YAZILIM GELİŞTİRME TALEPLERİNİN METİN MADENCİLİĞİ İLE SINIFLANDIRILMASI VE ÖNCELİKLENDİRİLMESİ

Murat Can TEKİN

YÜKSEK LİSANS TEZİ
Bilgisayar Mühendisliği Anabilim Dalı
Danışman: Dr. Öğr. Üyesi Volkan Tunalı

İstanbul
T.C. Maltepe Üniversitesi
Fen Bilimleri Enstitüsü
Haziran 2018

JÜRİ VE ENSTİTÜ ONAYI

Murat Can TEKİN'in "Yazılım Geliştirme Taleplerinin Metin Madenciliği İle Sınıflandırılması Ve Önceliklendirilmesi" başlıklı tezi 22/06/2018 tarihinde aşağıdaki jüri tarafından değerlendirilerek "Maltepe Üniversitesi Lisansüstü Eğitim ve Öğretim Yönetmeliği"nin ilgili maddeleri uyarınca, Bilgisayar Mühendisliği Anabilim Dalında Yüksek Lisans/~~Doktora~~ tezi **oy birliğiyle / ~~oy çokluğuyla~~** olarak kabul edilmiştir.

	Unvanı. Adı ve Soyadı	İmza
Üye (Tez Danışmanı)	: Dr. Öğr. Üyesi Volkan TUNALI	.....
Üye	: Doç. Dr. Turgay Tugay BİLGİN	.....
Üye	:Dr. Öğr. Üyesi Mehmet Ali Aksoy TÜYSÜZ	.....

Prof.Dr. İlter BÜYÜKDIĞAN

Enstitü Müdürü

maltepe üniversitesi	ETİK İLKE VE KURALLARA UYUM BEYANI	Doküman No	FR-178
		İlk Yayın Tarihi	01.03.2018
		Revizyon Tarihi	
		Revizyon No	00
		Sayfa	1/1

Revizyon Takip Tablosu

REVİZYON NO	TARİH	AÇIKLAMA
00	01.03.2018	İlk yayın.

ETİK İLKE VE KURALLARA UYUM BEYANI

22/06/2018

Bu tezin bana ait, özgün bir çalışma olduğunu; çalışmamın hazırlık, veri toplama, analiz ve bilgilerin sunumu olmak üzere tüm aşamalarından bilimsel etik ilke ve kurallara uygun davrandığımı; bu çalışma kapsamında elde edilmeyen tüm veri ve bilgiler için kaynak gösterdiğimi ve bu kaynaklara kaynakçada yer verdiğimi; çalışmanın Maltepe Üniversitesinde kullanılan "bilimsel intihal tespit programı" ile tarandığını ve öngörülen standartları karşıladığını beyan ederim.

Herhangi bir zamanda, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun saptanması durumunda, ortaya çıkacak tüm ahlaki ve hukuki sonuçlara razı olduğumu bildiririm.

Murat Can TEKİN

Hazırlayan İlgili Birim	Kalite Koordinatörü Dr. Öğr. Üyesi Şafak GÜNDÜZ	Kurumsal Yetkili Prof. Dr. Belma AKŞİT

(Dokuman No: FR-178; Yayın Tarihi: 01.03.2018; Revizyon Tarihi: ; Revizyon No:00)

Yazılım Geliştirme Taleplerinin Metin Madenciliği İle Sınıflandırılması Ve Önceliklendirilmesi

ORIJINALLIK RAPORU

%7	%5	%2	%4
BENZERLIK ENDEKSI	İNTERNET KAYNAKLARI	YAYINLAR	ÖĞRENCI ÖDEVLERI

BIRINCIL KAYNAKLAR

1	Submitted to Beykent Universitesi Öğrenci Ödevi	%2
2	tr.wikipedia.org İnternet Kaynağı	<%1
3	www.ermanc.sakarya.edu.tr İnternet Kaynağı	<%1
4	www.researchgate.net İnternet Kaynağı	<%1
5	GÖKER, Hanife and TEKEDERE, Hakan. "FATİH Projesine Yönelik Görüşlerin Metin Madenciliği Yöntemleri İle Otomatik Değerlendirilmesi", Gazi Üniversitesi Bilişim Enstitüsü, 2017. Yayın	<%1
6	polen.itu.edu.tr İnternet Kaynağı	<%1
7	docplayer.biz.tr İnternet Kaynağı	<%1

Dr. Öğr. Üyesi Volkan TUNALI

TEŞEKKÜR

Bu çalışmamın başından sonuna kadar bilgisiyle bana yol gösteren, her zaman yardım ve destek veren, yeri geldiğinde danışman kimliğini kenara bırakıp manevi abilik yapan değerli danışman hocam Dr. Öğr. Üyesi Volkan TUNALI'ya sonsuz saygılarımı ve teşekkürlerimi sunarım.

Ayrıca bu günlere kadar beni yetiştiren, eğitimimde maddi manevi desteğini esirgemeyen, her türlü nazımı çeken canım anneme ve canım babama; tezimin başlangıcında nişanlım bitiminde hayat arkadaşım olan eşim Ebru'ya sonsuz teşekkürlerimi sunarım.

Murat Can TEKİN

Haziran 2018

ÖZ

YAZILIM GELİŞTİRME TALEPLERİNİN METİN MADENCİLİĞİ İLE SINIFLANDIRILMASI VE ÖNCELİKLENDİRİLMESİ

Murat Can TEKİN
Yüksek Lisans Tezi
Bilgisayar Mühendisliği Anabilim Dalı
Danışman: Dr. Öğr. Üyesi Volkan TUNALI
Maltepe Üniversitesi Fen Bilimleri Enstitüsü, 2018

Kurumsal şirketlerde, yazılımlardaki hatalar ve değişiklik talepleri genellikle bir talep yönetim sistemi üzerinden BT birimine iletilir. Bu sistemde yer alan öncelik ve aciliyet bilgisi BT birimi için kritik öneme sahiptir. Ancak, talebi giren kişilerin inisiyatifine bırakılan öncelik kararı her zaman gerçekçi olmamaktadır. Örneğin, kritik olmayan ve düşük öncelikli bir değişiklik talebi yüksek öncelikli olarak girilebilmekte, bu da hatalı planlama ve müşteri memnuniyetsizliği ile sonuçlanabilmektedir. Bu çalışmada, iç müşteri talepleri metin madenciliği yöntemleriyle sınıflandırılarak taleplerin önem derecesi tahmin edilmeye çalışılmıştır. Sistemin eğitimi ve testi için kurumsal bir şirketin talep yönetim sisteminden alınan kayıtlar kullanılmıştır. Ham metin formundaki talep verisi üzerinde temizlik ve önişleme işlemlerinin ardından, doküman-terim matrisinin oluşturulmasında TFIDF ağırlıklandırma yönteminden yararlanılmıştır. Elde edilen veri seti üzerinde çeşitli sınıflandırma algoritmaları test edilmiş ve en yüksek başarıma %74,5 F-Skoru değeri ile Rasgele Orman algoritmasıyla ulaşılmıştır.

Anahtar Sözcükler: Talep Önceliklendirme, Yapay Öğrenme, Metin Sınıflandırma, Rasgele Orman.

ABSTRACT

CLASSIFICATION AND PRIORITIZATION OF SOFTWARE DEVELOPMENT DEMANDS WITH TEXT MINING

Murat Can TEKİN
Master Thesis
Master of Science in Computer Engineering Program
Thesis Advisor: Assist. Prof. Dr. Volkan TUNALI
Maltepe University Graduate School of Science And Engineering, 2018

In corporations, issues encountered in software and change demands are forwarded to the IT unit via a demand management system. The priority and severity information in this system has critical importance to the IT unit. However, the priority decision that is left to the individuals who create the demand records may not always be realistic. For instance, a non-critical and low-priority demand may be created with the highest priority, and this may lead to faulty planning and eventually to customer dissatisfaction. In this work, internal customer demands were classified using text mining techniques and their priorities were predicted. The system was trained and tested with the records extracted from the demand management system of a corporation. After cleaning and preprocessing the raw textual demand data, TFIDF weighting scheme was used when creating the document-term matrix. Several classification algorithms were tested on the data set obtained, and the highest performance was achieved by Random Forest algorithm with 74.5% F-Score.

Keywords: Demand Prioritization, Machine Learning, Text Classification, Random Forest.

İÇİNDEKİLER

JÜRİ VE ENSTİTÜ ONAYI 1
İLKE VE KURALLARA UYUM BEYANI 2
İNTİHAL RAPORU 3
TEŞEKKÜR 4
ÖZ 5
ABSTRACT 6
İÇİNDEKİLER 7
TABLOLAR LİSTESİ 9
ŞEKİLLER LİSTESİ 10
KISALTMALAR 11
ÖZGEÇMİŞ 12
BÖLÜM 1. GİRİŞ 13
Problem 13
Tezin Amacı ve Önemi 13
Sınırlıklar 14
BÖLÜM 2. İLGİLİ ÇALIŞMALAR 15
BÖLÜM 3. YÖNTEM 18
Veri Madenciliği 18
Veri Madenciliği Teknikleri 19
Sınıflandırma 19
Sınıflandırma Algoritmaları 20
Naive Bayes 20
Naive Bayes Multinomial 21
SMO (Sequential Minimal Optimization) 21
Karar Ağaçları (Decision Trees) 21
Rasgele Orman (Random Forest) 21
Rotasyon Ormanı (Rotation Forest) 22
Metin Madenciliği 22
Metin Madenciliği Süreci ve Kullanılan Yöntemler 23
Stemming (Kök Bulma) 23
Durdurma Kelimeleri Filtresi (Stopwords Filtering) 24
Terim Ağırlıklandırma 24

Terim Frekansı (TF – Term Frequency) 24
Ters Doküman Frekansı (IDF – Inverse Document Frequency) 24
Doküman Terim Matrisi (DTM – Document Term Matrix) 25
Seyrek Matris (Sparse Matrix) 25
Performans Ölçütleri 26
Kesinlik ve Hassasiyet (Precision & Recall) 26
F-Skoru (F-Score) 27
ROC-Alanı (ROC-Area) 27
Dengesiz Sınıflandırma Problemi 27
Aşırı Örnekleme (Oversampling) 28
Evren ve Örneklem 28
Veriler ve Toplanması 28
Taleplerin Metin Madenciliği ile Sınıflandırılması ve Önceliklendirilmesi 28
Kullanılan Materyaller ve Uygulama Akışı 28
Veri Seti 29
Veri Ön İşleme Uygulaması 30
PRETO 30
WEKA 31
BÖLÜM 4. DENEYSEL SONUÇLAR VE YORUMLAR 32
Deneysel Sonuçlar 32
Deney-1 32
Deney-2 33
Deney-3 34
Deney-4 35
Deney-5 36
Deney-6 37
Deney-7 38
Deney-8 39
Yorumlar 40
BÖLÜM 5. SONUÇ 42
EK'LER 43
Kaynakça 44

TABLOLAR LİSTESİ

Tablo 1 - Doküman Terim Matris Örneği 25
Tablo 2 - Seyrek Matris Örneği 25
Tablo 3 - Kesinlik ve Hassasiyet ile İlgili Bir Örnek 26
Tablo 4 - Veri Seti Örneği 29

ŞEKİLLER LİSTESİ

Şekil 1 - Veri Madenciliği Süreci (Hayri Sever, 2018) 19
Şekil 2 - Veri Madenciliği Yöntemleri 20
Şekil 3 - Metin Madenciliği Süreci 22
Şekil 4 - PRETO Ekran Görüntüsü 30
Şekil 5 - WEKA ARFF Dosya Örneği 31
Şekil 6 – Deney1 Algoritma Karşılaştırma Grafiği 32
Şekil 7 – Deney1 Algoritma Başarı Oranı Grafiği 33
Şekil 8 - Deney2 Algoritma Karşılaştırma Grafiği 33
Şekil 9 - Deney2 Algoritma Başarı Oranı Grafiği 34
Şekil 10 - Deney3 Algoritma Karşılaştırma Grafiği 34
Şekil 11 - Deney3 Algoritma Başarı Oranı Grafiği 35
Şekil 12 - Deney4 Algoritma Karşılaştırma Grafiği 35
Şekil 13 - Deney4 Algoritma Başarı Oranı Grafiği 36
Şekil 14 - Deney5 Algoritma Karşılaştırma Grafiği 36
Şekil 15 – Deney5 Algoritma Başarı Oranı Grafiği 37
Şekil 16 - Deney6 Algoritma Karşılaştırma Grafiği 37
Şekil 17 - Deney6 Algoritma Başarı Oranı Grafiği 38
Şekil 18 - Deney7 Algoritma Karşılaştırma Grafiği 38
Şekil 19 – Deney7 Algoritma Başarı Oranı Grafiği 39
Şekil 20 - Deney8 Algoritma Karşılaştırma Grafiği 39
Şekil 21 - Deney8 Algoritma Başarı Oranı Grafiği 40

KISALTMALAR

ARFF	:	Attribute-Relation File Format
BT	:	Bilgi Teknolojileri
CSV	:	Comma Seperated Variables
DN	:	Doğru Negatif
DP	:	Doğru Pozitif
DT	:	Decision Tree (Karar Ağacı)
DTM	:	Document Term Matrix (Doküman Terim Matrisi)
IDF	:	Inverse Document Frequency (Ters Doküman Frekansı)
KDD	:	Knowledge Discovery in Database (Veritabanlarında Bilgi Keşfi)
k-NN	:	k-Nearest Neighbors (k-En Yakın Komşu)
MLP	:	Multi Layer Perceptron (Çok Katmanlı Algılayıcı)
MNB	:	Multinomial Naive Bayes
NB	:	Naive Bayes
NLP	:	Natural Language Processing (Doğal Dil İşleme)
OCR	:	Optical Character Recognition (Optik Karakter Tanıma)
RF	:	Rotation Forest (Rotasyon Ormanı)
RO	:	Random Forest (Rasgele Orman)
ROC	:	Receiver Operating Characteristic (Alıcı Çalışma Karakteristiği)
SMO	:	Sequential Minimal Optimization (Sıralı Minimal Optimizasyon)
SVM	:	Support Vector Machines (Destek Vektör Makineleri)
TF	:	Term Frequency (Terim Frekansı)
WEKA	:	Waikato Environment for Knowledge Analysis
YN	:	Yanlış Negatif
YP	:	Yanlış Pozitif

ÖZGEÇMİŞ

Murat Can TEKİN

Bilgisayar Mühendisliği Anabilim Dalı

Eğitim

Derece	*Yıl*	*Üniversite, Enstitü, Anabilim/Anasanat Dalı*
Y.Ls.	2018	T.C. Maltepe Üniversitesi, Fen Bilimler Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı
Ls.	2013	İstanbul Ticaret Üniversitesi, Bilgisayar Mühendisliği Anabilim Dalı
Lise	2009	Bostancı Doğa Koleji

İş/İstihdam

Yıl	*Görev*
2017 - ...	Ergo Sigorta – Yazılım Uzmanı
2016- 2017	Liberty Sigorta – Yazılım Uzmanı
2014- 2016	Migros Ticaret A.Ş. – Yazılım Uzmanı

Kişisel Bilgiler

Doğum yeri ve yılı : İSTANBUL / 06.12.1991 Cinsiyet: Erkek
Yabancı diller : İngilizce
GSM / e-posta : 0(542) 597 10 17 / muratcantekin@hotmail.com

BÖLÜM 1. GİRİŞ

Bu bölümde; problem, tezin amacı ve önemi ve sınırlıklar hakkında bilgilere yer verilmiştir.

Problem

Kurumsal sektörde iç müşteri talepleri, firma içi hizmette önemli bir yere sahiptir. İç müşterinin yaptığı talepler doğrultusunda BT birimi bu talepleri karşılar ve gerekli çalışmayı yaparak hizmet kalitesini arttırmayı hedefler. Yapılan talepler önem derecesine göre sıralanmaktadır. İç müşteri sistemden talep açarken, genellikle talebinin hızlı tamamlanması için talebinin yüksek öncelikte olduğunu belirtir. Ancak açılan bu taleplerin önem derecesi yanlış seçildiği için adreslenecek talep, gerçek "yüksek" öncelikli taleplerin önüne geçebilmektedir. Yapılan bu yanlış seçim sonucu üretim, zaman ve iş gücü kaybına sebebiyet vermektedir.

Tezin Amacı ve Önemi

Rekabetin yoğun olduğu iş dünyasında firmalar müşteriye açılan sistemlerini en iyiye taşımak için çalışmalar yapar. Bu çalışmalar doğrultusunda müşteriye en iyi hizmeti vermeyi hedefler. Ancak müşterinin dokunduğu noktalarda alınan hatalar iç müşterilere bildirilir. İç müşteri birimleri ise bu hataları BT birimine iletir. Küçük çaplı firmalarda bu e-mail veya sözlü olarak yürütülebilmektedir. Ancak büyük çaplı firmalarda bu taleplerin sistem üzerinde tutulması gerekmektedir. Bunun için talep yönetim sistemi üzerinden talepte bulunurlar. Bu sistemde yer alan önceliklendirme ve aciliyet bilgisi BT birimi için önemlidir. BT birimi mevcut talepler arasında önceliği en yüksek öncelikte olanları sıraya alarak çözüme kavuşturmaya çalışır. Ancak iş biriminin talepleri yanlış önceliklendirmesi sonucunda oluşacak başarısız talep çözüm planlaması, firma içerisinde zaman ve iş gücü kayıp yaratmaktadır.

Veri madenciliğinin alt dalı metin madenciliği ile metinsel kaynaklar veri seti olarak kullanılarak, veri madenciliği işlemleri ile anlamlı bilgiler ortaya çıkartılır.

Bu çalışmada, iç müşteri talepleri, metin madenciliği yöntemleriyle sınıflandırılarak taleplerin önem derecesi tahmin edilmeye çalışılmıştır. Bu sayede, BT birimlerine iç müşteri taleplerinin önceliklerinin değerlendirilmesi ve doğru planlama

yapılması konusunda destekleyici bir sistem geliştirilmesi amaçlanmıştır. Bilindiği kadarıyla, yazılım geliştirme taleplerinin metin madenciliği kullanılarak önceliklendirilmesine yönelik, özellikle de Türkçe dilinin yapısal zorluklarını da dikkate alan benzer bir çalışma literatürde bulunmamaktadır.

Sınırlıklar

Bu çalışmada veri seti olarak kullanılmak üzere toplanan veriler sınıflara eşit dağılmamış olup sınırlı miktardadır. Mevcut verinin en temiz hali kullanılmıştır.

BÖLÜM 2. İLGİLİ ÇALIŞMALAR

Bu bölümde, metin madenciliği ile ilgili yapılan çalışmalardan bahsedilmektedir.

Sancar tarafından 2016 yılında yapılan bir çalışmada, metin madenciliği kullanılarak talep tanıma ve yönlendirme sistemini incelemiştir (Sancar, 2016). Bu çalışmada, dilekçeleri OCR yöntemiyle tarayarak içerisindeki cümleleri tanımlamış ve bu cümlelerde metin madenciliği uygulaması yapmıştır. NB, MNB, SVM, k-NN, Geri Yayılım Algoritması ve J48 algoritmalarını karşılaştırıp aralarından %87,6 ile MNB en başarılı sonucu vermiştir.

2014 yılında Kaşıkçı tarafından yapılan bir çalışmada, metin madenciliği ile e-ticaret sitelerinin belirlenmesi incelenmiştir (Kaşıkçı, 2014). Kaşıkçı bu çalışmada internet sitelerinin içerikleri metin madenciliği yöntemleri ile e-ticaret sitesi olup olmadığı bilgisi veren bir program geliştirmiştir. Uygulama içerisinde k-NN ve NB algoritmaları kullanılmıştır. Yapılan çalışmalar sonucunda k-NN algoritması ikili ağırlıklandırma yönetiminin %25 oranı ve k=7 değeriyle %91,83'lük bir başarı elde edilmiştir. Bu sonuca bağlı olarak her 100 dokümanda 8 adet hatalı sınıflandırma yapıldığı sonucuna ulaşılmıştır.

Kuzucu 2015 yılında yaptığı bir çalışmada, müşteri memnuniyeti belirlemek için metin madenciliği tabanlı bir yazılım aracını konu almıştır (Kuzucu, 2015). Mevcut şirket verileri üzerinden müşterilerin şikayet kayıtlarında mutlu veya mutsuz statüsünü şikayet kaydı girişi yaparken tahminleyecek bir çalışma yapmıştır. Bu çalışmada Naive Bayes algoritması kullanılmış olup, yapılan çalışmalar sonucu 0,8595'lik F-Skor başarısı elde edilmiştir.

2011 yılında Varol yaptığı çalışmada, metin madenciliği yöntemlerini kullanarak Türkçe dokümanlarda tür ve yazar tanıma konusunu ele almıştır (Varol, 2011). Çalışma konusunda veri seti olarak 7 şair ve bu şairlere ait 30 şiirden oluşan bir eğitim setini kullanmıştır. Bu çalışmada NB, MLP, SMO, RF, RO, DT gibi sınıflandırma algoritmaları incelenmiştir. Çalışma sonucunda RF, RO ve MLP %71 başarılı sınıflandırma yapmıştır.

Seçkin tarafından 2011 yılında yapılan bir çalışmada, metin madenciliğinde kullanılan yöntemlerin karşılaştırılması amacıyla siyasi parti liderlerinin grup genel

toplantı konuşmaları ile bir uygulama yapılmıştır (Seçkin, 2011). Bu çalışmada metin madenciliğinde kullanılan dilbilgisel ve istatiksel teknik ve algoritmalar, 3 siyasi parti liderine ait 10'ar toplamda 30 tane konuşmadan oluşan metinler veri seti olarak kullanılmıştır. Makine öğrenmesi yöntemlerinden NB, SVM, k-NN algoritması ve karar ağaçları kullanılmıştır. Yapılan çalışmalar sonucunda aralarında en başarılı olan makine öğrenme algoritmaları %100'e yakın değerle NB ve SVM olmuştur.

2000 yılında yapılan bir çalışmada Han ve Karypis, ağırlık merkezi tabanlı doküman sınıflandırma algoritmalarının analizini ele almıştır (Eui-Hong Han, 2000). Bu çalışmada doğrusal zamanlı ağırlık merkezi tabanlı doküman sınıflandırma algoritmasına yoğunlaşılmıştır. Yapılan deneylerde ağırlık merkezi tabanlı sınıflandırma algoritmalarının büyük boyutlu verisetleri üzerinde çoğunlukla NB, k-NN gibi algoritmalardan daha başarılı olduğu gözlemlenmiştir.

Yıldız 2007 yılında, metin sınıflandırmada yeni özellik çıkarımı üzerine bir çalışma yapmıştır (H.Kemal Yıldız, 2007). Türkçe'nin biçim birim yapısı kullanılarak türü bilinmeyen Türkçe bir metnin sınıflandırılması için özellik çıkarım yöntemi ele alınmıştır. Kelime gövdelerini baz alan bu çalışmada, veri setinde doküman içerisinde geçen terimlerin gövdelerinin her sınıfta kullanılma sıklığının toplam değerine göre ağırlıklandırma yöntemi kullanılmıştır. Kelime gövdelerini bulma yönteminde hızı arttırmak için Trie ağaç yapısından faydalanılmıştır. NB, SVM, k-NN, C4.5 ve RO sınıflandırma yöntemleri uygulanmıştır. En yüksek başarı oranı %96.25 ile NB yönteminde görülmüştür.

2013 yılında Fidan tarafından yapılan bir çalışmada, destek vektör makineleri ile doküman sınıflandırma konusu ele alınmıştır (Fidan, 2013). Çekirdek fonksiyonun etkileri ve parametreleri belirlenerek destek vektör makinelerinden ikiden fazla sınıfa iat veriler için üst seviye uzaya aktarılma işlemi yapılmıştır. LaSVM algoritmasının eşli çekirdek yöntemine uyumlu şekilde çalışması için ayarlanmıştır. Vektör makinelerinin karar sınırları için parametreler ayarlandıktan sonra veri eğitim seti ve test seti kümelerine ayrıştırılarak farklı kombinasyonlarda test çalışmaları yapılmıştır. Yapılan sınıflandırmada tahmin edilen sınıflandırma ve doğru sınıflandırma oranı ROC eğrisinin altında kalan alana göre değerlendirilmiştir. Sonuçlar ele alındığında çevrimiçi eşli sınıflandırma yönteminin iki ve üzeri sınıflı sınıflandırma yöntemlerine alternatif

olabileceğini göstermiştir. Ek olarak doğrusal eşli çekirdeklerin gauss eşli çekirdeklere göre daha iyi sonuç verdiği görülmüştür.

Parlak 2015 yılında yaptığı bir çalışmada, tıbbi dokümanların hastalıklara göre sınıflandırılması konusunu ele almıştır (Bekir Parlak, 2015). MEDLINE isimli tıbbi terimlerin bulunduğu veritabanında alt küme yöntemi ile tek etiketli çok sınıflı bir veri seti oluşturulup, bu veri setinde sayıca en yüksek çıkan ilk 10 hastalık çalışmada kullanılmıştır. Veri seti kök bulma işlemi yapılmış ve yapılmamış olarak iki farklı test verisi üzerinden değerlendirilmiştir. C4.5, Bayes ağı ve RO algoritmaları kullanılarak üç farklı sınıflandırma algoritması ile test edilmiştir. Bu üç algoritma içerisinde en yüksek sınıflandırıcı ve en yüksek başarı oranı Bayes ağı algoritması ile kök bulma işleminin yapılmadığı durumda sonuçlanmıştır.

2013 yılında Döven tarafından yapılan bir çalışmada, metin madenciliği ile dokümanlar arasındaki benzerliklerin bulunması araştırılmıştır (Döven, 2013). Yapılan çalışmada bir uygulama aracılığı ile sadece iki doküman arasında değil kullanıcının n adet seçebileceği doküman arasında benzerlik karşılaştırması yapılabilmektedir. Uygulama seçilen dokümanlar içerisinde bulunan dokümanda geçen cümleleri diğer dokümanlar içerisinde aratarak dokümanlarda tek tek karşılaştırma yaparak benzerlik hesaplaması yapmaktadır. Bu çalışmada kosinüs ve jaccard benzerlik yöntemlerinin en başarılı sonuçlar veren yöntemler olduğu tespit edilmiştir.

BÖLÜM 3. YÖNTEM

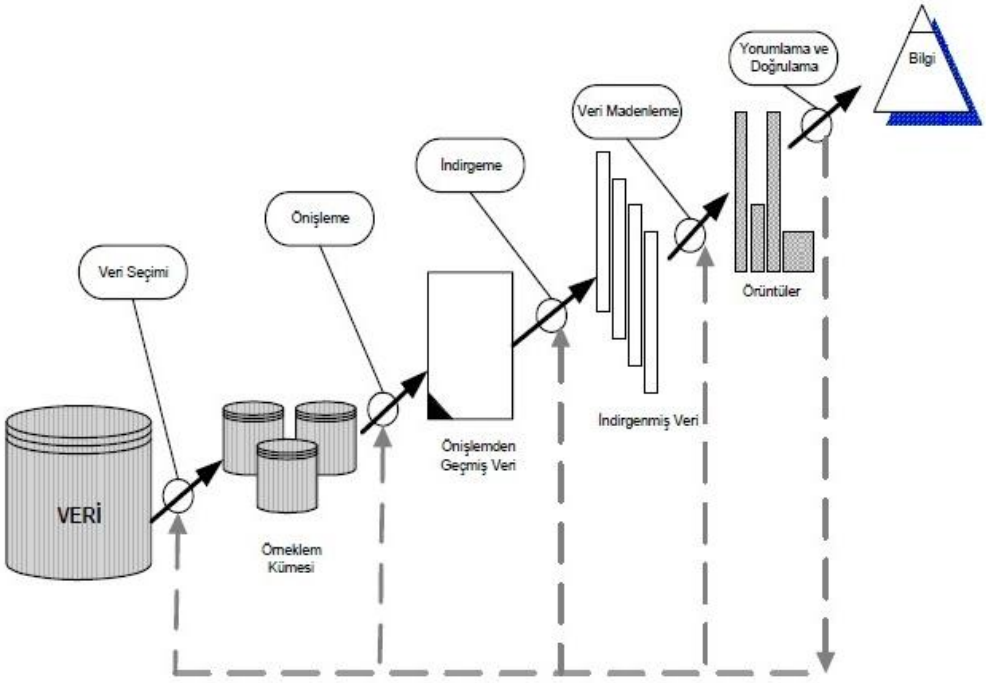
Bu bölümde, veri madenciliği, veri madenciliği teknikleri, sınıflandırma ve sınıflandırma algoritmaları ve metin madenciliği anlatılmaktadır. Ayrıca bu tez çalışmasında; metin madenciliği yöntemleri, performans ölçütleri, dengesiz sınıflandırma probleminden bahsedilmekte olup, çalışmanın araştırma modeli, evren ve örneklem, veriler ve toplanması ile ilgili bilgilere yer verilmiştir.

Veri Madenciliği

Günümüz teknoloji çağında yapılan her işlem tek başına anlamsız ham veriye dönüşmektedir. Gün geçtikçe artan teknolojik cihaz kullanımı, bu verilerin çok büyük boyutlara ulaşılmasına neden olmaktadır. Veriler topluluğu belirli bir boyuta geldiğinde, insan yeteneğiyle ölçümlenmesi için büyük bir zaman dilimine ihtiyaç duyulmaya başlanmaktadır. Bu zaman dilimi içerisinde yapılacak hataların fazlalığı da yapılan ölçümlerde yanlış sonuçlara sebep olabilir. Bu büyük veriler topluluğunun içerisinde istenilen deseni, verilerin birbirleri ile olan ilişkilerini, hatta geçmişte elde edilmiş veriler ile yakın geleceğe dönük tahminlerde bulunulabilir. Veri madenciliği, bu büyük veriler topluluğundan elde edilmek istenilen bilgilerin keşfinde analiz ve işlemlerin yapılması sürecidir (Arslantekin, 2003). Şekil 1'de bilgi keşfinin veri madenciliğiyle hangi adımlarla yapıldığı gösterilmektedir.

Özellikle büyük firmaların başvurduğu veri madenciliği, firmanın satışlarını arttırıcı faaliyetler konusunda önemli bir rol oynar. Örneğin; bir market zincirinden alışveriş yapan müşteri kasada ödeme yaparken markete ait avantaj kartıyla alışveriş yapmaktadır. Müşterinin verileri işlenerek daha sonra yapılan alışverişlerde müşteriyi alışveriş yapmaya yöneltecek kampanyalar sunulabilir.

Birçok kişi veri madenciliğinin popüler eş anlamı olan Veritabanlarında Bilgi Keşfi (KDD - Knowledge Discovery in Database) olarak adlandırır. Ayrıca; veriden bilgi madenciliği, bilgi çıkarma, veri ve desen analizi, veriyi nitelikli bilgiye dönüştürme yolu gibi kullanılan terimler de veri madenciliği ile benzer bir anlam taşımaktadır (Jiawei Han, 2012).



Şekil 1 - Veri Madenciliği Süreci (Hayri Sever, 2018)

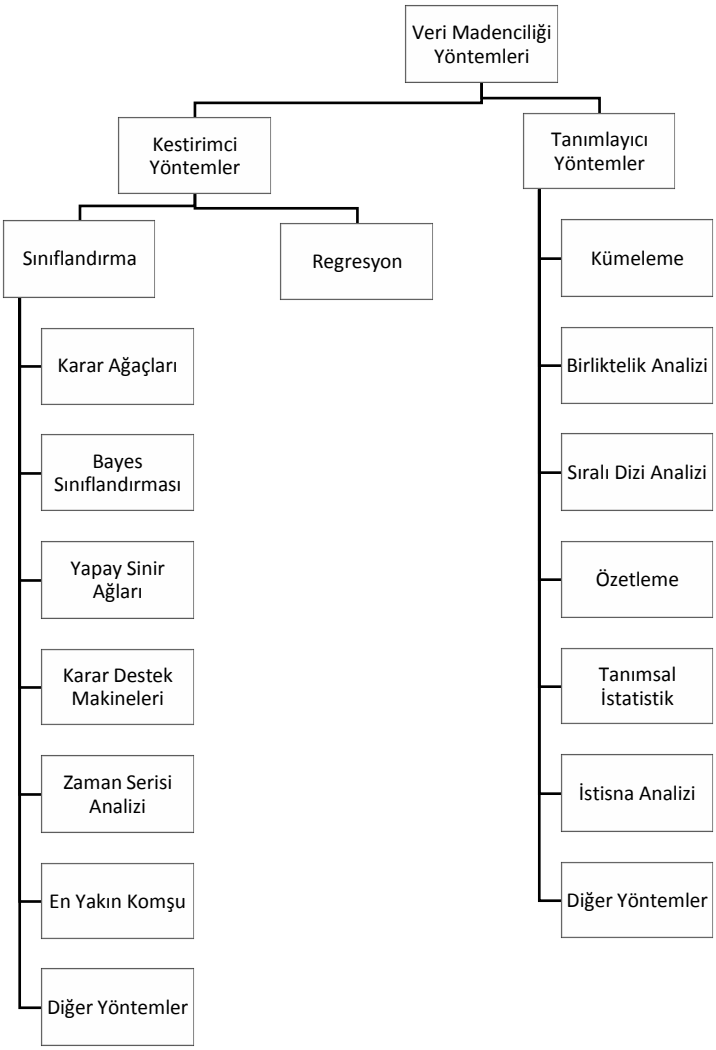
Veri Madenciliği Teknikleri

Veri madenciliği yapılırken hangi tekniklerin kullanıldığı Şekil-2'de gösterilmektedir. Bu teknikler kestirimci ve tanımlayıcı yöntemler olarak iki ana kategoriye ayrılmaktadır.

Sınıflandırma

Metinlerin sınıflandırılması, metin belgelerine önceden tanımlanmış sınıflardan uygun olanı eşleştirmeyi amaçlar (Mitchell, 1997).

Örneğin; Bir veri setinde türleri "politika", "spor", "sanat" olarak sınıflandırılmış haberler vardır. Bu haberler, yeni gelecek sınıflandırılmamış haberler için eğitici veri seti olarak da kullanılır. Yeni gelen bir haberin içeriğinde bulunan metinlere bakılarak uygun olan tanımlı sınıflardan birine yerleştirilir.



Şekil 2 - Veri Madenciliği Yöntemleri

Sınıflandırma Algoritmaları

Bu bölümde tezde kullanılan sınıflandırma algoritmalarından bahsedilmektedir.

Naive Bayes

Naive Bayes algoritması, veri kümesindeki değerlerin kombinasyonunu ve frekansını dikkate alarak bir olasılık kümesi oluşturan basit bir istatistiksel olasılık sınıflandırıcıdır (Tina R. Patil, 2013). Kolay, hızlı ve başarılı bir algoritma olduğu için en çok tercih edilen sınıflandırma algoritmalarındandır.

Naive Bayes Multinomial

Naive Bayes Multinomial algoritması, Naive Bayes algoritmasının metin belgeleri için özelleştirilmiş halidir. Naive Bayes belirli kelimelerin olup olmadığını belirlerken, Naive Bayes Multinomial, doküman içerisinde bulunan kelimerin tekrar etme sayılarından frekansını hesaplayarak olasılık kümesi oluşturmaktadır (Adrew McCallum, 1998). Her bir kelimenin tekrar etme sayısı bir başka kelimenin veya kelimelerin tekrar etme sayısına bağlı değildir.

SMO (Sequential Minimal Optimization)

Sıralı Minimal Optimizasyon algoritması, Destek Vektör Makinaları'nın (SVM - Support Vector Machines) kuadratik programlama problemini, herhangi bir ekstra matris depolaması olmadan ve her bir alt problem için yinelemeli nümerik bir rutine başvurmadan çabucak çözen basit bir algoritmadır (Platt, 1998).

Karar Ağaçları (Decision Trees)

Karar ağaçları, sınıflandırma yöntemlerinde yaygın olarak kullanılan eğiticili öğrenme yöntemidir. Adından da anlaşılacağı gibi ağaç görüntüsüne benzer bir yapıda olup dal ve yapraklardan oluşur. Ağaçta bulunan her bir düğüm testi temsil edip, test sonucunda dalları oluşturur. Test sonucu oluşan dalda yaprak varsa sınıflandırılmış verinin bir sınıfını belirtir (Kuzucu, 2015).

Rasgele Orman (Random Forest)

Adından da anlaşıldığı üzere bu algoritma, birden fazla karar ağacı üreterek bir karar ormanı oluşturur. Bu şekilde sınıflandırma değerinin yükseltilmesi hedeflenir. Çalışma mantığı şu şekildedir; ilk olarak birden fazla sınıflandırma ağacı oluşturulur, her ağaca sınıflandırma yapması için giriş vektörü verilir. Her ağacın sonucu arasında en yüksek oyu alan sınıf sonucu seçilir. Her ağaç, eğitim setinde bulunan örneklerden rasgele seçilerek yenisiyle değiştirilmesiyle oluşturulur. Her ağaç eğitim seti oluşturulurken verilerin üçte biri ağaç dışına ayrılır, geri kalanı ile sınıflandırma hatası hesap edilir (Türkoğlu, 2006).

Rotasyon Ormanı (Rotation Forest)

Rasgele Orman benzeri olan bu karar ormanında, Rotasyon Ormanı algoritması, son zamanlarda sınıflandırma hızını ve doğruluğunu arttırmayı amaçlayanların kullandığı yeni nesil bir toplu öğrenme algoritmasıdır (İsmail Çölkesen, 2014).

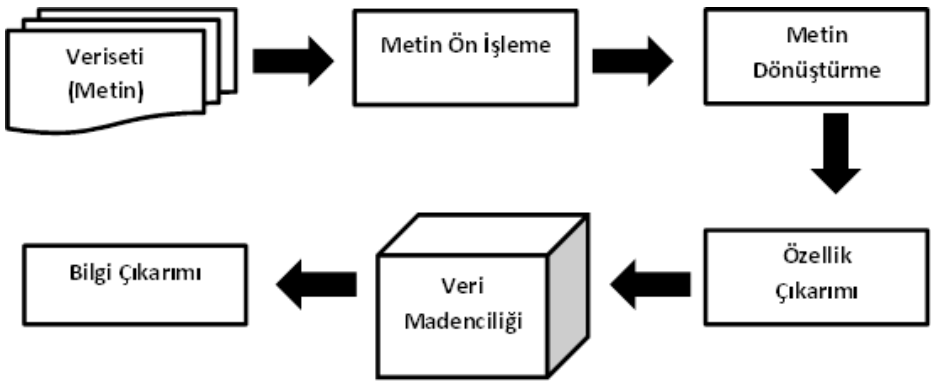
Metin Madenciliği

Günümüzde dijital ortama taşınmış ve dijital ortamda üretilmekte olan yazılı doküman miktarı artmaktadır. Elde edilen yazılı veriler üzerinden Doğal Dil İşleme (NLP – Natural Language Processing), yapay zeka, istatistik gibi teknikler kullanılarak metin madenciliği ile bilgi keşfi yapılmaktadır (Jiawei Han, 2012).

Bu yazılı veriler tek başına metin madenciliği için anlamsız olup kelime niteliğindedir. Ancak bir metin madenciliği uygulaması ile istenilen bilgiye ulaşılabilmektedir. Örneğin; bir firma hakkında yapılan olumlu/olumsuz yorumlarda bulunan kelimeleri doğal dil işleme ve metin madenciliği ile işleyerek o firmadan müşterilerin memnuniyet oranını çıkarılabilir. İş dünyasında bulunan verilerin yaklaşık %85'inin metin formatında olduğu tahmin edilmektedir (Andreas Hotho, 2005).

Metin madenciliği, doküman topluluğunun önişleme işlemleri sonrası çıkan ara sonuçların saklanması, bu ara sonuçların analizi için çeşitli tekniklerin kullanılması ve çıkan sonuçların görselleştirilmesi basamaklarından oluşmaktadır (James Sanger, 2002). Metin madenciliği süreci Şekil 3'te gösterilmiştir.

Metin madenciliği; yazar tanıma, metin sınıflandırma, duygu analizi, metin kümeleme, müşteri ilişkileri yönetimi ve bilimsel araştırmalar gibi çeşitli alanlarda kullanılmaktadır.



Şekil 3 - Metin Madenciliği Süreci

Bu tez çalışmasında metin madenciliğinde sıkça kullanılan ve başarılı olan sınıflandırma algoritmalarından Naive Bayes, Naive Bayes Multinomial, SMO ve karar ağacı algoritmalarından Rasgele Orman ve Rotasyon Ormanı kullanılmıştır.

Metin Madenciliği Süreci ve Kullanılan Yöntemler

Metin madenciliğinde temel olarak, yapısal olmayan verinin yapılandırılmış veriye dönüştürülmesi gerekir. Öncelikle veritabanından çekilecek veri setinin istenilen özellikte olması gerekmektedir. İstenilen özelliklerde toplanan veri seti üzerinde dizgeciklere ayırma (tokenization), durdurma sözcükleri filtreleme (stopword filtering), kök bulma (stemming), terim ağırlıklandırma (term weighting) ön işleme adımları yapılır (Andreas Hotho, 2005). Daha sonra veri seti NLP yöntemleri ile metin madenciliği algoritmalarıyla işlenmeye hazır hale getirilir. Yapılacak sınıflandırma, kümeleme, tahmin gibi algoritmalarla yorumlanabilir yapılandırılmış verilere ulaşılır.

Stemming (Kök Bulma)

Bir kelimeye gelen çoğul eki, fiil çekim ekleri gibi eklerin ayrılarak kelimenin kök halinin çıkartılması, kısacası kelimelerin en yalın eksiz hale çevrilmesi işlemidir (Volkan Tunalı, 2012). İngilizce çekimli bir dil olduğu için görece kolay olan stemming, Türkçe'nin eklemeli bir dil olmasından ötürü ayrıştırma işleminde daha zor olabilmektedir. Örneğin; "gidemezsin" ifadesi İngilizce'de "you can not go" olarak ifade edilir. "Gitmek" kelimesine eklenen olumsuzluk eki ile "gitmemek", şahıs olarak "sen" ise "gidemez-sin" olmaktadır, ancak İngilizce'de "go" ifadesi "not " kelimesi ile olumsuzlaştırılmış olup "you" kelimesiyle şahıs belirtilmektedir. Türkçe kelimelerde yükleme gelen son eklerle belirtilen noktalar, İngilizce'de ayrı ayrı kelime olarak geçmektedir. Bu nedenle Türkçe kelimelerin ancak parçalara -eklere- ayrılarak anlamı ortaya çıkmaktadır.

Zemberek, eklemeli Türk dilleri için geliştirilmiş, açık kaynak kodlu, platformdan bağımsız bir doğal dil işleme kütüphanesidir (Ahmet Afşin Akın, 2007). Zemberek, sözlük tabanlı bir kök bulma yöntemi kullanmakta olup, tanımlı bir kök ve ek sözlüğü kullanarak bu işlemi yapmaktadır.

Affix Stripping, yani ek çıkaran kök bulucu olarak da adlandırılan bir diğer yöntem ise Türkçe'nin kural tabanlı yapısını kullanarak stemming işlemini yapmaktadır. Bu yöntem, sözcüklerin morfolojik yapısını, eklerin sondan başa doğru çıkarılarak

yaklaşımıyla kelimenin kökünü bulacak şekilde geliştirilmiş bir doğal dil işleme yöntemidir (Gülşen Eryiğit, 2004).

Bu tez çalışmasında stemming adımında Zemberek ve Affix Stripping kullanılmıştır.

Durdurma Kelimeleri Filtresi (Stopwords Filtering)

Bir doğal dil işleme adımında dikkate alınmaması gereken kelimeler olabilir. Bu kelimeler cümlelerde yaygın olarak kullanılır ve cümleye küçük anlamsal zenginlikler katabilir, ancak tek başlarına bir bilgi anlam ifade etmezler. Örneğin; "ve", "veya", "yani" gibi kelimeler bir doküman içerisinde çok kez geçebilir, bu sebeple çıktılarda analiz çalışmalarında başarımı olumsuz etkileyebilir. Sonuç olarak durdurma kelimeri filtreleme işlemi, bu ve benzeri kelimelerin değerlendirme dışında bırakılmasıdır (Jure Leskovec, 2011). Durdurma kelimeleri, bir metin dosyasında tutulmaktadır. Bu tez çalışmasında; standart durdurma kelimelerine ek olarak, bu çalışmada işleme alınmayacak kelimeler de dosya içerisine eklenmiştir. Dosyanın içerdiği kelimeler ekler bölümünde yer almaktadır.

Terim Ağırlıklandırma

Kelimelerin kökleri elde edildikten sonra terim ağırlıklandırma işlemi yapılır. Bu işleme terimlerin doküman üzerindeki etkisi de denilebilir (Mehmet Fatih Karaca, 2012). Eğer terim doküman içerisinde bulunuyorsa terimin ağırlıklandırılmış değeri, bulunmuyorsa 0 değeri yazılır.

Terim Frekansı (TF – Term Frequency)

Ağırlıklandırmada sadece bulunduğu doküman üzerinden hesaplama yapılmaktadır. Terim doküman içerisinde kaç kez geçiyorsa ona göre ağırlıklandırılır. Eğer bir terim doküman içerisinde birden fazla kez geçiyorsa, o doküman için değerlidir.

Ters Doküman Frekansı (IDF – Inverse Document Frequency)

Ters doküman ağırlıklandırması, bir terimin geçtiği doküman sayısına göre hesaplanmaktadır. Bir terim ne kadar az dokümanda geçiyorsa ayırt ediciliği yani IDF değeri o kadar yüksektir. Benzer şekilde bir terim ne kadar çok dokümanda geçiyorsa

ayırt ediciliği yani IDF değeri o kadar düşük olur. Yani sonuç olarak; bir terimin frekansı IDF değeri ile çarpılarak normalize edilmiş olur.

Doküman Terim Matrisi (DTM – Document Term Matrix)

Metin madenciliğinde birbirinden bağımsız bulunan metinler doküman olarak nitelendirilir. Bu dokümanlar içerisinde geçen her bir kelime ise terim olarak kabul edilir. Bu terimlerin hangi dokümanda olup olmadığını sayısal halde belirten matris Doküman Terim Matrisi olarak adlandırılır. Örneğin; "bugün süt, yumurta, ekmek aldım." cümlesi ile "yarın eve tavuk, süt, bal almalıyım." cümlelerinin doküman terim matrisi üzerinde gösterimi Tablo 1'de gösterilmiştir.

Tablo 1 - Doküman Terim Matris Örneği

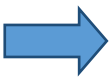
	bugün	süt	yumurta	ekmek	almak	yarın	ev	tavuk	bal
Doküman 1	1	1	1	1	1	0	0	0	0
Doküman 2	0	1	0	0	1	1	1	1	1

Seyrek Matris (Sparse Matrix)

Bu matrisin metin madenciliğinde kullanılmasının sebebi, her terimin (kelimenin) her dokümanda geçmemesidir. Terim kolon sayısının fazla olması ve her dokümanda belli başlı terimlerin bulunmasından dolayı, bulunmayan terimlerin sıfır olarak gösteriminin matris şeklidir. Kısacası seyrek matris, sıfır olmayan çok az elemanlı matristir (Randolph E. Bank, 2001). Normal bir matris tanımı için o matrisin boyutları kadar hafıza rezerve edilir. Ancak seyrek matriste iki boyutlu bir dizi ile aynı matris daha küçük bir hafıza rezerve edilerek tanımlanabilir.

Tablo 2 - Seyrek Matris Örneği

0	0	0	0	9	0
0	8	0	0	0	0
4	0	0	2	0	0
0	0	0	0	0	5
0	0	2	0	0	0



Satır	Sütun	Değer
5	6	6
0	4	9
1	1	8
2	0	4
2	2	2
3	5	5
4	2	2

Tablo 2'de görülen örnekte ilk satırında matrisin kaç satır ve sütun olduğunu ve sıfırdan farklı değerlerin sayısı yer almaktadır. Devam eden satırlarda ise matrisin içinde bulunan sıfırdan farklı değerlerin koordinatlarını ve değerlerini göstermektedir.

Performans Ölçütleri

Yapılan tahminlerin kalitesini ölçmek için bazı parametrik ölçütler kullanılmaktadır.

Kesinlik ve Hassasiyet (Precision & Recall)

Kesinlik, bir ölçme aletinin aynı fiziksel boyuta ait tekrarlanan çeşitli ölçümlerde aynı değeri verebilme ölçütüdür (Powers, 2007). Örneğin; bir kanser tarama testinde yapılan ölçümde pozitif olarak bulunmuş kişilerin yüzde kaçının gerçekten de pozitif olduğunun bilgisini sağlamaktadır.

Hassasiyet, yapılan bir ölçümde getirilen doğru bilginin, yapılan çeşitli ölçümlerle getirilmesi gereken doğru sonuçlara oranını ölçmektedir (Powers, 2007). Örneğin; bir kanser tarama testinde yapılan ölçümde pozitif olan kişilerin yüzde kaçı pozitif doğru olarak bulunmuş bilgisini bize sağlar.

Bu kavramları bir örnek üzerinden anlatmak gerekirse:

Tablo 3 - Kesinlik ve Hassasiyet ile İlgili Bir Örnek

Varolan / Tahmini Değer	Kanserli	Kanserli Değil	Toplam
Kanserli	50	10	60
Kanserli Değil	5	100	105
Toplam	55	110	

	Kanserli	Kanserli Değil
Kanserli	DP	YP
Kanserli Değil	YN	DN

Tablo 3'te görüldüğü gibi kanserli ve kanserli olmayan hastalar belirtilmiştir. Bu değerler varolan ve tahmin edilen değer olarak matris şeklinde gösterilmiştir.

Yapılan ölçümde pozitif olarak bulunmuş kişilerin yüzde kaçının gerçekten de pozitif olduğunu hesaplamak için Denklem 1 ve 2 kullanılmaktadır.

$$\text{Kesinlik}=\frac{DP}{DP+YP}=\frac{50}{50+10}=0{,}83 \qquad (1)$$

$$\text{Hassasiyet} = \frac{DP}{DP+YN} = \frac{50}{50+5} = 0{,}90 \quad (2)$$

Burada kesinlik değeri 50 gerçek kanserli hastanın, tahmini 60 kanserli hastaya oranıyla gerçekten yüzde kaçının kanserli olduğunu hesaplanmıştır. Hassasiyet değeri ise; 50 gerçek kanserli hastanın, yapılan ölçümde çıkan 55 kanserli hastada yüzde kaçının doğru bulunduğu hesaplanmıştır.

F-Skoru (F-Score)

F-Skoru, hassasiyet ve kesinlik değerlerinin harmonik ortalamasıdır ve Denklem 3'teki gibi hesaplanmaktadır.

$$\text{F-Skoru} = 2\,\frac{Kesinlik\ x\ Hassasiyet}{Kesinlik+Hassasiyet} = 2\,\frac{pr}{p+r} \quad (3)$$

Yukarıdaki örnek üzerinden örneklemek gerekirse:

$$\text{F-Skoru} = 2\,\frac{0{,}83\ .\ 0{,}9}{0{,}83+0{,}9} = 2\,\frac{0{,}747}{1{,}73} = 0{,}86$$ lık bir F-Skoru elde edilmiştir.

ROC-Alanı (ROC-Area)

ROC alanı veya diğer adıyla ROC eğrisi, sınıflandırıcı başarımı değerlendirmede kullanılan etkili bir ölçüttür ve Denklem 4'teki gibi hesaplanır:

$$\text{ROC Alanı} = 0.5\,\left(\frac{DP}{DP+YN} + \frac{DN}{DN+YP}\right) \quad (4)$$

Dengesiz Sınıflandırma Problemi

Bir veri seti üzerinden test yapılırken, sınıflandırılmış verilerde eşit bir dağılım olmayabilir. Ancak mevcut algoritmalar bu verisetlerinde bulunan verilerin sınıflarına eşit dağıldığını varsaymaktadır (Haibo He, 2009). Sınıflara eşit dağılmamış verisetlerinde çıkan sonuçlar iyi bir referans olmayabilir. Bu sorunun önlenmesi için bir kaç yöntem bulunmaktadır. Bunlar;

- Daha fazla veri toplayarak, sayıca az olan sınıfın oranını arttırmak.
- Doğruluk değeri yerine; belirleyicilik, kesinlik, F-Skoru gibi performans ölçütlerini dikkate alarak değerlendirmek.

- Azınlık olan sınıfa ait veriler üzerinden (tek sınıf sınıflandırması) makine öğrenmesi yaparak, test edilecek verinin her iki sınıftan gelmesine karşın gelen verinin eğitilmiş veriye uymaması veya uyması durumuna bağlı olarak sınıflandırma yapılır (Elif Kartal, 2017).

Aşırı Örnekleme (Oversampling)

Test edilecek veride azınlık olan sınıfın sayıca çok olan sınıfın sayısına yaklaştırılmasıdır. Aşırı örnekleme yöntemi ile azınlık sınıfına ait olan mevcut veri çoğaltılarak dengeleme yapılır (Mayuri S. Shelke, 2017).

Bu tezde dengesiz sınıflandırma iyileştirmesi için aşırı örnekleme kullanılarak mevcut veriler çoğaltılmıştır.

Evren ve Örneklem

Bir çok firmanın kendine ait talep sistemi bulunmaktadır. Bu sistemdeki iç müşteri talepleri bir evren olarak düşünülebilir. Bu veri evreni üzerinden hareketle aynı sektörde faaliyet gösteren firmalar ele alınırsa çok daha iyi sonuçlar elde edilebilir. Ancak mevcut alınabilecek örneklem adı altında veri seti sadece bir firmaya aittir.

Örneklem içerisinde önceden eğitilmiş verilerde "düşük", "orta", "yüksek" sınıflar bulunmaktadır. Bu sınıflandırmada mevcut veri dengesiz olduğu için, 43 adet "düşük", 150 adet "orta", 151 adet "yüksek" öncelikli talep bulunmaktadır.

Veriler ve Toplanması

Bu tezde kullanılan veri seti, özel bir firmanın iç müşteri taleplerinin takibi ve çözümü için kullanılan bir program üzerinden alınmıştır. Mevcut ham veri içerisinde bulunan kişisel bilgiler temizlenerek kullanıma hazır hale getirilmiştir.

Taleplerin Metin Madenciliği ile Sınıflandırılması ve Önceliklendirilmesi

Taleplerin metin madenciliği ile sınıflandırılması ve önceliklendirilmesi için yapılan çalışmalar ve uygulama bu bölümde anlatılmaktadır.

Kullanılan Materyaller ve Uygulama Akışı

Uygulamada ele alınacak veri seti, talep ve proje takip programından alınmış verilerdir. Bu veri seti üzerinde ön işleme ve temizleme işlemleri için Visual Studio 2013 uygulamasında C# diliyle geliştirilen ön işleme uygulaması oluşturulup kullanılmıştır. Elde edilen ön işlenmiş veri PRETO uygulaması ile, metin madenciliğinde ham veriden bilgi çıkarımı yapılabilecek seviyeye getirerek doküman

terim matrisi hazırlanmıştır. Bu doküman terim matrisini WEKA uygulamasına hazır hale dönüştürmek için Visual Studio 2013 uygulamasında C# diliyle geliştirilmiş uygulama kullanılmıştır. WEKA uygulamasına hazır olan veri, program içerisinde bulunan makine öğrenmesi algoritmaları ile test edilmiştir.

Veri Seti

Uygulamada kullanılan veri setinde iç müşterilerin taleplerinin başlığı, açıklaması ve sınıfı ele alınmıştır. Bu veri setinde bulunan ham veri setinden bir parça veri örneği Tablo 4'te gösterilmiştir.

Tablo 4 - Veri Seti Örneği

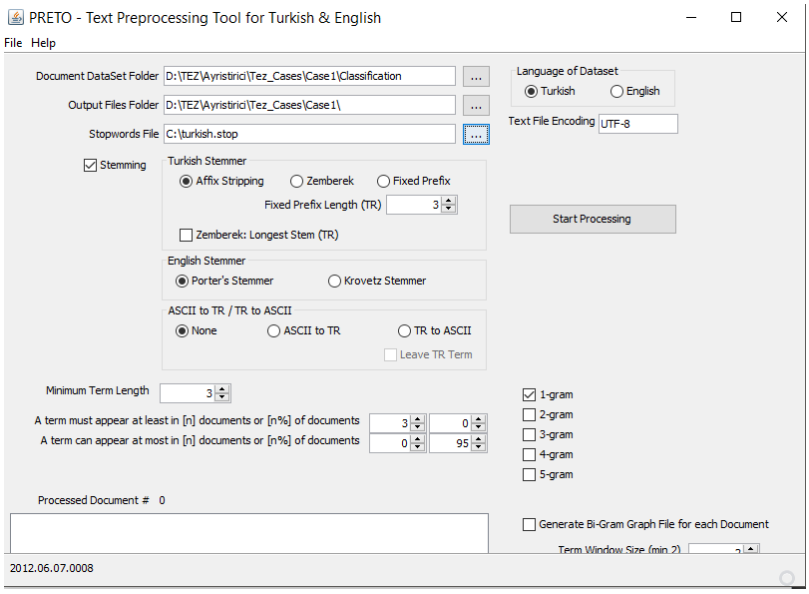
Başlık	**Açıklama**	**Sınıf**
TARİH FORMATINDA HATA	BT SİSTEM LİSTE SORGULAR/POLİÇE HASAR LİSTESİ YENİ alanında tarih aralıkları ile rapor alındığında exelde ki tarih formatlarında hata olmakta ve düzeltilememkte düzeltilmesi için desteğinizi rica ederiz.	Düşük
SAĞLIK REASÜRANS RAPORU REASÜRANS KOLONU YUVARLAMA HK.	SAĞLIK REASÜRANS RAPORU REASÜRANS KOLONUNDAKİ TUTARLAR LE OLUŞLAN FİŞLERDEKİ TUTARLAR ARASINDA KURUŞ FARKLARI BULUNMAKTADIR. KONTROLÜNÜ VE RAPORUN FİŞE YANSIYAN ŞEKLİ İLE DÜZENLENMESİNİ RİCA EDERİZ.	Orta
Acente Tecdit Listesi	Merhaba, Acente tecdit listesinden bölge kodu olarak Bankasürans departmanı görünmemektedir. İlgili hatanın giderilmesi için desteğinizi rica ederim. Saygılarımla,	Orta
346 - Mercedes Kasko baz update	K03 G04 T 59,535 Binde 59,535 olan bazın 54 olarak acilen düzeltilmesi gerekmektedir. Teşekkürler	Yüksek

Veri Ön İşleme Uygulaması

Veri setinde bulunan gürültü ve kirli veriyi ön işleme yöntem ile minimuma indirmek gerekmektedir. Visual Studio 2013 üzerinde yazılan bir C# konsol uygulaması ile önce işlenmemiş ham veri seti çekilmiştir. Daha sonra bu veri seti üzerinde bulunan harf dışında her türlü veri silinmiştir. Böylece hem kişisel bilgiler (TCKN, VKN gibi) temizlenmiş hem de yapılacak ölçümlerin tutarlı olması sağlanmıştır. Bu işlem sonrasında tekrar bu program ile veri setinde bulunan veriler sınıflarına göre 3 farklı klasöre, her talep bir txt olacak şekilde kaydedilmiştir. Böylece mevcut işlenmiş veri PRETO'ya hazır hale gelmiştir.

PRETO

PRETO, Türkçe metinler üzerinde metin madenciliği ön işleme uygulamaları için geliştirilmiş açık kaynak kodlu bir araçtır. Program içerisinde stemming, durdurma kelimeleri filtreleme, istatistiksel terim filtreleme, alınacak terimler için n-gram ile belirleme gibi metin madenciliği ön işleme adımları mevcuttur. PRETO'ya verilen veri seti üzerinde işlemler yaptıktan sonra çıktı olarak doküman terim matrisi (docbyterm), ağırlıklandırılmış doküman terim matrisi (docbyterm.tfidf) ve normalize edilmiş ağırlıklandırılmış doküman terim matrisi (docbyterm.tfidf.norm) vermektedir. PRETO'ya ait ekran görüntüsü Şekil 4'te verilmiştir.



Şekil 4 - PRETO Ekran Görüntüsü

Mevcut veri PRETO aracılığıyla docbyterm.mat, docbyterm.tfidf.mat, docbyterm.tfidf.norm.mat formatında dosyalara dönüştürülerek kısmen sınıflandırma algoritmaları uygulanmaya hazırlanmış hale gelmiştir. PRETO uygulamasından çıkan çıktıların WEKA uygulamasına hazır hale gelmesi için Visual Studio 2013 üzerinde C# konsol uygulaması yazılmıştır. Bu uygulama, mevcut veri üzerinde yaptığı değişiklikler ile WEKA dosya formatı olan ARFF formatına çevirmektedir.

WEKA

"Waikato Environment for Knowledge Analysis" baş harflerinden oluşan WEKA, metin madenciliği makine öğrenmesi algoritmaları barındıran Java tabanlı açık kaynak kodlu bir araçtır.

WEKA programı içerisinde 49 doküman ön işleme aracı, 76 sınıflandırma/regresyon algoritması, 8 kümeleme algoritması, 15 öznitelik/alt küme değerlendirici, 10 özellik seçimi için arama algoritması ve ilişkilendirme kuralı bulmak için 3 adet algoritma bulunmaktadır (Abd-ur-Rehman, 2009).

WEKA dosyaları ARFF ve CSV formatında dosyaları destekler. ARFF dosyasında ilk satırda @relation veri setinin tanımı tutulmaktadır. Sonraki satırlarda @attribute yani özellikler tutulmaktadır. Metin madenciliğinde her bir terim/kelime bir attribute olarak değerlendirilmektedir. Terimler de tanımlandıktan sonra @data bölümünde mevcut veri setinin terimlere göre bilgileri yer almaktadır. Örnek bir ARFF dosyası Şekil 5'te gösterilmiştir.

```
@relation havatahmini
@attribute nem numeric
@attribute sıcaklık numeric
@attribute basınç numeric
@attribute tahmin numeric
@data
53,25,1013,1
41,22,1011,-1
54,18,1012,-1
67,23,1000,1
```

Şekil 5 - WEKA ARFF Dosya Örneği

BÖLÜM 4. DENEYSEL SONUÇLAR VE YORUMLAR

Bu bölümde, mevcut veri setinin farklı özelliklerle önişleme işlemlerinden geçtikten sonra, farklı algoritmalarla çıkan deneysel sonuçlar ve sonuçlara ait yorumlara yer verilmiştir.

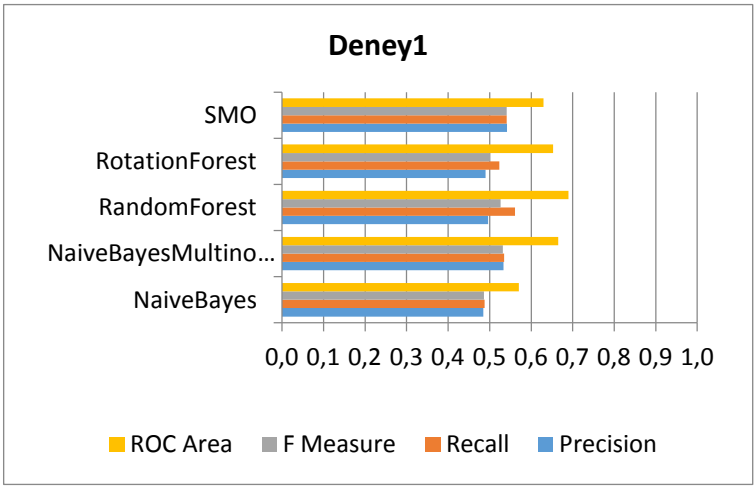
Deneysel Sonuçlar

Bu bölümde tez çalışmasında yapılan testlerin özellikleri ve çıktılarına yer verilmiştir.

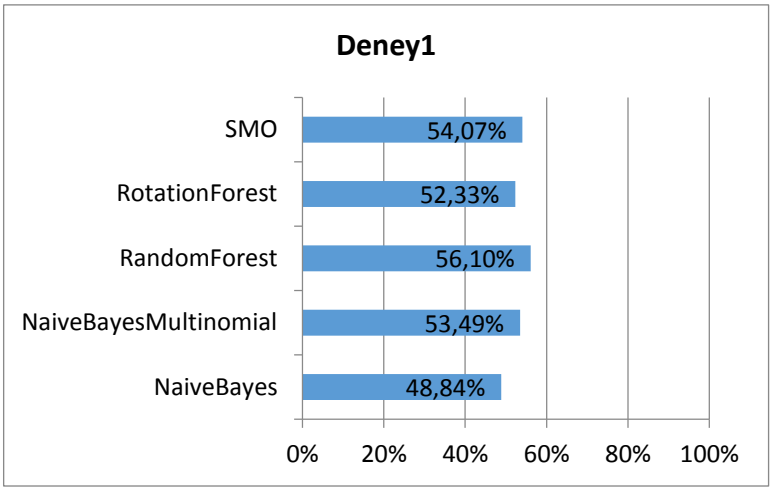
Deney-1

Bu deneyde 43 düşük, 150 orta, 151 yüksek öncelikli sınıflandırılmış veri seti bulunmaktadır. Bu veri seti Affix Stripping ile 1-gram kelimelik stemming işlemine sokulmuştur. Sonucunda 482 adet terim elde edilmiştir.

WEKA uygulamasında bulunan 5 farklı algoritma varsayılan parametreler ile 10 kat çapraz doğrulama yöntemi ile test edildiğinde şekildeki değerler elde edilmiştir.



Şekil 6 – Deney1 Algoritma Karşılaştırma Grafiği

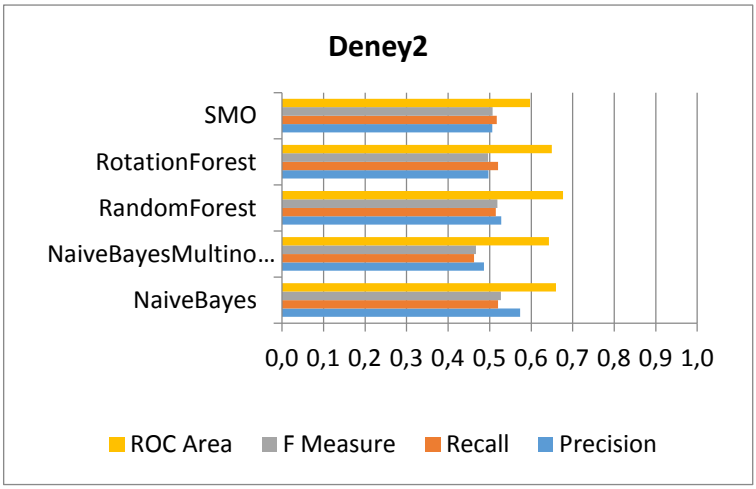


Şekil 7 – Deney1 Algoritma Başarı Oranı Grafiği

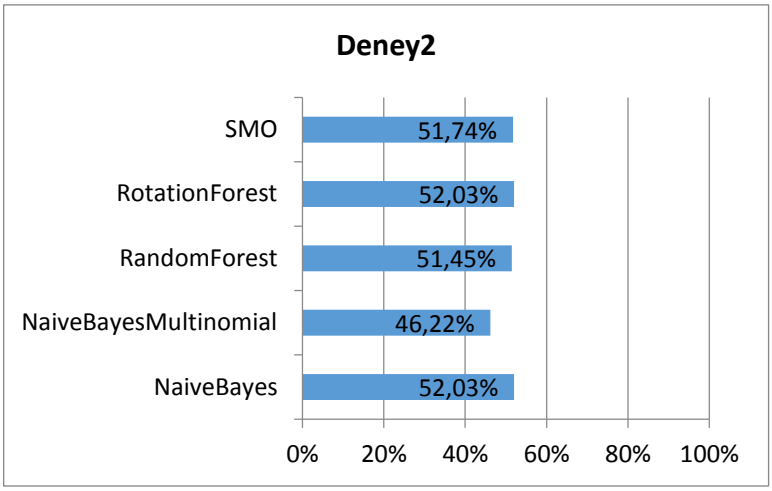
Deney-2

Bu deneyde 43 düşük, 150 orta, 151 yüksek öncelikli sınıflandırılmış veri seti bulunmaktadır. Bu veri seti Affix Stripping ile 2-gram kelimelik stemming işlemine sokulmuştur. Sonucunda 220 adet terim elde edilmiştir.

WEKA uygulamasında bulunan 5 farklı algoritma değerleri değiştirilmeden 10-Çapraz Doğrulama yöntemi ile test edildiğinde şekildeki değerler elde edilmiştir.



Şekil 8 - Deney2 Algoritma Karşılaştırma Grafiği

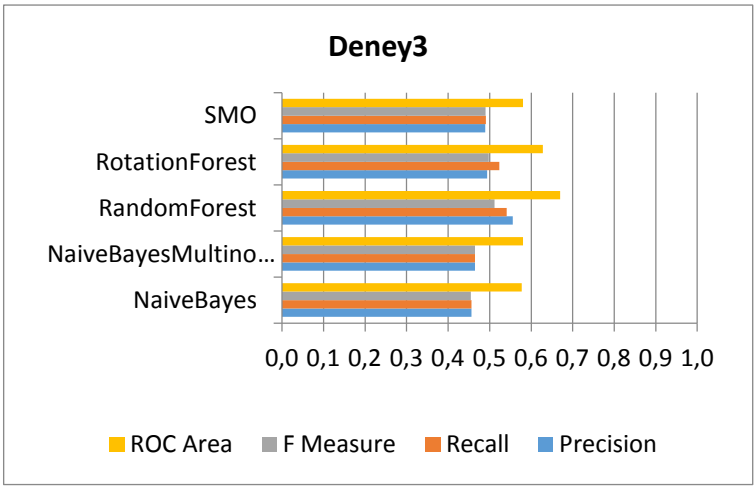


Şekil 9 - Deney2 Algoritma Başarı Oranı Grafiği

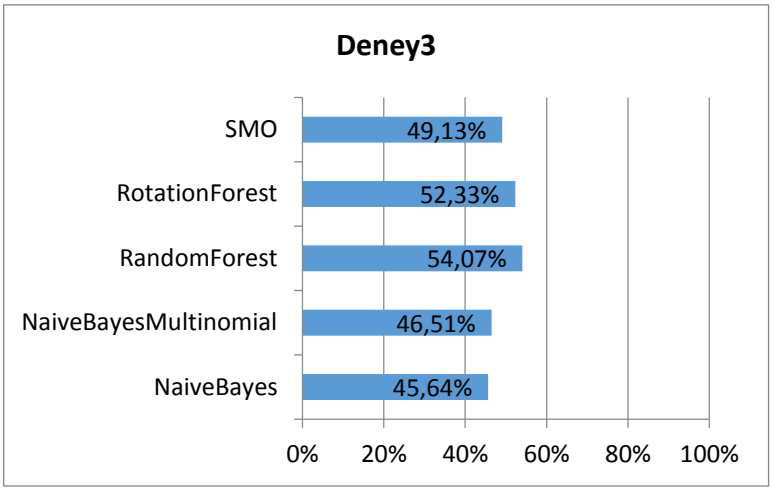
Deney-3

Bu deneyde 43 düşük, 150 orta, 151 yüksek öncelikli sınıflandırılmış veri seti bulunmaktadır. Bu veri seti Zemberek ile 1-gram kelimelik stemming işlemine sokulmuştur. Sonucunda 344 adet terim elde edilmiştir.

WEKA uygulamasında bulunan 5 farklı algoritma değerleri değiştirilmeden 10-Çapraz Doğrulama yöntemi ile test edildiğinde şekildeki değerler elde edilmiştir.



Şekil 10 - Deney3 Algoritma Karşılaştırma Grafiği

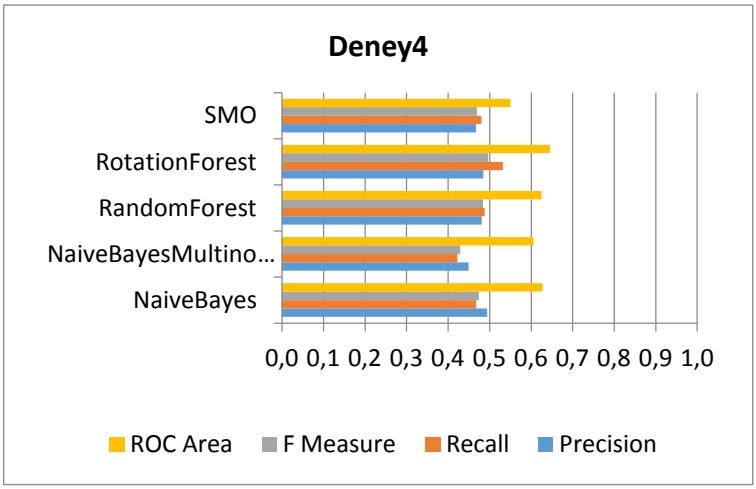


Şekil 11 - Deney3 Algoritma Başarı Oranı Grafiği

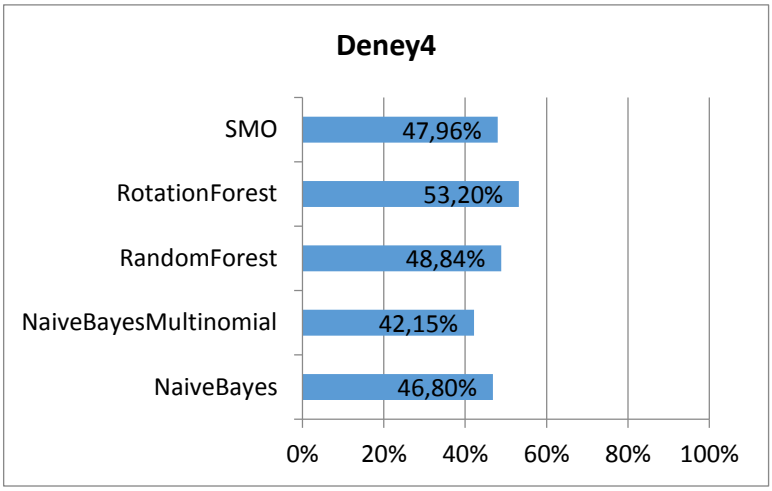
Deney-4

Bu deneyde 43 düşük, 150 orta, 151 yüksek öncelikli sınıflandırılmış veri seti bulunmaktadır. Bu veri seti Zemberek ile 2-gram kelimelik stemming işlemine sokulmuştur. Sonucunda 292 adet terim elde edilmiştir.

WEKA uygulamasında bulunan 5 farklı algoritma değerleri değiştirilmeden 10-Çapraz Doğrulama yöntemi ile test edildiğinde şekildeki değerler elde edilmiştir.



Şekil 12 - Deney4 Algoritma Karşılaştırma Grafiği

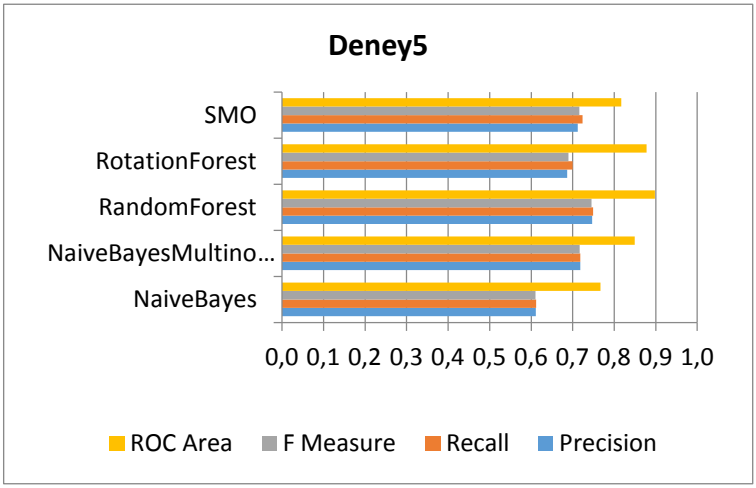


Şekil 13 - Deney4 Algoritma Başarı Oranı Grafiği

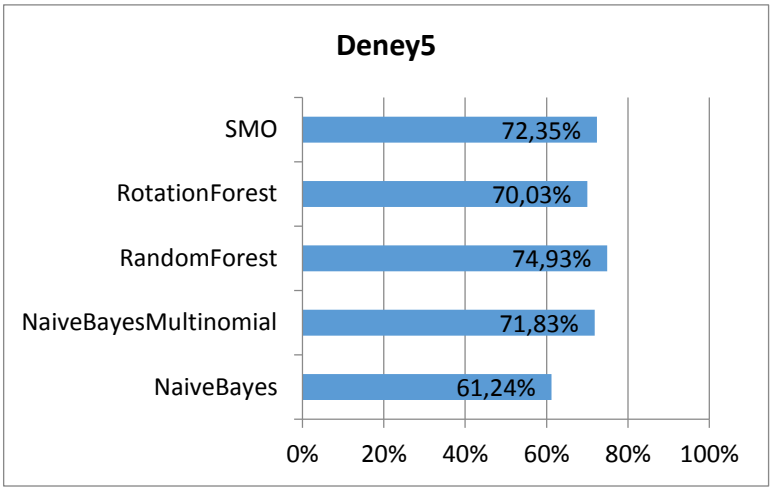
Deney-5

Bu deney veri setinde 43 adet bulunan düşük sınıflandırılmış veri 3 kere çoklanarak ele alınmıştır. Orta ve yüksek sınıflandırılmış veriler ise eşitlik olması için 129 adete düşürülmüştür. Bu veri seti Affix Stripping ile 1-gram kelimelik stemming işlemine sokulmuştur. Sonucunda 664 adet terim elde edilmiştir.

WEKA uygulamasında bulunan 5 farklı algoritma değerleri değiştirilmeden 10-Çapraz Doğrulama yöntemi ile test edildiğinde şekildeki değerler elde edilmiştir.



Şekil 14 - Deney5 Algoritma Karşılaştırma Grafiği

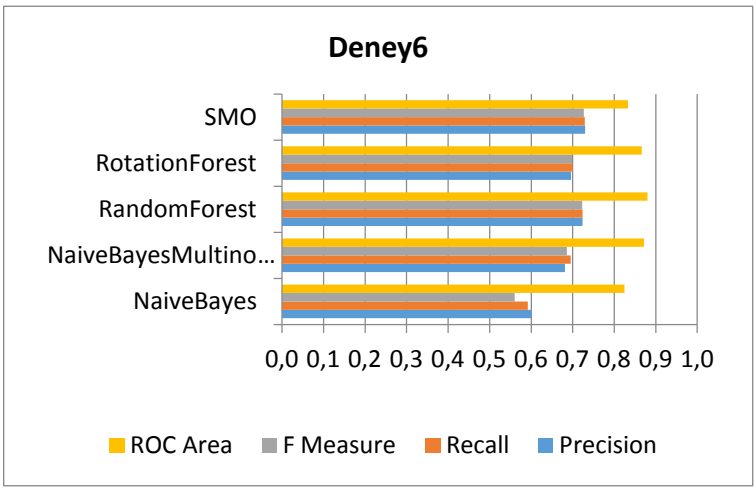


Şekil 15 – Deney5 Algoritma Başarı Oranı Grafiği

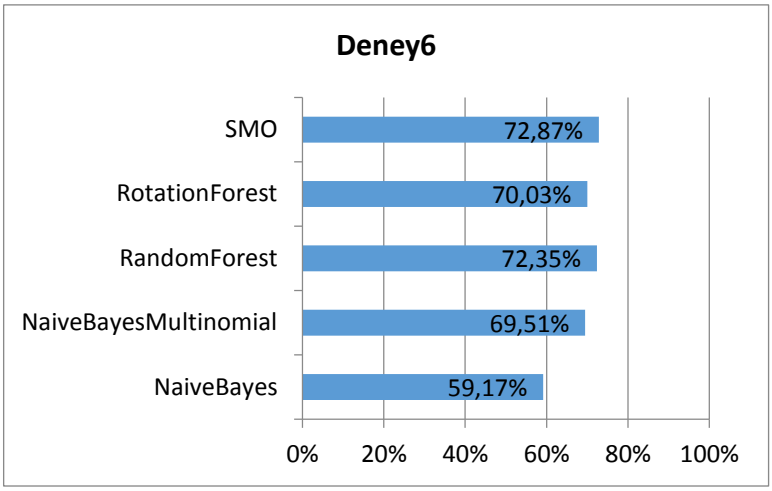
Deney-6

Bu deney veri setinde 43 adet bulunan düşük sınıflandırılmış veri 3 kere çoklanarak ele alınmıştır. Orta ve yüksek sınıflandırılmış veriler ise eşitlik olması için 129 adete düşürülmüştür. Bu veri seti Affix Stripping ile 2-gram kelimelik stemming işlemine sokulmuştur. Sonucunda 914 adet terim elde edilmiştir.

WEKA uygulamasında bulunan 5 farklı algoritma değerleri değiştirilmeden 10-Çapraz Doğrulama yöntemi ile test edildiğinde şekildeki değerler elde edilmiştir.



Şekil 16 - Deney6 Algoritma Karşılaştırma Grafiği

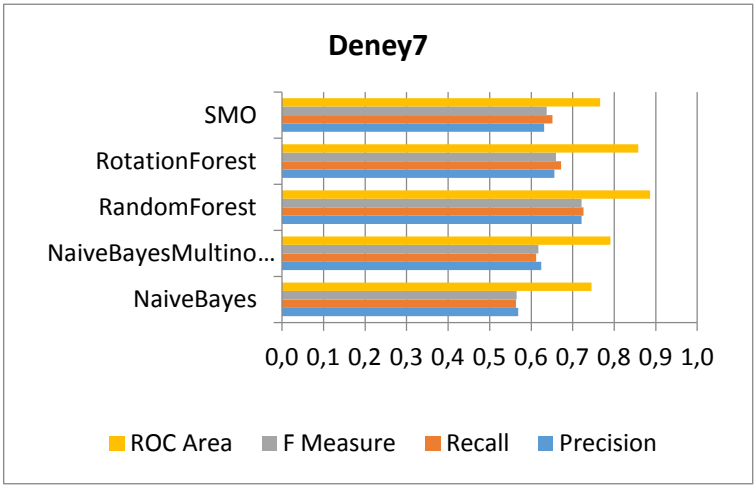


Şekil 17 - Deney6 Algoritma Başarı Oranı Grafiği

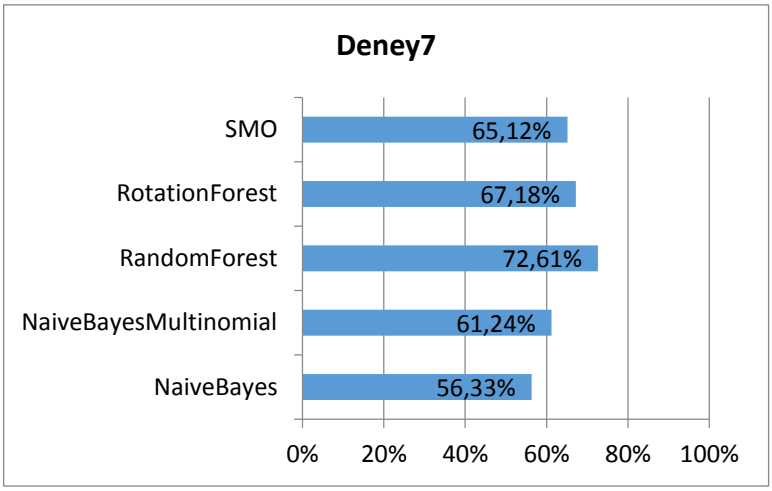
Deney-7

Bu deney veri setinde 43 adet bulunan düşük sınıflandırılmış veri 3 kere çoklanarak ele alınmıştır. Orta ve yüksek sınıflandırılmış veriler ise eşitlik olması için 129 adete düşürülmüştür. Bu veri seti Zemberek ile 1-gram kelimelik stemming işlemine sokulmuştur. Sonucunda 428 adet terim elde edilmiştir.

WEKA uygulamasında bulunan 5 farklı algoritma değerleri değiştirilmeden 10-Çapraz Doğrulama yöntemi ile test edildiğinde şekildeki değerler elde edilmiştir.



Şekil 18 - Deney7 Algoritma Karşılaştırma Grafiği

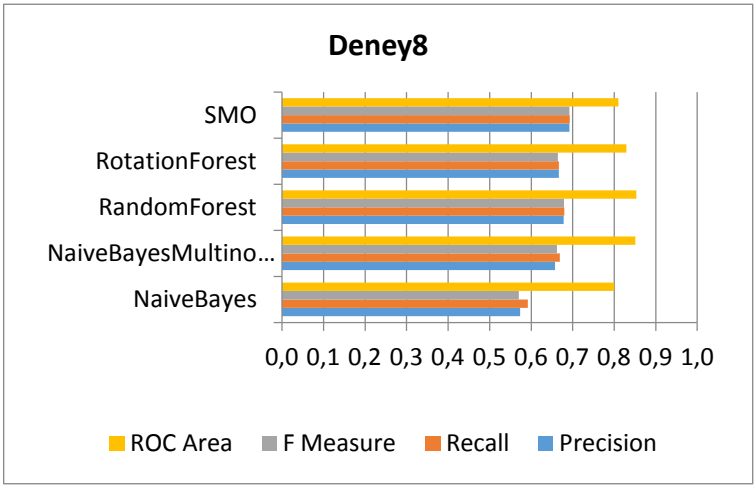


Şekil 19 – Deney7 Algoritma Başarı Oranı Grafiği

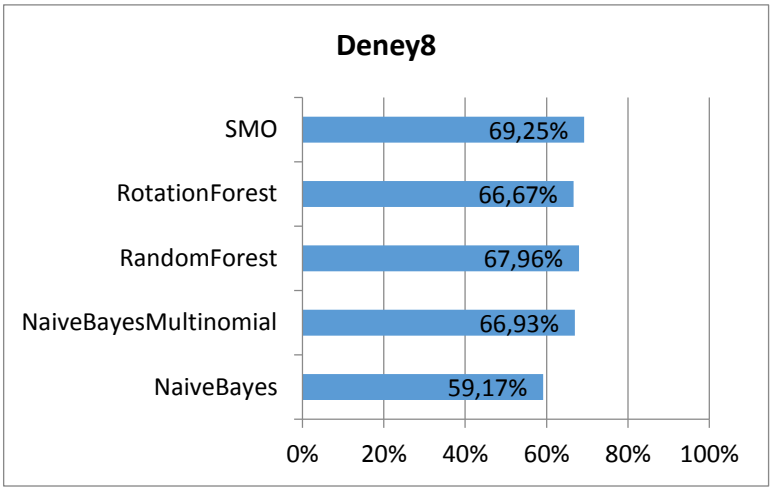
Deney-8

Bu deney veri setinde 43 adet bulunan düşük sınıflandırılmış veri 3 kere çoklanarak ele alınmıştır. Orta ve yüksek sınıflandırılmış veriler ise eşitlik olması için 129 adete düşürülmüştür. Bu veri seti Zemberek ile 2-gram kelimelik stemming işlemine sokulmuştur. Sonucunda 871 adet terim elde edilmiştir.

WEKA uygulamasında bulunan 5 farklı algoritma değerleri değiştirilmeden 10-Çapraz Doğrulama yöntemi ile test edildiğinde şekildeki değerler elde edilmiştir.



Şekil 20 - Deney8 Algoritma Karşılaştırma Grafiği



Şekil 21 - Deney8 Algoritma Başarı Oranı Grafiği

Yorumlar

Deney-1 sonuçlarında mevcut veriden elde edilen %56,10 doğru sınıflandırma ve 0,526 F-Skoru ile Rasgele Orman algoritması bu test için en iyi sonucu vermiştir.Test edilen 344 verinin 193 adedi doğru sınıflandırılmış olup, 151 adet veri yanlış sınıflandırılmıştır.

Deney-2 sonuçlarında mevcut veriden elde edilen %52,03 doğru sınıflandırma ve 0,527 F-Skoru ile NaiveBayes algoritması bu test için en iyi sonucu vermiştir.Test edilen 344 verinin 179 adedi doğru sınıflandırılmış olup, 165 adet veri yanlış sınıflandırılmıştır.

Deney-3 sonuçlarında mevcut veriden elde edilen %54,07 doğru sınıflandırma ve 0,512 F-Skoru ile Rasgele Orman algoritması bu test için en iyi sonucu vermiştir.Test edilen 344 verinin 186 adedi doğru sınıflandırılmış olup, 158 adet veri yanlış sınıflandırılmıştır.

Deney-4 sonuçlarında mevcut veriden elde edilen %53,20 doğru sınıflandırma ve 0,496 F-Skoru ile Rotasyon Ormanı algoritması bu test için en iyi sonucu vermiştir.Test edilen 344 verinin 183 adedi doğru sınıflandırılmış olup, 161 adet veri yanlış sınıflandırılmıştır.

Yapılan ilk 4 test deneyinde dengesiz sınıflandırma olmuştur. En az veriye sahip düşük sınıfında bulunan 43 veri 3 kere çoklanarak 129 adet düşük sınıflı veriye çıkarılmıştır. Orta ve yüksek sınıfların sayısı ise 129 adete düşürülmüştür. Böylelikle

tüm sınıflarda eşit sayıda veri bulunması sağlanmıştır. Sonraki yapılan 4 test bu veri setini baz alarak yapılmıştır.

Deney-5 sonuçlarında mevcut veriden elde edilen %74,93 doğru sınıflandırma ve 0,745 F-Skoru ile Rasgele Orman algoritması bu test için en iyi sonucu vermiştir.Test edilen 387 veride 290 adedi doğru sınıflandırılmış veri olup, 97 adet veri yanlış sınıflandırılmıştır.

Deney-6 sonuçlarında mevcut veriden elde edilen %72,35 doğru sınıflandırma ve 0,723 F-Skoru ile Rasgele Orman algoritması bu test için en iyi sonucu vermiştir.Test edilen 387 veride 280 adedi doğru sınıflandırılmış veri olup, 107 adet veri yanlış sınıflandırılmıştır.

Deney-7 sonuçlarında mevcut veriden elde edilen %72,61 doğru sınıflandırma ve 0,721 F-Skoru ile Rasgele Orman algoritması bu test için en iyi sonucu vermiştir.Test edilen 387 veride 281 adedi doğru sınıflandırılmış veri olup, 106 adet veri yanlış sınıflandırılmıştır.

Deney-8 sonuçlarında mevcut veriden elde edilen %69,25 doğru sınıflandırma ve 0,692 F-Skoru ile SMO algoritması bu test için en iyi sonucu vermiştir.Test edilen 387 veride 268 adedi doğru sınıflandırılmış veri olup, 119 adet veri yanlış sınıflandırılmıştır.

Yapılan deneysel çalışmalarda; dengesiz sınıflandırılmış algoritmaların dengelenmiş sınıflara göre daha düşük başarılı olduğu görülmüştür. Sonuçları incelediğimizde, Deney-5 başarı oranı olarak en iyi sonucu elde etmiş olup, Rasgele Orman algoritması bu çalışmada en iyi sonucu veren algoritma olmuştur.

BÖLÜM 5. SONUÇ

Yapılan bu çalışmada metin madenciliği ile yazılım geliştirme taleplerinin öncelik sınıflandırması ele alınmıştır. Çalışmada metin madenciliğinde sıkça kullanılan algoritmalarının aralarında karşılaştırma yapılmış ve en iyi sonucun elde edilmesi hedeflenmiştir. En iyi sınıflandırma algoritmasının Rasgele Orman olduğu gözlemlenmiştir. Ele alınan veri setinde sınıf elemanlarının sayıca eşit olmaması, dengesiz sınıflandırma sorunlarının sonuçları nasıl etkilediğini ve hangi yöntem ile sınıfların dengelendiği açıklamıştır. Yapılan testlerde ilk dört test ile sonraki dört test arasındaki başarı oranı farkı ve F-Skoru değerleri dengesiz sınıflandırmanın etkisini göstermiştir.

Bu çalışmada ele alınan veri setinin kısıtlı olması sebebiyle sonuçlar birbirine yakın çıkmıştır. Kullanılacak veri setinin daha fazla olması ve sınıflara eşit dağınıklıkta olması ile daha iyi sonuçlar elde edilebilir. Farklı metin madenciliği ön işleme yöntemleri ve teknikleri ile veri işlenerek, elde edilen işlenmiş veri farklı algoritmalar kullanılarak denenebilir.

İleri dönemlerde yapılacak çalışmalarda, talepler önceliklendirilirken elle sınıflandırılması yerine otomatikleştirilmesi BT birimi için daha iyi planlama yapılmasında yardımcı olabilir.

EK'LER

acaba	biz	dün	kez	olduğu	söyledi	özel
acil	bizden	eden	ki	olduğunu	tam	üzerine
altmış	bizi	elli	kim	on	tarafından	üç
altı	bizim	en	kimden	ona	tek	şey
ama	bu	en gibi	kime	ondan	trilyon	şeyden
ancak	bugün	eski	kimi	onlar	tüm	şeyi
arasında	buna	etti	konusunda	onlardan	var	şeyler
artık	bunda	eğer	kırk	onlari	ve	şimdi
aynı	bundan	gibi	mi	onların	veya	şu
bana	bunu	göre	milyar	onu otuz	ya	şuna
bazı	bunun	gün	milyon	ortaya	yani	şunda
başka	böyle	hem	mu	pek	yaptığı	şundan
belki	bütün	hep	mü	sadece	yapılan	şunu
ben	büyük	hepsi	mı	sanki	yedi	şöyle
benden	da	her	nasıl	sekiz	yeni	bey
beni	daha	hiç	ne	seksen	yer	hanım
benim	dahi	iki	neden	sen	yetmiş	hn
beş	de	ile	nedeniyle	senden	yine	merhaba
bile	dedi	ilk	nerde	seni	yirmi	selam
bin	defa	ise	nerede	senin	yok	sorun
bir	devam	iyi	nereye	siyasi	yüz	hata
biri	değil	için	niye	siz	yüzde	login
birkaç	diye	iş	niçin	sizden	zaman	problem
birkez	diğer	kadar	o	sizi	çok	sıkıntı
birlikte	doksan	karşı	olan	sizin	çünkü	poliçe
birşey	dokuz	katrilyon	olarak	son	önce	teklif
birşeyi	dört	kendi	oldu	sonra	önemli	basım

Ek–1 Durdurma Kelimeleri

KAYNAKÇA

Abd-ur-Rehman, S. M. (2009). WEKA & KNIME Open Source Machine Learning Tools.

Adrew McCallum, K. N. (1998). A Comparison of Event Models for Naive Bayes Text Classification.

Ahmet Afşin Akın, M. D. (2007). Zemberek, an Open Source Nlp Framework for Turkic Languages.

Andreas Hotho, A. N. (2005). A Brief Survey of Text Mining.

Arslantekin, S. (2003). Veri Madenciliği ve Bilgi Merkezleri. s. 369.

Bekir Parlak, A. K. (2015). Tıbbi Dokümanların Hastalıklara Göre Sınıflandırılması.

Döven, S. (2013). Metin madenciliği ile dokümanlar arasındaki benzerliklerin bulunması.

Elif Kartal, Z. Ö. (2017). Dengesiz Veri Setlerinde Sınıflandırma.

Eui-Hong Han, G. K. (2000). Centroid-Based Document Classification Algorithms: Analysis & Experimental Results.

Fidan, Ü. (2013). Destek Vektör Makineleri ile Doküman Sınıflandırma.

Gülşen Eryiğit, E. A. (2004). An Affix Stripping Morphological Analyzer for Turkish.

H.Kemal Yıldız, M. G. (2007). Metin Sınıflandırmada Yeni Özellik Çıkarımı.

Haibo He, E. A. (2009). Learning from Imbalanced Data.

Hayri Sever, B. O. (2018). *Veri Yapıları ve Algoritmalar*. 2018 tarihinde https://akagun.wordpress.com/bilisim-teknolojileri/2-hafta/ adresinden alındı

İsmail Çölkesen, T. Y. (2014). Rotasyon Orman Algoritması ile Yüksek Çözünürlüklü Multispektral Uydu Görüntülerinin Sınıflandırılması.

James Sanger, R. F. (2002). *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data.*

Jiawei Han, M. K. (2012). *Data Mining Concepts and Techniques Third Edition.*

Jure Leskovec, A. R. (2011). *Mining of Massive Datasets.*

Kaşıkçı, T. (2014). Metin madenciliği ile e-ticaret sitelerinin belirlenmesi.

Kuzucu, K. (2015). Müşteri Memnuniyeti Belirlenmesi için Metin Madenciliği Tabanlı Bir Yazılım Aracı.

Mayuri S. Shelke, D. R. (2017). A Review on Imbalanced Data Handling Using Undersampling and Oversampling Technique.

Mehmet Fatih Karaca, S. G. (2012). ColumnREADY: Internet gazeteleri köşe yazılarını hazırlama uygulama yazılımı.

Mitchell, T. M. (1997). *Machine Learning.*

Platt, J. C. (1998). Fast Training of Support Vector Machines using Sequential Minimal Optimization.

Powers, D. M. (2007). Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation.

Randolph E. Bank, C. C. (2001). Sparse Matrix Multiplication Package.

Sancar, Y. (2016). Metin Madenciliği Kullanılarak Talep Tanıma ve Yönlendirme Sistemi.

Seçkin, K. (2011). Metin Madenciliğinde Kullanılan Yöntemlerin Karşılaştırılması: Siyasi Parti Liderlerinin Grup Genel Toplantı Konuşmaları ile Bir Uygulama.

Swets, J. A. (1996). Signal Detection Theory And Roc Analysis In Psychology And Diagnostics Collected Papers.

Tina R. Patil, M. S. (2013). Performance Analysis of Naive Bayes and J48.

Türkoğlu, F. (2006). Melez Yaklaşımlarla Türkçe Dokümanlarda Yazar Tanıma.

Varol, M. (2011). Metin Madenciliği Yöntemlerini Kullanarak Türkçe Dokümanlarda Tür ve Yazar Tanıma.

Veri Bilimcisi. (2017, 07 14). *Doğruluk Ölçümü (Accuracy Measure)*. 04 24, 2018 tarihinde Veri Bilimcisi: https://veribilimcisi.com/2017/07/14/dogruluk-olcumu-nasil-yapilir-accuracy-measure/ adresinden alındı

Volkan Tunalı, T. B. (2012). Türkçe Metinlerin Kümelenmesinde Farklı Kök Bulma Yöntemlerinin Etkisinin Araştırılması.