86	0.88	0.12	0.14	0.88	0.86	0.87
Model 1.2.500	91	9	86	14	0.89	0.11	0.91	0.86	0.14	0.09	0.87	0.91	0.89
Model 1.2.1000	**99**	**1**	**86**	**14**	**0.93**	**0.07**	**0.99**	**0.86**	**0.14**	**0.01**	**0.88**	**0.99**	**0.932**
Model 1.2.5000	100	0	85	15	0.93	0.07	1	0.85	0.15	0	0.87	1	0.93
Model 1.2.10000	100	0	85	15	0.93	0.07	1	0.85	0.15	0	0.87	1	0.93
Yöntem (Bernoulli Naive Bayes)	dp	yn	dn	yp	Doğruluk	Hata Oranı	TPR	SPC	FPR	FNR	PPV	NPV	F-ölçütü
Model 1.3	***39***	***61***	***99***	***1***	***0.69***	***0.31***	***0.39***	***0.99***	***0.01***	***0.61***	***0.98***	***0.62***	***0.558***
Model 1.3.100	100	0	83	17	0.92	0.08	1	0.84	0.16	0	0.86	1	0.925
Model 1.3.200	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 1.3.300	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 1.3.400	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 1.3.500	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 1.3.1000	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 1.3.5000	100	0	83	17	0.92	0.08	1	0.83	0.17	0	0.85	1	0.919
Model 1.3.10000	100	0	85	15	0.93	0.07	1	0.85	0.15	0	0.87	1	0.93
Yöntem (Multinomial Naive Bayes)	dp	yn	dn	yp	Doğruluk	Hata Oranı	TPR	SPC	FPR	FNR	PPV	NPV	F-ölçütü
Model 1.4	***97***	***3***	***91***	***9***	***0.94***	***0.06***	***0.97***	***0.91***	***0.09***	***0.03***	***0.92***	***0.97***	***0.944***
Model 1.4.100	76	24	86	14	0.81	0.19	0.76	0.86	0.14	0.24	0.84	0.78	0.798
Model 1.4.200	86	14	84	16	0.85	0.15	0.86	0.84	0.16	0.14	0.84	0.86	0.85
Model 1.4.300	89	11	84	16	0.87	0.13	0.89	0.84	0.16	0.11	0.85	0.88	0.87
Model 1.4.400	95	5	84	16	0.9	0.1	0.95	0.84	0.16	0.05	0.86	0.94	0.903
Model 1.4.500	97	3	84	16	0.91	0.09	0.97	0.84	0.16	0.03	0.86	0.97	0.912
Model 1.4.1000	100	0	86	14	0.93	0.07	1	0.86	0.14	0	0.88	1	0.936
Model 1.4.5000	100	0	86	14	0.93	0.07	1	0.86	0.14	0	0.88	1	0.936
Model 1.4.10000	100	0	86	14	0.93	0.07	1	0.86	0.14	0	0.88	1	0.936

Ek 3 : Multinomial NB Model 0.4 Sınıflandırma Sonucu

Multinomial NB Etkisiz Kelimeler Mevcut			
Dosya Adı	Öngörü	Dosya Adı	Öngörü
n300.txt	normal	s300.txt	spam
n301.txt	normal	s301.txt	spam
n302.txt	normal	s302.txt	spam
n303.txt	normal	s303.txt	spam
n304.txt	normal	s304.txt	spam
n305.txt	normal	s305.txt	spam
n306.txt	normal	s306.txt	spam
n307.txt	normal	s307.txt	spam
n308.txt	normal	s308.txt	spam
n309.txt	normal	s309.txt	spam
n310.txt	normal	s310.txt	spam
n311.txt	normal	s311.txt	spam
n312.txt	normal	s312.txt	spam
n313.txt	normal	s313.txt	spam
n314.txt	normal	s314.txt	spam
n315.txt	normal	s315.txt	spam
n316.txt	normal	s316.txt	spam
n317.txt	normal	s317.txt	spam
n318.txt	normal	s318.txt	spam
n319.txt	normal	s319.txt	spam
n320.txt	normal	s320.txt	spam
n321.txt	normal	s321.txt	spam
n322.txt	normal	s322.txt	spam
n323.txt	normal	s323.txt	spam
n324.txt	normal	s324.txt	spam
n325.txt	normal	s325.txt	spam
n326.txt	normal	s326.txt	spam
n327.txt	normal	s327.txt	spam
n328.txt	normal	s328.txt	spam
n329.txt	normal	s329.txt	spam
n330.txt	normal	s330.txt	spam
n331.txt	normal	s331.txt	spam
n332.txt	normal	s332.txt	spam
n333.txt	normal	s333.txt	spam
n334.txt	normal	s334.txt	spam
n335.txt	normal	s335.txt	spam
n336.txt	normal	s336.txt	spam
n337.txt	normal	s337.txt	spam
n338.txt	normal	s338.txt	spam
n339.txt	normal	s339.txt	spam
n340.txt	normal	s340.txt	spam
n341.txt	normal	s341.txt	spam
n342.txt	normal	s342.txt	spam
n343.txt	normal	s343.txt	spam
n344.txt	normal	s344.txt	spam
n345.txt	normal	s345.txt	spam
n346.txt	normal	s346.txt	spam
n347.txt	normal	s347.txt	spam
n348.txt	normal	s348.txt	spam
n349.txt	normal	s349.txt	spam

n350.txt	normal	s350.txt	spam
n351.txt	normal	s351.txt	spam
n352.txt	normal	s352.txt	spam
n353.txt	normal	s353.txt	spam
n354.txt	normal	s354.txt	spam
n355.txt	normal	s355.txt	spam
n356.txt	normal	s356.txt	spam
n357.txt	normal	s357.txt	spam
n358.txt	normal	s358.txt	spam
n359.txt	normal	s359.txt	spam
n360.txt	normal	s360.txt	spam
n361.txt	normal	s361.txt	spam
n362.txt	normal	s362.txt	spam
n363.txt	normal	s363.txt	spam
n364.txt	normal	s364.txt	spam
n365.txt	normal	s365.txt	spam
n366.txt	normal	s366.txt	spam
n367.txt	normal	s367.txt	spam
n368.txt	normal	s368.txt	spam
n369.txt	normal	s369.txt	spam
n370.txt	normal	s370.txt	spam
n371.txt	normal	s371.txt	spam
n372.txt	normal	s372.txt	spam
n373.txt	normal	s373.txt	spam
n374.txt	normal	s374.txt	spam
n375.txt	normal	s375.txt	spam
n376.txt	normal	s376.txt	spam
n377.txt	normal	s377.txt	spam
n378.txt	normal	s378.txt	spam
n379.txt	normal	s379.txt	spam
n380.txt	normal	s380.txt	spam
n381.txt	normal	s381.txt	spam
n382.txt	normal	s382.txt	normal
n383.txt	normal	s383.txt	spam
n384.txt	normal	s384.txt	spam
n385.txt	normal	s385.txt	spam
n386.txt	normal	s386.txt	spam
n387.txt	normal	s387.txt	normal
n388.txt	normal	s388.txt	spam
n389.txt	normal	s389.txt	normal
n390.txt	normal	s390.txt	normal
n391.txt	normal	s391.txt	spam
n392.txt	normal	s392.txt	spam
n393.txt	normal	s393.txt	normal
n394.txt	normal	s394.txt	normal
n395.txt	normal	s395.txt	normal
n396.txt	normal	s396.txt	normal
n397.txt	normal	s397.txt	normal
n398.txt	normal	s398.txt	spam
n399.txt	normal	s399.txt	spam
Pozitif Adedi	**100**	**Negatif Adedi**	**91**

Ek 4 : Bernoulli NB Model 0.3 Sınıflandırma Sonucu

Bernoulli NB Etkisiz Kelimeler Mevcut			
Dosya Adı	**Öngörü**	**Dosya Adı**	**Öngörü**
n300.txt	spam	s300.txt	spam
n301.txt	spam	s301.txt	spam
n302.txt	spam	s302.txt	spam
n303.txt	spam	s303.txt	spam
n304.txt	spam	s304.txt	spam
n305.txt	normal	s305.txt	spam
n306.txt	spam	s306.txt	spam
n307.txt	spam	s307.txt	spam
n308.txt	spam	s308.txt	spam
n309.txt	spam	s309.txt	spam
n310.txt	spam	s310.txt	spam
n311.txt	spam	s311.txt	spam
n312.txt	spam	s312.txt	spam
n313.txt	spam	s313.txt	spam
n314.txt	normal	s314.txt	spam
n315.txt	spam	s315.txt	spam
n316.txt	spam	s316.txt	spam
n317.txt	normal	s317.txt	spam
n318.txt	spam	s318.txt	spam
n319.txt	normal	s319.txt	spam
n320.txt	normal	s320.txt	spam
n321.txt	normal	s321.txt	spam
n322.txt	normal	s322.txt	spam
n323.txt	spam	s323.txt	spam
n324.txt	normal	s324.txt	spam
n325.txt	normal	s325.txt	spam
n326.txt	normal	s326.txt	spam
n327.txt	spam	s327.txt	spam
n328.txt	spam	s328.txt	spam
n329.txt	spam	s329.txt	spam
n330.txt	normal	s330.txt	spam
n331.txt	normal	s331.txt	spam
n332.txt	normal	s332.txt	spam
n333.txt	spam	s333.txt	spam
n334.txt	normal	s334.txt	spam
n335.txt	spam	s335.txt	spam
n336.txt	spam	s336.txt	spam
n337.txt	normal	s337.txt	spam
n338.txt	spam	s338.txt	spam
n339.txt	spam	s339.txt	spam
n340.txt	normal	s340.txt	spam
n341.txt	normal	s341.txt	spam
n342.txt	spam	s342.txt	spam
n343.txt	spam	s343.txt	spam
n344.txt	normal	s344.txt	spam
n345.txt	spam	s345.txt	spam
n346.txt	spam	s346.txt	spam
n347.txt	spam	s347.txt	spam
n348.txt	spam	s348.txt	spam

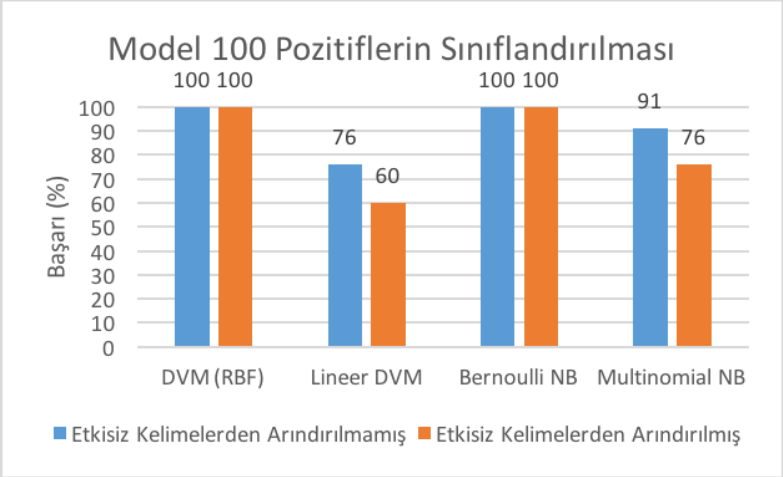
n349.txt	normal	s349.txt	spam
n350.txt	spam	s350.txt	spam
n351.txt	spam	s351.txt	spam
n352.txt	normal	s352.txt	spam
n353.txt	spam	s353.txt	spam
n354.txt	spam	s354.txt	spam
n355.txt	spam	s355.txt	spam
n356.txt	spam	s356.txt	spam
n357.txt	spam	s357.txt	spam
n358.txt	normal	s358.txt	spam
n359.txt	normal	s359.txt	spam
n360.txt	spam	s360.txt	spam
n361.txt	normal	s361.txt	spam
n362.txt	spam	s362.txt	spam
n363.txt	spam	s363.txt	spam
n364.txt	normal	s364.txt	spam
n365.txt	spam	s365.txt	spam
n366.txt	spam	s366.txt	spam
n367.txt	normal	s367.txt	spam
n368.txt	spam	s368.txt	spam
n369.txt	spam	s369.txt	spam
n370.txt	normal	s370.txt	spam
n371.txt	spam	s371.txt	spam
n372.txt	spam	s372.txt	spam
n373.txt	normal	s373.txt	spam
n374.txt	normal	s374.txt	spam
n375.txt	normal	s375.txt	spam
n376.txt	spam	s376.txt	spam
n377.txt	normal	s377.txt	spam
n378.txt	normal	s378.txt	spam
n379.txt	normal	s379.txt	spam
n380.txt	normal	s380.txt	spam
n381.txt	normal	s381.txt	spam
n382.txt	normal	s382.txt	normal
n383.txt	spam	s383.txt	spam
n384.txt	spam	s384.txt	spam
n385.txt	normal	s385.txt	spam
n386.txt	normal	s386.txt	spam
n387.txt	spam	s387.txt	spam
n388.txt	normal	s388.txt	spam
n389.txt	normal	s389.txt	spam
n390.txt	spam	s390.txt	spam
n391.txt	normal	s391.txt	spam
n392.txt	spam	s392.txt	spam
n393.txt	normal	s393.txt	spam
n394.txt	normal	s394.txt	spam
n395.txt	spam	s395.txt	spam
n396.txt	spam	s396.txt	spam
n397.txt	spam	s397.txt	spam
n398.txt	spam	s398.txt	spam
n399.txt	spam	s399.txt	spam
Pozitif Adedi	**42**	**Negatif Adedi**	**99**

Ek 5 : Bernoulli NB Model 0.3.10000

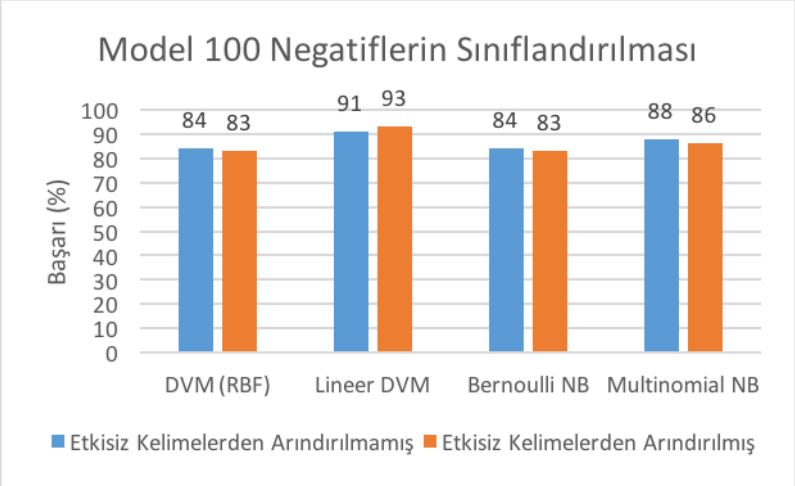
Bernoulli NB Etkisiz Kelimeler Mevcut			
Dosya Adı	**Öngörü**	**Dosya Adı**	**Öngörü**
n300.txt	normal	s300.txt	spam
n301.txt	normal	s301.txt	spam
n302.txt	normal	s302.txt	spam
n303.txt	normal	s303.txt	spam
n304.txt	normal	s304.txt	spam
n305.txt	normal	s305.txt	spam
n306.txt	normal	s306.txt	spam
n307.txt	normal	s307.txt	spam
n308.txt	normal	s308.txt	spam
n309.txt	normal	s309.txt	spam
n310.txt	normal	s310.txt	spam
n311.txt	normal	s311.txt	spam
n312.txt	normal	s312.txt	spam
n313.txt	normal	s313.txt	spam
n314.txt	normal	s314.txt	spam
n315.txt	normal	s315.txt	spam
n316.txt	normal	s316.txt	spam
n317.txt	normal	s317.txt	spam
n318.txt	normal	s318.txt	spam
n319.txt	normal	s319.txt	spam
n320.txt	normal	s320.txt	spam
n321.txt	normal	s321.txt	spam
n322.txt	normal	s322.txt	spam
n323.txt	normal	s323.txt	spam
n324.txt	normal	s324.txt	spam
n325.txt	normal	s325.txt	spam
n326.txt	normal	s326.txt	spam
n327.txt	normal	s327.txt	spam
n328.txt	normal	s328.txt	spam
n329.txt	normal	s329.txt	spam
n330.txt	normal	s330.txt	spam
n331.txt	normal	s331.txt	spam
n332.txt	normal	s332.txt	spam
n333.txt	normal	s333.txt	spam
n334.txt	normal	s334.txt	spam
n335.txt	normal	s335.txt	spam
n336.txt	normal	s336.txt	spam
n337.txt	normal	s337.txt	spam
n338.txt	normal	s338.txt	spam
n339.txt	normal	s339.txt	spam
n340.txt	normal	s340.txt	spam
n341.txt	normal	s341.txt	spam
n342.txt	normal	s342.txt	spam
n343.txt	normal	s343.txt	spam
n344.txt	normal	s344.txt	spam
n345.txt	normal	s345.txt	spam
n346.txt	normal	s346.txt	spam
n347.txt	normal	s347.txt	spam
n348.txt	normal	s348.txt	spam
n349.txt	normal	s349.txt	spam

n350.txt	normal	s350.txt	normal
n351.txt	normal	s351.txt	spam
n352.txt	normal	s352.txt	spam
n353.txt	normal	s353.txt	spam
n354.txt	normal	s354.txt	spam
n355.txt	normal	s355.txt	spam
n356.txt	normal	s356.txt	spam
n357.txt	normal	s357.txt	spam
n358.txt	normal	s358.txt	spam
n359.txt	normal	s359.txt	spam
n360.txt	normal	s360.txt	spam
n361.txt	normal	s361.txt	spam
n362.txt	normal	s362.txt	spam
n363.txt	normal	s363.txt	spam
n364.txt	normal	s364.txt	spam
n365.txt	normal	s365.txt	spam
n366.txt	normal	s366.txt	spam
n367.txt	normal	s367.txt	spam
n368.txt	normal	s368.txt	spam
n369.txt	normal	s369.txt	spam
n370.txt	normal	s370.txt	spam
n371.txt	normal	s371.txt	spam
n372.txt	normal	s372.txt	spam
n373.txt	normal	s373.txt	spam
n374.txt	normal	s374.txt	spam
n375.txt	normal	s375.txt	spam
n376.txt	normal	s376.txt	spam
n377.txt	normal	s377.txt	spam
n378.txt	normal	s378.txt	spam
n379.txt	normal	s379.txt	spam
n380.txt	normal	s380.txt	spam
n381.txt	normal	s381.txt	spam
n382.txt	normal	s382.txt	normal
n383.txt	normal	s383.txt	spam
n384.txt	normal	s384.txt	normal
n385.txt	normal	s385.txt	spam
n386.txt	normal	s386.txt	spam
n387.txt	normal	s387.txt	normal
n388.txt	normal	s388.txt	spam
n389.txt	normal	s389.txt	normal
n390.txt	normal	s390.txt	normal
n391.txt	normal	s391.txt	normal
n392.txt	normal	s392.txt	spam
n393.txt	normal	s393.txt	normal
n394.txt	normal	s394.txt	normal
n395.txt	normal	s395.txt	normal
n396.txt	normal	s396.txt	spam
n397.txt	normal	s397.txt	normal
n398.txt	normal	s398.txt	spam
n399.txt	normal	s399.txt	normal
Pozitif Adedi	**100**	**Negatif Adedi**	**88**

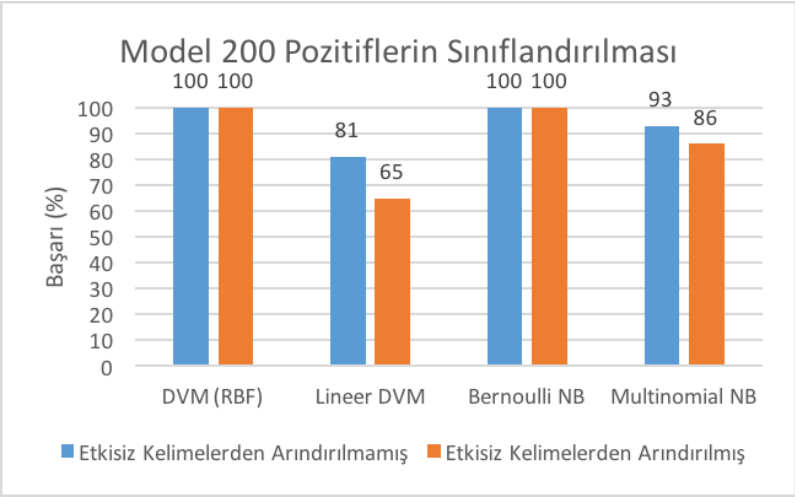
Ek 6 : Etkisiz kelimelerin eğitim kümesinden arındırılıp arındırılmama durumlarının sınıflandırma başarısına etkisi



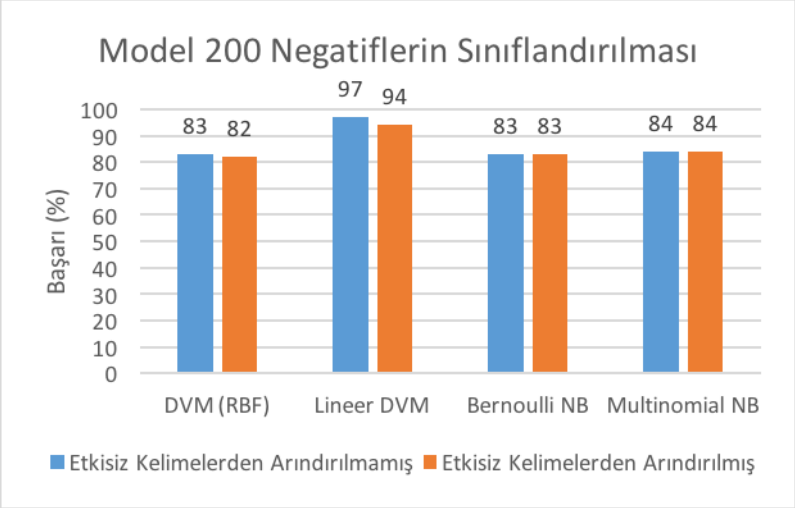
Şekil 15 : İstenilen e-postaların Model 100 ile sınıflandırma başarısı



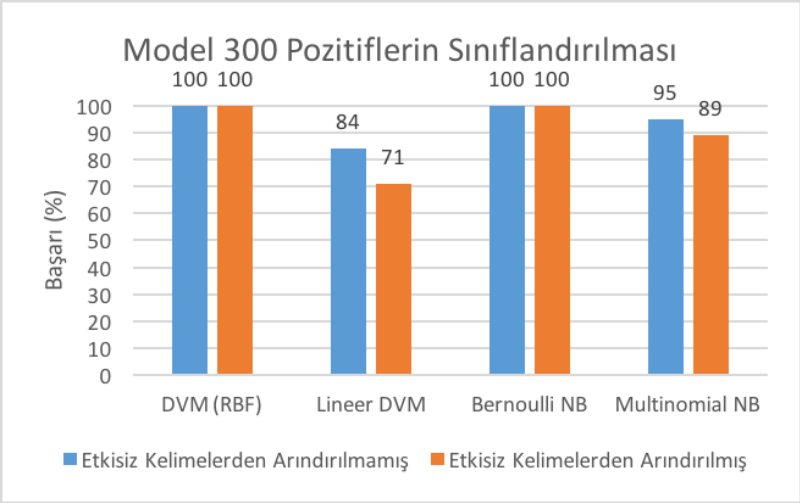
Şekil 16 : İstenmeyen e-postaların Model 100 ile sınıflandırma başarısı



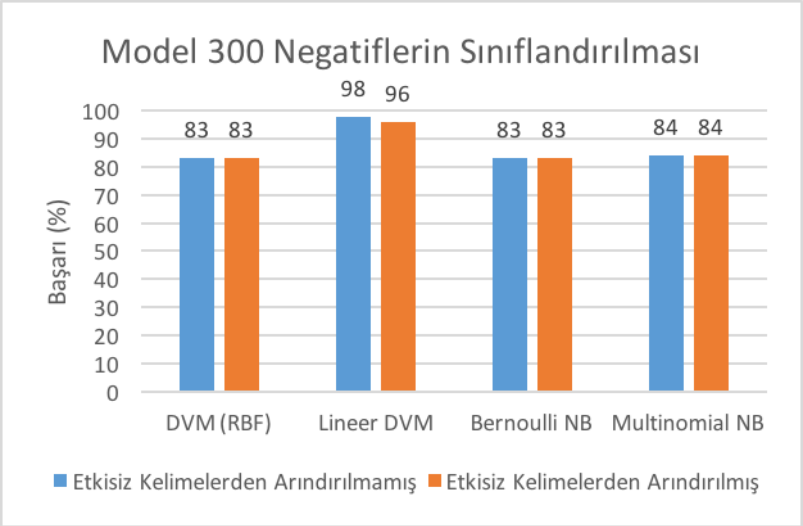
Şekil 17 : İstenilen e-postaların Model 200 ile sınıflandırma başarısı



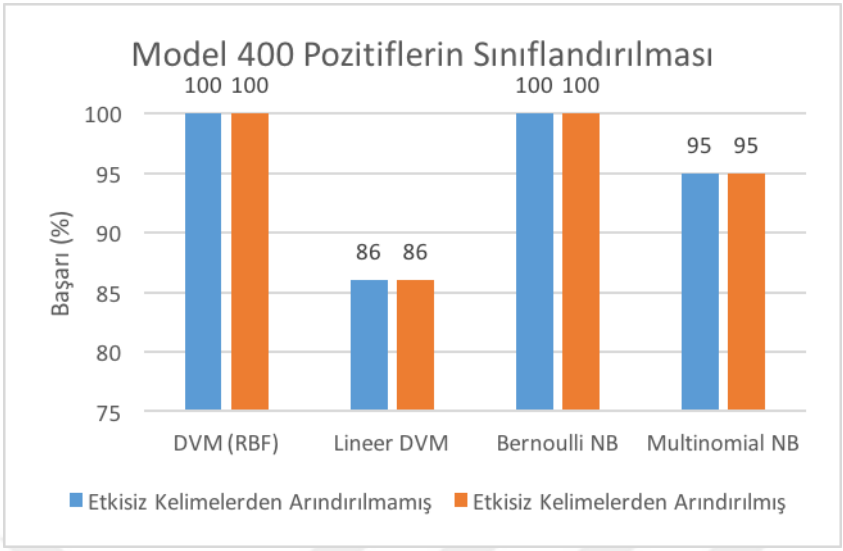
Şekil 18 : İstenmeyen e-postaların Model 200 ile sınıflandırma başarısı



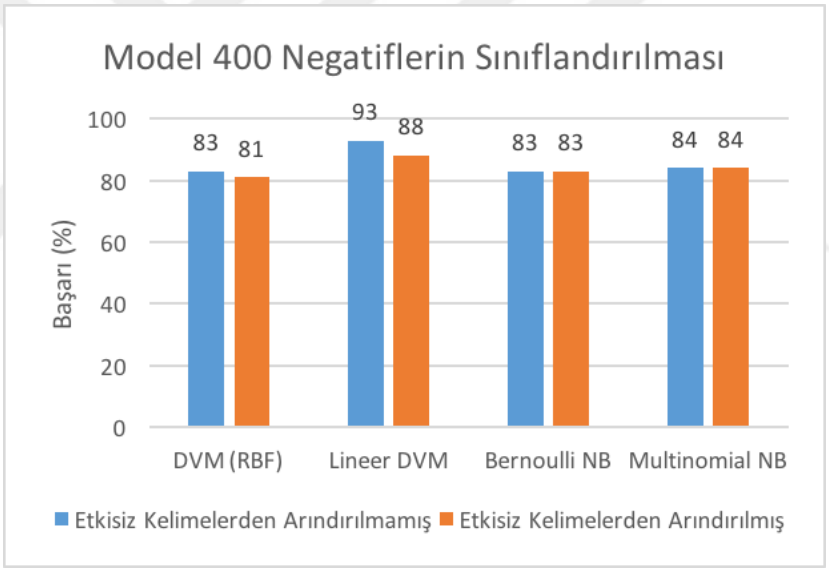
Şekil 19 : İstenilen e-postaların model 300 ile sınıflandırılma başarısı



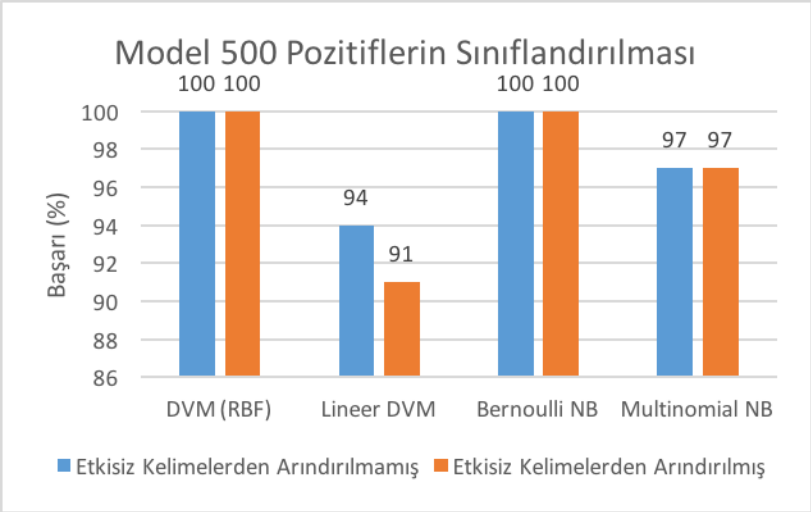
Şekil 20 : İstenmeyen e-postaların model 300 ile sınıflandırılma başarısı



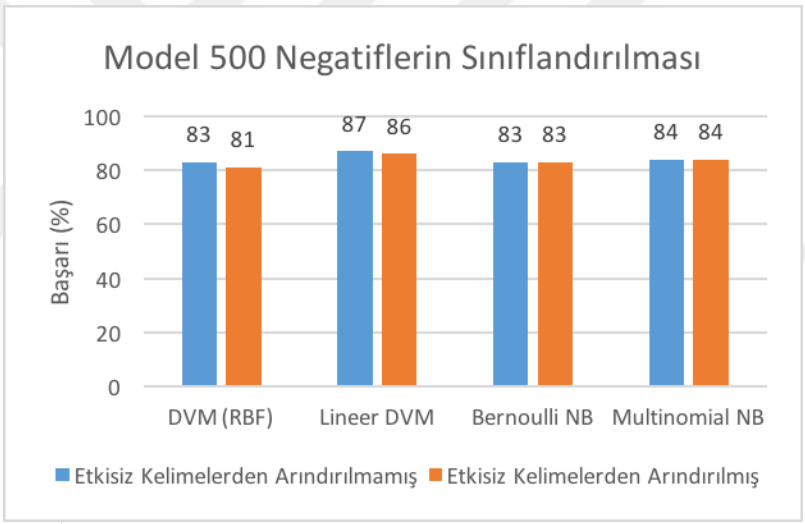
Şekil 21 : İstenilen e-postaların model 400 ile sınıflandırılma başarısı



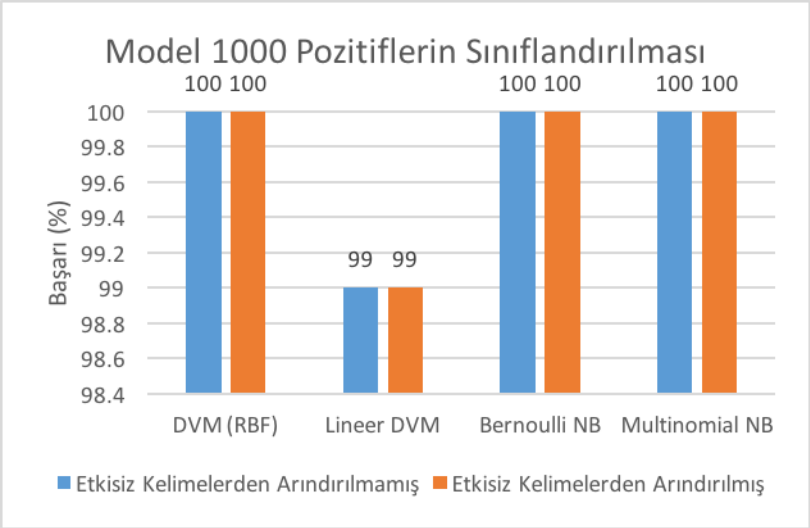
Şekil 22 : İstenmeyen e-postaların model 400 ile sınıflandırılma başarısı



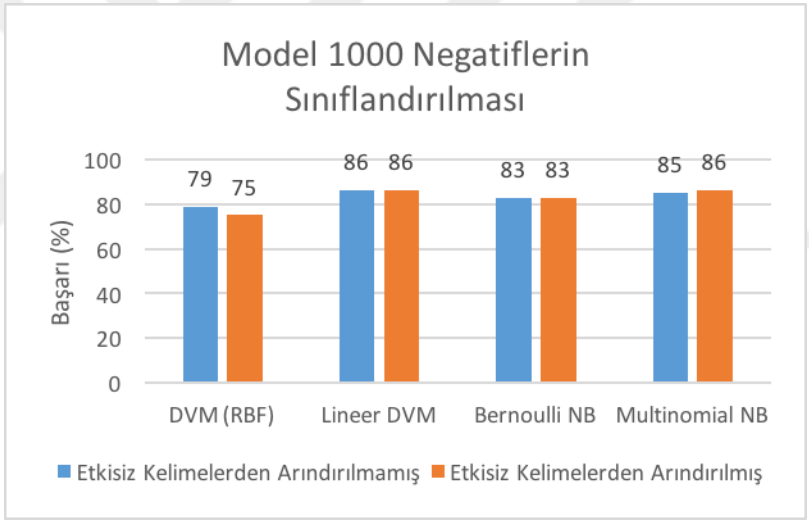
Şekil 23 : İstenilen e-postaların model 500 ile sınıflandırılma başarısı



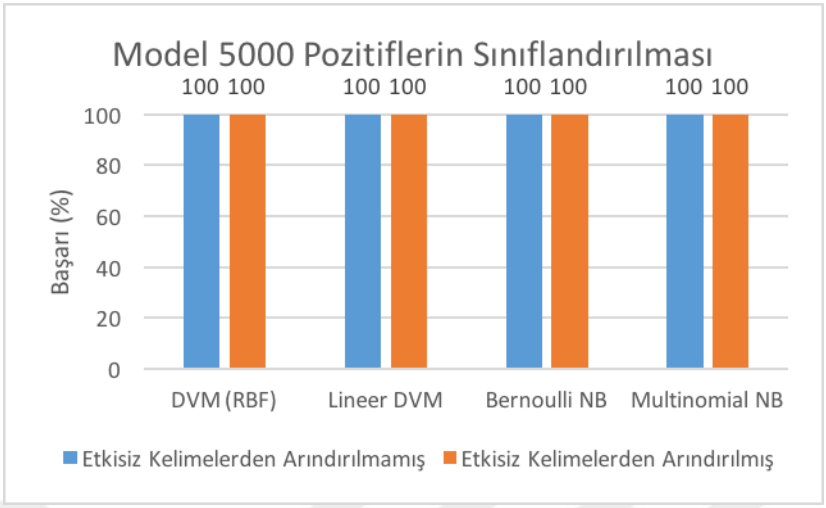
Şekil 24 : İstenmeyen e-postaların model 500 ile sınıflandırılma başarısı



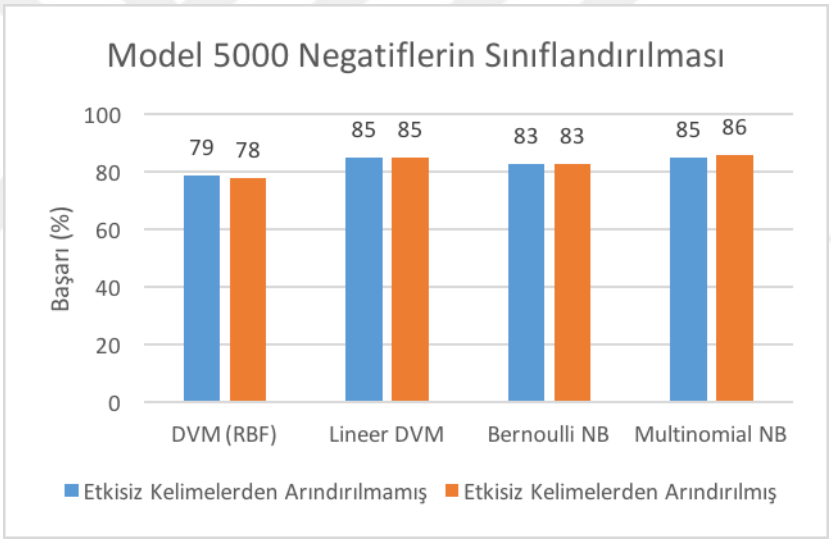
Şekil 25 : İstenilen e-postaların model 1000 ile sınıflandırılma başarısı



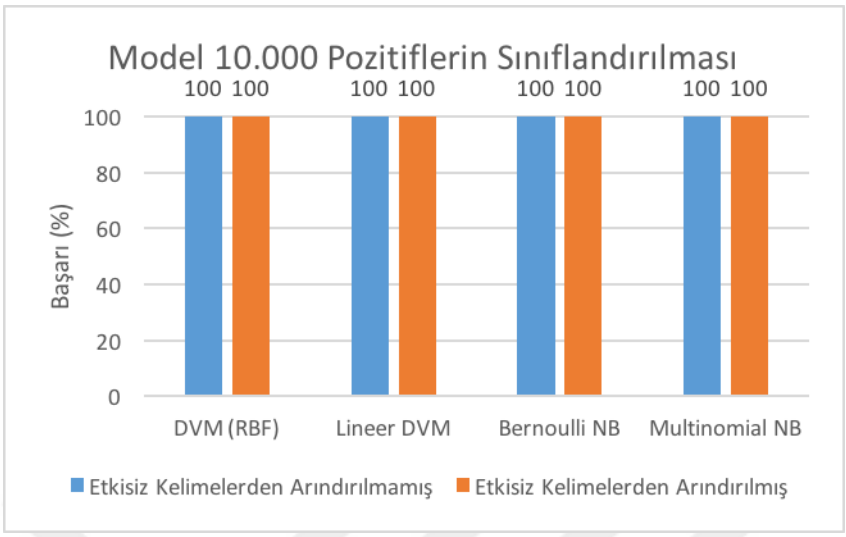
Şekil 26 : İstenmeyen e-postaların model 10000 ile sınıflandırılma başarısı



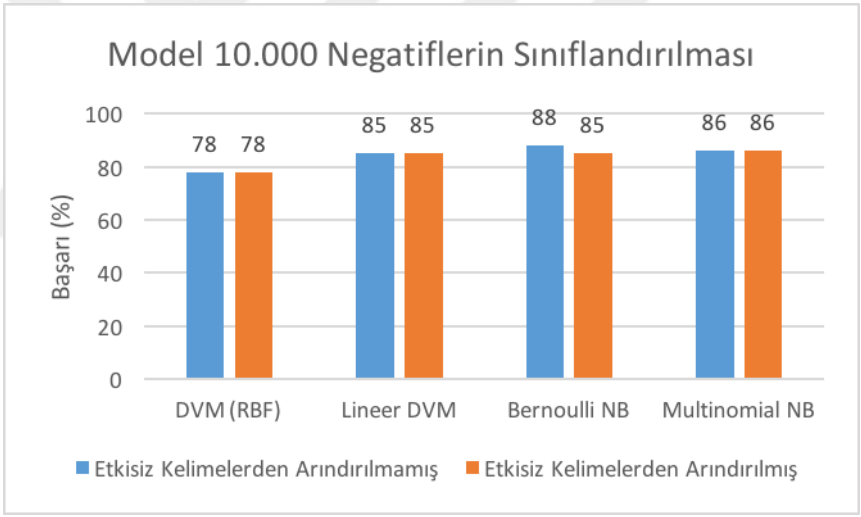
Şekil 27 : İstenen e-postaların model 5.000 ile sınıflandırılma başarısı



Şekil 28 : İstenmeyen e-postaların model 5.000 ile sınıflandırılma başarısı



Şekil 29 : İstenen e-postaların model 10.000 ile sınıflandırılma başarısı



Şekil 30 : İstenmeyen e-postaların model 10.000 ile sınıflandırılma başarısı

KAYNAKÇA

Aldatıcı E-Posta Mesajları Hk. *- Bilgi İşlem Daire Başkanlığı - Marmara Üniversitesi*. (2015, 10 07). 6 27, 2017 tarihinde bidb.marmara.edu.tr: https://bidb.marmara.edu.tr/notice/aldatici-e-posta-mesajlari-hk/ adresinden alındı

Allman, E. (2003). ***Spam, spam, spam, spam, spam, the ftc, and spam***. *Queue , 1* (6), 62-69.

Amasyalı, M. F. (2006). ***Farklı Özellik Vektörleri ile Türkçe Dokümanların Yazarlarının Belirlenmesi***. *Türkiye Yapay Sinir Ağları Sempozyumu* .

Androutsopoulos, I. K. (2000). ***An Evaluation of Naive Bayesian Anti-Spam Filtering***. *Proceedings of the workshop on Machine Learning in the New Information Age, G. Potamias, V. Moustakis and M. van Someren (eds.), 11th European Conference on Machine Learning* , (s. 9-17). Barcelona, Spain.

Androutsopoulos, I. K. (2000, Temmuz 24). ***An experimental comparison of naive Bayesian and keyword-based anti-spam filtering with personal e-mail messages.*** *Proceeding SIGIR '00 Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval* , 160-167.

Balaban E.M., K. E. (2015). ***Veri Madenciliği ve Makine Öğremesi***. İstanbul, Taksim/Beyoğlu, Türkiye: Çağlayan Kitabevi.

Bayramlı, B. (Dü.). (2015, 12 0). ***Regresyon, Ridge, Lasso, Çapraz Sağlama, Regülarize Etmek*** . 5 18, 2018 tarihinde https://burakbayramli.github.io/dersblog/stat/stat_regular/regresyon_ridge_lasso_capraz_saglama_regularize_etmek.html. adresinden alındı

Bellhouse, D. R. (2004). The Reverend Thomas Bayes, FRS: ***a biography to celebrate the tercentenary of his birth.*** *Statistical Science , 19* (1), 3-43.

Bernstein, M. N. (2017, 3 28). ***RBFKernel***. 7 4, 2017 tarihinde http://pages.cs.wisc.edu/~matthewb/pages/notes/pdf/svms/RBFKernel.pdf adresinden alındı

Bluszcz, J. F. (2016, 7 8). ***Application of Support Vector Machine Algorithm in E-Mail Spam Filtering***. 2017, 6, 8.

Boser, B. E. (1992, 7). ***A training algorithm for optimal margin classifiers***. *In Proceedings of the fifth annual workshop on Computational learning theory. ACM.* , 144-152.

Bottou, L. C. (1994). ***Comparison of classifier methods: a case study in handwritten digit recognition***. *Pattern Recognition, 1994. Vol. 2-Conference B: Computer Vision &*

Image Processing., Proceedings of the 12th IAPR International. Conference on. 2, s. 77-82. IEEE.

Bougé, K. (2015, 07 01). (X. Ren, Düzenleyen) 08 01, 2018 tarihinde https://github.com/shanzhenren: https://github.com/shanzhenren/ClusType/blob/master/candidate_generation/stopwords/stop-words-collection-2014-02-24/stop-words/stop-words_turkish_1_tr.txt adresinden alındı

Broomhead, D. S. (1998). ***Radial basis functions, multi-variable functional interpolation and adaptive networks***. *Royal Signals and Radar Establishment Malvern (United Kingdom)* .

Can, F. K. (2008). ***Information retrieval on Turkish texts***. *Journal of the American Society for Information Science and Technology , 59* (4), s. 407-421.

Challenge/Response Authentication Services - SpamHelp. (spamhelp.org) 08 2017, 08 tarihinde spamhelp.org: http://www.spamhelp.org/services/listings/challenge-response/ adresinden alındı

Christopher D. Manning, P. R. (2008). ***Scoring, term weighting and the vector space model*** (Cilt 6). Cambridge University Press.

clrc.rhul.ac.uk. (2012, 1 1). *Royal Holloway, University of London : Department of Computer Science*. 6 8, 2017 tarihinde clrc.rhul.ac.uk: http://www.clrc.rhul.ac.uk/people/vlad/ adresinden alındı

Cohen, W. W. (1996). **Learning rules that classify e-mail**. *AAAI spring symposium on machine learning in information access , 18*, 25.

columbia.edu. (2015, 3 1). ***Vladimir Vapnik*** | *Data Science Institute*. 6 8, 2017 tarihinde columbia.edu: http://datascience.columbia.edu/vladimir-vapnik adresinden alındı

Cortes, C. &. (1995). ***Support-vector networks***. *Machine learning , 20* (3), 273-297.

Çalış, K. G. (2013, 6 1). ***Reklam İçerikli Epostaların Metin Madenciliği Yöntemleri İle Otomatik Tespiti***. *Bilişim Teknolojileri Dergisi* , 1-7.

D. Jude Hemanth, S. S. (2018). ***Computational Vision and Bio Inspired Computing*** . India: Springer.

Düşgör Gökhan. (2016, 04 26). ***Cryptolocker Virüsü*** | *İstanbul Üniversitesi*. 07 27, 2017 tarihinde buyamer.istanbul.edu.tr: http://buyamer.istanbul.edu.tr/?p=8248 adresinden alındı

Drucker, H. D. (1999). ***Support vector machines for spam categorization***. *IEEE Transactions on Neural networks , 10* (5), s. 1048-1054.

Duan, K. B. (2005). ***Which is the best multiclass SVM method? An empirical study***. . *In International workshop on multiple classifier systems* , 278-285.
Dumais, S. (1998). ***Using SVMs for text categorization***. *IEEE Intelligent Systems , 13* (4), s. 21-23.

Ergin S., E. G. (2012). ***Turkish anti-spam filtering using binary and probabilistic models***. *AWER Procedia Information Technology and Computer Science* , 1007-1012.

FAQs | Hormel Foods. (1998, 12 2). (hormelfoods.com) 7 26, 2017 tarihinde hormelfoods.com: https://www.hormelfoods.com/faqs/ adresinden alındı
Géron, A. (2017). *Hands-On Machine Learning with Scikit-Learn and TensorFlow* . O'Reilly Media.

Goldberg, Y. &. (2008, 6). ***splitSVM: fast, space-efficient, non-heuristic, polynomial kernel computation for NLP applications***. *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies* , 237-240.

Google. ***Google Code Archive - Long-term storage for Google Code Project Hosting***. 08 01, 2018 tarihinde code.google.com: https://code.google.com/archive/p/stop-words/ adresinden alındı

guinnessworldrecords.com. (2016, 2 14). ***Oldest electronic spam*** | *Guinness World Records*. 6 2, 2017 tarihinde guinnessworldrecords.com: http://www.guinnessworldrecords.com/world-records/oldest-electronic-spam adresinden alındı

Haltaş, A. a. (2016). ***Medlıne Veritabanı Üzerinde Bulunan Tıbbi Dökümanların Kanser Türlerine Göre Otomatik Sınıflandırılması.*** *International Journal of Informatics Technologies , 9* (2), 181.

Jonathan Zdziarski, W. Y. (2006). **Approaches To Phishing Identification Using Match And Probabilistic Digital Fingerprinting Techniques**.

Juafsky, D. ***Naive Bayes.*** https://web.stanford.edu/class/cs124/lec/naivebayes.pdf adresinden alındı

Jurafsky, D. (2016, 12 20). ***Naive Bayes and Text Classification.*** 6 2, 2017 tarihinde stanford.edu: http://web.stanford.edu/class/cs124/lec/naivebayes.pdf adresinden alındı

Korting, T. S. (2015, 10 25). ***How Support Vector Machines (SVM) algorithm works*** *by Thales Sehn Korting on Prezi.* 6 8, 2017 tarihinde prezi.com: https://prezi.com/jdtqiauncqww/how-support-vector-machines-svm-algorithm-works/ adresinden alındı

Law, M. (2011, 1 3). ***A Simple Introduction to Support Vector Machine**s*. 6 8, 2017 tarihinde ufl.edu/: https://www.cise.ufl.edu/class/cis4930sp11dtm/notes/intro_svm_new.pdf adresinden alındı

Levine, J. (2010). ***DNS blacklists and whitelists***. *RFC 5782* .

listendata.com. (2017, 1). ***Support Vector Machine Simplified using R.*** 6 8, 2017 tarihinde listendata.com: http://www.listendata.com/2017/01/support-vector-machine-in-r-tutorial.html adresinden alındı

Malvina Nissim, J. B. (2016, 08 05). 04 10, 2018 tarihinde ***Learning from Data Lecture 4: Vector Space Models K-Nearest Neighbor (+ Support Vector Machines***): http://esslli2016.unibz.it/wp-content/uploads/2015/10/lecture4.pdf adresinden alındı

matt komorowski - ***a history of storage cos**t*. (2014, 3 9). (mkomo.com, Prodüktör) 7 27, 2017 tarihinde mkomo.com: http://www.mkomo.com/cost-per-gigabyte adresinden alındı

Mayda, İ. a. (2016). ***Yazar Tanımada Köşe Yazısı ve Tweet'lerin Çapraz Kullanımı Cross Usage of Articles and Tweets on Author Identification.*** *IEEE* .

McCormick, C. (2013, 8 15). ***Radial Basis Function Network (RBFN) Tutorial*** *Chris McCormick* . 6 8, 2017 tarihinde mccormickml.com: http://mccormickml.com/2013/08/15/radial-basis-function-network-rbfn-tutorial/ adresinden alındı

McMichael, T. (2018, 02 09). ***SPAM detection and reverse DNS lookups***. 04 07, 2018 tarihinde microsoft.com: https://blogs.technet.microsoft.com/timmcmic/2018/02/09/office-365-spam-detection-and-reverse-dns-lookups/ adresinden alındı

Metsis, V. A. (2006, 7 27). ***Spam Filtering with Naive Bayes*** – Which Naive Bayes ? *Third Conference on Email and Anti-Spam* , 28-69.

Mihalcea, R. (2015, 04 02). *http://web.eecs.umich.edu/~mihalcea/498IR/.* 04 12, 2018 tarihinde http://web.eecs.umich.edu/: http://web.eecs.umich.edu/~mihalcea/498IR/Lectures/VectorSpaceModel.pdf adresinden alındı

Monty Python - Spam. (2007, 2 14). (youtube.com, Prodüktör) 6 2, 2016 tarihinde youtube.com: https://www.youtube.com/watch?v=anwy2MPT5RE adresinden alındı

Nahayo, Y. a. (2016). ***Birleşik modellemeli ve modellemesiz metin-bağımsız konuşmacı tanıma için SVM, K-NN ve NBC sınıflandırıcıların başarımı***. *Sakarya Üniversitesi Fen Bilimleri Enstitüsü Dergisi 20.1* , 1- 6.

Ng, A. (2015, 10 19). *http://cs229.stanford.edu/notes/cs229-notes3.pdf.* 6 8, 2017 tarihinde http://cs229.stanford.edu/notes/cs229-notes3.pdf: http://cs229.stanford.edu/notes/cs229-notes3.pdf adresinden alındı

Nicolosi, N. (2018, 11 7). ***Feature Selection Methods for Text Classification.***
Oguz Kaynar, Y. G. (2016). Oto Kodlayıcı Tabanlı Derin Öğrenme Makinaları ile Spam Tespiti.
Okur, B. Ç. (2013). ***İngiliz alfabesi kullanılarak yazılmış Türkçe metinlerin Türk alfabesine göre yeniden oluşturulması***. *Signal Processing and Communications Applications Conference (SIU)* , 1-4.

openspf.org. (2018). ***What_is_SPF***. 4 7, 2018 tarihinde openspf.org: http://www.openspf.org/FAQ/What_is_SPF adresinden alındı

Pedregosa, F. V. (2011, 10 12). ***Scikit-learn: Machine learning in Python*** . *Journal of Machine Learning Research* , 2825-2830.

Press, 2. C. (Dü.). (2009, 04 07). **Support vector machines: The linearly separable case** . 5 18, 2018 tarihinde https://nlp.stanford.edu/IR-book/html/htmledition/support-vector-machines-the-linearly-separable-case-1.html adresinden alındı

Raschka, S. (2014, 10 4). ***Naive Bayes and Text Classification*** *I.* 6 6, 2017 tarihinde arxiv.org: https://arxiv.org/pdf/1410.5329.pdf adresinden alındı

Rosasco, L. V. (2014). ***Are loss functions all the same?*** 1063-1076.

S. Ergin, E. S. (2011). ***TurkishEmail*** . 6 6, 2017 tarihinde anadolu.edu.tr: http://ceng.anadolu.edu.tr/par/TurkishEmail.zip adresinden alındı

Sahami, M. D. (1998). ***A Bayesian Approach to Filtering Junk E-Mail***. *AAAI Workshop on Learning for Text Categorization , 62*, 98-105.

sans.edu. (-). ***Honeypots: A Security Manager's Guide to Honeypots***. 6 8, 2017 tarihinde sans.edu: http://www.sans.edu/cyber-research/security-laboratory/article/honeypots-guide adresinden alındı

Schlitt, W. &. (2006). ***Sender policy framework (SPF) for authorizing use of domains in e-mail*** .

scikit-learn.org. (2016, 11 11). *sklearn.naive_bayes.BernoulliNB - scikit-learn 0.18.1 documentation.* 6 7, 2017 tarihinde scikit-learn.org: http://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.BernoulliNB.html adresinden alındı

scikit-learn.org. (2016, 11 11). *sklearn.naive_bayes.MultinomialNB - scikit-learn 0.18.1 documentation.* 6 8, 2017 tarihinde scikit-learn.org: http://scikit-

learn.org/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html adresinden alındı

scikit-learn.org. (2016, 11 11). *sklearn.svm.LinearSVC — scikit-learn 0.18.1 documentation.* 6 8, 2017 tarihinde scikit-learn.org: http://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC adresinden alındı

scikit-learn.org. (2016, 11 11). *sklearn.svm.SVC - scikit-learn 0.18.1 documentation.* 6 8, 2017 tarihinde scikit-learn.org: http://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html adresinden alındı

scipy.org. (tarih yok). *numpy.argmax*. scipy.org: https://docs.scipy.org/doc/numpy/reference/generated/numpy.argmax.html adresinden alındı

Segal, R. e. (2004). SpamGuru: ***An Enterprise Anti-Spam Filtering System***. *CEAS* .

Shekhar, A. (2018, 03 10). ***What Are L1 and L2 Loss Functions****?* 5 17, 2018 tarihinde etslearnai.com: https://www.letslearnai.com/2018/03/10/what-are-l1-and-l2-loss-functions.html adresinden alındı

Sivanadyan, T. (2003). ***Spam? not any more! detecting spam emails using neural networks***. University of Wisconsin. University of Wisconsin.

SPAM® Family Of Products. (1998, 12 2). (hormelfoods.com, Prodüktör) 6 2, 2017 tarihinde hormelfoods.com: http://www.hormelfoods.com/Brands/BrandWall/SPAM-Family-of-Products adresinden alındı

spam.com. (2012, 3 1). *Canned Meat Varieties & Snack Flavors* | ***SPAM® Meat***. 6 2, 2017 tarihinde spam.com: http://www.spam.com/varieties adresinden alındı

stanford.edu. (2009, 4 7). ***Naive Bayes text classification***. 6 6, 2017 tarihinde stanford.edu: https://nlp.stanford.edu/IR-book/html/htmledition/naive-bayes-text-classification-1.html adresinden alındı

stanford.edu. (2009, 4 7). ***The Bernoulli model.*** 6 8, 2017 tarihinde stanford.edu: https://nlp.stanford.edu/IR-book/html/htmledition/the-bernoulli-model-1.html adresinden alındı

Stephen Boyd, L. V. (2004). ***Convex Optimization***. Cambridge University Press 2004.

ŞEKER, Ş. E. (2008, 5 26). ***Özetleme Fonksiyonları*** *(Hash Function) & Bilgisayar Kavramları* . 7 30, 2017 tarihinde sadievrenseker.com: http://bilgisayarkavramlari.sadievrenseker.com/2008/05/26/ozetleme-fonksiyonlari-hash-function/ adresinden alındı

Tüfekci, P. U. (2012). ***Text classification of web based news articles by using Turkish grammatical features***. *In Signal Processing and Communications Applications Conference (SIU)* (s. 1-4). IEEE.

Türkmen, H. İ. (2011). ***Konuşma Dili Kullanılarak Demografik Bilgilerin Sınıflandırılması Demographic Information Classification Exploliting Spoke***. *19th Signal Processing and Communications Applications Conference.* IEEE.

Taşkın KAVZOĞLU, İ. Ç. (2010, 07 01). ***Destek Vektör Makineleri ile Uydu Görüntülerinin Sınıflandırılmasında Kernel Fonksiyonlarının Etkilerinin İncelenmes***i . *Harita Dergisi* (144), s. 73-82.

Taha, H. A. (2000). **Yöneylem Araştırması**. (Ş. A. Esnaf, Çev.) İstanbul: Literatür Yayıncılık Dağıtım.

Tantuğ, A. a. (2006). ***Performance Analysis of Naïve Bayes Classification, Support Vector Machines and Neural Networks for Spam Categorization***. *Applied Soft Computing Technologies: The Challenge of Complexity* , s. 495-504.

Tekin, R. Y. (2013). ***K-means ve YSA temelli Hibrit Bir Model ile Epileptik EEG İşaretlerinin Sınıflandırılması***. *Elektrik Elektronik Bilgisayar Semp. Elazığ.* Fırat Üniversitesi.

Ting-Fan Wu, C.-J. L. (2004, 8 04). ***Probability Estimates for Multi-class Classification by Pairwise Coupling***. *Journal of Machine Learning Research* , s. 975-1005.

totalsend.com. (2014, 5 8). ***A Beginners Guide to Email Blacklisting***. totalsend.com: https://www.totalsend.com/a-beginners-guide-to-email-blacklisting adresinden alındı

Wu, C. H. (2009). **Behavior-based spam detection using a hybrid method of rule-based techniques and neural networks**. *Expert Systems with Applications , 36* (3), 4321-4330.

WW II Encyclopedia -- SPAM. (2002, 11 7). (skylighters.org) 6 2, 2017 tarihinde skylighters.org: http://www.skylighters.org/encyclopedia/spam.html adresinden alındı

yardimmerkezi.net. (2013, 9 23). *Yardım Merkezi - **MX kaydı, SPF kaydı, PTR kaydı nedir? Ne işe yarar** ?* 6 2, 2017 tarihinde yardimmerkezi.net: https://www.yardimmerkezi.net/mx-kaydi-spf-kaydi-ptr-kaydi-nedir-ne-ise-yarar/ adresinden alındı

Zink, T. (2016, 07 05)**. *The outbound IP and HELO format for Office 365*** . 04 06, 2018 tarihinde microsoft.com: https://blogs.msdn.microsoft.com/tzink/2016/07/15/the-outbound-ip-and-helo-format-for-office-365/ adresinden alındı