EGE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

(DOKTORA TEZİ)

GÖRÜŞ SINIFLANDIRMA İÇİN MAKİNE ÖĞRENMESİ ALGORİTMALARINA DAYALI BİR YÖNTEM TASARIMI VE GERÇEKLEŞTİRİMİ

Aytuğ ONAN

Tez Danışmanı : Prof. Dr. M. Serdar KORUKOĞLU

İkinci Danışmanı : Yrd. Doç. Dr. Hasan BULUT

Bilgisayar Mühendisliği Anabilim Dalı

Sunuş Tarihi : 28.07.2016

Bornova-İZMİR

2016

Aytuğ ONAN tarafından Doktora tezi olarak sunulan "**Görüş Sınıflandırma İçin Makine Öğrenmesi Algoritmalarına Dayalı Bir Yöntem Tasarımı ve Gerçekleştirimi**" başlıklı bu çalışma EÜ Lisansüstü Eğitim ve Öğretim Yönetmeliği ile EÜ Fen Bilimleri Enstitüsü Eğitim ve Öğretim Yönergesi'nin ilgili hükümleri uyarınca tarafımızdan değerlendirilerek savunmaya değer bulunmuş ve **28.07.2016** tarihinde yapılan tez savunma sınavında aday **oybirliği**/~~oyçokluğu~~ ile başarılı bulunmuştur.

Jüri Üyeleri:		**İmza**
Jüri Başkanı	**: Prof. Dr. Serdar KORUKOĞLU**	**.....**
Raportör Üye:	**Prof. Dr. Aybars UĞUR**	**.....**
Üye	**: Doç. Dr. Vecdi AYTAÇ**	**.....**
Üye	**: Yrd. Doç. Dr. Serkan BALLI**	**.....**
Üye	**: Yrd. Doç. Dr. Korhan KARABULUT**	**.....**

EGE ÜNİVERSİTESİ FEN BİLİMLERİ ENSTİTÜSÜ

ETİK KURALLARA UYGUNLUK BEYANI

EÜ Lisansüstü Eğitim ve Öğretim Yönetmeliğinin ilgili hükümleri uyarınca Doktora Tezi olarak sunduğum "**Görüş Sınıflandırma İçin Makine Öğrenmesi Algoritmalarına Dayalı Bir Yöntem Tasarımı ve Gerçekleştirimi**" başlıklı bu tezin kendi çalışmam olduğunu, sunduğum tüm sonuç, doküman, bilgi ve belgeleri bizzat ve bu tez çalışması kapsamında elde ettiğimi, bu tez çalışmasıyla elde edilmeyen bütün bilgi ve yorumlara atıf yaptığımı ve bunları kaynaklar listesinde usulüne uygun olarak verdiğimi, tez çalışması ve yazımı sırasında patent ve telif haklarını ihlal edici bir davranışımın olmadığını, bu tezin herhangi bir bölümünü bu üniversite veya diğer bir üniversitede başka bir tez çalışması içinde sunmadığımı, bu tezin planlanmasından yazımına kadar bütün safhalarda bilimsel etik kurallarına uygun olarak davrandığımı ve aksinin ortaya çıkması durumunda her türlü yasal sonucu kabul edeceğimi beyan ederim.

28 / 07 / 2016

İmzası

Aytuğ ONAN

ÖZET

GÖRÜŞ SINIFLANDIRMA İÇİN MAKİNE ÖĞRENMESİ ALGORİTMALARINA DAYALI BİR YÖNTEM TASARIMI VE GERÇEKLEŞTİRİMİ

ONAN, Aytuğ

Doktora Tezi, Bilgisayar Mühendisliği Anabilim Dalı
Tez Danışmanı: Prof. Dr. M. Serdar KORUKOĞLU
İkinci Danışmanı: Yrd. Doç. Dr. Hasan BULUT
Temmuz 2016, 137 sayfa

Görüş sınıflandırma, doğal dil işleme, makine öğrenmesi ve istatistik disiplinlerinden, yöntem, teknik ve araçların kullanılması ile metin belgelerinde yer alan öznel bilgilerin belirlenmesine yönelik bir araştırma alanıdır. Bu tez çalışması kapsamında, görüş madenciliği, bir metin sınıflandırma problemi olarak ele alınarak makine öğrenmesi yöntemleri aracılığıyla etkin görüş sınıflandırma yöntemleri geliştirilmiştir. Geliştirilen yöntemler üç temel çatı altında incelenebilir. Birincisi, metin sınıflandırmada karşılaşılan en önemli problemlerden biri olan yüksek boyutluluk ve seyrekliği ortadan kaldırmak amacıyla temel filtre tabanlı öznitelik seçim yöntemlerini etkin bir şekilde birleştiren genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçimi yöntemidir. İkinci olarak, sınıflandırıcı topluluğunda yer alan temel öğrenme algoritmalarının, topluluk çıktısına, doğru sınıflandırma başarımlarına göre katkı koymalarına yönelik, çok amaçlı diferansiyel gelişim algoritmasına dayalı ağırlıklı oylama sınıflandırıcı topluluğu birleştirme kuralıdır. Üçüncü olarak ise, topluluk öğrenmesi sürecinin, topluluk budama aşamasında, uygun öğrenme algoritmalarının seçilmesine yönelik yöntem geliştirilmesidir. Geliştirilen sınıflandırıcı topluluğu budama yönteminde, ortak kümeleme ve metasezgisel aramadan yararlanılmıştır. Geliştirilen yöntemlere dayalı yeni bir sınıflandırma mimarisi önerilmiştir. Bu mimari ile geliştirilen yöntemler etkin bir şekilde birleştirilerek mevcut ve geliştirilen yöntemlerin bireysel performanslarına kıyasla daha iyi başarım elde edilmiştir.

Anahtar sözcükler: Görüş sınıflandırma, sınıflandırıcı topluluğu, çoğunluk oylaması, çok amaçlı eniyileme, öznitelik seçimi, topluluk budama, sıra birleştirme.

ABSTRACT

THE DESIGN AND IMPLEMENTATION OF A METHOD FOR OPINION CLASSIFICATION BASED ON MACHINE LEARNING ALGORITHMS

ONAN, Aytuğ

Ph.D. in Computer Engineering
Supervisor: Prof. Dr. M. Serdar KORUKOĞLU
Co-Supervisor: Assist. Prof. Dr. Hasan BULUT
July 2016, 137 pages

Opinion classification, which utilizes methods, techniques and tools from natural language processing, machine learning and statistics, is a research field to determine subjective information in the text documents. In this thesis, opinion mining is regarded as a text classification problem to build an efficient opinion classification scheme based on machine learning methods. The developed methods can be grouped into three categories. First, a feature selection method based on genetic rank aggregation, which integrates individual filter-based feature selection methods in an effective way, is presented to overcome the high dimensionality and sparsity problems encountered in text classification. Secondly, a classifier ensemble combination rule based on multi-objective differential evolution algorithm is presented so that the base learning algorithms of the classifier ensemble can contribute to the final outcome of the ensemble according to their predictive performance. Thirdly, an ensemble pruning scheme is presented to obtain an appropriate subset of classifiers from the ensemble. In the proposed pruning scheme, consensus clustering and metaheuristic search are utilized. A novel classification architecture is proposed based on the developed methods. With this framework, developed methods are combined in an effective way and the predictive performance of the framework is enhanced compared to the individual performances of the standard and the developed methods.

Keywords: Opinion classification, classifier ensemble, majority voting, multi-objective optimization, feature selection, ensemble pruning, rank aggregation.

TEŞEKKÜR

Bu tez çalışması süresince, bilimsel bilgi birikimi, deneyim ve önerilerinden yararlandığım tez danışmanlarım Sayın Prof. Dr. M. Serdar KORUKOĞLU'na ve Sayın Yrd. Doç. Dr. Hasan BULUT'a, tez izleme toplantılarında, tezin gelişimini izleyen ve değerli önerilerde bulunan Muğla Sıtkı Koçman Üniversitesi Bilişim Sistemleri Mühendisliği Bölümü Öğretim Üyesi Sayın Yrd. Doç. Dr. Serkan BALLI'ya ve Ege Üniversitesi Bilgisayar Mühendisliği Bölümü Öğretim Üyesi Sayın Doç. Dr. Vecdi AYTAÇ'a teşekkürlerimi sunarım.

Tez sürecinde desteklerini esirgemeyen aileme ve doktora tez çalışmasını, BİDEB 2211- Yurt İçi Lisansüstü (Doktora) Burs Programı kapsamında destekleyen TÜBİTAK Bilim İnsanı Destekleme Daire Başkanlığına teşekkür ederim.

İÇİNDEKİLER

Sayfa

İÇİNDEKİLER (devam)

Sayfa

İÇİNDEKİLER (devam)

Sayfa

İÇİNDEKİLER (devam)

Sayfa

İÇİNDEKİLER (devam)

Sayfa

ŞEKİLLER DİZİNİ

ŞEKİLLER DİZİNİ (devam)

Şekil Sayfa

ÇİZELGELER DİZİNİ

Çizelge Sayfa

ÇİZELGELER DİZİNİ (devam)

Çizelge Sayfa

KISALTMALAR DİZİNİ

Kısaltmalar	Açıklama
ACC	Doğru sınıflandırma oranı
AVP	Olasılıklar ortalaması
BES	Bagging sınıflandırıcı seçimi
BFS	En iyi önce arama algoritması
BLR	Bayesci lojistik regresyon
BWWV	En iyi-en kötü ağırlıklı oylama
CHI	Ki-kare ölçütü
EBA	İyileştirilmiş Borda sıra birleştirme
EM	Beklenti maksimizasyonu
ESM	Model kütüphanesinden topluluk seçimi algoritması
EWA	Üstel ağırlıklı birleştirme
GA	Genetik algoritma
Gain	Kazanç oranı
GS	Açgözlü arama
HRA	En yüksek sıra birleştirme
IG	Bilgi kazancı
KNN	K-en yakın komşu algoritması

KISALTMALAR DİZİNİ (devam)

Kısaltmalar	Açıklama
LDA	Lineer diskriminant analizi
LR	Lojistik regresyon
LRA	En düşük sıra birleştirme
MA	Ortalama sıra birleştirme
MAJ	Çoğunluk oylaması
MAP	Maksimum olasılık
MEA	Medyan sıra birleştirme
MIP	Minimum olasılık
NB	Naive Bayes
PCC	Pearson korelasyon katsayısı
PRP	Olasılıklar çarpımı
PS	Olasılıklı önem ölçütü
PSO	Parçacık sürüsü eniyilemesi
QBWWV	İkinci dereceden en iyi-en kötü ağırlıklı oylama
RBF	Radyal tabanlı fonksiyon ağları
RL	Relief algoritması
RORA	Gürbüz sıra birleştirme

KISALTMALAR DİZİNİ (devam)

Kısaltmalar	Açıklama
RRA	Round Robin sıra birleştirme
RWV	Ölçeklenmiş ağırlıklı oylama
SOM	Öz-düzenlemeli harita
SSA	Kararlılık seçimi
SU	Simetrik belirsizlik katsayısı
SVM	Destek vektör makineleri
SWV	Basit ağırlıklı oylama
TF	Terim sıklığı
TP	Terim varlığı

1. GİRİŞ

Görüş madenciliği (*duygu analizi*) (*opinion mining, sentiment analysis*), doğal dil işleme, makine öğrenmesi ve istatistik disiplinlerinden, yöntem, teknik ve araçların kullanılması ile metin belgelerinde yer alan görüş, duygu ve tutum gibi öznel bilgilerin belirlenmesine yönelik bir araştırma alanıdır. Web, hızla artan içeriği ile en önemli bilgi kaynaklarının başında yer almaktadır (Bhatia and Khalid, 2008). Web, metin, video gibi birçok farklı multimedya içeriğine sahip bir kaynaktır. Web'in sunduğu bilgi, hükümet, iş organizasyonu ve bireysel karar vericiler için değerli ve kullanışlı bir kaynaktır. Görüş sınıflandırma, belirli bir konuya yönelik kamusal görüşün belirlenmesi, belirli bir ürüne ilişkin pazar analizinin yapılması gibi birçok farklı alanda başarıyla uygulanabilmektedir (Zhang et al., 2009). Belirli bir konuya ilişkin kamusal görüşün belirlenmesi, politika ve siyasi uygulamaların şekillenmesinde etkili olabilir. Benzer şekilde, belirli bir ürüne yönelik görüş madenciliği ile ilgili ürüne ilişkin genel görüş ya da ürünün beğenilen/beğenilmeyen özellikleri çıkarılabilir. Görüşler, karar verme sürecinde etkileme gücü en yüksek unsurların başında gelmektedir. Kişilerin belirli bir konu ya da ürüne ilişkin görüşlerinin belirlenmesi, yönetim bilimleri, siyaset bilimi, ekonomi başta olmak üzere, birçok farklı disiplin için oldukça yararlıdır (Liu, 2012). Görüş madenciliği ile veri yapılandırılmış hale getirilerek, karar destek sistemleri ve bireysel karar vericiler için önemli bir kaynak işlevi görür (Fersini et al., 2014).

Görüş madenciliği, uygulandığı ayrıntı seviyesine göre, temel olarak belge seviyesi görüş madenciliği (*document-level opinion mining*), tümce seviyesi görüş madenciliği (*sentence-level opinion mining*) ve varlık/özellik seviyesi görüş madenciliği (*entity/aspect-level opinion mining*) olmak üzere üç temel sınıf altında incelenmektedir (Liu, 2012). Belge seviyesi görüş madenciliğinde, metin belgesinin genel görüş kutbunun (duygu yönü) belirlenmesi amaçlanır. Buna karşılık olarak, tümce seviyesi görüş madenciliğinde ise metin belgesinde yer alan belirli bir tümcenin duygu yönünün (pozitif, negatif ya da tarafsız bir ifade) belirlenmesi hedeflenir. Varlık/özellik seviyesi görüş madenciliğinde ise belirli bir varlığın (ürün ya da üzerinde görüş bildirilen başka bir kavram) farklı özelliklerine ilişkin duygu yönünün belirlenmesi söz konusudur.

Görüş madenciliği alanında gerçekleştirilen çalışmaları, sözlüğe dayalı yöntemler ve makine öğrenmesine dayalı yöntemler olmak üzere, iki temel bölüm altında incelemek mümkündür (Medhat et al., 2014). Sözlüğe dayalı yöntemlerde,

görüş sınıflandırma işlemi, daha önceden oluşturulmuş görüş sözcükleri içeren bir sözlüğe dayalı olarak gerçekleştirilir. Sözlüğe dayalı yöntemler, sözlük tabanlı ve derlem tabanlı yöntemler olmak üzere iki sınıf altında incelenmektedir. Sözlük tabanlı yöntemler ile metnin görüş kutbu belirlenirken, metinde geçen sözcük ve tümcelerin anlamsal oryantasyonlarına dayalı bir hesaplama gerçekleştirilir (Taboada et al., 2011). Derlem tabanlı yöntemlerde ise istatistiksel ya da semantik yöntemler aracılığıyla görüş kutbu belirlenmektedir.

Makine öğrenmesine dayalı yöntemler ile görüş madenciliği gerçekleştirilirken, görüş kutbu etiketlenmiş veri seti, makine öğrenmesi algoritmalarından birine uygulanarak sınıflandırma modeli oluşturulmakta; ardından, bu model, yeni örneklerin sınıflandırılmasında kullanılmaktadır. Makine öğrenmesine dayalı yöntemler ise öğreticili (*supervised*), yarı-öğreticili (*semi-supervised*) ve öğreticisiz (*unsupervised*) yöntemler olmak üzere üç temel sınıf altında incelenmektedir (Liu, 2012).

Öğreticili öğrenme yöntemleri, temel makine öğrenmesi yöntemlerinin, derlemdeki alana özgü etiketler ile eğitilmesi ile sınıflandırma modelini oluşturur. Öğreticili öğrenme yöntemlerinin düzgün bir şekilde çalışabilmesi için eğitim için kullanılan veri setinin yeterince büyük olması gerekmektedir.

Yarı-öğreticili öğrenme yöntemleri, öğreticili öğrenme yöntemlerinin, eğitim verisi ile test verisi arasındaki dağılışlar farklı olduğunda etkin sonuç vermemeleri, alana özgü olmaları; belirli bir alan için oluşturulan sınıflandırma modelinin, başka bir alanda uygulanabilmesi için, sınıflandırıcının yeniden eğitilmesini gerektirmesi gibi kısıtlarını ortadan kaldırmayı amaçlar (Lin, 2011).

Öğreticisiz/Zayıf Öğreticili öğrenme yöntemleri, öğreticili ya da yarı-öğreticili görüş madenciliği yöntemlerinde karşılaşılan alan bağımlılığı problemini ortadan kaldırmak için kullanılmaktadır. Öğreticisiz öğrenme yöntemlerinde, duygu sözlüğü, görüş kutbu belirlemek amacıyla ön bilgi olarak verilmektedir. Alan bağımsız duygu sözlükleri oluşturmanın, alana özgü derlem oluşturmadan daha az maliyetli olmasından faydalanılmaktadır (Lin, 2011).

Metin ve web madenciliğinde sıklıkla kullanılan makine öğrenmesi sınıflandırıcıları arasında, karar ağacı algoritmaları (C4.5, ID3 vb.), kural tabanlı sınıflandırma algoritmaları, istatistiksel sınıflandırıcılar (Naive Bayes algoritması), doğrusal sınıflandırıcılar, destek vektör makineleri ve yapay sinir

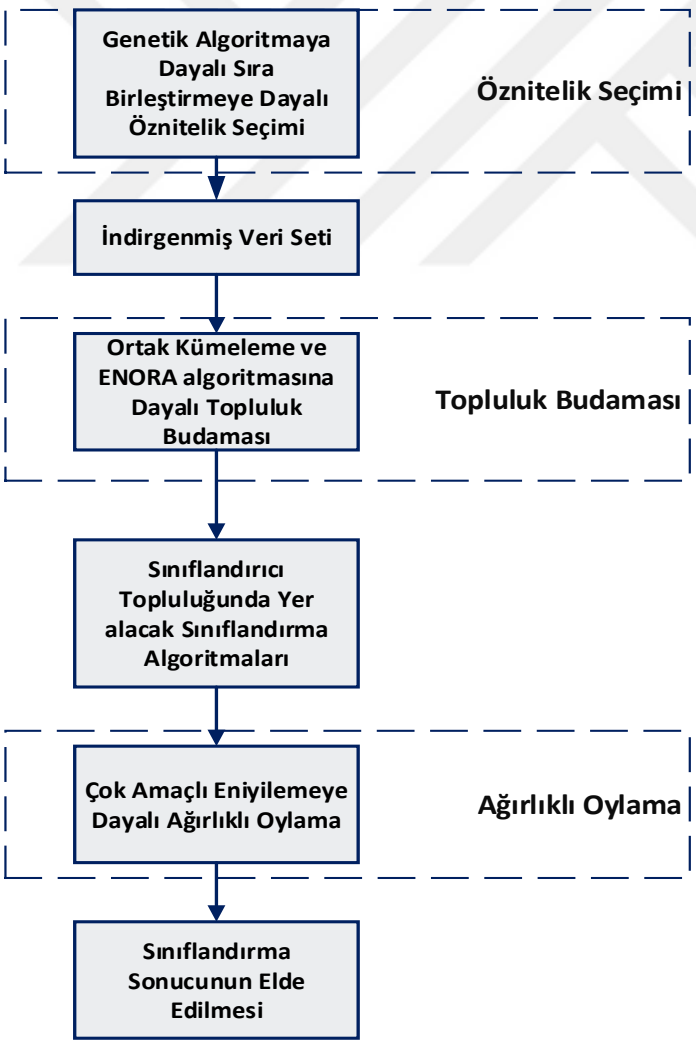
ağları gibi yöntemler yer almaktadır (Aggarwal and Zhai, 2012). Metin belgelerinin makine öğrenmesi sınıflandırıcıları ile işlenebilmesi için, metin özelliklerinin çıkarılması ve uygun dilsel özellikler kullanılarak temsil edilmesi gerekmektedir. Terim varlığı (*term presence*), terim sıklığı (*term frequency*), tümcenin ögeleri (*part of speech*), görüş sözcükleri, kalıplar, olumsuzluk bildiren ifadeler gibi farklı dilsel özelliklere başvurulmaktadır (Medhat et al., 2014). Makine öğrenmesi algoritmaları, uygun bir temsil yöntemi aracılığıyla temsil edilen etiketli veri seti ile eğitilmekte ve sınıflandırma modeli, öğreticili öğrenme (*supervised learning*) ile oluşturulmaktadır.

Yüksek sınıflandırma başarımına sahip etkin bir sınıflandırma modelinin geliştirilebilmesi için, veri setini uygun bir biçimde temsil edecek yöntemin belirlenmesi ve etkin bir öznitelik seçim yöntemi (*feature selection method*) aracılığıyla, bilgi vericiliği yüksek özniteliklerin seçilmesi, oldukça yüksek önem taşımaktadır. Metin madenciliği, yüksek boyutluluk (*high dimensionality*) ve çok sayıda gereksiz/ilgisiz öznitelik içerme problemleri ile karşı karşıyadır (Aggarwal and Zhai, 2012). Metin belgelerinin temsilinde sıklıkla uygulanan yöntemlerden birisi sözcük torbası (*bag of words*) yaklaşımıdır. Sözcük torbası yaklaşımı kullanılarak metin temsilinde, her bir örnek sabit boyutlu bir vektör aracılıyla temsil edilir. Vektörün her bir bileşeni, ilgili örneğin karşılık gelen öznitelik için aldığı değeri belirtir. Sözcük torbası yaklaşımı, yaygın kullanımı ve basit yapısına karşın, yüksek boyutluluk problemine sahiptir. Bu nedenle, hem ölçeklenebilirlik hem de sınıflandırma başarımı bakımından, öznitelik seçimi yöntemi uygulanması önemli duruma gelmektedir. Basit yapısı ve kısmen yüksek başarımından ötürü filtre tabanlı öznitelik seçim yöntemleri, metin madenciliğinde sıklıkla uygulanmaktadır (Medhat et al., 2014; Aggarwal and Zhai, 2012).

Sınıflandırıcı toplulukları (*classifier ensemble, ensemble of classifiers*), sınıflandırma işleminin tek bir sınıflandırma algoritması ile elde edilen sonuç yerine birden fazla sınıflandırma algoritmasının eğitilmesi ve sonucun, farklı sınıflandırma algoritmalarının çıktılarının birleştirilmesi ile elde edilmesine yönelik bir makine öğrenmesi araştırma alanıdır (Dietterich, 2000). Sınıflandırıcı toplulukları ile temel öğrenme (sınıflandırma) algoritmalarından daha yüksek sınıflandırma başarımına sahip yöntemler geliştirilmesi hedeflenmektedir. Etkin bir sınıflandırıcı mimarisi geliştirilmesindeki en önemli noktalardan biri, temel öğrenme algoritması olarak hangi sınıflandırıcıların kullanılacağının belirlenmesidir. Etkin bir sınıflandırıcı topluluğu mimarisi geliştirilebilmesi için, sınıflandırıcı topluluğuna katılan öğrenme algoritmalarının çıktıları ve yapıları

bakımından çeşitliliğin sağlanması gereklidir (Dietterich, 2000). Etkin sınıflandırıcı topluluğu tasarımındaki bir diğer önemli unsur, sınıflandırıcı topluluğunun çıktılarının birleştirilmesinde kullanılacak kuralın belirlenmesidir (Moreno-Seco et al., 2006). Temel öğrenme algoritmalarının birleştirilmesinde en sık kullanılan yöntemler arasında, ağırlıksız oylama (*unweighted voting*) ve ağırlıklı oylama (*weighted voting*) yöntemleri gelmektedir.

Bu tez çalışması kapsamında, etkin bir metin sınıflandırma mimarisi tasarlanması hedeflenmiştir. Bu doğrultuda, öğreticili öğrenmeye dayalı bir sınıflandırıcı topluluğu mimarisi önerisinde bulunulmuştur. Şekil 1.1'de tez kapsamında geliştirilen sınıflandırıcı topluluğuna dayalı mimarinin genel yapısı sunulmaktadır. Tez kapsamında geliştirilen mimari, öznitelik seçimi, sınıflandırıcı topluluğu birleştirme kuralları ve sınıflandırıcı seçimi problemlerine odaklanarak, değinilen konularda etkin çözüm önerileri getirmektedir.



Şekil 1.1. Geliştirilen görüş sınıflandırma mimarisinin genel yapısı

Tez çalışmasının temel katkıları şu şekilde özetlenebilir:

- **Etkin bir öznitelik seçim yöntemi geliştirilmesi:** Basit yapısı ve etkin başarımı nedeniyle, metin madenciliğinde en sık kullanılan öznitelik seçim yöntemleri, temel filtre tabanlı öznitelik seçim yöntemleridir. Tez çalışması kapsamında, yedi farklı filtre tabanlı öznitelik seçim yöntemi (bilgi kazancı, ki-kare ölçütü, kazanç oranı ölçütü, simetrik belirsizlik katsayısı, Pearson korelasyon katsayısı, Relief-F algoritması ve olasılıklı önem ölçütü) ile elde edilen öznitelik sıralamaları birleştirilerek etkin bir öznitelik seçim yöntemi geliştirilmiştir. Geliştirilen öznitelik seçim yönteminde, değinilen öznitelik seçim yöntemleri ile elde edilen sıralamalar, genetik algoritma aracılığıyla bileştirilmiştir. Geliştirilen genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçim yöntemi, metin madenciliği alanında, sıra birleştirme problemine dayalı öznitelik seçiminin kullanıldığı ilk çalışmadır (Onan and Korukoğlu, 2015a).

- **Etkin bir sınıflandırıcı topluluğu birleştirme kuralı geliştirilmesi:** Etkin bir sınıflandırıcı topluluğu geliştirilmesinde, sınıflandırıcı çıktılarının birleştirilmesinde kullanılacak birleştirme kuralı oldukça önemlidir. Tez çalışması kapsamında, ağırlıklı oylama birleştirme kuralında, sınıflandırıcılara uygun değerler atanması problemi, bir eniyileme problemi olarak ele alınmıştır. Metasezgisel yöntemler aracılığıyla, sınıflandırıcının belirli bir çıktı sınıfındaki başarımını yansıtacak şekilde ağırlık değerleri atanması için, çok amaçlı diferansiyel gelişim algoritmasına dayalı bir ağırlıklı oylama mimarisi sunulmuştur. Geliştirilen yöntem, çok amaçlı diferansiyel gelişim algoritmasının, sınıflandırıcı topluluğu ağırlık ayarlamada kullanıldığı ilk çalışmadır (Onan et al., 2016).

- **Etkin bir sınıflandırıcı topluluğu budama yöntemi geliştirilmesi:** Sınıflandırıcı topluluğu budama, sınıflandırıcı topluluğunda yer alan öğrenme algoritmalarından bir bölümünün budanması ile topluluğun sınıflandırma başarımının artırılması ve hesaplama maliyetinin azaltılmasını amaçlayan bir yöntemdir. Etkin bir makine öğrenmesi sınıflandırıcısı tasarımında, sınıflandırıcı topluluğunda yer alan temel öğrenme algoritmalarından hangilerinin kullanılacağının belirlenmesi büyük önem taşımaktadır. Tez çalışması kapsamında,

kümelemeye dayalı ve rastsal sınıflandırıcı topluluğu budama yöntemlerini bir araya getiren, melez bir sınıflandırıcı topluluğu budama yöntemi geliştirilmiştir. Geliştirilen yöntemin kümeleme aşamasında, ortak kümeleme, metasezgisel arama aşamasında ENORA algoritması kullanılmıştır. Geliştirilen yöntem, ortak kümelemenin, sınıflandırıcı topluluğu budamada kullanıldığı ilk çalışmadır. Bunun yanı sıra, ENORA algoritması da ilk kez sınıflandırıcı topluluğu budamada kullanılmıştır.

Tezin ikinci bölümünde, tez çalışmasına ilişkin konularda gerçekleştirilen önceki çalışmalar aktarılmaktadır. Üçüncü bölümde, tez kapsamında geliştirilen yöntemlere ilişkin temel kavramlar ve yöntemler tanıtılmaktadır. Dördüncü bölümde, tez kapsamında geliştirilen öznitelik seçim mimarisi, sınıflandırıcı topluluğu budama mimarisi ve çok amaçlı eniyilemeye dayalı ağırlıklı oylama yöntemi anlatılmaktadır. Beşinci bölümde, geliştirilen yöntemler ile elde edilen deneysel sonuçlar aktarılmaktadır. Son bölümde ise, tez çalışmasının sonuçları ve katkılarına yer verilmektedir.

2. ÖNCEKİ ÇALIŞMALAR

Bu bölümde, tezin yakından ilgili olduğu, metin madenciliğinde öznitelik seçimi, görüş madenciliğinde sınıflandırıcı toplulukları kullanımı, sınıflandırıcı topluluklarında ağırlıklı oylama yöntemleri ve sınıflandırıcı topluluk budama gibi konulara ilişkin önceki çalışmalara yer verilmektedir.

2.1 Öznitelik Seçimine İlişkin Önceki Çalışmalar

Tan and Zhang (2008) çalışmalarında, Çince görüş bildiren metin belgeleri için dört farklı öznitelik seçimi yöntemini (belge sıklığı, ki-kare ölçütü, karşılıklı bilgi ölçütü ve bilgi kazancı ölçütü), beş temel öğrenme algoritması (ağırlık merkezi sınıflandırıcısı, K-en yakın komşu, Naive Bayes algoritması ve destek vektör makineleri gibi) üzerinde değerlendirmiştir. Chen et al. (2009) çalışmalarında, Naive Bayes algoritması ile çok sınıflı metin sınıflandırma için iki farklı öznitelik seçim ölçütü geliştirmiştir. Bu çalışmada, iki sınıflı sınıflandırma problemleri için kullanılan ölçütler, çok sınıflı problemlere uyarlanmıştır. Wang et al. (2011) çalışmalarında, görüş sınıflandırma için Fisher diskriminant oranına (*Fisher's discriminant ratio*) dayalı bir öznitelik seçim yöntemi geliştirmiştir. Geliştirilen öznitelik seçim yönteminin sınıflandırma başarımı, destek vektör makineleri aracılığıyla değerlendirilmiştir. Mesleh (2011) çalışmasında, Arapça metin belgelerinin sınıflandırılmasında öznitelik seçim yöntemlerinin etkinliğini değerlendirmiştir. Çalışmada, ki-kare ölçütü, bilgi kazancı, karşılıklı bilgi, olasılık oranı gibi öznitelik seçim yöntemleri destek vektör makineleri aracılığıyla değerlendirilmiştir. Deneysel sonuçlar, Arapça metin belgelerinde sınıflandırma başarımı bakımından en iyi sonuçların genellikle ki-kare ölçütü ile elde edildiğini göstermektedir. Uysal and Gunal (2012) çalışmalarında, metin sınıflandırma için olasılık tabanlı bir öznitelik seçim yöntemi geliştirmiştir. Bu çalışmada, belirli bir sınıfta yer alan, ancak diğer sınıflarda yer almayan, ayırt ediciliği yüksek terimlere, yüksek değerler atanması amaçlanmaktadır. Buna karşılık, çok sayıda sınıfta yer alan ilgisiz/gereksiz özniteliklere ise düşük değerler atanması amaçlanmaktadır. Bu doğrultuda, tüm sınıflarda yüksek görülme sıklığına sahip özniteliklere düşük, belirli sınıflarda yüksek görülme sıklığına sahip özniteliklere ise yüksek ağırlık değerleri atanmaktadır. Gunal (2012) çalışmasında, metin sınıflandırma için filtre ve sarmalama tabanlı öznitelik seçim yöntemlerini bir arada kullanan melez bir öznitelik seçim yöntemi geliştirmiştir. Deneysel çalışmalar, farklı öznitelik seçim yöntemlerinin birleştirilmesi ile elde edilen melez mimarinin, tek bir öznitelik seçim yöntemi kullanılmasına kıyasla daha

etkin sonuçlar verdiğini göstermiştir. Sheydaei et al. (2015) çalışmalarında, metin sınıflandırma için, birliktelik kuralları madenciliğine dayalı bir öznitelik seçimi geliştirmiştir. Bu çalışmada, kümeleme analizi aracılığıyla, bilgi vericiliği yüksek öznitelikler gruplara atanmıştır. Javed et al. (2015) çalışmalarında, metin sınıflandırma için iki aşamalı bir öznitelik seçim yöntemi geliştirmiştir. Bu yöntemde, öznitelikler üzerinde bilgi kazancı ya da başka bir öznitelik seçim yöntemi kullanılarak, özniteliklere ilişkin bir sıralama listesi elde edilmektedir. Ardından, öznitelik altkümesi değerlendirme aracılığıyla son öznitelik kümesi belirlenmektedir.

Farklı öznitelik seçim yöntemleri, özniteliklerin bilgi vericiliklerini, farklı ölçütler ya da yöntemler aracılığıyla değerlendirmektedir. Bu nedenle, farklı öznitelik seçim yöntemleri sonucu elde edilen öznitelik sıralamaları birbirinden farklı olmaktadır. Bu sıralamaların, sıra birleştirme yöntemleri kullanılarak, daha iyi bir sıralama elde etmek üzere birleştirilmesi, öznitelik seçimi alanında üzerinde çalışılan araştırma konularından biridir.

Jong et al. (2004) çalışmalarında, evrimsel bir öznitelik seçim yönteminin farklı çalışmalarında elde edilen öznitelik sıralamalarını, sıra birleştirmesi ile birleştiren melez bir öznitelik seçim yöntemi geliştirmiştir. Geliştirilen öznitelik seçim yönteminde, alıcı işletim karakteristiği eğrisine (*ROC curve*) dayalı genetik algoritma kullanılmaktadır. Prati (2012) çalışmasında, dört temel sıra birleştirme yönteminin (Borda, Concordet, Schulze ve Markov zinciri yöntemleri) öznitelik seçimindeki etkinliklerini değerlendirmiştir. Çalışmada kullanılan veri setlerinde, öznitelik sıralamaları elde edilebilmesi için, bilgi kazancı, kazanç oranı, simetrik belirsizlik katsayısı, ki-kare ölçütü, OneR algoritması ve ReliefF algoritması kullanılmıştır. Deneysel çalışmalarda, en yüksek başarım Schulze sıra birleştirme yöntemi ile elde edilmiştir. Dittman et al. (2013) gen seçimi için sıra birleştirme yöntemlerinin başarımlarını değerlendirmiştir. Genlerin bilgi vericiliklerine göre sıralanmasında, yirmi beş farklı öznitelik sıralama yöntemi kullanılmıştır. Bu yöntemler arasında, ROC eğrisi altındaki alan (*area under ROC curve*), sapma (*deviance*), F-ölçütü (*F-measure*), geometrik ortalama ve Gini indeksi gibi yöntemler yer almaktadır. Farklı sıra birleştirme yöntemlerinin öznitelik seçimindeki etkinliklerinin değerlendirilmesi için ortalama sıra birleştirme, medyan sıra birleştirme, en yüksek sıra birleştirme, en düşük sıra birleştirme, kararlılık seçimi, üstel ağırlıklı sıra birleştirme, iyileştirilmiş Borda sıra birleştirme ve Round Robin sıra birleştirme gibi temel sıra birleştirme yöntemleri kullanılmıştır. Deneysel çalışmalar, sıra birleştirmeye dayalı öznitelik seçim

yöntemlerinin genellikle filtre-tabanlı öznitelik seçim yöntemlerine kıyasla daha yüksek başarım gösterdiğini ortaya koymaktadır. Ancak, karşılaştırılan sıra birleştirme yöntemleri arasında, istatistiksel olarak anlamsal bir performans farklılığı elde edilememiştir. Bir başka çalışmada, Relief algoritması, korelasyon tabanlı öznitelik seçimi ve bilgi kazancına dayalı melez bir öznitelik seçim yöntemi geliştirilmiştir (Bouaguel et al., 2013a). Geliştirilen öznitelik seçim yönteminde, değinilen öznitelik seçim yöntemleri ile elde edilen farklı sıralamalar genetik algoritmaya dayalı sıra birleştirme kullanılarak birleştirilmiştir. Wald et al. (2012) çalışmalarında, gen seçiminde, melez ve filtre tabanlı öznitelik seçim yöntemlerinin başarımını değerlendirmiştir. Çalışmada, bireysel filtre tabanlı öznitelik seçim yöntemleri olarak, ROC eğrisi altındaki alan, olasılık oranı, kat değişim oranı (*fold change ratio*), sinyal gürültü oranı (*signal to noise ratio*) ve bilgi kazancı kullanılmıştır. Bireysel filtre tabanlı öznitelik seçim yöntemleri ile elde edilen sıralama listelerinin birleştirilmesinde, ortalama sıra birleştirme yöntemi kullanılmıştır. Deneysel çalışmalar, bilgi kazancı ve kat değişim oranına dayalı öznitelik sıralamaların, sıra birleştirme ile bir araya getirilmesinin, bireysel öznitelik seçim yöntemlerine kıyasla daha yüksek başarım elde ettiğini göstermektedir. Bouaguel et al. (2013b) çalışmalarında, Relief, Pearson korelasyon katsayısı ve karşılıklı bilgi öznitelik seçim yöntemlerini sıra birleştirme aracılığıyla birleştiren bir öznitelik seçim yöntemi geliştirmiştir. Bir diğer çalışmada, bilgi kazancı, ki-kare ölçütü ve simetrik belirsizlik katsayısı öznitelik seçim yöntemlerini, Borda sıra birleştirme yöntemi ile birleştiren bir öznitelik seçim yöntemi geliştirilmiştir (Sarkar et al., 2014).

2.2 Sınıflandırıcı Topluluklarına İlişkin Önceki Çalışmalar

Makine öğrenmesine dayalı yöntemler, görüş madenciliğinde başarılı sonuçlar vermekte ve sıklıkla uygulanmaktadır. Makine öğrenmesine dayalı görüş madenciliği alanındaki güncel çalışmalar, görüş sınıflandırıcılarının tahmin etme başarımının artırılmasında, sınıflandırıcı topluluğu yaklaşımının kullanılmasının uygun olduğunu göstermektedir (Prabowo and Thelwall, 2009; Xia et al., 2011; Wang et al., 2014). Görüş sınıflandırma, makine öğrenmesi ve doğal dil işlemenin önemli bir araştırma alanıdır. Bu nedenden ötürü, görüş sınıflandırma alanındaki çalışmalar, çalışma alanının dilsel ya da makine öğrenmesine yönelik özelliklerine odaklanabilmektedir. Örneğin, Xi et al. (2011) çalışmalarında, metin belgelerinin temsilinde sözcük türü, sözcükler arası ilişkiler, TF-IDF temsili gibi dilsel özelliklerin, sınıflandırıcı toplulukları ile bir arada kullanılmasının görüş sınıflandırma alanındaki etkinliğini değerlendirmiştir. Bu çalışmada, Naive Bayes

algoritması, maksimum entropi sınıflandırıcısı, destek vektör makineleri gibi temel öğrenme algoritmaları (sınıflandırıcılar), sabit kural sınıflandırıcı topluluğu birleştirme kuralı (*fixed-rule output combination*) ve ağırlıklı sınıflandırıcı topluluğu birleştirme kuralları gibi çeşitli yöntemler ile birleştirilmiştir. Benzer şekilde, Wang et al. (2014) tarafından gerçekleştirilen çalışmada, metin temsilinde farklı N-gram temsil modelleri (1-gram ve 2-gram), terim sıklığı, terim varlığı ile TF-IDF temsili, Bagging, rastgele alt uzay (*Random Subspace*) ve destekleme (*boosting*) gibi temel sınıflandırıcı topluluğu yöntemleri ile bir arada kullanılarak değerlendirilmiştir. Wang et al. (2015) görüş sınıflandırma için, sözcük türü analizi (*part of speech analysis*) ve rastgele alt uzay (*Random Subspace*) yöntemlerine dayalı bir sınıflandırma mimarisi sunmuştur. Bu çalışmada, içerik sözlüğü ve fonksiyon sözlüğüne ilişkin parametreler aracılığı ile sınıflandırıcı topluluğunun performansı iyileştirilmiştir. Başka bir çalışmada, sözlük, sözcük torbası ve duygu bildiren ifadeler aracılığıyla farklı öznitelik kümeleri elde edilerek, bu öznitelik kümelerinin performansı, sınıflandırıcı toplulukları kullanılarak değerlendirilmiştir (Da Silva et al., 2014). Bu çalışmada, Naive Bayes, destek vektör makineleri, rastgele orman (*random forest)* ve lojistik regresyon sınıflandırıcıları temel öğrenme algoritmaları olarak kullanılmıştır. Augustyniak et al. (2016) tarafından geliştirilen sözlük tabanlı görüş sınıflandırma mimarisinde, özniteliklere olasılık ve sıklık oranlarına dayalı olarak değer atanmıştır. Bir başka çalışmada, görüş bildiren metinlerin temsili için gizli konu öznitelik kümesine (*latent topic feature set*) ve sözcükler arası ilişkilere dayalı iyileştirilmiş bir yöntem önerilmiştir (Zhang and He, 2015). Bu çalışmada, geliştirilen yöntem aracılığıyla temsil edilen metin belgeleri, sınıflandırıcı toplulukları kullanılarak sınıflandırılmıştır. Zhang et al. (2015) çalışmalarında, sözcükler arası anlamsal ilişkilere dayalı bir görüş sınıflandırma yöntemi önermiştir. Bu çalışmada, kümeleme analizi kullanılarak, benzer öznitelikler gruplandırılmıştır. Ardından, metin belgeleri destek vektör makineleri ile sınıflandırılmıştır. Lima et al. (2015) çalışmalarında, Twitter mesajlarını sınıflandırmak için sözcük tabanlı ve makine öğrenmesine dayalı yöntemleri bir arada kullanan bir sınıflandırma mimarisi önerisinde bulunmuştur.

Görüş sınıflandırma alanındaki çalışmaların bir bölümü, makine öğrenmesine dayalı yöntemlerin iyileştirilmesi ile sınıflandırma başarımının artırılmasına odaklanmıştır. Bu bağlamda, Prabowo and Thelwall (2009) çalışmalarında, görüş sınıflandırmada birden çok sınıflandırıcı kullanılmasının sınıflandırma başarımını artırıp artırmadığını deneysel olarak incelemiştir. Deneysel analizde, genel sorgu tabanlı, kural tabanlı, istatistik tabanlı

sınıflandırıcılar ile destek vektör makineleri farklı sınıflandırıcı topluluğu mimarileri elde etmek amacıyla bir arada kullanılmıştır. King et al. (2015) çalışmalarında, Naive Bayes, lojistik regresyon, karar ağacı ve destek vektör makineleri gibi temel sınıflandırma algoritmalarının, oylama, destekleme (*Boosting*), yığılmış genelleme (*Stacking*) gibi farklı sınıflandırıcı topluluğu yöntemleri ile bir araya getirilmesinin etkinliğini irdelemiştir. Wang et al. (2013) çalışmalarında, aşırı öğrenme makinesi (*extreme learning machine*) ve sezgisel bulanık küme teorisine dayalı bir görüş sınıflandırma mimarisi sunmuştur. Bir başka çalışmada, etkin bir görüş sınıflandırma mimarisi oluşturmak amacıyla Bayes model ortalaması (*Bayesian model averaging*) sınıflandırıcı topluluğu birleştirme kuralına dayalı bir yöntem geliştirmiştir. Sınıflandırıcı topluluğu oluşturulmasındaki en önemli noktalardan birisi sınıflandırıcı topluluğunda yer alacak sınıflandırma algoritmalarının belirlenmesidir. Bu doğrultuda, Lin et al. (2015) tarafından en uygun sınıflandırıcı altkümesinin seçilmesine yönelik bir algoritma geliştirilmiştir.

2.3 Ağırlıklı Oylama Yöntemlerine İlişkin Önceki Çalışmalar

Sınıflandırıcı topluluğu tasarımında en önemli unsurlardan birisi, sınıflandırma algoritmalarının birleştirilmesinde kullanılacak sınıflandırıcı topluluğu birleştirme kuralının belirlenmesidir. Çoğunluk oylaması (*majority voting*), basit yapısına karşın, etkin ve sıklıkla kullanılan sınıflandırıcı topluluğu birleştirme kurallarından biridir. Bunun yanı sıra, sınıflandırıcı topluluklarında birleştirme kurallarına yönelik güncel araştırma çalışmaları, ağırlıklı oylama yöntemlerinin, sınıflandırıcı topluluklarının başarımını iyileştirebildiğini göstermektedir (Ekbal and Saha, 2011a; Ekbal and Saha, 2011b; Kim et al., 2011). Ağırlıklı oylamaya dayalı güncel çalışmalardan bir bölümü, doğrudan belirli bir uygulama alanına dayalı olarak geliştirilirken, çalışmaların bir bölümü, daha genel bir yapıda tasarlanmıştır.

Ağırlıklı oylamaya dayalı sınıflandırıcı topluluklarının tasarlandığı alanlardan birisi, bir doğal dil işleme araştırma alanı olan varlık ismi çıkarımı (*named entity recognition*) dır. Ekbal and Saha (2011a) tarafından gerçekleştirilen çalışmada, sınıflandırıcıların çıktı sınıflarındaki başarımlarına dayalı olarak sınıflandırıcı ağırlığı atanmasında genetik algoritmalar kullanılmıştır. Bir başka çalışmada, sınıflandırıcı topluluğunda yer alan sınıflandırma algoritmalarının ağırlık değerleri, çok amaçlı benzetilmiş tavlama algoritması kullanılarak eniyilenmiştir (Ekbal and Saha, 2011b). Saha and Ekbal (2013) tarafından

gerçekleştirilen diğer bir çalışmada, Naive Bayes, karar ağacı, saklı Markov modeli (*hidden Markov model*), maksimum entropi, koşullu rastgele alan (*conditional random field*) ve destek vektör makineleri gibi sınıflandırma algoritmaları temel öğrenme algoritmaları olarak alınmıştır. Bu sınıflandırma algoritmaları ile ikili oylama ve gerçek oylama birleştirme kurallarının etkinlikleri irdelenmiştir. Ekbal and Saha (2013) tarafından gerçekleştirilen çalışmada ise çok amaçlı benzetilmiş tavlamaya dayalı iki aşamalı bir sınıflandırma mimarisi sunulmuştur. Bu mimaride, benzetilmiş tavlama algoritması, hem öznitelik seçimi hem de sınıflandırma algoritmalarına uygun ağırlık değerlerinin atanması amacıyla kullanılmıştır.

Ağırlıklı oylamaya dayalı sınıflandırıcı topluluklarının kullanıldığı alanlardan biri, tıbbi teşhis ve biyoenformatik alanıdır. Örneğin, Bashir et al. (2015a) tarafından geliştirilen çalışmada, meme kanseri teşhisinde ağırlıklı oylamaya dayalı sınıflandırıcı topluluğu kullanılmıştır. Geliştirilen mimaride, Naive Bayes, karar ağacı algoritması ve destek vektör makineleri gibi sınıflandırma algoritmaları kullanılmıştır. Sınıflandırma algoritmalarının ağırlık değerleri, bu algoritmaların eğitim verisi üzerindeki normalize edilmiş doğru sınıflandırma oranlarına göre hesaplanmıştır. Bashir et al. (2015b) tarafından geliştirilen diğer bir çalışmada, kalp hastalıklarının teşhisi için destekleme (*Boosting*) algoritmasına ve çok amaçlı eniyilemeye dayalı bir sınıflandırıcı topluluğu mimarisi geliştirilmiştir. Geliştirilen mimaride, çok amaçlı eniyileme kullanılarak, temel öğrenme algoritmalarının ağırlık değerleri eniyilenmiştir.

Ağırlıklı oylamaya dayalı sınıflandırıcı topluluklarının kullanıldığı alanlardan bir diğeri ise kredi derecelendirmedir. Xia et al. (2016) çalışmalarında, kredi derecelendirme için, ağırlıklı oylama ve öğreticili kümelemeye dayalı bir mimari geliştirmiştir. Geliştirilen mimaride, kümeleme analizi kullanılarak, orijinal veri setinden çeşitliliği yüksek eğitim alt kümeleri elde edilmektedir. Başka bir çalışmada ise evrimsel algoritmaya dayalı ağırlıklı oylama ve destek vektör makineleri kullanılarak, etkin bir uzaktan algılama sınıflandırıcısı geliştirilmiştir (Garcia-Gutierrez et al., 2015). Aburomman and Reaz (2016) çalışmalarında, ağırlıklı oylamaya dayalı bir saldırı tespit sınıflandırıcısı geliştirmiştir. Geliştirilen sınıflandırıcı mimarisinde, altı K-en yakın komşu algoritması ile altı destek vektör makineleri sınıflandırıcısı eğitilmiştir. Bu sınıflandırma algoritmaları, parçacık sürüsü eniyilemesi ve ağırlıklı çoğunluk oylaması gibi sınıflandırıcı topluluğu birleştirme kuralları kullanılarak bir araya getirilmiştir.

Daha önce değinildiği gibi, ağırlıklı oylamaya dayalı çalışmaların bir bölümü, belirli bir uygulama alanı ile sınırlı tutulmayan, genel yapıda tasarlanmış çalışmalardır. Örneğin, Kim et al. (2011) tarafından geliştirilen çalışmada, yeni bir ağırlıklı oylamaya dayalı sınıflandırıcı topluluğu birleştirme kuralı geliştirilmiştir. Bu çalışmada, sınıflandırma algoritmalarının ve sınıflandırılan örneklerin ağırlık vektörleri tutularak, sınıflandırılması zor olan örneklerin ve zor örneklerin sınıflandırılmasında daha başarılı olan sınıflandırıcıların ağırlık değerleri artırılmıştır. Bu şekilde, sınıflandırıcılara ilişkin eniyilenmiş bir ağırlık vektörü elde edilmiştir. Zhang et al. (2014) tarafından geliştirilen çalışmada, diferansiyel gelişim algoritmasına dayalı bir ağırlıklı oylama birleştirme kuralı geliştirilmiştir. Cao et al. (2015) tarafından geliştirilen çalışmada ise aşırı öğrenme makinelerine dayalı sınıflandırıcı topluluğunun birleştirilmesinde esnek hesaplamaya dayalı ağırlıklı oylamadan yararlanılmıştır.

2.4 Sınıflandırıcı Topluluğu Budamaya İlişkin Önceki Çalışmalar

Sınıflandırıcı topluluğu budama, sınıflandırmada mevcut öğrenme algoritmalarının tümünün kullanılması yerine, sınıflandırmanın, uygun bir sınıflandırıcı topluluğu altkümesi ile yapılması ile daha etkin sonuçlar alınmasına yönelik bir araştırma alanıdır. Ruta and Gabrys (2001) tarafından gerçekleştirilen çalışmada, genetik algoritma, tabu arama ve toplum tabanlı artırımlı öğrenme algoritmalarının sınıflandırıcı topluluğu budamadaki başarımları değerlendirilmiştir. Bu çalışmada, çoğunluk oylaması hatası uygunluk fonksiyonu olarak kullanılmıştır. Zhou et al. (2002) çalışmalarında, GASEN adını verdikleri, genetik algoritmaya dayalı bir sınıflandırıcı topluluğu budama yöntemi geliştirmiştir. Bu mimaride, öncelikle yapay sinir ağları eğitilmektedir. Ardından, yapay sinir ağlarına rastgele olarak atanan ağırlık değerleri genetik algoritma aracılığıyla eniyilenmekte ve sinir ağları, ağırlık değerlerine dayalı olarak seçilerek budanmış sınıflandırıcı topluluğu elde edilmektedir. Bir başka çalışmada, GASEN sınıflandırıcı topluluğu budama mimarisi, karar ağaçlarından oluşan sınıflandırıcı topluluklarını budamak amacıyla uygulanmıştır (Zhou and Tang, 2003). Sheen and Sirisha (2013) tarafından geliştirilen sınıflandırıcı topluluğu budama mimarisinde, harmoni arama algoritması kullanılmıştır. Geliştirilen budama yöntemi, kötü amaçlı yazılım tespitinde uygulanmıştır. Dai (2013) çalışmasında, rastsal açgözlü arama algoritmasına dayalı sınıflandırıcı topluluğu budama mimarisi geliştirmiştir. Mendialdua et al. (2015) tarafından geliştirilen çalışmada, sınıflandırıcı topluluğu altkümesi budanmasında, dağılım tahmini algoritması (*estimation of distribution algorithm*) kullanmıştır.

Aksela (2003) çalışmasında, üstel aramaya dayalı sınıflandırıcı topluluğu budama yöntemlerinin başarımını değerlendirmiştir. Bu amaçla, hatalar arası korelasyon, *Q*-istatistiği, karşılıklı bilgi gibi ölçütler ile arama algoritmalarının başarımları incelenmiştir.

Margineantu and Dietterich (1997) çalışmalarında, geri uyarlamaya dayalı indirgenmiş hata budama (*reduce error pruning with back-fitting*) ismini verdikleri ardışık bir sınıflandırıcı topluluğu budama yöntemi geliştirmiştir. Geliştirilen yöntemin ilk iki adımı, ileri doğru arama algoritması ile aynıdır. Üçüncü adımda ise sınıflandırıcı topluluğunun çoğunluk oylaması hatasını en aza indirgeyen sınıflandırma algoritmaları, topluluğa dâhil edilmektedir. Caruana et al. (2004) tarafından, model kütüphanesinden topluluk seçimi algoritması geliştirilmiştir. Model kütüphanesinden topluluk seçimi algoritmasında, temel öğrenme algoritmalarından oluşan bir sınıflandırıcı kütüphanesinden, belirli değerlendirme ölçütleri ve arama algoritmalarına dayalı olarak, sınıflandırıcı altkümesi elde edilmektedir. Partalas et al. (2012) çalışmalarında, sınıflandırıcı topluluğu budamada açgözlü arama algoritmalarının başarımlarını değerlendirmiştir. Karşılaştırmalı analizlerde, farklı arama yönleri, değerlendirme veri setleri ve değerlendirme ölçütleri dikkate alınmıştır.

Kotsiantis and Pintelas (2005) çalışmalarında, sıralamaya dayalı altı aşamalı bir sınıflandırıcı topluluğu budama yöntemi geliştirmiştir. Öncelikle, veri setinden örneklem alınmakta ve veri seti üç parçaya bölünmektedir. Bu parçalardan ikisi eğitim seti olarak, biri ise test seti olarak kullanılmaktadır. Her bir algoritma, *t*-testi kullanılarak, en yüksek doğru sınıflandırma oranına sahip sınıflandırma algoritması ile karşılaştırılmaktadır. İstatistiksel analiz sonucunda, $p<0.05$ değerini veren algoritmalar, sınıflandırıcı topluluğuna dâhil edilmemektedir. Bir başka çalışmada, her bir test örnekleminin, sınıflandırıcı topluluğundan en uygun sınıflandırma algoritması kullanılarak sınıflandırılmasına yönelik dinamik bir sınıflandırıcı seçimi yöntemi geliştirilmiştir (Swiderski et al., 2016).

Zhang and Cao (2014) tarafından geliştirilen çalışmada, sınıflandırıcı topluluğunda yer alan öğrenme algoritmaları, spektral kümeleme kullanılarak iki kümeye ayrılmaktadır. Algoritmaların kümelere atanmasında, doğru sınıflandırma oranı ve çeşitlilik kullanılmaktadır. Ardından, kümelerden birinde yer alan öğrenme algoritmaları, sınıflandırıcı topluluğunda tutulurken, diğerleri budanmaktadır. Lin et al. (2014) tarafından geliştirilen melez sınıflandırıcı

topluluğu budama mimarisinde, k-ortalama kümeleme algoritması, dinamik seçim ve ardışık arama yöntemleri bir arada kullanılmıştır.

3. TEMEL YÖNTEMLER

Bu bölümde, tezin kapsamını oluşturan öznitelik temsili, öznitelik seçim yöntemleri, sınıflandırma algoritmaları ve sınıflandırıcı toplulukları ile sınıflandırıcı topluluğu budama yöntemleri tanıtılmaktadır.

3.1 Öznitelik Temsili

Görüş sınıflandırmada makine öğrenmesi yöntemlerinin uygulanabilmesi için, görüş bildiren metinlerin uygun bir biçimde temsil edilmesi gerekmektedir. Farklı öznitelik temsil yöntemleri bulunmakla birlikte, görüş sınıflandırmada en yaygın kullanıma sahip temsil yapısı, sözcük torbası (*bag of words*) yaklaşımdır. Sözcük torbası yöntemi, $\{f_1, f_2, \ldots, f_m\}$ belirli bir metinde görülen m adet özniteliği, $n_i(d)$, f_i özelliğinin d belgesinde görülme sıklığını belirtmek üzere, her bir belgenin d: $(n_1(d), n_2(d),\ldots, n_m(d))$ şeklinde temsil edilmesidir. Sözcük torbası yaklaşımı, belirli bir metin belgesindeki sözcüğün görülme sıklığını temsil ederken, belgenin diğer yapısal özelliklerini (sözcüklerin sırası, sözcükler arası ilişkileri vb.) dikkate almaz. Sözcük torbası yaklaşımı kullanılarak metin temsilinde, her bir örnek sabit boyutlu bir vektör aracılığı ile temsil edilir. Vektörün her bir bileşeni, ilgili örneğin karşılık gelen öznitelik için aldığı değeri belirtir. Sözcük torbası yöntemi, oldukça basit yapısına karşın metin madenciliği için oldukça etkin bir temsil modeli olarak çalışmaktadır. Örneğin, "*Ayşe kitap okumayı ve resim yapmayı sever.*" ve "*Can futbol oynamayı ve TV izlemeyi sever.*" tümceleri sözcük torbası yöntemi ile temsil edildiğinde, öncelikle her iki tümcede geçen sözcüklere indeks değerleri atanır: *"Ayşe": 1, "kitap": 2, "okumayı": 3, "ve": 4, "resim": 5, "yapmayı": 6, "sever": 7, "Can": 8, "futbol":9, "oynamayı": 10, "TV": 11, "izlemeyi": 12.* Burada, her iki tümcede geçen toplam on iki sözcük bulunduğu için belirtilen temsil yapısı oluşmuştur. Ardından, her bir tümce 12 öznitelik içeren vektörler ile aşağıda belirtildiği gibi temsil edilebilir:

[1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0]

[0, 0, 0, 1, 0, 0, 1, 1, 1, 1, 1, 1]

N-gram temsil modeli, metin sınıflandırmada sıklıkla kullanılan modellerden biridir. N-gram, bir karakter katarının n adet karakter dilimidir. N-gram yöntemi, belge sınıflandırmada kullanılan temel hesaplamalı dilbilim yöntemlerinden biridir. Bu yöntem, belirli bir dile ilişkin dilbilgisi özellikleri, sözcük yapısı gibi dilsel özelliklere gereksinim duymadan çalışır. Çalışmalarda en

çok kullanılan N-gram modelleri, 1-gram (*unigram*), 2-gram (*bigram*) ve 3-gram (*trigram*)'dır. Örneğin, tümcemiz "*Doçent Doktor"* ise, bu tümcenin bazı n-gramları aşağıdaki şekildedir:

- 1-gramlar: *"D", "o", "ç", "e", "n", "t", "_", "D", "o", "k", "t", "o", "r"*
- 2-gramlar: *"Do", "oç", "çe", "en", "nt", "t_", "_D", "Do", "ok", "kt", "to", "or"*
- 3-gramlar: *"Doç", "oçe", "çen", "ent", "nt_", "t_D", "_Do", "Dok", "okt", "kto", "tor"*

Burada, altçizgi karakteri ("_") kullanılarak, boşluk karakteri temsil edilmektedir.

Sözcük torbası yöntemi kullanılarak metin temsilinde, belirli bir özniteliğin belirli bir metin belgesinde görülme sıklığı dikkate alınmaktadır. Belirli bir *t* sözcüğü için, *d* belgesi içerisindeki görülme sıklığı *TF(t, d)* terim sıklığı (*term frequency*) olarak tanımlanmaktadır. Bunun yanı sıra, bazı metin madenciliği çalışmalarında, belirli bir *t* sözcüğünün, *d* belgesi içerisinde görülüp görülmediğine ilişkin temsil olarak tanımlanabilen *TP(t, d)* terim varlığı (*term presence*) da kullanılmaktadır. Terim varlığı temsili kullanıldığında, belirli bir metin, metinde yer alan terimler 1 ve metinde yer almayan terimler 0 değerini alacak biçimde iki-değerli bir özellik vektörü ile belirtilmektedir. Pang et al. (2002) tarafından gerçekleştirilen çalışmada, görüş madenciliği için terim varlığının, terim sıklığına kıyasla daha iyi bir temsil yapısı sunduğu ortaya konmuştur. Bu, konuya dayalı metin sınıflandırma ile görüş kutbu sınıflandırma arasındaki farklılığı ortaya koymaktadır. Konuya dayalı metin sınıflandırmada, belirli bir terimin görülme sıklığı, sınıflandırma sonucunda daha yüksek bir belirleyiciliğe sahiptir (Pang and Lee, 2004).

Terim puanlama ise özniteliklerin görece etki değerlerini belirlemek için kullanılan bir yöntemdir. Metin sınıflandırmada en sık kullanılan terim puanlama yöntemlerinin başında TF-IDF (*term frequency–inverse document frequency*) yöntemi gelmektedir. Daha önce değinildiği gibi, bir terimin *TF(t, d)* terim sıklığı değeri, ilgili terimin belirli bir belgede geçme sıklığını göstermektedir. Ancak, birçok farklı belgede sıklıkla görülen ve yüksek terim puanına sahip olmasına

karşın belirleyici bir öneme sahip olmayan terimler bulunabilir. Bu nedenle, IDF (*inverse document frequency*) ölçütü ile bu durum ortadan kaldırılmaya çalışılmaktadır. Belirli bir *t* terimine ilişkin *IDF(t, d)* ölçütü değeri Eşitlik 3.1'e göre belirlenmektedir:

$$IDF(t,d) = \log \frac{N}{DF(t,d)} \tag{3.1}$$

Burada, *N*, toplam belge sayısını, *DF(t, d),* terimin görüldüğü belge sayısını temsil etmektedir. IDF ölçütü, 0 ile 1 arasında değerler almaktadır. *TF-IDF(t, d)* ölçütü değeri ise Eşitlik 3.2'ye göre hesaplanmaktadır:

$$TF - IDF(t,d) = TF(t,d) \times \log \frac{N}{DF(t,d)} \tag{3.2}$$

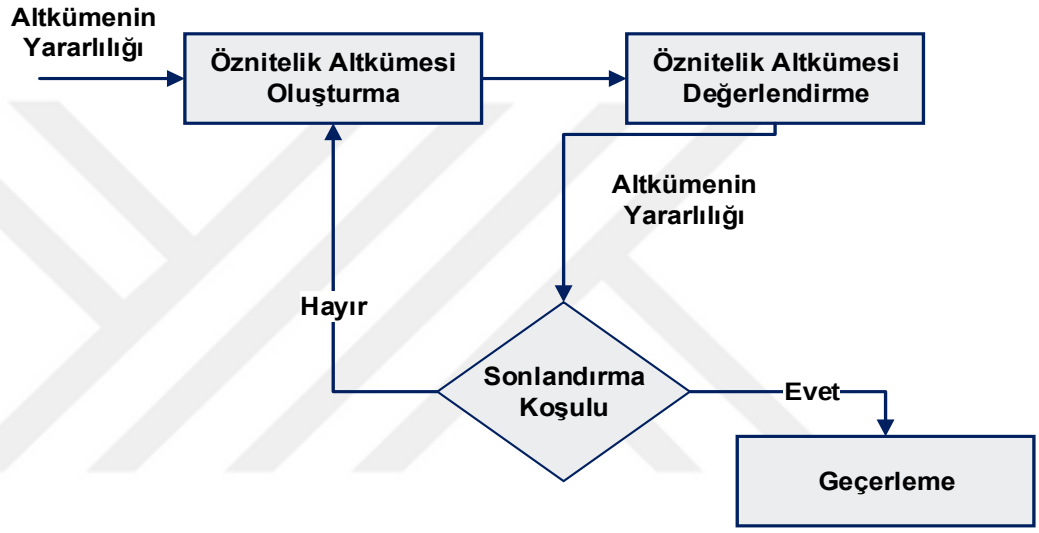
TF-IDF ölçütünün hesaplanmasında, terim sıklığı ve IDF ölçütlerinin çarpımı kullanılmaktadır. Yüksek bir TF-IDF değerine, belirli bir belgede yüksek terim sıklığına sahip ancak tüm belgeler arasındaki görülme sıklığı düşük bir terim ile ulaşılabilmektedir. Bu nedenle, TF-IDF ölçütü aracılığıyla birçok belgede geçen ancak herhangi bir ayırt ediciliği olmayan terimler ayıklanmaktadır.

Tez kapsamında geliştirilen yöntemlerin sınanması için görüş madenciliğine ilişkin dokuz farklı alandan veri setleri kullanılmıştır (Bkz. Bölüm 5.1). Bu veri setleri için uygun bir temsil yöntemi belirlenmesine yönelik olarak gerçekleştirilen deneysel çalışmalarda, terim sıklığı (TF), terim varlığı (TP) ve TF-IDF ölçütleri, 1-gram ve 2-gram temsilleri ile toplam altı farklı temsil ile değerlendirilmiştir. Veri temsillerine ilişkin deneysel analizlere göre, en yüksek doğru sınıflandırma oranı, terim sıklığı ve 1-gram yöntemleri bir arada kullanıldığında, ikinci en yüksek başarım ise terim sıklığı ve 2-gram yöntemleri bir arada kullanıldığında elde edilmektedir. TF-IDF ölçütü ile elde edilen sonuçlar ise kısmen daha düşük sonuçlardır (Onan and Korukoğlu, 2015b). Bu nedenle, tezin beşinci bölümünde aktarılan deneysel sonuçlarda, veri seti öncelikle, terim sıklığı ve 1-gram yöntemleri kullanılarak temsil edilmiştir.

3.2 Öznitelik Seçimi

Öznitelik seçimi (*feature selection*), yüksek boyutluluk sorununu ortadan kaldırmak ve etkin bir sınıflandırma modeli oluşturmak için sıklıkla kullanılan makine öğrenmesi yöntemleri arasındadır. Öznitelik seçimi, veri setindeki mevcut

öznitelikler arasından uygun bir alt kümenin seçilmesini amaçlar. Öznitelik seçimi ile veri setinde yer alan gereksiz/ilgisiz özniteliklerin ortadan kaldırılması ile hem sınıflandırmada kullanılan öğreticili öğrenme algoritmasının çalışma süresi kısaltılmış hem de daha iyi bir sınıflandırma modeli oluşturulmuş olur (Dash and Lui, 1997). Öznitelik seçimi, hedef modeli temsil etmek için gerekli en küçük öznitelik alt kümesinin belirlenmesi olarak tanımlanabilir (Kira and Rendell, 1992a). Başka bir deyişle, öznitelik seçimi, $M<N$ olmak üzere, N tane öznitelik içeren bir öznitelik alt kümesinin, belirli bir ölçüt fonksiyonuna dayalı olarak belirlenmesi sürecidir (Narendra and Fukunaga, 1977).



Şekil 3.1. Öznitelik seçimi sürecinin aşamaları (Dash and Lui, 1997)

Makine öğrenmesinde öznitelik seçimi yöntemleri kullanılmasının birçok faydası bulunmaktadır. Daha önce de değinildiği gibi, öznitelik seçimi, öğrenme algoritmasının çalışma süresini azaltmaktadır. Bunun yanı sıra, gereksiz, ilgisiz ve gürültülü özniteliklerin ortadan kaldırılması ile birlikte öğrenme algoritmasının verinin en önemli özniteliklerine odaklanması, basit ancak daha yüksek tahmin etme başarımına sahip bir sınıflandırma modeli oluşturulabilmesini olanaklı hale getirmektedir. Öznitelik seçiminin, doğru sınıflandırma oranını artırması ya da en azından önemli ölçüde düşürmemesi amaçlanır. İyi bir öznitelik seçim yönteminde, öznitelik seçimi uygulanarak indirgenmiş veri setinin sınıf dağılışı ile başlangıçtaki orijinal sınıfın sınıf dağılışlarının birbirine yakın olması beklenir (Koller and Sahami, 1996). Öznitelik seçimi süreci, temel olarak öznitelik altkümesi oluşturma, altküme değerlendirme, sonlandırma koşulu ve sonuç geçerleme olmak üzere dört aşamadan oluşmaktadır (Dash and Lui, 1997). Şekil 3.1'de öznitelik seçim sürecinin temel aşamaları gösterilmiştir.

3.2.1 Öznitelik altkümesi oluşturma

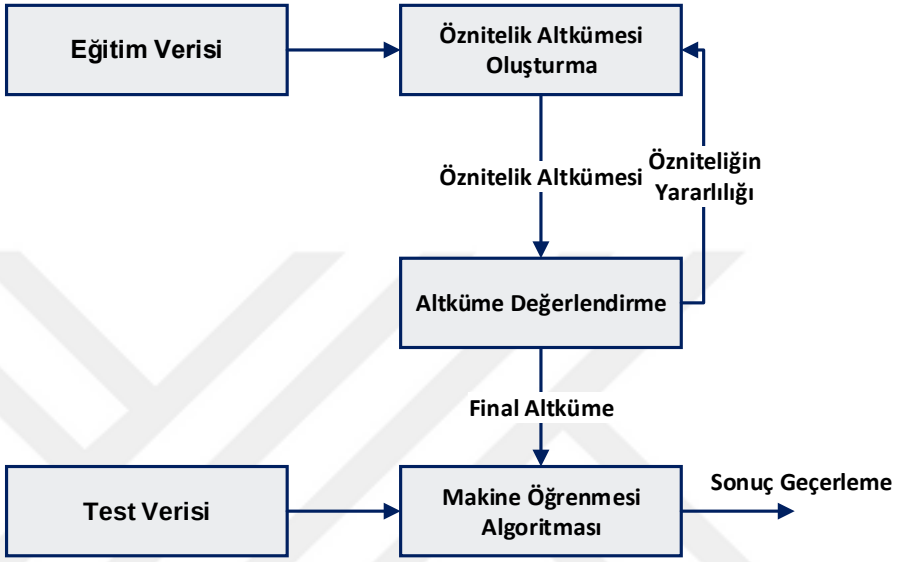
Öznitelik altkümesi oluşturma, orijinal öznitelik setinden değerlendirmede kullanılmak üzere altkümeler oluşturan bir arama sürecidir. Bu süreç, hiç öznitelik içermeyen, tüm öznitelikleri içeren ya da özniteliklerin rastgele bir altkümesini içeren bir küme ile başlatılır. Özniteliklerin tümünü ya da hiçbir özniteliği içermeyen kümeler ile başlanması durumunda, öznitelikler her bir yinelemede eklenir ya da kaldırılır. Ancak, aramanın rastgele bir altküme ile başlatılması durumunda öznitelikler yinelemeli olarak elenebilir, kaldırılabilir ya da rastgele olarak oluşturulabilir (Langley, 1994; Dash and Lui, 1997). Öznitelik altkümesi oluşturmada dikkat edilmesi gereken unsurlardan biri arama organizasyonudur. Öznitelik altkümesinin tam arama yöntemi ile aranması az sayıda öznitelik içeren veri setleri için bile oldukça maliyetli olmaktadır. Tam arama, N, veri setinde yer alan öznitelik sayısını temsil etmek üzere, 2^N olası altkümenin incelenmesini gerektirmektedir. Bu nedenle öznitelik seçiminde genellikle sezgisel arama yöntemleri kullanılmaktadır (Langley, 1994).

3.2.2 Öznitelik altkümesi değerlendirme

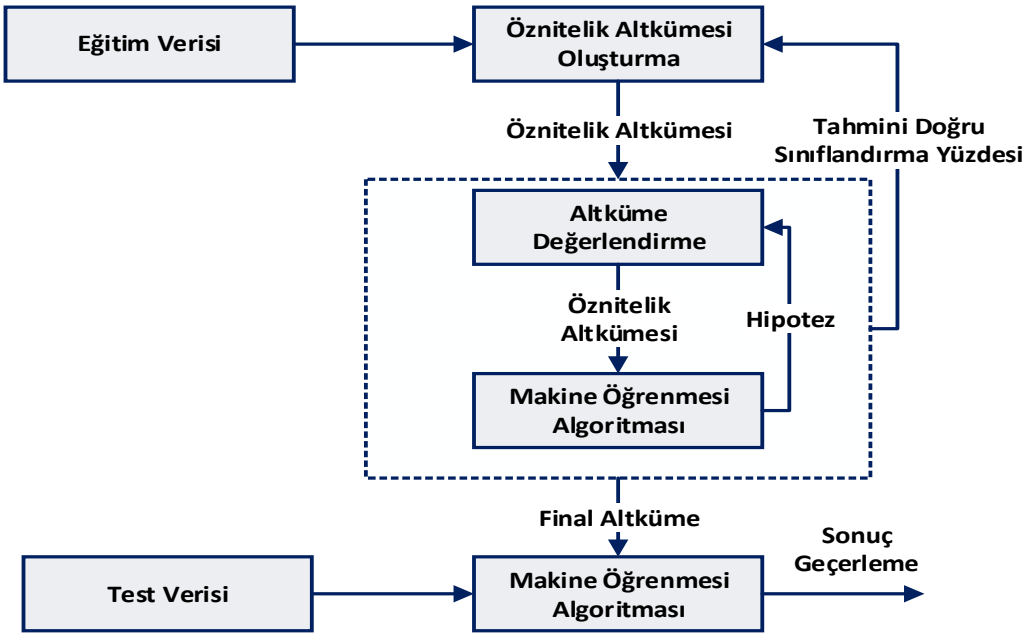
Öznitelik altkümesi değerlendirme, belirli bir altküme oluşturma süreci ile elde edilen öznitelik altkümesinin yararlılığını belirlemek için kullanılır. Öznitelik alt kümelerinin değerlendirilmesinde kullanılan stratejiye göre, öznitelik seçim yöntemleri, filtre tabanlı öznitelik seçim yöntemleri ve sarmalama tabanlı öznitelik seçim yöntemleri olmak üzere iki temel gruba ayrılmaktadır (Chandrashekar and Sahin, 2014). Filtre yöntemleri, öznitelik alt kümesinin yararlılığını incelemek için, verinin genel özelliklerine dayalı sezgiseller kullanır. Filtre yöntemlerinde, belirli bir öğrenme algoritmasının uygulanması ve bu algoritmanın başarımını eniyileyen öznitelik setinin seçilmesi yerine, hedef ile en çok doğrusal ilişkili öznitelikler ya da hedefle en yüksek ortak bilgiye sahip öznitelikler seçilir. Filtre yöntemlerinde, kullanılan sezgisele dayalı olarak, ilgili problem için kullanışlı/yararlı olan öznitelikler belirlenir ve önemsiz olanlar elenerek öznitelik seçimi gerçekleştirilmiş olur. Şekil 3.2'de filtre yöntemlerinin temel adımları özetlenmiştir.

Sarmalama yöntemlerinde ise öznitelik seçimi, belirli bir öğrenme algoritmasının performansına dayalı olarak gerçekleştirilir. Burada, seçilen öğrenme algoritmasının performansının eniyilenmesi amaçlanır. Sarmalama yöntemlerinde öznitelik seçimi belirli bir eğitim setinin, belirli bir öğrenme

algoritması ile elde ettiği sonuca dayalı olarak gerçekleştirildiğinden, filtre tabanlı yöntemlere kıyasla daha yüksek başarım elde edilebilir. Ancak, öznitelik seçiminde öğrenme algoritmalarının yinelemeli olarak çağrılmasını gerektirdiğinden, sarmalama tabanlı yöntemler daha uzun çalışma süresi gerektirir (Hall, 1999). Şekil 3.3'te sarmalama tabanlı öznitelik seçim yöntemlerinin genel yapısı özetlenmiştir.



Şekil 3.2. Filtre-tabanlı öznitelik seçimi sürecinin aşamaları (Tan, 2007)



Şekil 3.3. Sarmalama-tabanlı öznitelik seçimi sürecinin aşamaları (Tan, 2007)

Filtre tabanlı ve sarmalama tabanlı öznitelik seçim yöntemlerinin yanı sıra, melez öznitelik seçim yöntemleri de bulunmaktadır. Melez öznitelik seçim

yöntemleri, hem doğru sınıflandırma başarımını artırmayı, hem de öznitelik seçim sürecini hızlandırmayı amaçlayan yöntemlerdir. Filtre yöntemleri, sarmalama yöntemlerinden önce kullanılarak, özniteliklerin belirli bir ölçüte dayalı olarak sıralaması elde edilerek sarmalama yönteminin daha iyi bir başlangıç çözümü üzerinden işletilmesi ve böylelikle daha hızlı sonuca ulaşılması sağlanabilir (Zhu et al, 2007).

Filtre tabanlı yöntemler, öznitelik değerlendirmede kullanılan yönteme dayalı olarak bireysel öznitelik ölçütleri ve grup öznitelik ölçütleri olmak üzere iki temel sınıf altında incelenmektedir (Diao, 2014). Bireysel öznitelik değerlendirme ölçütleri, özniteliklerin uygunluklarını bireysel olarak değerlendirir. Bireysel öznitelik değerlendirme ölçütlerinde öznitelik alt kümesi, tüm özniteliklerin belirli bir ölçüte göre sıralanması ve eşik değeri aşan özniteliklerin seçilmesi ile oluşturulur. Bireysel öznitelik değerlendirme ölçütleri, öznitelik sıralama yöntemleri olarak da adlandırılmaktadır. Bireysel öznitelik değerlendirme ölçütleri, hesaplama etkinliği bakımından başarılı yöntemlerdir. Grup öznitelik ölçütleri ise aday öznitelik altkümelerinin değerlendirildiği yaklaşımlardır. Böylelikle, öznitelikler arası ilişkiler hesaba katılır (Diao, 2014).

3.2.3 Filtre tabanlı öznitelik seçim yöntemleri

Bu bölümde, geliştirilen öznitelik seçim yönteminde kullanılan, temel filtre tabanlı öznitelik seçim yöntemlerine değinilmektedir.

3.2.1.1 Bilgi kazancı

Bilgi kazancına dayalı öznitelik sıralama yönteminde (*information gain based feature ranking, IG*), bilgi kazancı ölçütü kullanılarak öznitelikler bireysel olarak değerlendirilir. Bilgi kazancı yöntemi, metin madenciliğinde sıklıkla uygulanan öznitelik yöntemleri arasındadır. Bilgi kazancı, entropi değerine dayalı olarak belirlenir. Belirli bir C sınıfının, belirli bir A özniteliğinin gözlenmesi ve gözlenmemesi durumlarındaki entropi değerleri Eşitlik 3.3 ve 3.4'e göre hesaplanır (Hall and Holmes, 2003):

$$H(C) = -\sum_{c \in C} p(c) \log_2 p(c) \tag{3.3}$$

$$H(C|A) = -\sum_{a \in A} p(a) \sum_{c \in C} p(c|a) \log_2 p(c|a) \tag{3.4}$$

Bilgi kazancı ölçütü, belirli bir *A* özniteliğinin görülmesi durumunda ortaya çıkan ek bilgiyi belirtmek için kullanılır. Her bir A_i özniteliği ile sınıf arasındaki bilgi kazancı Eşitlik 3.5'te belirtilen formüle göre hesaplanır (Hall and Holmes, 2003):

$$IG_i = H(C) - H(C|A_i) \tag{3.5}$$

Bilgi kazancına dayalı öznitelik seçiminde, her bir öznitelik ve sınıf için bilgi kazancı değerleri hesaplanarak bilgi kazancına dayalı sıralama elde edilir. Özniteliklere ilişkin sıralamanın elde edilmesinin ardından, bilgi vericiliği yüksek olan (bilgi kazancı değeri bakımından yüksek değerler veren) belirli sayıda öznitelik tutulurken, diğer öznitelikler ortadan kaldırılır.

3.2.1.2 Ki-kare ölçütü

Ki-kare ölçütüne dayalı öznitelik sıralama yönteminde (*chi-squared feature ranking, CHI*), ki-kare ölçütü kullanılarak öznitelikler bireysel olarak değerlendirilir. Ki-kare ölçütü, istatistikte sıklıkla kullanılan bir ölçüttür. Bu ölçüt, belirli bir *t* terimi ve *c* kategorisi arasında bağımsızlık olmaması durumunu ölçümlemek için kullanılır (Yang and Pedersen, 1997). Ki-kare ölçütü, Eşitlik 3.6 ve 3.7'ye göre hesaplanır:

$$\chi^2(c,t) = \frac{N \times (AD - BC)}{(A+C)(B+C)(A+B)(C+D)} \tag{3.6}$$

$$\chi^2_{\max}(t) = \max_i(\chi^2(t,c_i)) \tag{3.7}$$

Burada, *A*, *t* teriminin *c* kategorisine ait olduğu durum sayısını, *N*, toplam belge sayısını, *B*, *t* teriminin, *c* kategorisine ait olmadığı durum sayısını, *C*, *c* kategorisinin, *t* terimi olmaksızın görüldüğü durum sayısını, *D*, hem *t* teriminin hem de *c* kategorisinin gözlenmediği durum sayısını temsil etmektedir. Ki-kare ölçütü, her bir kategori ve terim için hesaplanmaktadır.

3.2.1.3 Kazanç oranı ölçütü

Kazanç oranı ölçütü (*gain ratio, Gain*), simetrik olmayan bir ölçüttür. Bilgi kazancı, yüksek değer içeren özniteliklere karşı taraflıdır. Kazanç oranı ölçütü, bilgi kazancının bu sorununu ortadan kaldırmayı amaçlar. *Y* tahmin edilmek istenen öznitelik ve *X* gözlemlenen öznitelik olmak üzere, kazanç oranı ölçütü, bilgi kazancının *X* özniteliğinin entropisine bölünmesi ile elde edilir (Hall and Smith, 1998):

$$GR = \frac{IG}{H(X)} \tag{3.8}$$

Kazanç oranı ölçütü, [0-1] aralığında değerler almaktadır. Kazanç oranının bir değerine sahip olması, *X* özniteliğinin *Y* özniteliğini tamamen tahmin edebildiğini, kazanç oranının sıfır olması ise *X* ve *Y* öznitelikleri arasında ilişki olmadığını göstermektedir. Bilgi kazancı ölçütünden farklı olarak, düşük değerler alan öznitelikler lehine taraflıdır.

3.2.1.4 Simetrik belirsizlik katsayısı

Simetrik belirsizlik katsayısı (*symetrical uncertainty coefficient, SU*), bilgi kazancının (*IG*) daha önce değinilen sorununu ortadan kaldırmak için, bilgi kazancının *X* ve *Y* özniteliklerinin entropi değerleri (sırasıyla *H(X)* ve *H(Y)*) toplamına bölünmesi ile elde edilir. Simetrik belirsizlik katsayısı, Eşitlik 3.9'a göre hesaplanır (Hall and Smith, 1998):

$$SU = 2 \times \left[\frac{IG}{H(Y) + H(X)} \right] \tag{3.9}$$

Simetrik belirsizlik katsayısı, [0-1] aralığında değerler almaktadır. Katsayı değerinin bire eşit olması, bir özniteliğe ait bilginin diğer özniteliği tamamen tahmin etmek için yeterli olduğunu, katsayı değerinin sıfıra eşit olması ise özniteliklerin birbirleriyle ilintisiz olduklarını göstermektedir. Kazanç oranı ölçütünde olduğu gibi, simetrik belirsizlik katsayısı da düşük değerler alan öznitelikler lehine taraflıdır.

3.2.1.5 Pearson korelasyon katsayısı

Pearson korelasyon katsayısı (*Pearson correlation coefficient, PCC*), iki değişken arasındaki korelasyonu ölçümlemek için kullanılır. Pearson korelasyon katsayısı, *X*, *x* değerlerini alan özniteliği, *Y*, *y* değerlerini alan sınıfı ve $\bar{x}$ ile $\bar{y}$ ilgili değerlerin *k* indeksine göre ortalama değerlerini temsil etmek üzere, Eşitlik 3.10'da belirtilen formüle göre hesaplanır (Guyon and Elisseeff, 2003):

$$R(i) = \left[\frac{\sum_{k=1}^{m} (x_{k,i} - \overline{x_i})(y_k - \bar{y})}{\sqrt{\sum_{k=1}^{m} (x_{k,i} - \overline{x_i})^2 \sum_{k=1}^{m} (y_k - \bar{y})^2}} \right] \tag{3.10}$$

3.2.1.6 Relief-F algoritması

Relief algoritması (RL), filtre tabanlı bir öznitelik seçim yöntemidir (Kira and Rendell, 1992b). Relief algoritmasında, özniteliklerin sıralanmasında kullanılan diğer istatistiksel ölçütlerin aksine, bağlamsal bilgi de göz önünde bulundurularak, öznitelikler arası sıkı bağlantıların dikkate alınması amaçlanır (Robnik-Sikonja and Kononenko, 2003). Relief algoritması, özniteliklerin yararlılığını, farklı sınıflarda ve aynı sınıfta yer alan örneklerin, öznitelik değerlerinin farklı olup olmamasına göre değerlendiren bir öznitelik seçimidir. Relief algoritması, aynı sınıfta yer alan örnekler için, farklı öznitelik değerleri alan, özniteliklerin yararlılığı düşürürken, birbirinden farklı sınıflarda yer alan örneklerin ilgili öznitelik değeri farklı olduğu takdirde, bu özniteliği değerli kılmaktadır (Kira and Rendell, 1992b). Bu doğrultuda, Relief algoritması, rastgele olarak seçilen herhangi bir R_i örneğinin, biri aynı sınıftan (H), diğeri farklı sınıftan (M) olmak üzere en yakın iki komşusunu belirler. Ardından, aynı sınıfta ve farklı sınıfta yer alan komşular ile ilgili örneğin öznitelik değerlerinin farklı/aynı olması durumuna göre, ilgili özniteliğin yararlılığı güncellenir. Ancak, Relief algoritması, yalnızca iki sınıflı problemlere uygulanabilmektedir. Relief-F algoritması, Relief algoritmasının, eksik ve gürültülü değerler içeren veri setleri ile çok sınıflı sınıflandırma problemlerinde etkin bir biçimde çalışabilmesi için geliştirilmiştir. Relief-F algoritmasında, Relief algoritmasında olduğu gibi, rastgele olarak seçilen herhangi bir R_i örneği üzerinden süreç işletilir. Ancak, burada en yakın iki komşu yerine, aynı sınıftan k, farklı sınıftan k tane en yakın komşu incelenir. Algoritma 3.1'de Relief-F algoritmasının genel işleyişi sunulmuştur. Burada, m, kullanıcı tanımlı parametre değeri olmak üzere, sürecin kaç kez yineleneceğini belirtmektedir. Öznitelikleri ise A_i, i=1,…,a olacak şekilde temsil edilmektedir.

Burada, A, veri setinde yer alan herhangi bir özniteliği, H_j, incelenmekte olan örneğin, aynı sınıfta yer alan k en yakın komşusunu, M_J, incelenmekte olan örneğin farklı sınıfta yer alan k en yakın komşusunu ve $W[A]$, A özniteliğinin yararlılık değerini temsil etmektedir. Relief-F algoritması, belirli bir özniteliğin yararlılığını, yinelemeli örneklem alarak belirler. Belirli bir özniteliğin değeri, farklı sınıfların en yakın örnekleri için, ilgili özniteliğin değeri kullanılarak değerlendirilir. Algoritma, rastgele olarak bir R_i örneğinin seçilmesi ile başlar. Ardından bu örneğin aynı sınıfta yer alan k en yakın komşusu (H_j) ve farklı sınıflarda yer alan k en yakın komşusu belirlenir (M_J(C)). Her bir öznitelik için, nitelik değerlendirmesi, aynı sınıfta yer alan ve farklı sınıflarda yer alan k en

yakın komşulara dayalı olarak yapılır. Süreç, kullanıcı tarafından belirlenen *m* parametresi kadar yinelenir (Robnik-Sikonja and Kononenko, 2003).

Algoritma 3.1 Relief-F algoritması (Robnik-Sikonja and Kononenko, 2003)

Girdi: Her bir eğitim örneği için öznitelik değerleri ve sınıf değerlerine ilişkin bir vektör

Çıktı: Özniteliklerin nitelik değerlendirmesi *W*

W[A] :=0,0;

for i:=1 to *m* do begin

Rastgele olarak bir R_i örneğini seç,

R_i örneğinin aynı sınıfta yer alan *k* en yakın komşusunu (H_j) belirle,

R_i örneğinin farklı sınıflarda yer alan *k* en yakın komşusunu (M_J(C)) belirle,

for A:=1 to a do

// Nominal öznitelikler için *diff* fonksiyonunun hesaplanması:

$$diff(A, i_1, i_2) = \begin{cases} 0 & \deg er(A, i_1) = \deg er(A, i_2) \\ 1 & aksi \quad takdirde \end{cases}$$

// Nümerik öznitelikler için *diff* fonksiyonunun hesaplanması:

$$diff(A, i_1, i_2) = \frac{\left|\deg er(A, i_1) - \deg er(A, i_2)\right|}{maks(A) - \min(A)}$$

$$W[A] := W[A] - \sum_{j=1}^{k} diff(A, R_i, M_j(C))/(m, k);$$

$$\sum_{c \neq class(R_i)} \left[\frac{P(C)}{1 - P(class(R_i))} \sum_{j=1}^{k} diff(A, R_i, M_j(C))/(m, k) \right]$$

end

end

3.2.1.7 Olasılıklı önem ölçütü

Olasılıklı önem ölçütü (*probabilistic significance measure, PS*), filtre tabanlı bir öznitelik sıralama yöntemidir (Ahmad and Dey, 2005). İki örnek için sınıf etiketleri farklı olduğu takdirde, bu iki örneğe ilişkin önemli/bilgilendirici özniteliğin almış olduğu değerlerin birbirinden farklı olması beklenir. Olasılıklı önem ölçütü, öznitelikleri iki yönlü bir fonksiyon sonucunda sıralar. Her bir A_i özniteliği için, özniteliğin sınıfa karşı ilişkisi ($AE(A_i)$) hesaplanır. Özniteliğin sınıfa karşı ilişkisi, A_i özniteliğinin tüm olası değerlerinin, sınıf kararlarına etkisini belirlemeyi amaçlar. Bunun yanı sıra, sınıf etiketleri değiştiğinde özniteliklerin değerlerinde meydana gelen değişimleri göz önünde bulundurmak için sınıfların özniteliklere karşı ilişkisi ($CE(A_i)$) hesaplanır. Belirli bir öznitelik hem ($AE(A_i)$)

hem ($CE(A_i)$) değeri yüksek olduğu takdirde, önemli bir öznitelik olarak kabul edilir.

Belirli bir A_i özniteliği için, özniteliğin sınıfa karşı ilişkisi ($AE(A_i)$), Eşitlik 3.11'e göre hesaplanır. Bu değer, özniteliğin ayırt edicilik gücünün (υ_i^r), özniteliğin tüm olası değerleri için ortalamasını alan bir fonksiyondur. k tane farklı öznitelik değeri alan bir özniteliğin "özniteliğin sınıfa karşı ilişkisi" değeri [0-1] aralığında değer alır (Ahmad and Dey, 2005).

$$AE(A_i) = \left(1/k \sum_{r=1,2,\ldots k} \upsilon_i^r \right) - 1.0 \tag{3.11}$$

Belirli bir A_i özniteliği için, sınıfların özniteliklere karşı ilişkisi ($CE(A_i)$) ise Eşitlik 3.12'ye belirlenir. Özniteliğin sınıf değerlerinden ayırt edilebilirliği ($\wedge_i^j$) dikkate alınır. Burada, m, toplam sınıf sayısını, j, incelenmekte olan sınıfı temsil etmektedir (Ahmad and Dey, 2005).

$$CE + (A_i) = (1/m) \times \left(\sum_{j=1,2,\ldots m} \wedge_i^j \right) - 1.0 \tag{3.12}$$

3.3 Sınıflandırma Algoritmaları

Sınıflandırma algoritmaları, metin madenciliği ve makine öğrenmesine dayalı görüş madenciliğinde başarıyla uygulanmaktadır. Metin ya da görüş sınıflandırmada sıklıkla kullanılan sınıflardan bazıları, Naive Bayes algoritması, destek vektör makineleri, radyal tabanlı fonksiyon ağları, K-en yakın komşu algoritması, karar ağacı algoritmaları, lojistik regresyon ve lineer diskirimant analizidir (Sebastiani, 2002). Bu bölümde, başlıca sınıflandırma algoritmalarının temel özellikleri aktarılmaktadır.

3.3.1 Naive Bayes algoritması

Naïve Bayes algoritması (NB), Bayes teoremine dayalı olasılık tabanlı bir sınıflandırıcıdır. Naïve Bayes sınıflandırıcısı, belirli bir sınıfa ait öznitelik değerinin etkisinin, diğer öznitelik değerlerinden bağımsız olduğunu varsayar. Bu varsayım, sınıf koşul bağımsızlığı olarak isimlendirilmektedir. Sınıf koşul bağımsızlığı, belirli bir özniteliğin var olmasının ya da olmamasının, başka bir özniteliğin var olmasını, olmamasını ya da değerini etkilemeyeceğini ifade eder. Sınıf koşul bağımsızlığı varsayımı, gerekli hesaplamaların daha kolay bir biçimde

gerçekleştirilmesini sağlamaktadır. Naïve Bayes sınıflandırıcısı, basit yapısı, hesaplama etkinliği ve yüksek doğru sınıflandırma başarımı nedeniyle metin madenciliği başta olmak üzere pek çok sınıflandırma çalışmasında başarı ile uygulanmaktadır. Basit yapısına ve temel aldığı sınıf koşul bağımsızlığı varsayımının doğru olmamasına karşın, daha karmaşık yapıya sahip ve hesaplama etkinliği düşük sınıflandırma algoritmalarından daha iyi sonuçlar verebilmektedir (Shmueli et al., 2010). Sınıflandırma alanında yapılan karşılaştırmalı çalışmalarda, Naïve Bayes sınıflandırıcısının, karar ağacı ve yapay sinir ağları yöntemleri ile karşılaştırılabilir sonuçlar verdiği gözlenmektedir (Han and Kamber, 2006).

Bayes teoremi Eşitlik 3.13'e göre işlemektedir:

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)} \tag{3.13}$$

Burada, *X,* öznitelik vektörü, *H* ise bir öznitelik vektörünün belirli bir sınıfa ait olma olasılığını temsil eden bir hipotezdir. *P(H|X)* ise ardıl olasılığı temsil eder.

Naïve Bayes sınıflandırma algoritması temel olarak şu şekilde çalışmaktadır (Han and Kamber, 2006):

1. *D'*nin veri setini temsil ettiği ve *D'*deki her *X'*in sınıf etiketinin belli olduğu varsayılmak üzere, *X, n* tane öznitelikten oluşan bir vektördür ve $X=(x_1, x_2, \ldots, x_n)$ şeklinde temsil edilmektedir.
2. $C_1, C_2, \ldots, C_m$ ile temsil edilen *m* tane sınıf olduğunu varsayalım. Naïve Bayes sınıflandırıcısı, bir *X* vektörünün herhangi bir C_i sınıfına ait olup olmadığını belirlemek için, bütün sınıflar için Eşitlik 3.14'e göre, $P(C_i|X)$ ardıl olasılığını hesaplar:

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)} \tag{3.14}$$

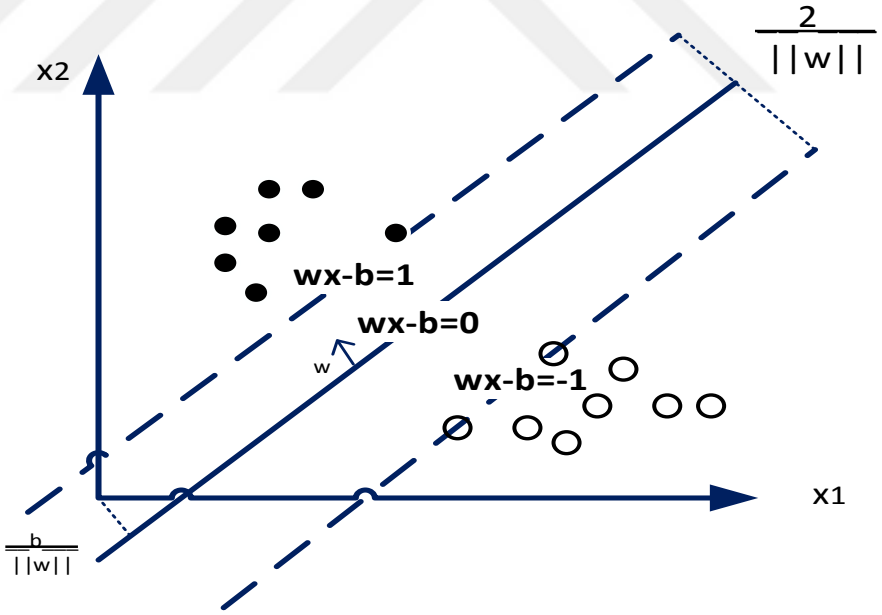
3. $P(X)$ değeri tüm sınıflar için sabit bir değer aldığından, yalnızca $P(X|C_i)$ ifadesinin maksimize edilmesi gerekmektedir.
4. $P(C_i)$ ifadesi C_i sınıfındaki elemen sayısının tüm eleman sayısına oranıdır.
5. $P(X|C_i)$ ifadesi ise, *X'*in *n* tane değer içeren bir öznitelik vektörü olduğu varsayılarak, Eşitlik 3.15'e göre hesaplanır:

$$P(X|C_i) = \prod_{k=1}^{n} P(x_k|C_i) \quad (3.15)$$

6. $P(X|C_i)\ P(C_i)$ ifadesini maksimum yapan C_i sınıfı, X vektörünün sınıfı olarak belirlenir.

3.3.2 Destek vektör makineleri

Destek vektör makineleri (*support vector machines, SVM*), hem doğrusal hem de doğrusal olmayan verilerin sınıflandırılmasında kullanılan temel algoritmalardan biridir. Destek vektör makineleri, doğrusal olmayan bir eşleme yöntemi kullanarak, orijinal verinin daha üst bir boyuta dönüştürülmesini sağlar. Bu yeni boyutta, veriyi en uygun şekilde ayıracak bir üst düzlem arar. Bu üst düzlem, bir sınıfın diğerinden ayrılmasını sağlayan karar sınırıdır. Uygun bir doğrusal olmayan eşleme yöntemi kullanılarak, iki sınıf içeren bir veri seti her zaman bir üst düzlem kullanılarak ayrılabilmektedir (Vapnik, 1995). Şekil 3.4'te iki sınıfa ait örnekler ile eğitilmiş bir destek vektör makinesi algoritması için elde edilen destek vektörleri ve üst düzlem gösterilmektedir.



Şekil 3.4. Destek vektörleri ve üst düzlem

Burada, iki farklı sınıfa ait örnekler (siyah içi dolu örnekler bir sınıfı ve içi boş örnekler diğer sınıfı temsil etmek üzere) bulunmaktadır. Her bir örnek iki özniteliğe ilişkin (X_1 ve X_2) değerlere sahiptir. Destek vektör makineleri, bu iki sınıfa ait örnekleri doğrusal bir üst düzlem aracılığı ile en uygun şekilde (iki sınıfı

birbirinden en uzak şekilde ayıracak biçimde) ayırmayı amaçlar. İki öznitelik içeren verilerin sınıflandırılmasında, örnekler doğrusal bir çizgi aracılığıyla; üç öznitelik içeren verilerin sınıflandırılmasında, örnekler bir düzlem aracılığıyla; dört ve daha çok öznitelik içeren veriler için ise üst düzlem aracılığıyla ayrılmaktadır.

İki sınıfın örneklerini birbirinden ayıran bir üst düzlem bulunmaktadır. Bu üst düzlem, Eşitlik 3.16'daki gibi temsil edilebilir:

$$W.X + b = 0 \tag{3.16}$$

Burada, *W*, ağırlık vektörünü ($W=\{w_1, w_2,\ldots,w_n\}$), *n*, öznitelik sayısını ve *b*, sapma değerini temsil etmektedir. İki öznitelikli durum için üst düzlem Eşitlik 3.17'deki gibi yeniden yazılabilir:

$$w_0 + w_1 x_1 + w_2 x_2 = 0 \tag{3.17}$$

Üst düzlemin üzerinde yer alan herhangi bir nokta Eşitlik 3.18'de ve üst düzlemin altında yer alan herhangi bir nokta Eşitlik 3.19'da verilen koşulları sağlamaktadır:

$$w_0 + w_1 x_1 + w_2 x_2 > 0 \tag{3.18}$$

$$w_0 + w_1 x_1 + w_2 x_2 < 0 \tag{3.19}$$

Destek vektör makineleri yönteminde, üst düzleme en yakın pozitif ve negatif örnekler arasındaki mesafe marjin genişliği olarak tanımlanmaktadır. Marjinin tarafları dikkate alınarak, Eşitlik 3.18 ve Eşitlik 3.19 aşağıda verildiği şekilde yeniden yazılabilir (Han and Kamber, 2006):

$$H_1 : w_0 + w_1 x_1 + w_2 x_2 \geq 1, (y_i = +1) \text{ için} \tag{3.20}$$

$$H_2 : w_0 + w_1 x_1 + w_2 x_2 \leq -1, (y_i = -1) \text{ için} \tag{3.21}$$

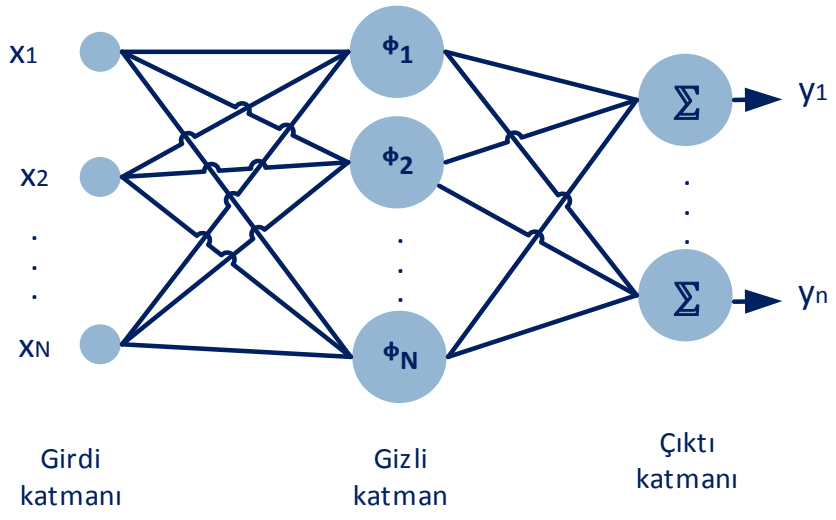
H_1 ya da H_2 ile verilen eşitlikleri sağlayan örnekler destek vektörleri olarak isimlendirilmektedir. Daha önce de değinildiği gibi, destek vektör makinelerinin temel amacı, bu destek vektörleri arasındaki uzaklığı en çok yapan kenarı bulmaktır. Bu kenar içinde bulunan ve destek vektörlerine eşit uzaklıkta olan bir üst düzlem, sınıfları birbirine en iyi biçimde ayırmaktadır.

Destek vektör makineleri, genelleştirme yetenekleri yüksek makine öğrenmesi yöntemleridir. Genelleştirme yeteneği, belirli bir öğrenme algoritmasının mevcut örneklerden hareketle, yeni durumlar ile karşılaşıldığında

uygun sonuçlar üretebilmesi olarak tanımlanabilir. Destek vektör makineleri, küresel en iyi sonucu bulmaya yöneliktir. Bunun yanı sıra, destek vektör makineleri eşitliklerinde yer alan mesafe parametresinin yanlış sınıflandırma hatasını kontrol etmesi sayesinde gürültülü ve aykırı değerler içeren veri setlerine karşı dayanıklıdır. Destek vektör makineleri yönteminin en temel iki sorunu ise eğitim için harcanan sürenin fazla olması ve parametrelerinin ayarlanmasıdır (Abe, 2010).

3.3.3 Radyal tabanlı fonksiyon ağları

Radyal tabanlı fonksiyon ağları (*radial basis function networks, RBF*), iki katmanlı ileri beslemeli bir yapay sinir ağı mimarisine sahiptir. Bu mimaride, girdi katmanı, çıktı katmanı ve girdi katmanı ile çıktı katmanı arasında tek bir gizli katman yer almaktadır. Gizli katmanda yer alan nöronların aktivasyonu için radyal tabanlı fonksiyonlar kullanılmaktadır (Bors, 2001). Radyal tabanlı fonksiyonların doğrusal ya da doğrusal olmayan modellerde ve tek ya da çok katmanlı ağ mimarilerinde uygulanması mümkün olsa da Broomhead and Lowe (1988)'de kullanılan yapıdan ötürü, radyal tabanlı fonksiyon ağları, tek gizli katman içeren bir yapıya sahiptir (Orr, 1996). Çıktı katmanında, gizli katmandan elde edilen sonuçlar birleştirilerek ağın çıktısı oluşturulmaktadır. Şekil 3.5'te radyal tabanlı fonksiyon ağı mimarisi gösterilmiştir.



Şekil 3.5. Radyal tabanlı fonksiyon ağı mimarisi

Radyal tabanlı fonksiyon ağının çıktı katmanındaki herhangi bir *k* nöronunun çıktısı Eşitlik 3.22'de verilen formüle göre hesaplanmaktadır:

$$y_k(x) = \sum_{j=1}^{m} w_{kj} h_j(x) \tag{3.22}$$

Burada, h_j (x), $j=1, \ldots, m,$ radyal biriminin girdi örüntüsü x ve radyal birim j ile çıktı nöronu k arasındaki w_{kj} ağırlığı için aktivasyon değerini belirtmektedir. Radyal birimin aktivasyon değeri, girdi örüntüsü ile gizli birim merkezi arasındaki uzaklığa bağlıdır (Tinos and Murta, 2009). Radyal tabanlı fonksiyon ağlarında kullanılmak üzere geliştirilmiş farklı aktivasyon fonksiyonları bulunmak ile birlikte, zaman serileri modellemesine ilişkin uygulamalarda genellikle ince tabaka kama modeli, örüntü sınıflandırmasına ilişkin uygulamalarda ise Gauss fonksiyonları seçilmektedir (Bors, 2001). Radyal tabanlı fonksiyon ağlarında öğrenme genellikle iki aşamalı olarak gerçekleşmektedir. Bu iki aşamalı yaklaşımda, ağın farklı katmanları birbirlerinden bağımsız olarak eğitilerek, öncelikle merkez ve ölçeklendirme parametrelerinin ve ardından çıktı katmanındaki ağırlık değerlerinin uyarlanması işlemleri gerçekleştirilmektedir (Schwenker et al., 2001). Radyal tabanlı fonksiyon ağ merkezlerinin eğitilmesinde genellikle, kümeleme analizi, vektör nicemlemesi, sınıflandırma ağacı algoritmaları, eğim düşümü algoritması gibi yaklaşımlar kullanılmaktadır.

3.3.4 K-en yakın komşu algoritması

K-en yakın komşu algoritması (*K-nearest neighbor algorithm, KNN*), sınıflandırma işlemini, sınıflandırılmak istenen örnek ile sınıflardaki örnekler arasındaki benzerliğe dayalı olarak gerçekleştiren bir yöntemdir. Eğitim örnekleri n boyutlu sayısal nitelikler ile belirtilir. Her örnek n boyutlu uzayda bir noktayı gösterir. Bu şekilde, tüm eğitim örnekleri n boyutlu örnek uzayda tutulur. Bilinmeyen bir örnek ile karşılaşıldığında, k-en yakın komşu algoritması, ilgili örneğe en yakın k tane eğitim örneğini (en yakın komşuyu) belirler.

Yakınlık, Öklit uzaklığı (*Euclidean distance*) ya da başka bir uzaklık ölçütü kullanılarak ölçümlenir. İki nokta ($X_1=x_{11}, x_{12}, \ldots, x_{1n}$) ve ($Y_1=y_{11}, y_{12}, \ldots, y_{1n}$) olmak üzere, bu iki nokta arasındaki Öklit uzaklığı Eşitlik 3.23 ile hesaplanır:

$$dist(X_1, X_2) = \sqrt{\sum_{i=1}^{n} (x_{1i} - x_{2i})^2} \tag{3.23}$$

Özniteliklerin değerlerinin doğrudan Eşitlik 3.23'te kullanılması, büyük değerlere sahip özniteliklerin (gelir vb.), küçük değerlere sahip öznitelikler (ikili öznitelikler vb.) üzerinde baskınlık kurmasına neden olabilir. Bu nedenle, ilgili öznitelik değerleri uzaklık formülünde kullanılmadan önce veri ayrıştırma

yöntemlerinden biri uygulanarak, belirli bir aralıkta değerler alacak şekilde düzenlenmektedir. Min-maks ayrıştırma yöntemi, tüm özniteliklerin *0* ile *1* arasında değerler almasını sağlamaktadır ve Eşitlik 3.24 ile hesaplanmaktadır:

$$v' = \frac{v - \min_A}{\max_A - \min_A} \tag{3.24}$$

Burada, *A*, özniteliği, *v* ilgili özniteliğin değerini, $\min_A$ ve $\max_A$ ise ilgili özniteliğin aldığı minimum ve maksimum değerleri temsil etmektedir (Han and Kamber, 2006).

K-en yakın komşu algoritmasının en temel özelliği, basit yapısı ve az sayıda parametre gerektirmesidir. Yeterince büyük eğitim setlerinin varlığında, k-en yakın komşu algoritması oldukça etkin sonuçlar verebilmektedir. Ancak, özellikle büyük eğitim veri setleri için en yakın komşuların bulunması oldukça maliyetli olabilmektedir. Bu maliyeti ortadan kaldırmak için temel bileşenler analizi gibi boyut azaltma yöntemleri, daha güçlü veri yapıları (arama ağaçları) gibi yöntemler kullanılabilmektedir (Shmueli et al., 2010). K-en yakın komşu algoritmasının diğer bir problemi de eğitim verisinde ilgisiz özniteliklerin bulunmasından, ciddi bir biçimde etkilenmesidir. K-en yakın komşu algoritması, ilgisiz özniteliklerin varlığında da sınıflandırma modeli oluşturabilmektedir. Ancak, böyle durumlarda eğitim için gereken süre oldukça artmaktadır (Aha et al., 1991).

3.3.5 C4.5 karar ağacı algoritması

C4.5 algoritması, sürekli ve kesikli değerler ile çalışabilen bir karar ağacı algoritmasıdır. Karar ağacı yöntemi, sınıflandırma ve tahmin etmede kullanılan önemli veri madenciliği teknikleri arasında yer almaktadır. Karar ağacı, girdisi olmayan bir kök düğüm ve her biri birer girdi alan iç düğümlerden oluşan yönlü bir ağaçtır. Çıktıları bir başka düğüm tarafından girdi olarak alınan düğümler iç ya da test düğümü, çıktıları bir başka düğüme girdi olmayan düğümler ise yaprak düğümler olarak adlandırılmaktadır. Karar ağacında her bir iç düğüm, örnek uzayını girdi öznitelik değerlerinin belirli bir fonksiyona tabi tutulmasına dayalı olarak iki ya da daha fazla parçaya ayırmaktadır (Rokach and Maimon, 2010). Karar ağacının iç düğümleri öznitelikler üzerinde gerçekleştirilen testleri, dallar test sonuçlarını ve her bir yaprak düğüm sınıf etiketini temsil etmektedir.

Karar ağaçlarının sınıflandırılmada kullanılmasında, karar ağaçlarının basit yapısı sayesinde, oluşturulan sınıflandırma modelinin kolay anlaşılabilir olması,

karar ağaçlarının parametrik olmaması sayesinde, bilgi keşfi için uygun bir yapı sunması, diğer sınıflandırma yöntemlerine kıyasla kısmen daha hızlı bir biçimde oluşturulması gibi özellikler rol oynamaktadır (Gehrke, 2003). Bunun yanı sıra, karar ağaçlarından kuralların elde edilmesi de oldukça kolay bir biçimde gerçekleştirilebilmektedir. Karar ağaçları hem kategorik hem de nümerik verilerin sınıflandırılmasında kullanılabilmektedir. Karar ağaçları, değinilen üstün özelliklerine karşın, birden fazla öznitelik içeren çıktıları olanaklı kılmamaları, kısmen değişken sonuçlar üretmeleri, test verisindeki küçük değişikliklere karşı bile duyarlı olmaları, nümerik veri setleri için karmaşık bir ağaç yapısı oluşturmaları gibi problemler ile karşı karşıya kalmaktadır (Zhao and Zhang, 2008).

Sınıflandırma ve tahmin etmede karar ağaçlarının kullanılması, eğitim verisinden karar ağacı modelinin oluşturulması, bu modelin, test verisi kullanılarak uygun sınama ölçütleri aracılığıyla değerlendirilmesi ve ilgili modelin gelecekteki değerleri tahmin etmede kullanılması şeklinde işlemektedir (Garcia-Laencina et al., 2010; Birant, 2011).

Karar ağaçlarında, kural düğümleri olarak hangi özniteliklerin seçileceğinin belirlenmesi önemlidir. C4.5 algoritmasında, öznitelik seçimi kazanç oranına (*gain ratio*) dayalı olarak belirlenmektedir (Han and Kamber, 2006). Bir *A* özniteliğinin bilgi kazancı *Gain(A)* ile temsil edilmek üzere, Eşitlik 3.25'e göre hesaplanmaktadır. Bir veri setinin entropi değeri *Info(D)* ile temsil edilmek üzere, Eşitlik 3.26'ya göre belirlenmektedir. Belirli bir *A* özniteliğinin değerlerine göre, örneklerin bu değerlerdeki dağılışlarının entropisi $Info_A(D)$ ile temsil edilmek üzere, Eşitlik 3.27'ye göre belirlenmektedir.

$$Gain(A) = Info(D) - Info_A(D) \tag{3.25}$$

$$Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i) \tag{3.26}$$

$$Info_A(D) = -\sum_{j=1}^{y} \frac{|D_j|}{|D|} \times Info(D_j) \tag{3.27}$$

Burada, *m*, sınıf sayısını, p_i, her sınıfın öncül olasılığını, *y*, *A* özniteliğinin aldığı farklı değer sayısını temsil etmektedir. Kazanç oranı Eşitlik 3.28'e göre, ayrıklaştırmada kullanılan *SplitInfo* ölçütü ise Eşitlik 3.29'a göre hesaplanmaktadır:

$$Gain\_Ratio(A) = \frac{Gain(A)}{SplitInfo(A)} \tag{3.28}$$

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right) \tag{3.29}$$

C4.5 algoritması, eksik öznitelik değerleri içeren eğitim veri setleri ile çalışabilmekte, karar ağacı oluşturma sırasında ya da sonrasında bazı düğümlerin/alt ağaçların silinmesi ile aşırı uygunluk problemini kaldırmakta, eğitim setindeki istisnai ve gürültülü değerlerin çıkarılmasını sağlamaktadır (Niuniu and Yuxun, 2010).

3.3.6 Lojistik regresyon

Lojistik regresyon (*logistic regression, LR*), istatistiksel bir sınıflandırma algoritmasıdır. Lojistik regresyon sınıflandırıcısında da Naive Bayes algoritmasında olduğu gibi eğitim setinde yer alan örneklere dayalı olarak bir sınıflandırma modeli oluşturulması ve yeni örneklerin en yüksek olasılık değerine sahip sınıfa atanması söz konusudur. Belirli bir *d* belgesi, (w_1, ..., w_M) şeklinde bir vektör ile temsil edilmek üzere, lojistik regresyon sınıflandırıcısı tarafından en yüksek olasılık değerini ($P(c_j|d)$) aldığı c_j sınıfına atanır. Naive Bayes algoritmasında, olasılık değerlerinin hesaplanması Bayes teoremine dayalı olarak yapılırken, lojistik regresyon sınıflandırıcısında $P(c_j|d)$ olasılık değeri doğrudan parametreler üzerinden hesaplanır. Lojistik regresyonda son olasılık $P(c_j|d)=P(c_j|w_1, ..., w_M)$ şu şekilde temsil edilir (Shatkay and Craven, 2012):

$$P(c_j|w_1,..., w_M) = \frac{e^{\left(\lambda_{j0}+\sum_{j=1}^{M}\lambda_{ji}w_i\right)}}{\sum_{k=1}^{K} e^{(\lambda_{k0}+\sum_{j=1}^{M}\lambda_{ki}w_i)}} \tag{3.30}$$

Burada, λ_{j0}, c_j sınıfı için ağırlık değerini temsil ederken $\lambda_{j1},\ldots, \lambda_{jM}$ bireysel terim olasılıklarına karşılık gelir. Bu parametreler, eğitim verilerinden öğrenilir. Parametreler eş zamanlı olarak tahmin edilerek, eğitim setindeki her bir *d* belgesine atanan sınıfa ilişkin koşullu olasılık değeri maksimize edilir. Lojistik regresyon sınıflandırıcısında belirli bir belgenin sınıflandırılmasında her bir c_j sınıfı için $\lambda_{j0} + \sum_{i=1}^{M} \lambda_{ji} w_i$ doğrusal fonksiyonu hesaplanır. Ardından Eşitlik 3.30'da verilen formüle dayalı olarak, her bir sınıf için olasılık değeri belirlenir. Naive Bayes algoritması da istatistiksel bir sınıflandırıcı olmasına karşın, sınıf koşul bağımsızlığı varsayımına dayanmaktadır. Bu nedenle, lojistik regresyon

sınıflandırıcısının Naive Bayes sınıflandırıcısına kıyasla daha yüksek başarıma sahip sınıflandırma modelleri oluşturması beklenmektedir.

3.3.7 Bayesci lojistik regresyon

Bayesci lojistik regresyon (*Bayesian logistic regression, BLR*), Laplace ve Gauss dağılımlarını kullanan, istatistik tabanlı bir sınıflandırma algoritmasıdır (Genkin et al., 2007). Bayesci lojistik regresyon algoritmasında, sınıflandırma modelinde aşırı uygunluk probleminin ortadan kaldırılması amaçlanır. Bunun yanı sıra, metin veri setleri için sıkı bir tahmin modeli oluşturulması amaçlanır. Metin madenciliği veri setlerinde görülen en önemli sorunlardan biri yüksek boyutluluk problemidir. Lojistik regresyon sınıflandırıcısının maksimum olabilirlik tahmin yöntemine dayalı olması, metin sınıflandırma problemlerinde lojistik regresyon yönteminin kullanılmasını zorlaştırmaktadır. Bayesci lojistik regresyon yöntemi ile metin madenciliğinde değinilen problemlerin ortadan kaldırılması amaçlanmaktadır.

3.3.8 Lineer diskirimant analizi

Lineer diskriminant analizi (*linear discriminant analysis, LDA*), Fisher lineer diskriminant fonksiyonu ya da başka bir diskriminant fonksiyon aracılığıyla sınıflandırma modelinin oluşturulduğu bir yöntemdir. Burada, LDA sınıflandırıcının eğitilmesi için her bir sınıfa ilişkin öncül olasılık değerleri tahmin edilir. LDA sınıflandırıcı için ortak kovaryans matrisi, sınıflara ilişkin koşullu kovaryans matrislerinin ağırlıklı ortalaması hesaplanarak elde edilir. Ardından, diskriminant fonksiyonunun katsayıları ve serbest terimleri belirlenir (Kuncheva, 2014). Sınıflandırmada lineer diskriminant analizi, basit yapısına karşın, etkin sonuçlar verebilmektedir.

3.4 Sınıflandırıcı Toplulukları

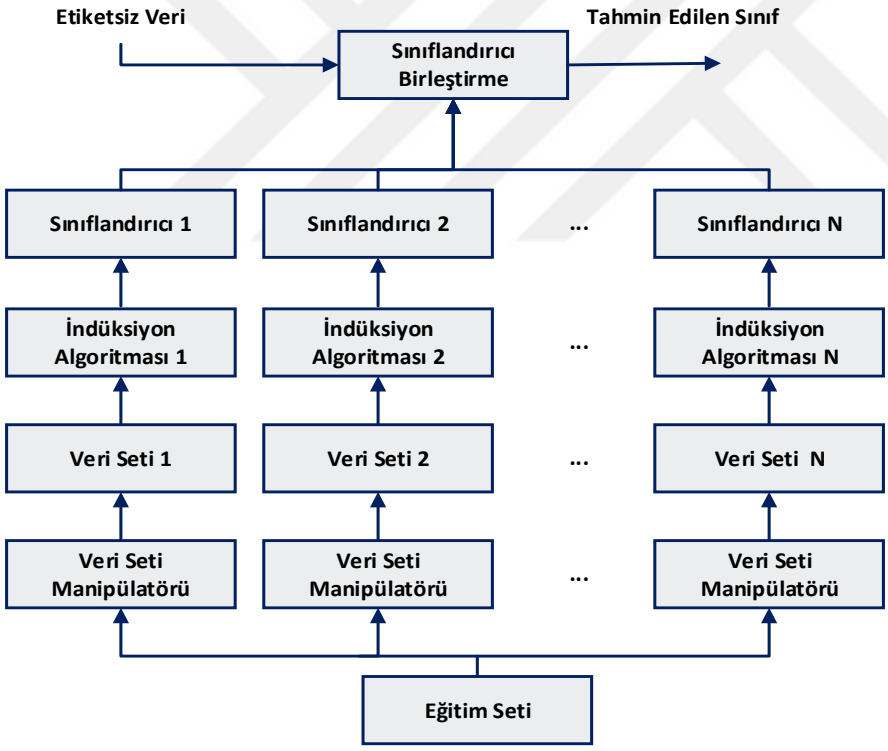
Sınıflandırıcı toplulukları yöntemi (*classifier ensemble method*), sınıflandırma işleminin, tek bir öğrenme algoritması ile elde edilen sonuç yerine birçok farklı öğrenme algoritmasının eğitilmesi ve sonucun, oluşturulan kararların birleştirilmesi ile elde edilmesine yönelik bir makine öğrenmesi yöntemidir (Dietterich, 2000). Sınıflandırıcı topluluklarının genelleştirme yetenekleri, tek bir sınıflandırıcıya kıyasla daha iyidir. Sınıflandırıcı topluluklarına dayalı sınıflandırma modelleri geliştirmenin, istatistiksel, hesaplama ve temsil

bakımından geçerli nedenleri bulunmaktadır (Dietterich, 2000). İstatistiksel açıdan bakıldığında, yeterli verinin varlığında farklı sınıflandırıcıların elde edilmesi mümkündür. Böylelikle, farklı sınıflandırıcılar bir araya getirilerek, yanlış bir karara varma riski azaltılmış olur. Bazı sınıflandırma algoritmaları yerel en iyi değerlere takılabilmektedir. Bunun yanı sıra, bazı sınıflandırma algoritmalarının başarımları, başlangıçta belirlenen parametre değerlerine dayalı olarak önemli farklılıklar gösterebilmektedir. Bu nedenle, hesaplama açısından bakıldığında, birbirlerinden bağımsız olarak eğitilmiş sınıflandırıcıların birleştirilmesi ile daha etkin sonuçlar elde edilmesi mümkündür. Öznitelik uzayında, farklı sınıflandırıcılar farklı bölgelerde yüksek başarım gösterebilir. Dolayısı ile bazı uygulamalarda farklı öznitelik temsil şekilleri elde edilerek, her bir öznitelik setinde farklı sınıflandırıcılar elde edilebilir (Polikar, 2006).

Sınıflandırıcı topluluğu mimarisi, temel olarak, eğitim seti, eğitim setinden girdi öznitelikleri ve hedef öznitelikleri doğrultusunda sınıflandırma modeli oluşmasını sağlayan algoritma, sınıflandırıcılar arası çeşitlilik sağlayan bileşen ve farklı sınıflandırıcı çıktılarını birleştiren bileşen olmak üzere dört bileşenden oluşur (Rokach, 2010). Sınıflandırıcı toplulukları, farklı sınıflandırıcıların oluşturulmasında kullanılan ayrıntı seviyesine dayalı olarak, veri seviyesi, öznitelik seviyesi, sınıflandırıcı seviyesi ve sınıflandırıcı birleştirme seviyesi olmak üzere beş temel sınıfa bölünmektedir (Kuncheva, 2014). Veri seviyesinde, farklı veri altkümeleri kullanılarak, öznitelik seviyesinde, farklı öznitelik altkümeleri kullanılarak, sınıflandırıcı seviyesinde farklı temel sınıflandırma algoritmaları kullanılarak ve sınıflandırıcı birleştirme seviyesinde farklı sınıflandırıcı birleştirme kuralları kullanılarak sınıflandırıcı toplulukları oluşturulması amaçlanır. Sınıflandırıcı topluluklarına ilişkin bir diğer sınıflandırma ise sınıflandırıcı topluluklarının oluşturulmasında kullanılan yapıdır (Rokach, 2010). Sınıflandırıcı toplulukları, toplulukların oluşturulmasında kullanılan yapıya dayalı olarak, bağımlı ve bağımsız yöntemler olmak üzere iki temel sınıf altında incelenmektedir. Bağımlı yöntemlerde, bir sınıflandırıcının çıktısı, daha sonraki sınıflandırıcıların oluşturulması için kullanılırken; bağımsız sınıflandırıcı topluluklarında, sınıflandırıcılar birbirlerinden bağımsız olarak oluşturulduktan sonra, bu sınıflandırıcıların çıktıları belirli bir yönteme dayalı olarak birleştirilmektedir. Bu bölümün geri kalanında temel sınıflandırıcı toplulukları ve sınıflandırıcı topluluğu birleştirme kuralları sunulmaktadır.

3.4.1 Bağımlı sınıflandırıcı toplulukları

Bağımlı sınıflandırıcı toplulukları (*dependent methods*), artırımlı toplu öğrenme (*incremental batch learning*) ve model güdümlü örneklem seçimi (*model guided instance selection*) olmak üzere iki temel sınıf altında incelenmektedir (Provost and Kolluri, 1997; Rokach, 2010). Artırımlı toplu öğrenmeye dayalı yöntemlerde, belirli bir yinelemede elde edilen sınıflandırma sonucu, bir sonraki yinelemede öğrenme algoritmasına ön bilgi olarak verilmektedir. Bu nedenle, bir sonraki yinelemenin sınıflandırıcısı oluşturulurken, mevcut eğitim setine ek olarak, bir önceki sınıflandırıcının sonuçları göz önünde bulundurulmaktadır. Bu yaklaşımda, en son yinelemede oluşturulan sınıflandırıcı, son sınıflandırıcı olarak kullanılmaktadır. Model güdümlü örneklem seçimi yönteminde ise bir önceki yinelemede oluşturulan sınıflandırıcılar, bir sonraki yinelemenin eğitim setini ayarlamak için kullanılır (Provost and Kolluri, 1997).

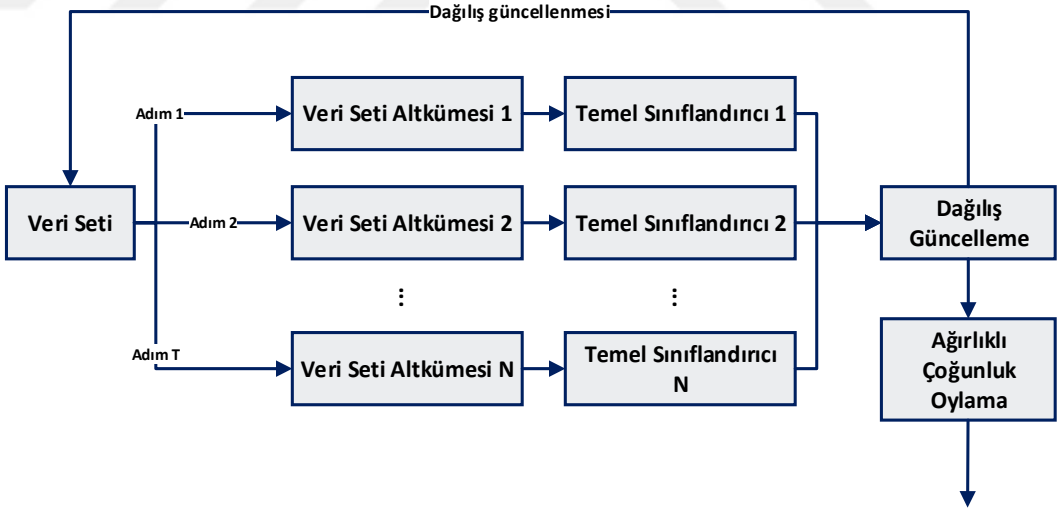


Şekil 3.6. Model güdümlü örneklem seçimi yönteminin genel yapısı (Rokach, 2010)

Şekil 3.6'da model güdümlü örneklem seçim yönteminin temel mimarisi sunulmuştur. Destekleme (*Boosting*) ve adaptif destekleme (*AdaBoost*) algoritmaları, bağımlı öğrenme yöntemine dayalı (model güdümlü örneklem seçimine dayalı) algoritmalardır.

3.4.1.1 AdaBoost algoritması

Destekleme (*Boosting*) algoritması, temel zayıf öğrenme algoritmalarının doğru sınıflandırma başarımlarının iyileştirilmesi için kullanılan, bağımlı öğrenme yöntemine dayalı bir sınıflandırıcı topluluğu yöntemidir. Öğrenme algoritmaları, sınıflandırma performanslarına göre, zayıf ve güçlü sınıflandırıcılar olmak üzere ikiye ayrılmaktadır. Bir sınıflandırma problemi için rastgele tahminden biraz iyi doğru sınıflandırma oranı elde eden sınıflandırma algoritmaları, zayıf sınıflandırıcılar olarak adlandırılırken, sınıflandırma problemi için oldukça yüksek doğru sınıflandırma oranlarına sahip sınıflandırma kuralları ya da hipotezler oluşturan yöntemler ise güçlü sınıflandırıcılar olarak tanımlanmaktadır (Zheng and Xue, 2009). Destekleme algoritması, belirli bir zayıf öğrenme algoritmasının, çeşitli farklı dağılışlara sahip eğitim verisi üzerinde yinelemeli olarak çalıştırılması ve ardından zayıf öğrenme algoritmaları ile elde edilen sınıflandırıcılar birleştirilerek tek bir güçlü sınıflandırma modelinin oluşturulması ilkesine dayanmaktadır. AdaBoost (*Adaptive Boosting, adaptif destekleme*) algoritması, destekleme algoritmasının, yinelemeli bir sürece dayalı olarak iyileştirilmesine yönelik olarak geliştirilmiş bir algoritmadır (Freund and Schapire, 1996).



Şekil 3.7. AdaBoost algoritmasının genel mimarisi (Wang et al., 2014)

AdaBoost algoritmasında, sınıflandırılması zor olan örneklere daha fazla odaklanılarak başarımın artırılması amaçlanır. Eğitim setindeki, her bir örneğe bir ağırlık değeri atanır. Algoritmanın her bir yinelemesinde, doğru sınıflandırılmayan örneklere ilişkin ağırlık değerleri artırılırken, doğru sınıflandırılmış örneklerin ağırlık değerleri azaltılır. Böylelikle, zayıf öğrenme

algoritması, eğitim setinde sınıflandırılması zor olan örnekler için daha fazla yineleme gerçekleştirmeye ve daha fazla sınıflandırıcı oluşturmaya odaklanır (Rokach, 2010). Bunun yanı sıra, her bir sınıflandırma algoritmasına da ağırlık değerleri atanır. Bu ağırlık değerleri, ilgili sınıflandırıcının doğru sınıflandırma oranını ölçümlemek için kullanılır. Bu nedenle, daha doğru sınıflandıran sınıflandırma algoritmalarının ağırlıkları daha yüksek tutulur. Şekil 3.7'de AdaBoost algoritmasının genel mimarisi, Algoritma 3.2'de AdaBoost algoritmasının eğitim aşaması ve Algoritma 3.3'te AdaBoost algoritmasının genel işleyişi sunulmaktadır.

Algoritma 3.2 AdaBoost algoritmasının eğitim aşaması (Kuncheva, 2014)

Eğitim: $Z=\{z_1, \ldots, z_N\}$ Etiketli Veri Seti

1. Topluluk boyutu (L) ve temel sınıflandırma modelini belirle.
2. $w^1 = [w_1^1, ..., w_N^1],\ w_j^1 \in [0,1],\ \sum_{j=1}^{N} w_j^1 = 1$

   $w_j^1 = \frac{1}{N},\ j = 1, ..., N$
3. $k = 1, ..., L$ için:
   a. w^k dağılışını kullanarak Z veri setinden S_k örneklemini al.
   b. D_k sınıflandırıcısını S_k örneklemi üzerinde eğit.
   c. k adımındaki ağırlıklı topluluk hatasını aşağıdaki eşitliğe göre belirle:

   $$\varepsilon_k = \sum_{j=1}^{N} w_j^k l_k^j$$

   (D_k sınıflandırıcısın z_j'yi yanlış sınıflandırması durumunda $l_k^j = 1$, doğru sınıflandırması durumunda 0 değeri alır.)

   d. Eğer $\varepsilon_k = 0$ ise; $w_j^k = \frac{1}{N}$ olacak şekilde yeniden ağırlık değerleri atanır ve devam edilir. Aksi takdirde, $\varepsilon_k \in (0,0.5)$ olmak üzere ağırlıklar güncellenir:

   $$\beta_k = \frac{\varepsilon_k}{1-\varepsilon_k}$$

   $$w_j^{k+1} = \frac{w_j^k \beta_k^{(1-l_k^j)}}{\sum_{i=1}^{N} w_i^k \beta_k^{(1-l_k^i)}} \quad (j=1, ..., N)$$

4. $D=\{D_1, ..., D_L\}$ ve $\beta_1, ..., \beta_L$ değerlerini döndür.

Algoritma 3.3 AdaBoost algoritmasının genel işleyişi (Kuncheva, 2014)

İşleyiş: Her bir x nesnesi için:

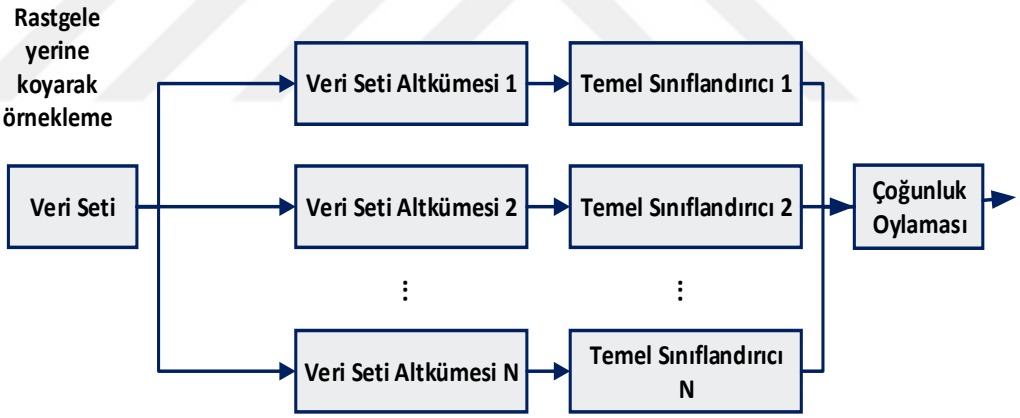
1. Nesneyi, $D=\{D_1, \ldots, D_L\}$ sınıflandırıcılarını kullanarak sınıflandır.
2. ω_t sınıfı için destek değerini aşağıdaki formüle göre belirle:

$$\mu_t(x) = \sum_{D_k(x)=\omega_t} \ln\left(\frac{1}{\beta_k}\right)$$

3. En yüksek destek değerine sahip sınıfı x nesnesi için sınıf etiketi olarak belirle.

3.4.2 Bağımsız sınıflandırıcı toplulukları

Bağımsız sınıflandırıcı topluluklarında, orijinal veri setinden birkaç farklı veri seti oluşturularak, sınıflandırıcılar bu veri setleri üzerinde eğitilir. Orijinal veri setinden elde edilen yeni veri setleri, birbirlerinden tamamen farklı örnekler içerebileceği gibi, birbirleriyle çakışan örnekler içerecek biçimde de oluşturulabilir. Bagging, Dagging ve rastgele alt-uzay (*Random Subspace*) algoritmaları, bağımsız öğrenme yöntemine dayalı algoritmalardır.



Şekil 3.8. Bagging algoritmasının genel mimarisi (Wang et al., 2014)

3.4.2.1 Bagging algoritması

Bagging (*Bootstrap aggregating*) algoritmasında sınıflandırıcı topluluğu, eğitim setinin farklı örneklemleri üzerinde eğitilen sınıflandırıcılar bir araya getirilerek oluşturulur (Breiman, 1996). Sınıflandırıcı topluluğunun etkin sonuçlar verebilmesi için gerekli olan çeşitlilik, her bir sınıflandırıcının eğitiminde farklı eğitim setleri kullanılarak gerçekleştirilmeye çalışılır. Eğitim setleri genellikle orijinal problem dağılımından rastgele olarak oluşturulur. Veri setinden örneklerin seçilmesinde basit rastgele yerine koyarak örnekleme (*simple random sampling*

with replacement) yöntemi kullanılır. Sınıflandırıcıların çıktıları, çoğunluk oylaması ya da ağırlıklı oylama yöntemi kullanılarak birleştirilir.

Bagging yöntemi için önemli noktalardan biri, temel öğrenme algoritması olarak kullanılacak sınıflandırıcının kararsız (*unstable*) bir yapıya sahip olması; yani, farklı örneklemlere dayalı eğitim setleri ile eğitildiğinde, sınıflandırıcı çıktısında kayda değer bir farklılık oluşturmasıdır (Kuncheva, 2014). Aksi takdirde, birbirinin aynısı olan sınıflandırıcıların oluşturduğu bir sınıflandırıcı topluluğu elde edileceğinden, sınıflandırma başarımında iyileştirme elde edilemeyebilir. Şekil 3.8'de Bagging algoritmasının genel mimarisi, Algoritma 3.4'te ise genel işleyişi sunulmaktadır.

Algoritma 3.4 Bagging algoritmasının genel işleyişi (Kuncheva, 2014)

Eğitim: Z={z_1, …, z_N} Etiketli Veri Seti

1. Topluluk boyutu (L) ve temel sınıflandırma modelini belirle.
2. Z veri setinden L adet önyüklemeli örneklem oluştur ve her bir D={D_1, …, D_L} sınıflandırıcısını, oluşturulan örneklemlerden birini kullanarak eğit.

İşleyiş: Her bir x nesnesi için:

1. Nesneyi, D={D_1, …, D_L} sınıflandırıcılarını kullanarak sınıflandır.
2. Herhangi bir D_i sınıflandırıcı tarafından nesneye atanan etiketi, ilgili sınıf olarak kullanılan bir oy olarak kabul ederek, x nesnesini toplamda sınıflandırıcılar tarafından en çok oylanan sınıfa ata.

3.4.2.2 Dagging algoritması

Dagging algoritmasında da Bagging algoritmasında olduğu gibi, eğitim setinden farklı örneklemlerin oluşturulması ve bu örneklemler üzerinde sınıflandırma algoritmalarının eğitimi söz konusudur (Ting and Witten, 1997). L eğitim seti $L=\{(y_n, x_n), n=1,...,N'\}$ için y_n, n. örneğinin sınıf değerini ve x_n, ilgili örneğin öznitelik değer vektörünü temsil etmek üzere, bu eğitim setinden iki farklı yönteme dayalı olarak örneklemler alınabilir. Önyüklemeli örneklem oluşturulduğu takdirde, L eğitim setinden basit rastgele yerine koyarak örnekleme yöntemi ile $N \leq N'$ olmak üzere, K tane N boyutlu alt küme oluşturulur. Ayrık örneklem (*disjoint samples*) oluşturulduğunda ise L eğitim seti yerine yeniden koymadan $KN \leq N'$ olacak şekilde K tane N boyutlu alt kümeye dönüştürülür. Dagging algoritmasında da sınıflandırıcıların çıktıları çoğunluk oylama yöntemi aracılığıyla birleştirilir. Dagging algoritmasında sınıflandırıcıların eğitilmesinde önyüklemeli örneklem yerine; ayrık örneklemler kullanılması özellikle veri setindeki örnek sayısına dayalı olarak kötü zaman performansı gösteren yöntemlerde oldukça kullanışlı olabilmektedir (Witten et al., 2011). Algoritma

3.4'teki genel eğitim ve işleyiş yapısı örneklem oluşturma dışında Dagging algoritmasında da aynı şekilde işlemektedir.

3.4.2.3 Rastgele alt-uzay algoritması

Rastgele alt-uzay (*Random Subspace*) algoritması, Bagging algoritmasında olduğu gibi eğitim setinden örneklemler elde edilerek sınıflandırıcıların eğitildiği bir sınıflandırıcı oluşturma yöntemidir (Ho, 1998). Ancak, rastgele alt-uzay yönteminde, Bagging algoritmasından farklı olarak, eğitim setinin değiştirilmesi, örnek uzayı yerine, öznitelik uzayı üzerinde gerçekleştirilmektedir. S eğitim setindeki her bir örnek X_j p-boyutlu bir vektör ($X_j=(x_{j1}, x_{j2},...,x_{jp})$ olmak üzere, rastgele alt-uzay algoritması ile S eğitim setinden $p^*<p$ olmak üzere, p^* tane öznitelik rastgele olarak seçilir. Böylelikle, p-boyutlu öznitelik uzayından, p^* boyutlu rastgele seçilmiş örnekler elde edilmiş olur. Değiştirilmiş eğitim seti, $S'=(X_1', X_2',...,X_n')$, p^*-boyutlu X_j' ($(X_j'=(x_{j1},x_{j2},...,x_{jp^*})$) örneklerinden oluşmaktadır. Ardından, temel sınıflandırma algoritmaları, aynı boyutlara sahip alt uzaylar S^i ($i=1,2,\ldots,B$) üzerinde eğitilerek, sınıflandırma algoritmaların sonuçları oylamaya dayalı olarak belirlenir. Rastgele alt-uzay yöntemi ile veri boyutuna kıyasla daha küçük eğitim örnekleri elde edildiği için, alt-uzaylar üzerinde eğitilen sınıflandırıcılar daha küçük problemler üzerinde çalışmış olur. Bunun yanı sıra, rastgele alt-uzay yöntemi ile özellikle oldukça fazla gereksiz öznitelik içeren veri setlerinde etkin sonuçlar alınması mümkündür (Panov and Džeroski, 2007). Algoritma 3.5'te rastgele alt-uzay algoritmasının genel işleyişi sunulmaktadır.

Algoritma 3.5 Rastgele alt-uzay algoritması genel işleyişi (Panov and Džeroski, 2007)

Girdi: *S: eğitim seti, B:* alt-uzay sayısı, *p*:* alt-uzay boyutu
Çıktı: *E:* Sınıflandırıcı Topluluğu

$E \leftarrow 0$
$i = 1,..., B$ için:
a. Eğitim seti (S) ve alt-uzay boyutuna (p^*) dayalı olarak rastgele alt-uzayı (S^i) seç.
b. Seçilen rastgele alt-uzayı (S^i) üzerinde C^i sınıflandırıcısını eğit.
c. C^i sınıflandırıcısını, E sınıflandırıcı topluluğuna dahil et.
E sınıflandırıcı topluluğunu döndür.

3.4.3 Sınıflandırıcı topluluğu birleştirme kuralları

Sınıflandırıcı topluluğu birleştirme kuralları, sınıflandırıcı topluluklarının başarımında en önemli bileşenlerden biridir (Moreno-Seco et al., 2006). Temel sınıflandırma algoritmalarının çıktılarının birleştirilmesine yönelik yöntemler,

oylamaya dayalı yöntemler ve üst öğrenme yöntemleri olmak üzere iki sınıf altında incelenmektedir.

3.4.3.1 Oylama yöntemleri

Sınıflandırıcı topluluğu tasarımında önemli noktalardan birisi, birleştirme kuralının belirlenmesidir. Temel sınıflandırma algoritmalarının çıktılarının birleştirilmesinde en sık kullanılan yöntemlerden birisi, oylama (*voting*) yöntemidir. Oylama yöntemi, temel olarak ağırlıksız (*unweighted*) ve ağırlıklı (*weighted*) oylama yöntemleri olmak üzere ikiye ayrılmaktadır. Başlıca ağırlıksız oylama yöntemleri, minimum olasılık (*minimum probability, MIP*), maksimum olasılık (*maximum probability, MAP*), çoğunluk oylaması (*majority voting, MAJ*), olasılıklar çarpımı (*product of probability, PRP*) ve olasılıklar ortalaması (*average of probabilities, AVP*) yöntemleridir. Başlıca ağırlıklı oylama yöntemleri ise, basit ağırlıklı oylama (*simple weighted voting, SWV*), ölçeklenmiş ağırlıklı oylama (*re-scaled weighted voting, RWV*), en iyi-en kötü ağırlıklı oylama (*best-worst weighted voting, BWWV*) ve ikinci dereceden en iyi-en kötü ağırlıklı (*quadratic best-worst weighted voting, QBWWV*) oylamadır (Moreno-Seco et al., 2006).

Temel sınıflandırıcıların birleştirilmesinde en çok kullanılan yöntem çoğunluk oylamasıdır. Burada, *k* tane temel sınıflandırma algoritmasının çıktıları çoğunluk oylamasına tabi tutularak en yüksek oyu alan çıktının, sınıflandırıcı topluluğunun çıktısı olarak belirlenmesi söz konusudur. Çoğunluk oylamasının yanı sıra, başka ağırlıksız oylama yöntemleri de bulunmaktadır. LC_i (X), hesaplanan en yüksek olasılık değerini temsil etmek üzere, sınıflandırıcı topluluğunun çıktısı Eşitlik 3.31'e göre belirlenmektedir (Dehzangi and Karamizadeh, 2011):

$$H(X) = \arg_{i=1,\ldots n} \max(LC_i(X)) \tag{3.31}$$

Burada, *n*, sınıf sayısını temsil etmektedir. Bu gösterime dayalı olarak olasılıklar ortalaması, olasılıklar çarpımı, maksimum olasılık, minimum olasılık ve çoğunluk oylaması birleştirme kurallarına ilişkin eşitlikler sırasıyla Eşitlik 3.32-3.36'da verilmiştir (Dehzangi and Karamizadeh, 2011):

$$LC_i(X) = \frac{1}{m}\sum_{j=1}^{m} P_j(w_i|x) \tag{3.32}$$

$$LC_i(X) = \frac{1}{m}\prod_{j=1}^{m} P_j(w_i|x) \tag{3.33}$$

$$LC_i(X) = \max_{j=1,\ldots m} \left\{ P_j(w_i|x) \right\} \tag{3.34}$$

$$LC_i(X) = \min_{j=1,\ldots m} \left\{ P_j(w_i|x) \right\} \tag{3.35}$$

$$H(X) = \arg_{i=1,\ldots m} \max \left\{ S_i = \sum_{j=1}^{m} I(h_j(X) = Y) \right\} \tag{3.36}$$

Burada, *m*, sınıflandırıcı topluluğunda yer alan temel sınıflandırma algoritması sayısını temsil etmektedir.

Basit ağırlıklı oylama (*simple weighted voting*) yönteminde, her bir sınıflandırıcının ağırlık değerleri (w_k), ilgili sınıflandırıcının eğitim setindeki performansına dayalı olarak Eşitlik 3.37'ye göre belirlenmektedir:

$$w_k = \frac{a_k}{\sum_l a_l} \tag{3.37}$$

Burada, a_k, eğitim setinde *k* temel öğrenme algoritması ile elde edilen doğru sınıflandırma oranını, *l*, sınıflandırıcı topluluğunda yer alan toplam sınıflandırıcı (temel öğrenme algoritması) sayısını ve a_l, belirli bir *l* temel öğrenme algoritması ile elde edilen doğru sınıflandırma oranını temsil etmektedir.

Ölçeklenmiş ağırlıklı oylama (*re-scaled weighted voting*) yönteminde, eğitim setinde *N/M* ya da daha az doğru sınıflandırma oranına sahip sınıflandırıcılara ağırlık değeri olarak sıfır atanmaktadır. Böylelikle, belirli bir eşik değerin altında başarıma sahip sınıflandırıcıların sınıflandırıcı topluluğunun çıktısında etkili olması engellenmektedir. Bu yöntemde her bir sınıflandırıcının ağırlık değeri Eşitlik 3.37'ye göre ölçeklenmektedir. Burada, e_k, belirli bir D_k sınıflandırıcısının hatalı tahmin sayısını, *N*, örnek sayısını ve *M*, sınıf sayısını temsil etmek üzere Eşitlik 3.38'e göre hesaplanmaktadır (Moreno-Seco et al., 2006):

$$a_k = \max \left\{ 0{,}1 - \frac{M.e_k}{N(M-1)} \right\} \tag{3.38}$$

En iyi-en kötü ağırlıklı oylama (*best-worst weighted voting*) yönteminde, sınıflandırıcı topluluğundaki en iyi ve en kötü performansa sahip sınıflandırıcılar belirlenir. En iyi sınıflandırıcının ağırlık değerine bir değeri atanırken, en kötü sınıflandırıcının ağırlık değeri sıfır olarak verilir. Diğer sınıflandırıcıların ağırlık değerleri Eşitlik 3.39'a göre belirlenir:

$$a_k = 1 - \frac{e_k - e_B}{e_W - e_B} \tag{3.39}$$

Burada, e_B, tüm sınıflandırıcılar içerisindeki en düşük hata sayısını, e_w, tüm sınıflandırıcılar içerisindeki en yüksek hata sayısını belirtmektedir (Moreno-Seco et al., 2006).

İkinci dereceden en iyi-en kötü ağırlıklı oylama (*quadratic best-worst weighted vote*) yönteminde, e_B, tüm sınıflandırıcılar içerisindeki en düşük hata sayısını, e_w, tüm sınıflandırıcılar içerisindeki en yüksek hata sayısını belirtmek üzere ağırlık değerleri Eşitlik 3.40'a göre hesaplanmaktadır:

$$a_k = \left(\frac{e_W - e_k}{e_W - e_B}\right)^2 \tag{3.40}$$

3.4.3.2 Yığılmış genelleme algoritması

Yığılmış genelleme algoritması (*Stacked generalization, Stacking*), en bilinen üst öğrenmeye dayalı (*meta learning*) sınıflandırıcı topluluğu yöntemleri arasındadır (Wolpert, 1992). Bagging, rastgele alt-uzay, AdaBoost gibi sınıflandırıcı topluluğu yöntemlerinde, aynı temel öğrenme algoritmaları birleştirilerek, sınıflandırma modeli oluşturulurken, yığılmış genelleme algoritmasında, farklı temel öğrenme algoritmaları bir araya getirilmektedir. Algoritma 3.6'da yığılmış genelleme algoritmasının genel işleyişi özetlenmiştir.

Algoritma 3.6 Yığılmış genelleme algoritması genel işleyişi (Zhou, 2012)

Girdi: - D veri seti $D=\{(x_1, y_1), (x_2, y_2),\ldots,(x_m,y_m)\}$;
- Birinci seviye temel öğrenme algoritmaları $L_1, \ldots, L_T$.
- Üst-seviye öğrenme algoritması L.

Süreç:
for $t=1,\ldots,T$:
  $h_t=L_t(D)$ *% Birinci seviye temel öğrenme algoritması L_t'nin D veri seti üzerinde uygulanması*
end;
  D'= Ø; % Yeni (üst-seviye) veri setinin oluşturulması.
for $i=1,\ldots,m$:
  for $i=1,\ldots,T$:
    $z_{it}=h_t(x_i)$ *% x_i eğitim örneğinin h_t ile sınıflandırılması.*
  end;
  $D'=D' \cup \{((z_{i1},z_{i2},\ldots,z_{iT}),y_i)\}$
end;
h'=L(D') % İkinci seviye öğrenme algoritması h', L'nin D' üzerinde eğitilmesi ile elde edilir.

Çıktı: $H(x)=h'(h_1(x),\ldots,h_T(x))$

Farklı temel öğrenme algoritmalarından sınıflandırıcı topluluğu elde edilmesinde, iki seviyeli bir yapı benimsenmektedir. Bu yapıda, temel sınıflandırma algoritmaları, üst seviye öğrenme algoritması aracılığı ile birleştirilmektedir. Öncelikle, eğitim seti üzerinde temel öğrenme algoritmaları eğitilerek, sınıflandırıcılara ilişkin tahminler elde edilmektedir. Temel öğrenme algoritmalarının çıktılarına dayalı olarak, başlangıçtaki veri seti ile aynı sınıflara sahip bir üst seviye veri seti oluşturulmaktadır. Üst seviye veri setinin oluşturulmasında, *k*-kat çapraz geçerleme yöntemine benzer bir süreç izlenmektedir (Kuncheva, 2014). Üst seviye örneklemi, her bir temel öğrenme algoritmasının çıktıları ile her bir örneğe ilişkin doğru sınıf etiketleri birleştirilerek oluşturulmaktadır. Ardından, üst seviye öğrenme algoritması, yeni oluşturulan veri seti üzerinde eğitilerek, sınıflandırıcı topluluğunun çıktısı belirlenmektedir.

3.5 Sınıflandırıcı Topluluğu Budama

Sınıflandırıcı toplulukları (*classifier ensembles, ensemble of classifiers*), sınıflandırma işleminin, tek bir öğrenme algoritması yerine birden fazla öğrenme algoritmasının eğitilmesi ile elde edilen sonuçların birleştirilmesine dayalı olarak yapılmasına yönelik bir makine öğrenmesi araştırma alanıdır (Dietterich, 2000). Topluluk öğrenmesi (*ensemble learning*), topluluğu oluşturan her bir modelin, belirli bir probleme ilişkin öğrenme sürecine tabi tutulduğu ve toplulukta yer alan modellerin, son çıktıyı elde etmek üzere bir araya getirildiği bir yöntemdir (Mendes-Moreira et al., 2012). Topluluk öğrenmesi yaklaşımı, sınıflandırma ve regresyon gibi öğreticili öğrenme yöntemlerinin yanı sıra, kümeleme analizi gibi öğreticisiz öğrenme yöntemlerinde de uygulanabilmektedir (Strehl and Ghosh, 2003).

Topluluk öğrenmesi süreci, topluluk oluşturma (*ensemble generation*), topluluk budama (*ensemble pruning*) ve topluluk birleştirme (*ensemble integration*) olmak üzere üç temel aşamadan oluşmaktadır (Roli et al., 2001; Mendes-Moreira et al., 2012). Topluluk öğrenmesi sürecinin topluluk oluşturma aşamasında, sınıflandırıcı topluluğunda yer alacak öğrenme algoritmaları oluşturulmaktadır. Öğrenme algoritmalarının oluşturulması, homojen ve heterojen olmak üzere iki farklı şekilde gerçekleştirilir. Homojen sınıflandırıcı topluluklarında, aynı öğrenme algoritması, parametre değerlerinin farklı alınması, öğrenme sürecinin rastsallaştırılması, eğitim setlerinin farklılaştırılması, girdi özniteliklerinin değiştirilmesi gibi çeşitli yöntemler kullanılarak farklılaştırılmaktadır (Tsoumakas et al., 2008). Daha önce değinilen, Bagging ve

destekleme (*Boosting*) algoritmaları, homojen sınıflandırıcı topluluğu yaklaşımlarıdır. Heterojen sınıflandırıcı topluluklarında ise aynı veri seti üzerinde farklı öğrenme algoritmalarının eğitilmesi sonucu sınıflandırma modeli elde edilmektedir. Böylelikle, farklı öğrenme algoritmalarının güçlü yönleri ile zayıf yönlerini telafi eden bir sınıflandırma modeli oluşturulması amaçlanmaktadır. Heterojen sınıflandırıcı toplulukları, farklı öğrenme algoritmaları kullanılarak oluşturulduğundan, sınıflandırıcı topluluğunun çeşitliliğinin (*diversity*) yüksek olması beklenmektedir. Sınıflandırıcı toplulukları tasarımında, topluğun çeşitliliğinin yüksek tutulması ile sınıflandırma başarımının artırılması amaçlanmaktadır (Gashler et al., 2008). Topluluk öğrenmesi sürecinin topluluk birleştirme aşamasında, temel öğrenme algoritmaları belirli bir sınıflandırıcı topluluğu birleştirme kuralı kullanılarak birleştirilmektedir.

Sınıflandırıcı topluluğu budama (*ensemble pruning, selective ensemble, ensemble thinning, ensemble selection*), sınıflandırıcı topluluğunda yer alan öğrenme algoritmalarından bir bölümünün, topluluğun sınıflandırma başarımını artırmak ve hesaplama maliyetini azaltmak amacıyla budanması ile daha etkin bir sınıflandırıcı topluluğu elde edilmesine yönelik bir süreçtir (Zhou et al., 2002). Topluluk budama yöntemleri, üstel (*exponential*), rastsal (*randomized*), ardışık (*sequential*), sıralı (*ranked*) ve kümelemeye dayalı yöntemler olmak üzere beş grupta incelenmektedir (Mendes-Moreira et al., 2012).

Üstel topluluk budama algoritmalarında, K tane öğrenme algoritması içeren bir topluluktan k tane öğrenme algoritması içeren bir uygun bir altküme seçilmesi için olası 2^K-1 altküme dikkate alınır. Bu nedenle, üstel yöntemlerde, arama uzayı oldukça büyüktür. Uygun altkümenin belirlenmesi, NP-tam (*NP-complete*) bir problem olduğundan, oldukça yüksek hesaplama maliyetine sahiptir ve yalnızca az sayıda öğrenme algoritması içeren topluluklar için uygundur (Tamon and Xiang, 2000; Martinez-Munoz and Suarez, 2006).

Rastsal topluluk budama algoritmalarında, sınıflandırıcı topluluğundan, uygun öğrenme algoritmalarını içeren bir altküme seçilmesine yönelik arama uzayının daha etkin bir biçimde incelenebilmesi için metasezgisel yöntemler kullanılır. Genetik algoritmalar, tabu arama algoritması, toplum tabanlı artırımlı öğrenme, stokastik arama gibi metasezgisel yöntemler sınıflandırıcı topluluğu budaması için kullanılan yöntemler arasındadır (Ruta and Gabrys, 2001; Zhou and Tang, 2003; Partalas et al., 2006).

Ardışık topluluk budama algoritmalarında, arama süreci mevcut çözümün yeni öğrenme algoritmaları eklenerek ya da mevcut öğrenme algoritmaları çıkarılarak gerçekleştirilir. İleriye doğru yapılan ardışık topluluk budamada, arama sürecine hiçbir öğrenme algoritması içermeyen boş sınıflandırıcı topluluğu ile başlanır ve her bir yinelemede sınıflandırma modeline yeni öğrenme algoritmaları eklenir. Geriye doğru yapılan ardışık topluluk budamada ise sınıflandırıcı topluluğunda yer alan tüm öğrenme algoritmalarını içeren model ile başlanır ve her bir yinelemede öğrenme algoritmaları, sınıflandırıcı topluluğundan kaldırılır. İleri-geri yönlü budamada ise, arama sürecinde hem yeni öğrenme algoritmalarının eklenmesi hem de çıkarılması söz konusudur (Mendes-Moreira et al., 2012).

Sıralı topluluk budama algoritmalarında, sınıflandırıcı topluluğunda yer alan öğrenme algoritmaları, belirli bir ölçüt kullanılarak sıralanır. Ardından, sıralı listenin en yüksek ölçüt değerine sahip *k* tane öğrenme algoritması sınıflandırıcı topluluğunda tutulurken, geride kalan öğrenme algoritmaları budanır. Sıralı topluluk budama algoritmalarında, yalnızca ölçüt değerine dayalı olarak karar verildiğinden, bu yöntemler hesaplama etkinliği bakımından etkin algoritmalardır. Ancak, doğru sınıflandırma başarımı bakımından yüksek başarım vermemektedir (Pinto, 2013).

Kümelemeye dayalı budama yöntemlerinde ise belirli bir kümeleme algoritması kullanılarak (*k*-ortalama vb.), sınıflandırıcı topluluğunda yer alan öğrenme algoritmaları, sınıflandırmada gösterdikleri performansa göre, kümelere atanmaktadır. Ardından, her bir kümeden birer öğrenme algoritması seçilerek, budanmış sınıflandırıcı topluluğu oluşturulmaktadır (Mendes-Moreira et al., 2012).

3.5.1 Sınıflandırıcı topluluğu budama algoritmaları

Bu bölümde, tez çalışması kapsamında geliştirilen sınıflandırıcı topluluğu budama algoritmasının başarımının sınanması için kullanılan, temel sınıflandırıcı topluluğu budama algoritmaları tanıtılmaktadır.

3.5.1.1 Model kütüphanesinden topluluk seçimi

Model kütüphanesinden topluluk seçimi algoritması (*ensemble selection from libraries of models, ESM*), en temel sınıflandırıcı topluluğu budama

algoritmalarından biridir (Caruana et al., 2004). Burada, temel öğrenme algoritmalarından oluşan bir kütüphaneden belirli ölçütlere ve arama algoritmalarına göre, uygun öğrenme algoritmalarını içeren bir altküme seçilmesi amaçlanır. Öncelikle, fazla ve farklı makine öğrenmesi algoritmaları kullanılarak temel sınıflandırma modelleri oluşturulur. Ardından, ileriye doğru arama algoritması belirli bir ölçüt fonksiyonu ile bir arada kullanılarak, sınıflandırıcı altkümesi elde edilmektedir. Caruana et al. (2004) tarafından gerçekleştirilen deneysel çalışmalarda, destek vektör makineleri, yapay sinir ağları, k-en yakın komşu algoritması, karar ağaçları gibi birçok farklı makine öğrenmesi yaklaşımına dayalı sınıflandırma algoritmaları, farklı parametre değerleri ile farklılaştırılarak model kütüphanesi oluşturulmuştur. Ardından, ileriye doğru arama kullanılarak etkin altküme elde edilmiştir.

ESM algoritması, birçok farklı değerlendirme ölçütü ile bir arada kullanılabilecek basit bir yapıya sahiptir. Temel öğrenme algoritmalarının değerlendirilmesinde ise doğru sınıflandırma oranı, karesel ortalama hata (*root mean squared error*), ortalama çapraz entropi (*mean cross entropy*), geri çağırma (*recall*), duyarlılık (*precision*), F-ölçütü (*F-score*), ROC alanı (*ROC area*) gibi farklı değerlendirme ölçütleri kullanılmıştır.

Algoritma 3.7 Model kütüphanesinden topluluk seçimi (Sun and Pfahringer, 2011)

Girdiler: A, L, F, E, r
A, model kütüphanesini oluşturmak üzere kullanılan temel öğrenme algoritmaları
L, model kütüphanesi
F, eğitim seti
E, sınıflandırıcı topluluğu
r, geçerleme seti oranı

1. T←Basit_Rastgele_Örnekleme(F,r) // T setinin basit rastgele örnekleme ile elde edilmesi
2. H←F-T // H, geçerleme setinin elde edilmesi
3. E←boş sınıflandırıcı topluluğu
4. L←Model_Kütüphanesi_Oluştur(A,T) //Temel öğrenme algoritmaları ve T seti kullanılarak model kütüphanesinin elde edilmesi
5. Temel sınıflandırma algoritmalarına ağırlık değerleri atanması (Sınıflandırma algoritmalarının geçerleme setindeki başarımlarına göre, ileriye doğru arama ile)
6. E←L'deki sıfırdan büyük ağırlık değerine sahip temel sınıflandırma algoritmaları
7. E'yi döndür.

ESM algoritması, hiçbir öğrenme algoritması içermeyen sınıflandırıcı topluluğu ile başlar. Ardından, model kütüphanesinde yer alan öğrenme algoritmalarından biri, geçerleme setindeki başarımına dayalı olarak sınıflandırıcı topluluğuna dâhil edilir. Model kütüphanesinde yer alan öğrenme algoritmalarının sınıflandırıcı topluluğuna eklenmesi işlemi belirli bir yineleme boyunca ya da kütüphanede yer alan tüm algoritmalar incelenineye dek sürdürülür. Yinelemeli

süreç tamamlandığında, sınıflandırıcı topluluğunda yer alacak öğrenme algoritmaları belirlenmiş olur (Caruana et al., 2004). Algoritma 3.7'de model kütüphanesinden topluluk seçimi algoritmasının genel işleyişi özetlenmiştir.

3.5.1.2 Bagging sınıflandırıcı seçimi

Bagging sınıflandırıcı seçimi algoritması (*Bagging ensemble selection, BES*), Bagging ve model kütüphanesinden topluluk seçimi (ESM) algoritmalarına dayalı bir sınıflandırıcı topluluğu budama yöntemidir (Sun, 2014). ESM algoritması, birçok temel sınıflandırıcı topluluğu yöntemine kıyasla daha yüksek sınıflandırma başarımı elde etmektedir (Caruana et al., 2004). Ancak, bazı durumlarda, ESM algoritmasının geçerleme setine aşırı uygunluk göstermesi (*overfitting*), oluşturulan sınıflandırıcı topluluğunun sınıflandırma başarımının düşmesine yol açmaktadır. ESM algoritması ile yapılan deneysel çalışmalarda geçerleme ve test setlerindeki sınıflandırma başarımları incelendiğinde, model kütüphanesindeki temel öğrenme algoritması sayısı arttıkça geçerleme setindeki başarımın arttığı gözlenmiştir. Buna karşın, benzer artış örüntüsü, belirli bir eşik değerinden sonra, test setinde alınamamaktadır (Sun, 2014). Temel ESM algoritmasının aşırı uygunluk problemini ortadan kaldırmak amacıyla, Caruana et al. (2004) tarafından, ESM algoritmasında üç temel değişiklik önerisinde bulunulmuştur. Bunlardan birincisi, sınıflandırıcıların basit rastgele seçim aracılığıyla seçilmesidir. Basit rastgele seçim ile temel sınıflandırma algoritmalarının birden çok kez seçilmesi olanaklı hale getirilmektedir. İkinci öneri, ESM algoritmasına hiçbir öğrenme algoritması içermeyen sınıflandırıcı topluluğu ile başlamak yerine, model kütüphanesinde yer alan temel öğrenme algoritmalarının geçerleme seti üzerindeki başarımlarına göre sıralanması ve en yüksek başarıma sahip N tane algoritmayı içeren sınıflandırıcı topluluğu ile sürece başlanmasıdır. Üçünü öneri ise sınıflandırıcı seçiminin gruplandırılmış temel öğrenme algoritmaları ile yapılmasıdır. Burada, model kütüphanesinde yer alan temel öğrenme algoritmaları K tane gruba rastgele atanarak, seçim işlemi her bir grup içerisinde sürdürülür. ESM algoritması genel yapısı itibariyle basit bir yapıya sahiptir. Ancak, algoritmanın aşırı uygunluk problemini ortadan kaldırmak amacıyla getirilen öneriler, algoritmayı daha parametrik bir duruma getirmektedir. Bunun yanı sıra, ESM algoritmasının performansı önemli ölçüde geçerleme seti olarak kullanılacak setin büyüklüğüne bağlıdır. Uygun geçerleme setinin belirlenmesi, algoritmaya ek bir yük getirmektedir (Sun, 2014).

Bagging algoritması, temel öğrenme algoritmalarının kararsız (*unstable*) olması durumunda, sınıflandırıcı çıktısında kayda değer bir farklılık oluşturularak, etkin bir sınıflandırma başarımı elde edilmesini amaçlayan temel bir sınıflandırıcı topluluğu yöntemidir (Breiman, 1996; Kuncheva, 2014). Sun (2014) çalışmasında, ESM algoritmasının kararsız bir temel öğrenme algoritması olarak ele alınması durumunda, Bagging yöntemi ile bir arada kullanılması ile algoritmanın daha önce değinilen sorunlarının ortadan kaldırılarak, sınıflandırma başarımının iyileştirileceğini savunmuştur. BES algoritmasında, grup dışında kalan örnekler geçerleme seti olarak kullanılarak, geçerleme seti oranı belirlenmesi gereksinimi ortadan kaldırılmıştır. BES algoritmasının temel işleyişi, Algoritma 3.8'de özetlenmiştir.

Algoritma 3.8 Bagging sınıflandırıcı seçimi algoritması (Sun, 2011)

Girdiler: S, E, T
S, eğitim seti
E, sınıflandırıcı seçimi algoritması
T, Bootstrap örnekleme sayısı

for i=1←T **do**
    S_b←*S*'den Bootstrap örnekleme yöntemi ile yerine yeniden koyarak örneklem elde et
    S_{oob}←Örneklem dışı kalan örnekler
    E'de yer alan temel öğrenme algoritmalarını S_b kullanarak eğit
    E_i←Temel öğrenme algoritmalarının S_{oob} setindeki performansına göre seçim yap
    U=U∪ E_i
endfor
U'yu döndür.

3.5.1.3 LibD3C algoritması

LibD3C algoritması, k-ortalama (*k-means*) kümeleme algoritmasına, dinamik seçime ve ardışık aramaya dayalı melez bir topluluk budama yöntemidir (Lin et al., 2014). Algoritmada, öncelikle k-ortalama algoritması kullanılarak model kütüphanesinde yer alan temel öğrenme algoritmalarının bir bölümü budanarak, aday sınıflandırıcı kümesi elde edilir. Ardından, aday sınıflandırma algoritmaları, ileriye ya da geriye doğru yapılan topluluk budaması ile incelenir. Arama sürecinin tamamlanması ile elde edilen sınıflandırıcı kümesi, çoğunluk oylaması birleştirme kuralı ile birleştirilerek sınıflandırıcı topluluğunun çıktısı oluşturulur. Ω={1,…,*c*} sınıf etiketlerini, $x^{\rightarrow}\epsilon R^n$, Ω'de etiketli *n* özniteliği, $S = \{(x_1^{\rightarrow}, \omega_1), .., (x_m^{\rightarrow}, \omega_m)\}$ eğitim setini ve $h_i(x_j^{\rightarrow}, \omega_j)$, h_i sınıflandırıcısının $(x_j^{\rightarrow}, \omega_j) \in S$ örneği için oluşturduğu çıktıyı temsil etmek üzere, sınıflandırıcının ilgili örneği doğru sınıflandırması durumunda $h_i(x_j^{\rightarrow}, \omega_j)$ bir aksi takdirde sıfır değerini alır. Belirli bir h_i sınıflandırıcısının *S* eğitim setindeki tüm örnekler üzerindeki performansı bir $y_i^{\rightarrow}$ vektörü ile temsil edilir. Ardından, k-ortalama

algoritması, sınıflandırma algoritmalarını benzer başarım gösteren sınıflandırıcılar aynı gruplarda yer alacak şekilde altkümelere ayırmak üzere uygulanır. $H = \{h_1, \dots, h_T\}$ sınıflandırıcıları, k tane kümeye, $\sum_{i=1}^{T} min_j D(h_i, h_j)$ koşulunu sağlayan k tane nokta belirlenerek atanır. K-ortalama algoritması, küme sayısı (k) parametresinin kullanıcı tarafından girilmesini gerektirmektedir.

LibD3C algoritmasında, k sayısı $\sum_{i=1}^{T} min_j D(h_i, h_j)$ için elde edilen değer kötüleşinceye kadar artırılarak, en uygun küme sayısı belirlenir. Kümeleme algoritması ile algoritmaların başarımlarına göre gruplara atanmasının ardından, ardışık arama aracılığıyla aday sınıflandırıcılar değerlendirilir. Ardışık arama aşamasında, ileriye doğru ya da geriye doğru arama gerçekleştirilebilir. Arama sürecine hiçbir sınıflandırma algoritması içermeyen boş sınıflandırıcı topluluğu ile ya da model kütüphanesinde yer alan tüm sınıflandırma algoritmalarını içeren set ile başlanır. Öncelikle, sınıflandırma algoritmaları kümeleme aşamasındaki performanslarına dayalı olarak sıralanır. Ardından, ileri doğru arama ile her bir altkümenin en yüksek başarım gösteren sınıflandırıcısı sınıflandırıcı topluluğuna eklenirken, en kötü başarıma sahip sınıflandırıcı budanır. Bu süreç, yinelemeli olarak sürdürülür. Arama sürecinde değerlendirme ölçütü olarak hangi ölçütün kullanılacağı oldukça önemlidir. LibD3C algoritmasında değerlendirme ölçütü olarak, değerlendiriciler arası uyuşma (*inter-rater agreement*) ve çoğunluk oylaması hatası (*majority voting error*) kullanılmıştır.

3.6 Ortak Kümeleme

Kümeleme analizi (*cluster analysis*), veri setinde yer alan nesnelerin, kendileri ile aynı küme içerisindeki nesneler ile olabildiğince benzer ve diğer kümelerde yer alan nesneler ile olabildiğince farklı olacak biçimde gruplara (kümelere) atanmasına yönelik, öğreticisiz (*unsupervised*) bir öğrenme biçimidir (Tan et al., 2005). Küme toplulukları (*cluster ensembles*), sınıflandırıcı ya da regresyon topluluklarında olduğu gibi, temel algoritmaların birleştirilmesi ile daha yüksek başarıma sahip bir topluluk elde edilmesi amaçlanır. Ortak kümeleme (*consensus clustering*), kümeleme işleminin tek bir kümeleme algoritması sonucu elde edilen çıktı yerine küme toplulukları aracılığıyla yapılması ile daha kararlı ve gürbüz kümeleme sonuçları elde edilmesine yönelik bir araştırma alanıdır (Ghosh and Acharya, 2011). Ortak kümeleme kullanmanın başlıca gerekçeleri şu şekilde sıralanabilir (Vega-Pons and Ruiz-Shulcloper, 2011; Ghosh and Acharya, 2011):

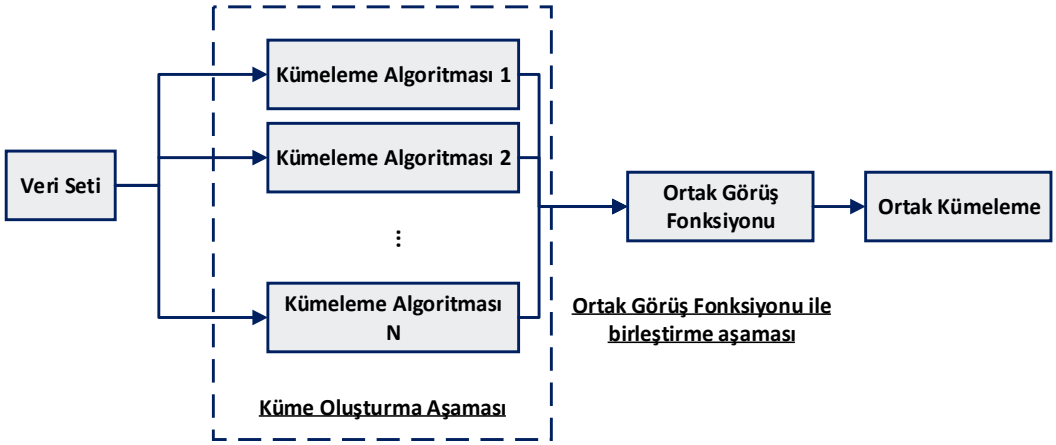
- Kümeleme algoritması sonucu elde edilen çıktının kalitesini iyileştirmesi,

- Temel kümeleme algoritmalarının güçsüz yönlerini ortadan kaldırarak daha gürbüz bir kümeleme yöntemi elde etmesi,

- Tek bir kümeleme algoritması ile çözüm getirilemeyen kümeleme problemlerine uygun çözümler üretebilmesi,

- Gürültülü ve aykırı değerler içeren veri setlerine daha az duyarlı olması.

Ortak kümeleme, küme oluşturma (*cluster generation*) ve ortak görüş fonksiyonu (*consensus function*) olmak üzere iki temel aşamadan oluşur. Küme oluşturma aşamasında, farklı kümeleme algoritmaları kullanılması, farklı küme başlangıç noktaları ya da parametre değerleri alınması, veri setinden farklı eğitim seti altkümeleri elde edilmesi, veri setinin farklı öznitelik setleri üzerinde çalışılması gibi yollarla çeşitlilik sağlanır. Ardından, ortak görüş fonksiyonu kullanılarak, küme oluşturma aşamasında elde edilen bölümlemeler birleştirilerek küme topluluğunun çıktısı elde edilir (Ghaemi et al., 2009). Şekil 3.9'da ortak kümelemenin genel mimarisi sunulmuştur.



Şekil 3.9. Ortak kümelemenin genel mimarisi (Vega-Pons and Ruiz-Shulcloper, 2011)

Küme topluluklarının başarımlarındaki en önemli etkenlerden biri, hangi ortak görüş fonksiyonunun (*consensus function*) kullanılacağının belirlenmesidir. Belirlenen ortak görüş fonksiyonunun, temel kümeleme algoritması olarak kullanılan yöntemler sonucu elde edilen farklı küme bölümlemelerini etkin bir biçimde birleştirmesi amaçlanmaktadır. Ortak görüş fonksiyonları, medyan

bölümleme temelli yaklaşımlar (*median partition*) ve nesnelerin birlikte görülmesine dayalı yaklaşımlar (*object co-occurence*) olmak üzere iki temel sınıf altında incelenmektedir (Vega-Pons and Ruiz-Shulcloper, 2011). Medyan bölümleme temelli yöntemlerde küme topluluğundan elde edilen son bölümleme medyan bölümleme problemi olarak ele alınır. Burada, küme topluluğunda yer alan temel kümeleme algoritmaları ile elde edilen N bölümleme ile arasındaki benzerliği en çok yapan bir son bölümleme belirlenmesi amaçlanır. Medyan bölümleme temelli yöntemlerde, herhangi iki bölümleme arasındaki benzerliği ölçümlemek amacıyla, normalize edilmiş karşılıklı bilgi (*normalized mutual information*), fayda fonksiyonu (*utility function*), Fowlkes-Mallows indeksi ve saflık (*purity*) gibi ölçütler kullanılmaktadır (Strehl and Ghosh, 2002; Mirkin, 2001; Fowlkes and Mallows, 1983).

Nesnelerin birlikte görülmesine dayalı yaklaşımlarda ise, her bir nesnenin küme topluluğunun son bölümlemesinde hangi küme etiketini taşıyacağının belirlenmesi amaçlanır. Bu doğrultuda, belirli bir nesnenin herhangi bir kümede kaç kez görüldüğü ya da iki nesnenin aynı kümede kaç kez birlikte görüldüğü dikkate alınır. Ardından, küme topluluğunun son bölümlemesi oylama aracılığıyla belirlenir. Yeniden etiketleme ve oylama (*relabelling and voting*), birlikte görülme matrisi (*co-association matrix*) ve çizge tabanlı (*graph based*) yöntemler, nesnelerin birlikte görülmesine dayalı yaklaşımlardır (Vega-Pons and Ruiz-Shulcloper, 2011). Yeniden etiketleme ve oylama yönteminde, küme topluluğunda yer alan temel kümeleme algoritmaları ile elde edilen farklı bölümlemeler üzerinde öncelikle ortak bir küme etiketlemesi yapılır. Ardından, oylama ile küme topluluğunun son bölümlemesi elde edilir. Birlikte görülme matrisine dayalı yöntemlerde, öncelikle küme topluluğunda yer alan temel kümeleme algoritmaları ile elde edilen bölümlemelere dayalı olarak birlikte görülme matrisi oluşturulur. Ardından, birlikte görülme matrisi üzerinde benzerlik tabanlı bir kümeleme algoritması uygulanarak, küme topluluğunun son bölümlemesi elde edilir. Çizge tabanlı yöntemlerde ise farklı kümeleme algoritmaları sonucu elde edilen bölümlemelere dayalı olarak ağırlıklı bir çizge oluşturulur. Ardından, çizge kesim noktasını (*graph cut*) en aza indirgeyen bölümleme, küme topluluğunun son bölümlemesi olarak alınır (Vega-Pons and Ruiz-Shulcloper, 2011).

3.6.1 Kümeleme algoritmaları

Küme toplulukları oluşturulabilmesi için farklı kümeleme algoritmalarının temel kümeleme algoritmaları olarak kullanılması ve bu algoritmalar ile elde edilen bölümlemelerin birleştirilmesi gerekmektedir. Bu bölümde, geliştirilen ortak kümeleme ve metasezgisel aramaya dayalı sınıflandırıcı topluluğu budama yönteminde temel kümeleme algoritması olarak kullanılan yöntemlere değinilmektedir.

K-ortalama (*K-means*) algoritması, basit yapısı, gerçekleştiriminin kolay olması ve etkin sonuçlar vermesi nedeniyle, kümelemede en çok kullanılan bölümleyici kümeleme yöntemlerindendir (Jain, 2010). K-ortalama algoritmasında, küme sayısının girdi parametresi olarak girilmesi gerekmektedir. Algoritma, küme merkezini temsil etmek üzere *k* tane noktanın rastgele seçimi ile başlar. Ardından veri setinde yer alan nesneler, nesne ve küme merkezi arasındaki benzerliğe dayalı olarak, kendilerine en yakın kümelere atanır. Küme merkezleri, her bir küme için nesnelerin ortalama değerleri hesaplanarak güncellenir. Süreç, küme merkezleri değişmeyinceye dek sürdürülür. K-ortalama algoritması, birbirlerinden iyi ayrılmış verilerin bölümlenmesinde etkindir. Algoritma, *n*, veri setinde yer alan nesne sayısını, *k*, küme sayısını ve *t*, yineleme sayısını temsil etmek üzere, $O(tkn)$ hesaplama karmaşıklığına sahiptir (Han and Kamber, 2006). Algoritma, genellikle yerel minimuma takılmaktadır (Selim and Ismail, 1984). Bunun yanı sıra, gürültülü ve aykırı değerlere duyarlı bir algoritmadır (Theodoridis and Koutroumbas, 1999).

K-ortalama algoritması ile elde edilen kümelemenin başarımı önemli ölçüde başlangıçta rastgele seçilen küme merkezlerine bağlıdır. K-ortalama++ (*K-means++*) algoritmasında başlangıç küme merkezleri, sezgisel bir fonksiyon aracılığıyla belirlenerek kümeleme başarımının iyileştirilmesi amaçlanır. K-means++ algoritmasında öncelikle veri noktaları arasından biri rastgele küme merkezi olarak seçilir. Ardından, her bir veri noktası ile en yakın mevcut küme merkezi arasındaki uzaklık belirlenir. Yeni küme merkezi, ağırlıklı bir olasılık dağılımı kullanılarak, veri noktasının küme merkezine uzaklığının karesine orantılı olacak şekilde hesaplanır (Arthur and Vassilvitskii, 2007). Süreç, *k* tane küme merkezinin seçimi tamamlanıncaya dek sürdürülür. Küme merkezlerinin seçiminin tamamlanmasının ardından, K-ortalama algoritmasının adımları işletilir.

Beklenti maksimizasyonu algoritması (*expectation maximization, EM*), veri nesnelerinin kümelere atanması işlemini olasılık yöntemlerine göre yapan iki aşamalı, yinelemeli bir algoritmadır (Dempster et al., 1977). Beklenti maksimizasyonu algoritması, beklenti ve maksimizasyon olmak üzere iki temel aşamadan oluşmaktadır. Algoritma, her bir veri nesnesini, ilgili nesnenin belirli bir kümeye ait olma olasılığını temsil eden üyelik derecesine göre kümelere atar (Han and Kamber, 2006). Algoritmada, sonlu Gauss karışım modeli kullanılarak parametre değerleri yinelemeli olarak tahmin edilir (Jin and Han, 2010). Algoritmada öncelikle tahminsel parametre değerleri belirlenir. Ardından, bu parametre değerleri, beklenti ve maksimizasyon aşamalarında, her bir örneğin olasılık üyelik değerleri, tahminsel parametre değerlerine göre hesaplanarak güncellenir (Dempster et al., 1977).

Öz-düzenlemeli harita algoritması (*self-organizing maps, SOM*), veri setinde boyut indirgemeye dayalı, bölümleyici bir kümeleme algoritmasıdır (Kohonen, 2001). SOM algoritması, kümeleme sonuçlarını kolay yorumlanabilir ve düşük boyutlu bir yapıda sunmaktadır. SOM algoritmasında, veri noktaları, harita üzerindeki ızgara noktalarına yakınlıklarına göre, düşük boyutlu topolojik bir harita üzerinde temsil edilmektedir.

3.7 Arama Algoritmaları

Bu bölümde, tez çalışmasının sınıflandırıcı topluluğu birleştirme kuralı ve sınıflandırıcı topluluğu budama aşamalarında kullanılan algoritmalar tanıtılmaktadır. Sınıflandırıcı topluluğu model kütüphanesinde yer alan sınıflandırma algoritmalarından, uygun bir sınıflandırıcı altkümesi elde edilmesi bir arama sürecidir. Arama sürecinin, tam arama (*exhaustive search*) yöntemi ile aranması, az sayıda sınıflandırıcı içeren, sınıflandırıcı toplulukları için bile oldukça maliyetli olmaktadır. Tam arama, *n,* sınıflandırıcı topluluğunda yer alan temel sınıflandırma algoritması sayısını temsil etmek üzere, 2^n olası altkümenin incelenmesini gerektirmektedir. Sınıflandırıcı topluluğundan uygun bir sınıflandırıcı altkümesi elde edilmesi problemi, öznitelik altkümesi oluşturma sürecine benzerlik göstermektedir. Her iki süreçte de metasezgisel arama algoritmalar sıklıkla uygulanmaktadır. Tez çalışmasının sınıflandırıcı budama aşamasında, en iyi önce arama algoritması, genetik algoritma, açgözlü arama algoritması, parçacık sürüsü eniyilemesi, diferansiyel gelişim algoritması ve ENORA algoritması gibi arama algoritmalarının etkinliği değerlendirilmiştir. Tez çalışması kapsamında geliştirilen, çok amaçlı eniyilemeye dayalı ağırlıklı oylama

yönteminde ise çok amaçlı parçacık sürüsü eniyilemesi ve çok amaçlı diferansiyel gelişim algoritmaları kullanılmıştır.

3.7.1 En iyi önce arama algoritması

En iyi önce arama algoritması (*best first search, BFS*), basit sezgisel arama algoritmalarından biridir. En iyi önce arama algoritmasında, arama yolu üzerinde aynı yoldan geri dönmeye (*backtracking*) izin verilir (Rich and Knight, 1991; Hall, 1999). En iyi önce arama algoritmasında, açgözlü tepe tırmanma (*greedy hill climbing*) algoritmasındaki gibi mevcut sınıflandırma algoritması altkümesi üzerinde yerel değişiklikler yapılarak, arama uzayında ilerlenir. En iyi önce arama algoritmasında, arama işlemi mevcut düğümün çocukları arasından en iyi değere sahip olanın seçilmesi şeklinde devam ettirilir. Ancak, burada aynı yoldan geri dönme özelliği sayesinde, daha az umut verici bir arama yoluna ulaşıldığı takdirde, daha önceki, iyi alt kümeye geri dönülerek, arama süreci bu konumdan devam ettirilebilir (Hall, 1999). Algoritma 3.9'da en iyi önce algoritmasının genel yapısı özetlenmiştir.

Algoritma 3.9 BFS algoritması (Hall, 1999)

1. Başlangıç durumunu içeren AÇIK liste ile başla, KAPALI listeyi boş olarak oluştur ve başlangıç durumunu ENİYİ'ye ata,
2. AÇIK listeden en yüksek değerlendirme değerine sahip durumu al ve *s*'ye ata,
3. *s*'yi AÇIK listeden kaldır ve KAPALI listeye ekle,
4. Eğer *s*'nin değerlendirme değeri, ENİYİ'nin değerlendirme değerinden büyük ya da eşitse, *s'yi* ENİYİ'ye ata,
5. *s*'nin AÇIK ya da KAPALI listeden yer almayan her bir çocuk düğümü *t*'yi değerlendir ve AÇIK listeye ekle,
6. Eğer düğümlerin genişletilmesi ile ENİYİ değeri değiştiyse Adım 2'ye git.
7. ENİYİ'yi döndür.

3.7.2 Genetik algoritmalar

Genetik algoritmalar (*genetic algorithms, GA*), evrimsel hesaplamaya dayalı metasezgisel yöntemlerdir. Genetik algoritmaların sınıflandırıcı seçiminde kullanılması ile sabit uzunlukta bir ikili alt küme dizisi elde edilmesi amaçlanır. Bu dizideki her bir pozisyonda yer alan değer, belirli bir sınıflandırma algoritmasının ilgili altkümede var olma ya da olmama durumlarına karşılık gelir. Algoritma 3.10'da genetik algoritmanın sınıflandırma seçimi arama sürecinde kullanılmasına ilişkin temel adımlar özetlenmektedir.

Algoritma 3.10 GA algoritması (Hall, 1999)

1. Rastgele olarak oluşturulmuş *P* başlangıç toplumu ile başla,
2. *P* toplumunda yer alan her bir birey *x* için uygunluk fonksiyonu (e(x)) değerini hesapla,
3. *P* toplumunun bireyleri üzerinde ($p(x) \propto e$(x)) olacak şekilde bir *p* olasılık dağılımı tanımla,
4. *p* olasılık dağılımına dayalı olarak *P* toplumundan iki birey (*x* ve *y*) seç,
5. *x* ve *y* üzerinde çaprazlama operatörü uygulayarak yeni bireyler (*x'* ve *y*') oluştur,
6. Yeni bireyler (*x'* ve *y*') üzerinde mutasyon operatörü uygula,
7. *x'* ve *y'*'yi bir sonraki nesle (*P*') ekle,
8. Eğer $|P'| < |P|$ ise Adım 4'e git,
9. $P \leftarrow P'$
10. İşlenmesi gereken daha fazla nesil olması durumunda Adım 2'ye git,
11. *P* toplumunda yer alan en yüksek uygunluk değerine sahip bireyi döndür.

Genetik algoritma, bir sonraki neslin çaprazlama, mutasyon gibi genetik operatörler kullanılarak oluşturulduğu yinelemeli bir süreçtir. Çaprazlama ile atalardaki seçili genler üzerinde işlem yapılarak yeni yavrular oluşturulur. Mutasyon operatörü ile çaprazlama sonucu oluşan yavrular rastgele olarak değiştirilerek, çözümün yerel uygun değere takılması engellenir. Toplum içinden hangi kromozomların çaprazlama için seçileceğinin belirlenmesinde her bir kromozomun uygunluk değeri dikkate alınmaktadır (Obitko, 1998). Bu nedenle, yüksek uygunluk değerine sahip sınıflandırma algoritması altkümelerinin çaprazlama ve mutasyon ile yeni sınıflandırma algoritması altkümelerini oluşturmak üzere seçilme olasılıkları daha yüksektir.

3.7.3 Açgözlü arama algoritması

Açgözlü arama algoritması (*greedy search, GS*), arama sürecinde kullanılan hızlı bir sezgisel yöntemdir. Açgözlü arama ile, *n*, sınıflandırıcı içeren bir sınıflandırıcı topluluğu için incelenmesi gereken altküme sayısı n^2'ye düşürülür. Açgözlü arama algoritmaları, arama uzayındaki tüm sınıflandırma algoritmalarını içeren ya da hiçbir sınıflandırma algoritması içermeyen bir küme ile başlatılabilir. İleri doğru açgözlü aramada (*forward greedy search*), boş altküme ile başlanır. Ardından, her bir adımda, bir sınıflandırma algoritması seçilerek, bu algoritma, mevcut altkümeye eklenir. Bu yöntem, ardışık ileri seçim (*sequential forward selection*) olarak da adlandırılmaktadır. Bu arama yöntemi, birbirleriyle yüksek ilişkili, az sayıda sınıflandırma algoritması içeren bir altküme elde edilmesini sağlar. Aramanın geriye doğru yapılması durumunda ise model kütüphanesinde yer alan tüm sınıflandırma algoritmalarını içeren bir küme ile başlanır. En kötü başarıma sahip sınıflandırma algoritmasının çıkarılması, daha yüksek bir değerlendirme değeri oluşturduğu sürece, süreç sürdürülür. Algoritma 3.11'de ileri doğru açgözlü aramanın temel adımları özetlenmiştir.

Algoritma 3.11 Açgözlü arama algoritması (Gütlein, 2006)

1. $S_{current}$'a boş küme olarak al,
2. S_{best}'i boş küme olarak al,
3. Sonlandırma koşulu sağlanıncaya dek aşağıdaki adımları yinele:
4. $S_{current}$'ta yer almayan her bir a sınıflandırıcısı için:
   a. Eğer sınıflandırıcının $S_{current}$'a eklenmesi sonucu elde edilen değerlendirme değeri, S_{best}'ten büyükse, $S_{best}= S_{current} \cup$ a olacak şekilde güncelle.
   b. $S_{current} =S_{best}$
5. Eğer S_{best} değişmediyse dur.
6. S_{best}'i döndür.

3.7.4 Parçacık sürü eniyilemesi algoritması

Parçacık sürü eniyilemesi algoritması (*particle swarm optimization, PSO*), bir eniyileme problemini, *P* adet parçacıktan oluşan bir toplum aracılığıyla çözmeye çalışan metasezgisel bir yöntemdir.

Algoritma 3.12 PSO algoritması (Diao, 2014)

1. Parametrelere ilk değerlerini ata,
2. Rastgele üretilmiş çözüm ile başla,
3. Belirtilen maksimum nesil sayısına ulaşılıncaya değin aşağıdaki adımları yinele:
   a. Mevcut altkümeyi ve yerel en iyi altkümeyi güncelle.
   b. Toplumdaki 1. Bireyden başlayarak tüm bireyler için aşağıdaki adımları gerçekleştir:

$$0 \le r_{d1}, r_{d2} \le 1$$

$$v_i = w_g v_i + c_1 r_{d1} d(\overline{B}^{p^i}, B^{p^i}) + c_2 r_{d2} d(\overline{B}^{p^i}, B^{p^i})$$

B^{p^i}'yi v_i'ye dayalı olarak $\overline{B}^{p^i}$'ye doğru hareket ettir

   c. $$w_g = w_{\min} + (1 - \frac{g}{g_{\max}})(w_{\max} - w_{\min})$$ şeklinde güncelle.

Bu yöntemde, parçacıklar çözüm uzayı etrafında pozisyon ve hızlarına dayalı olarak hareket eder. Belirli bir parçacığın hareketi, hem ilgili parçacığın mevcut en iyi pozisyonuna hem de sürüdeki en iyi mevcut çözüme dayalı olarak yönlendirilir. Bireysel en iyi pozisyon, parçacıklar daha iyi konumlara ulaştığında güncellenir. Yöntem, sınıflandırma algoritması seçiminde kullanıldığında, belirli bir aday sınıflandırma algoritması altkümesinin (B^{p^i}) hız parametresi v_i, değiştirilecek sınıflandırma algoritması sayısını temsil edecek şekilde, Eşitlik 3.41'e göre belirlenir (Diao, 2014):

$$v_i = w_g v_i + c_1 r_{d1} d(\overline{B}^{p^i}, B^{p^i}) + c_2 r_{d2} d(\overline{B}^{p^i}, B^{p^i}) \quad (3.41)$$

Burada, w_g, giderek azalan eylemsizlik ağırlığını, c_1 ve c_2, hızlanma katsayılarını, r_{d1} ve r_{d2}, [0-1] aralığında uniform rastgele sayıları, $d(\overline{B}^{p^i}, B^{p^i})$, mevcut en iyi altküme ve mevcut altküme arasındaki uzaklığı temsil etmektedir. Algoritma 3.12'de parçacık sürü eniyilemesi algoritmasının temel adımları özetlenmiştir. Burada, $p^i \in P$ parçacık sürüsünde yer alan parçaları, $\overline{B}^{p^i}$, p^i tarafından bulunan yerel en iyi altkümeyi, c_1 ve c_2, sırasıyla yerel en iyi altkümeyi ve mevcut altkümeye ilişkin hızlanma katsayılarını temsil etmektedir.

Algoritma 3.13 CMDPSOFS Algoritması (Xue et al., 2013)

Girdi: Eğitim ve test seti.

Çıktı: Egemen olmayan çözüm seti, eğitim ve test seti doğru sınıflandırma oranları.

1. Başla,
2. Parçacık sürüsüne ilk değerlerini ata ve başlat,
3. Lider Kümesi ve Arşive ilk değerlerini ata,
4. Lider Kümesinde yer alan her bir eleman için yoğunluk mesafesini hesapla,
5. Maksimum yineleme sayısına ulaşılmadığı sürece aşağıdaki adımları yinele:
    a. Her bir parçacık için belirtilen adımları uygula:
        i. Parçacık için Lider Kümesinden ikili tur seçimine ve yoğunluk mesafesine dayalı olarak bir lider (*gbest*) seç,
        ii. Parçacığın hız ve pozisyonunu güncelle,
        iii. Mutasyon operatörü uygula,
        iv. Parçacığı amaç fonksiyonu kullanarak (örnek sayısı ve doğru sınıflandırma oranı bakımından)
        v. Parçacığın *pbest* konumunu güncelle,
    b. Egemen olmayan çözüm setini belirleyerek Lider Kümeyi güncelle,
    c. Lider Kümesinde yer alan her bir eleman için yoğunluk mesafesini hesapla,
6. Arşivde yer alan çözümlerin test seti üzerinde doğru sınıflandırma yüzdelerini elde et,
7. Arşivde yer alan çözümlerin test ve eğitim seti üzerindeki doğru sınıflandırma yüzdelerini döndür.

3.7.5 Çok amaçlı parçacık sürüsü eniyilemesi algoritması

Çok amaçlı sınıflandırıcı ağırlığı belirlemede kullanılan ilk yöntem, çok amaçlı parçacık sürüsü eniyileme algoritması olan CMDPSOFS (PSO based multiobjective feature selection approach) algoritmasıdır (Xue et al., 2013). Algoritma 3.13'te CMDPSOFS algoritmasının temel adımları sunulmuştur. CMDPSOFS, yoğunluk mesafesi, mutasyon ve egemen set kavramlarına dayalı bir algoritmadır. Algoritmada, her bir parçacığın egemen olmayan çözüm seti olası liderler olarak lider kümesinde tutulmaktadır. Lider (*gbest*) ise lider kümesindeki adaylar içerisinden yoğunluk mesafelerine ve ikili turnuva seçimine dayalı olarak seçilmektedir. Yoğunluk faktörü parametresi ile egemen olmayan

çözüm setinden hangi çözümlerin lider kümesine ekleneceğinin belirlenmesi için kullanılmaktadır. Turnuva seçimi ile lider kümesinde yer alan iki lider rastgele olarak seçilmekte bunlar içerisinden en düşük yoğunluk mesafesine sahip olan lider adayı, *gbest* olarak seçilmektedir. Mutasyon operatörü kullanılarak sürünün çeşitliliği artırılmaktadır.

3.7.6 ENORA algoritması

ENORA algoritması (*Elitist Pareto-based multi-objective evolutionary algorithm*), seçkinciliğe dayalı çok amaçlı bir evrimsel eniyileme algoritmasıdır (Jimenez et al., 2014). ENORA algoritması, μ, toplum büyüklüğünü ve λ, toplumdaki çocuk sayısını temsil etmek üzere, (μ+ λ) hayatta kalma oranına sahiptir. Hayatta kalma oranının bu şekilde hesaplanması, μ adet en iyi çocuk ve ata bireyin hayatta kalmasını olanaklı kılmaktadır. ENORA algoritması, seçkinciliğin yanı sıra, ikili turnuva seçimi (*binary tournament selection*), genetik çeşitlenme (*recombination*) ve mutasyon operatörlerini kullanmaktadır. ENORA algoritması, *N* bireyden oluşan bir toplumun oluşturulması ile başlar. Algoritmanın her bir yinelemesinde, ikili turnuva seçimi operatörü kullanılarak, toplumdan iki ata seçilir. Böylelikle, topluluk sıralaması fonksiyonuna göre en iyi iki birey, ata olarak seçilmiş olur.

Algoritma 3.14 ENORA algoritmasının genel yapısı (Jimenez et al., 2014)

Girdiler: *T, N*
T, yineleme sayısı
N, toplumdaki birey sayısı

1. *P* toplumunu *N* tane birey ile oluştur,
2. *P* toplumunun her bir bireyini değerlendir,
3. *t*←0
4. **while** *t*<*T* **do**
5. *Q*←*Ø*
6. *i*←0
7. **while** *i*<*N* **do**
8. *Ata1*←İkili turnuva seçimi ile toplumdan seçim
9. *Ata2*←İkili turnuva seçimi ile toplumdan seçim
10. *Çocuk1, Çocuk2*←Varyasyon *Ata1, Ata2*
11. *Çocuk1'*i değerlendir
12. *Çocuk2'yi* değerlendir
13. *Q*← *Q*∪{*Çocuk1, Çocuk2*}
14. *i*←*i+2*
15. **endwhile**
16. *R*← *P*∪*Q*
17. *P*←*R'*den en iyi *N* tane bireyin topluluk sıralaması fonksiyonuna göre seçilmesi
18. *t*←*t+1*
19. **endwhile**
20. *P'*den egemen olmayan çözüm kümesini döndür.

Algoritma 3.15 İkili turnuva seçimi (Jimenez et al., 2014)

Girdiler: *P*
P, toplum

1. *I*←*P*'den rastgele seçim
2. *J*←*P*'den rastgele seçim
3. *I* ve *J* bireylerinin, topluluk sıralaması fonksiyonuna göre değerlerini belirle ve iyi olanı döndür.

Algoritma 3.16 Topluluk sıralaması fonksiyonu (Jimenez et al., 2014)

Girdiler: *P, I, J*
P, toplum
I, birey
J, birey

1. Eğer *rank(P,I)*<*rank(P,J)* ise,
2. Doğru döndür.
3. Eğer *rank(P,J)*<*rank(P,I)* ise,
4. Yanlış döndür.
5. *k(P,I)*>*k(P,J)* döndür.

Algoritma 3.14'te ENORA algoritmasının genel yapısı, Algoritma 3.15'te algoritmada kullanılan ikili turnuva seçimi operatörü ve Algoritma 3.16'da topluluk sıralaması fonksiyonunun genel işleyişi sunulmaktadır.

Topluluk sıralaması fonksiyonunda, toplumdaki herhangi iki birey *I* ve *J*'den topluluk sıra değeri düşük olan, daha iyi birey olarak kabul edilir. Herhangi bir *I* bireyinin *P* toplumundaki sıra değeri (*rank*(*P*,*I*)) ile temsil edilmek üzere, *I* bireyinin *P* toplumundaki *J* bireylerine karşı egemen olmama seviyesini temsil etmektedir ve *slot(I)* ve *slot(J)* fonksiyonu değerleri eşit olacak şekilde hesaplanmaktadır. *slot(I)* fonksiyonu, *I* bireyinin ait olduğu arama uzayını temsil etmek üzere, Eşitlik 3.42'ye göre hesaplanmaktadır (Jimenez et al., 2014):

$$slot(I) = \sum_{j=1}^{n-1} d^{\,j-1} \left\lfloor d \frac{\alpha_j^I}{\pi/2} \right\rfloor \tag{3.42}$$

$$\alpha_j^I = \begin{cases} \pi/2 \quad , \quad h_j^I = 0 \\ \arctan\left(\dfrac{h_{j+1}^I}{h_j^I}\right) \quad , \quad h_j^I \neq 0 \end{cases} \tag{3.43}$$

Burada, $d = \lfloor \sqrt[n-1]{N} \rfloor$ ve h_j^I, [0, 1] aralığında normalize edilmiş amaç fonksiyonunu temsil etmektedir. Herhangi iki bireyin aynı sıra değerine sahip olması durumunda, daha yüksek kalabalık uzaklığına sahip olan daha iyi bir birey olarak kabul edilir. Kalabalık uzaklığı, Eşitlik 3.44'e göre hesaplanır (Jimenez et al., 2014):

$$k(P,I)=\begin{cases}\infty \quad , \quad f_j^{\,I}=f_j^{\max}, f_j^{\,I}=f_j^{\min}, \exists j \\ \sum_{j=1}^{n}\dfrac{f_j^{\sup_j^I}-f_j^{\inf_j^I}}{f_j^{\max}-f_j^{\min}}, Aksi \quad takdirde\end{cases} \tag{3.44}$$

Burada, $f_j^{\max}$ ve $f_j^{\min}$, sırasıyla, toplumda yer alan herhangi bir bireyin aldığı maksimum ve minimum amaç fonksiyonu değerlerini, $f_j^{\sup_j^I}$ ve $f_j^{\inf_j^I}$, sırasıyla, j amaç fonksiyonu için, I bireyinden daha yüksek değer elde eden en yakın bireyin değerini ve j amaç fonksiyonu için, I bireyinden daha düşük değer elde eden en yakın bireyin değerini temsil etmektedir.

3.7.7 Diferansiyel gelişim algoritması

Diferansiyel gelişim algoritması (*differential evolution*), toplum-tabanlı bir sezgisel algoritmadır (Storn and Price, 1997). Diferansiyel gelişim algoritması, diğer evrimsel algoritmalar ile ortak bazı özelliklere sahip olmak ile birlikte, algoritmada arama sürecine mevcut toplumun uzaklık ve yön bilgileri dâhil edilmektedir (Engelbrecht, 2007). Evrimsel algoritmalarda, çeşitlilik genellikle çaprazlama (*crossover*) ve mutasyon (*mutation*) operatörleri aracılığıyla sağlanmaktadır. Evrimsel algoritmalarda genellikle öncelikle çaprazlamanın uygulanması, ardından mutasyon operatörünün uygulanması şeklinde bir yapı benimsenir. Ancak, diferansiyel gelişim algoritmasında öncelikle mutasyon operatörü uygulanarak, çaprazlama operatörü tarafından yavruların oluşturulmasında kullanılacak deneme vektörü elde edilmektedir. Bunun yanı sıra, mutasyon oranının belirlenmesinde önceden bilinen herhangi bir olasılık dağılım fonksiyonu kullanılmamaktadır. Mutasyon oranı, mevcut toplumdaki bireylerin farklarına dayalı olarak belirlenmektedir (Engelbrecht, 2007). Tek amaçlı eniyileme problemlerinin çözümünde kullanılan temel diferansiyel gelişim algoritmasında, mutasyon operatörü uygulanarak öncelikle mevcut toplumdaki her bir birey için bir deneme vektörü elde edilmektedir (Engelbrecht, 2007; Price et al., 2005). Ardından, çaprazlama operatörü, deneme vektörü üzerinde uygulanarak yavrular oluşturulmaktadır. Her bir bireyden ($x_i(t)$), deneme vektörü ($u_i(t)$) şu şekilde oluşturulmaktadır. Toplumdan $i \neq i_1$ olacak şekilde bir hedef vektörü ($x_{i1}(t)$) seçilir. Ardından toplumdan iki birey (x_{i2} ve x_{i3}) $i \neq i_1 \neq i_2 \neq i_3$ olacak şekilde rastgele olarak seçilir. Deneme vektörü, Eşitlik 3.45'e göre belirlenir (Engelbrecht, 2007):

$$u_i(t)=x_{i1}(t)+\beta(x_{i2}(t)-x_{i3}(t)) \tag{3.45}$$

Burada, $\beta \in (0, \infty)$ ölçek faktörüdür.

Diferansiyel gelişim algoritmasının diğer bir temel operatörü çaprazlama operatörüdür. Çaprazlama ile yavru ($x_i'(t)$), deneme vektörü ($u_i(t)$) ile ata vektörün ($x_i(t)$) ayrık birleşimi ile elde edilmektedir. Eşitlik 3.46'da diferansiyel gelişim algoritması çaprazlama operatörü verilmiştir:

$$x_{ij}^{'}(t) = \begin{cases} u_{ij}(t)\, if & j \in J \\ x_{ij}(t) \quad aksi \quad takdirde \end{cases} \tag{3.46}$$

Burada, $x_{ij}(t)$, $x_i(t)$ vektörünün *j.* elemanını J, çaprazlama noktaları kümesini temsil etmektedir. Çaprazlama noktalarının belirlenmesinde, Binom ve üstel çaprazlama gibi yöntemler kullanılmaktadır.

3.7.8 Çok amaçlı diferansiyel gelişim algoritması

Diferansiyel gelişim algoritması, çok amaçlı eniyileme problemlerinin çözümünde başarı ile uygulanmaktadır. Geleneksel diferansiyel gelişim algoritmaları, tek amaçlı eniyileme problemlerini çözmek amacıyla geliştirilmiştir. Diferansiyel gelişim algoritmasının çok amaçlı problemlerde uygulanabilmesi için egemen olmayan çözüm seti elde edilebilecek şekilde geliştirilmesi gerekmektedir (Psychas et al., 2015). Çok amaçlı diferansiyel gelişim algoritmasında iki önemli etken çeşitliliğin nasıl sağlanacağı ve seçkinciliğin nasıl yapılacağıdır (Mezura-Montes et al., 2008). Çeşitliliğin sürdürülebilmesi için toplumdaki bireylerin birbirlerine olan yakınlıklarının belirlenmesi gereklidir. Yakınlığın ölçümlenmesinde, kalabalık mesafesi (*crowding distance*) ya da uygunluk paylaşımı (*fitness sharing*) gibi çeşitli uzaklık ölçütleri kullanılmaktadır (Deb et al., 2002; Goldberg and Richardson, 1987). Seçkinciliğin yapılabilmesi için ise egemen olmayan çözümlerin yer aldığı bir arşiv tutulabilir (Mezura-Montes et al., 2008). Sınıflandırıcılara uygun ağırlık değerlerinin atanmasında, çok amaçlı diferansiyel gelişim algoritmasına dayalı (MODE) algoritması kullanılmıştır (Psyschas et al., 2015). Algoritma 3.17'de MODE algoritmasının genel yapısı verilmiştir.

Bu algoritmada, performans iyileştirmeye yönelik olarak birkaç farklı mutasyon operatörü geliştirilmiştir. Sınıflandırıcı topluluğunda ağırlık ataması problemine ilişkin deneysel çalışmalarda en yüksek doğru sınıflandırma başarımı MODE1 mutasyon operatörü ile elde edildiğinden, bu mutasyon operatörü kullanılmıştır. MODE algoritması, mutasyon operatörü ile başlamaktadır. Mutasyon operatörü ile toplumdaki her bir birey için bir deneme vektörü oluşturulur. MODE1 mutasyon operatörü, Pareto-optimal çözümler kümesi ve

Pareto-optimal olmayan çözümler kümesinden bireyleri dahil edecek şekilde Eşitlik 3.47'ye göre işlemektedir (Psychas et al., 2015):

$$u_i(t) = Pareto_{i1}(t) + \beta(x_{i2}(t) - x_{i3}(t)) \tag{3.47}$$

Burada, $Pareto_i$, Pareto-optimal çözümler kümesinden rastgele çözümleri ve x_i, toplumun rastgele çözümlerini temsil etmektedir. Bu mutasyon operatöründe, hedef vektörü, Pareto-optimal çözümler kümesi ve toplumdaki diğer vektörler arasından rastgele olarak seçilmektedir. Lider ise Pareto-optimal çözümler arasından rastgele seçilmektedir. Diferansiyel gelişim algoritmasında, çaprazlama operatörü genellikle deneme vektörü üzerinde uygulanarak yavrular oluşturulmaktadır. Ancak, MODE algoritmasında, çaprazlama operatörü kullanılmamıştır. Çözüm kalitesini iyileştirmek için, deneme vektörleri üzerinde belirli bir yineleme sayısı boyunca değişken komşuluk araması algoritması uygulanmaktadır.

Algoritma 3.17 MODE Algoritması (Psychas et al., 2015)

1. İlk değer ata,
2. Birey sayısını belirle,
3. İlk toplumu oluştur,
4. Toplumun her bir amaç fonksiyonu için değerlendir,
5. Mutasyon operatörünü seç,
6. Pareto-optimal çözüm kümesine ilk değer ata,
7. Maksimum yineleme sayısına ulaşılıncaya dek aşağıdaki adımları yinele:
    a. Maksimum ata birey sayısına ulaşılıncaya dek aşağıdaki adımları yinele:
        i. Ata birey vektörünü seç,
        ii. Mutasyon operatörü ile deneme vektörünü oluştur,
        iii. Bireyleri her bir amaç fonksiyonuna dayalı olarak değerlendir,
        iv. Her bir deneme vektörüne değişken komşuluk arama algoritmasını uygula,
        v. Yavrunun ata üzerinde egemen olması durumunda, ata ile yavruyu bir sonraki kuşak için değiştir.
    b. Pareto-optimal çözüm kümesini güncelle.
8. Pareto-optimal çözüm kümesini döndür.

4. GELİŞTİRİLEN YÖNTEMLER

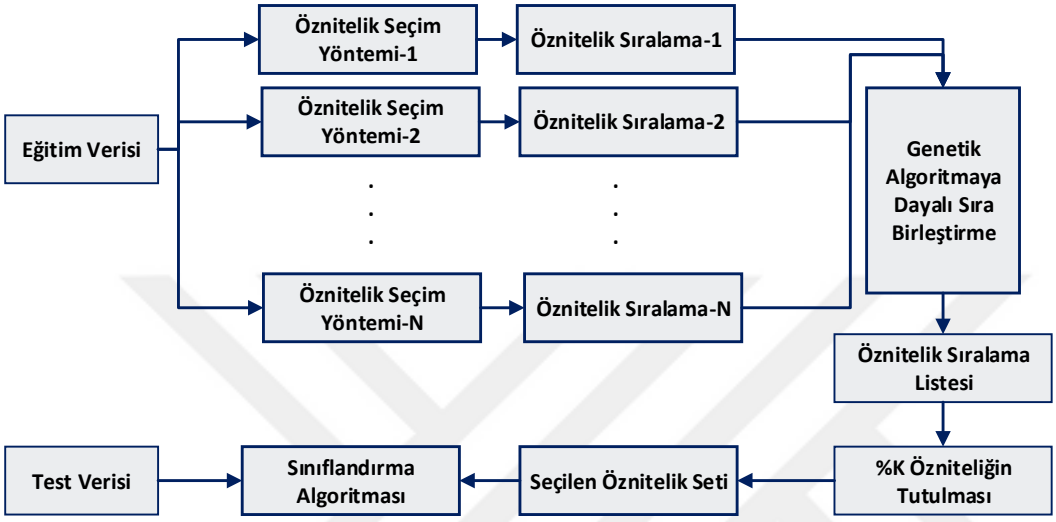
Bu bölümde, tez çalışması kapsamında geliştirilen genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçim yöntemi (GA), sınıflandırıcı topluluğunda yer alan temel sınıflandırma algoritmalarına uygun ağırlık değerleri atanabilmesi için geliştirilen çok amaçlı eniyilemeye dayalı sınıflandırma mimarisi ve geliştirilen sınıflandırıcı topluluğu budama yöntemi ile tez kapsamında geliştirilen görüş sınıflandırma mimarisi tanıtılmaktadır. Genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçim yönteminde, yedi farklı filtre-tabanlı öznitelik seçim yöntemi ile elde edilen sıralamalar birleştirilerek, öznitelik seçimi işlemi birleştirilmiş sıralama üzerinden gerçekleştirilmektedir. Geliştirilen çok amaçlı eniyilemeye dayalı ağırlıklı oylama yönteminde, sınıflandırıcılara uygun ağırlık değerleri atanması problemi, bir eniyileme problemi olarak ele alınarak, sınıflandırma algoritmalarının her bir çıktı sınıfındaki başarımlarına göre, uygun ağırlık değerleri ile sınıflandırıcı topluluğunun çıktısına katkı koymaları işlemi, çok amaçlı diferansiyel gelişim algoritması aracılığıyla gerçekleştirilmektdir. Geliştirilen sınıflandırıcı topluluğu budama yönteminde ise ortak kümeleme ve çok amaçlı evrimsel arama yöntemleri birlikte kullanılarak, model kütüphanesinden etkin sınıflandırıcı topluluğu altkümesinin belirlenmesi amaçlanmaktadır.

4.1 Genetik Algoritma İle Sıra Birleştirmeye Dayalı Öznitelik Seçim Yöntemi

Bu bölümde, tez çalışması kapsamında geliştirilen genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçim yöntemi (GA) sunulmaktadır (Onan and Korukoğlu, 2015a). Öznitelik sıralama yöntemleri ile farklı değerlendirme ölçütlerini temel alan değişik sıralamalar elde edilmektedir. Farklı öznitelik sıralama yöntemleri ile elde edilen değişik sıralamaların uygun bir biçimde birleştirilmesi ile etkin bir öznitelik sıralaması elde edilebilir (Dwork et al., 2001). Bu bölümde, farklı öznitelik sıralama yöntemlerinin sonuçlarından genetik algoritma ve sıra birleştirme yöntemleri kullanılarak tek bir sıralama listesi elde edilmesi açıklanmaktadır.

Şekil 4.1'de öznitelik seçim yönteminin temel mimarisi sunulmaktadır. Öncelikle, eğitim verisi üzerinde, farklı öznitelik sıralama yöntemlerine ilişkin öznitelik sıralama listeleri elde edilmektedir. Burada, yedi farklı öznitelik sıralama yöntemi (bilgi kazancı, ki-kare ölçütü, kazanç oranı ölçütü, simetrik belirsizlik

katsayısı, Pearson korelasyon katsayısı, Relief-F algoritması ve olasılıklı önem ölçütü) kullanılarak sıralamalar elde edilmiştir. Ardından, bu sıralamalardan genetik algoritmaya dayalı sıra birleştirme yöntemi kullanılarak tek bir sıralama elde edilmekte ve toplam öznitelik sayısına ilişkin farklı oransal değerlerde en yüksek sıralama derecesi alan öznitelikler tutularak öznitelik seçim işlemi tamamlanmaktadır.



Şekil 4.1. Geliştirilen öznitelik seçim yöntemi mimarisi (GA)

Sıra birleştirme problemi, bir eniyileme problemi olarak ele alındığında, amaç, tüm bireysel sıralamalara olabildiğince yakın yeni bir sıralama elde edilmesi haline gelir (Pihur et al., 2007). Eşitlik 4.1'de eniyileme problemine ilişkin matematiksel ifade verilmektedir:

$$\Phi(\delta) = \sum_{i=1}^{m} w_i d(\delta, L_i) \tag{4.1}$$

Burada, δ, $k=|L_i|$ uzunluğundaki önerilen sıralı listeyi, L_i, i sıralı listesini, w_i, ilgili sıralı listenin önem derecesini ve d, listeler arasındaki uzaklığı ölçümlemek için kullanılan uzaklık ölçütünü temsil etmektedir. Eniyileme probleminin amacı, L_i sıralı listeleri ile arasındaki toplam uzaklığı en aza indirgeyen δ* listesinin Eşitlik 4.2'ye göre belirlenmesidir:

$$\delta^* = \arg\min \sum_{i=1}^{m} w_i d(\delta, L_i) \tag{4.2}$$

Burada, listeler arasındaki uzaklığın ölçümlenmesinde genellikle Spearman uzaklığı (*Spearman footrule distance*) ya da Kendall'ın tau uzaklığı (*Kendall's tau distance*) kullanılmaktadır. Çalışmada, listeler arası uzaklığı ölçümlemek için

Spearman uzaklığı kullanılmıştır. Sperman uzaklığı, Eşitlik 4.3'e göre hesaplanmaktadır (Pihur et al., 2007):

$$S(\delta, L_i) = \sum_{t \in L_i \cup \delta} \left| r^{\delta}(t) - r^{L_i}(t) \right| \tag{4.3}$$

Burada $r^{\delta}(t)$ ve $r^{L_i}(t)$, herhangi bir t elemanının sırasıyla δ ve L_i listelerindeki derecesini(sırasını) temsil etmektedir.

Sıra birleştirme probleminin az sayıda bileşen içeren listeler için dahi kaba kuvvet yöntemler ile çözülmesi oldukça maliyetli olabilmektedir. Bu nedenle, problemin genetik algoritma ya da başka bir metasezgisel algoritma ile ele alınması uygundur. Problemin çözümünde kullanılan genetik algoritmada genellikle gezgin satıcı problemlerinin çözümünde kullanılan yol temsili (*path representation*) kullanılmıştır. Bu temsilde, her bir kromozom *n* uzunluğunda bir permütasyon içermektedir. Her bir elemanın konumu, ilgili elemanın sıralamasını (derecesini) temsil etmektedir. Örneğin, sekiz elemandan oluşan 3-2-4-1-7-5-8-6 şeklindeki bir sıralı liste (3 2 4 1 7 5 8 6) şeklinde temsil edilmektedir. Genetik algoritma ile mevcut listeler ile uzaklığı en az olan yeni bir liste elde edilmesi amaçlandığından uygunluk fonksiyonu olarak Eşitlik 4.3'te belirtilen Spearman uzaklığı kullanılmıştır. Toplum büyüklüğü, *n*, sıralama işlemi uygulanacak nesne sayısını ve *k*, parametre olmak üzere, $S=k.n$ (k=20) bireyden oluşacak şekilde seçilmiştir (Aledo et al., 2013). Genetik algoritmada seçim ile bir sonraki nesildeki yavruları oluşturacak bireylerin toplumdan nasıl seçileceği ve her birinin ne kadar yavru oluşturacağı belirlenir. Seçimin çok sıkı ya da zayıf yapılması, çeşitliliğin azalması ya da evrimin yavaşlaması gibi sorunlara neden olabilir (Mitchell, 1999). Seçim sırasında doğal seçim sürecine benzer şekilde, yüksek uygunluk değerine sahip bireylerin seçilme olasılıklarının daha yüksek olması beklenir. Genetik algoritmada, seçim mekanizması olarak, turnuva seçim yöntemi (k=2) kullanılmıştır. Turnuva seçiminde, toplumdan rastgele olarak *k* tane birey seçilmektedir. Ardından, bu *k* birey uygunluk değerlerine göre karşılaştırılmakta ve içlerinden en yüksek uygunluk değerine sahip olan seçilmektedir. Algoritmanın her bir kuşağında, $k.n$ kromozom çifti rastgele olarak seçilmekte ve bu çiftler arasından en iyi (en düşük) uygunluk değerine sahip bireyler seçilmektedir. Seçilen *S* tane bireye çaprazlama ve mutasyon operatörleri uygulanarak yavru bireyler oluşturulur. Çaprazlama, iki ata çözümden yavrunun oluşturulması sürecidir. Çaprazlamanın ardından mutasyon uygulanarak, algoritmanın yerel en iyiye takılması engellenmekte ve toplumda çeşitliliğin sürdürülmesi sağlanmaktadır. Yol temsilinde, genetik algoritmalarda kullanılan temel çaprazlama ve mutasyon operatörleri kullanılamamaktadır (Larranaga et al.,

1999). Literatürde, yol temsili ile çalışabilen birçok çaprazlama (kısmen haritalanmış çaprazlama, sıra çaprazlama, sıra tabanlı çaprazlama, konum tabanlı çaprazlama gibi) ve birçok mutasyon (yer değiştirme mutasyonu, sinyal değiştirme mutasyonu gibi) bulunmaktadır (Larranaga et al., 1999). Ancak, genetik algoritmaların sıra birleştirmede kullanılmasında en etkin çaprazlama ve mutasyon operatörünün konum tabanlı çaprazlama ve ekleme mutasyon operatörü olduğu deneysel olarak belirlenmiştir (Aledo et al., 2013). Dolayısı ile bu çalışmada belirtilen genetik algoritma operatörleri kullanılmaktadır.

Konum tabanlı çaprazlama (*position based crossover*) yönteminde, rastgele bir konum seti seçilerek, bu konumdaki değerler her iki atada korunurken, diğer konumlar diğer atadaki kısmi sıralama dikkate alınarak doldurulmaktadır. Örneğin (1 2 3 4 5 6 7) ve (4 3 7 6 5 2 1) ataları için {2, 3, 5} konumları seçilmiş olsun. Bu durumda öncelikle yavrular (* 2 3 * 5 * *) ve (* 3 7 * 5 * *) olarak oluşturulmakta ve ardından diğer bileşenler diğer atada kullanılmamış olan değerler kısmi sıraları ile yazılarak oluşturulmaktadır. Böylelikle, (4 2 3 7 5 6 1) ve (1 3 7 2 5 4 6) şeklinde iki yeni yavru oluşturulmaktadır (Syswerda, 1991). Ekleme mutasyon operatöründe (Fogel, 1988) ise rastgele bir eleman seçilerek bu eleman sıralamadan kaldırılmakta ardından rastgele başka bir konuma eklenmektedir. Örneğin (1 2 3 4 5 6 7) üzerinde ekleme mutasyonu yapıldığını ve {5} konumun seçildiğini varsayalım. Bu durumda, öncelikle 5 kaldırılıp (1 2 3 4 6 7) elde edilmekte, ardından {5} rastgele olarak bir konuma yerleştirilerek mutasyon tamamlanmaktadır (1 5 2 3 4 6 7 gibi).

Algoritma 4.1'de genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçim yönteminin genel aşamaları özetlenmiştir.

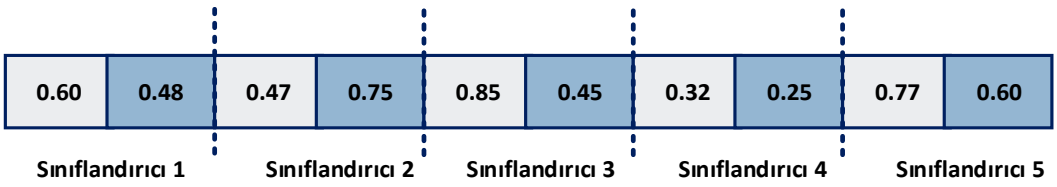
Algoritma 4.1 Genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçimi aşamaları

1. Bilgi kazancı, ki-kare ölçütü, kazanç oranı ölçütü, simetrik belirsizlik katsayısı, Pearson korelasyon katsayısı, Relief-F algoritması ve olasılıklı önem ölçütü filtre tabanlı öznitelik seçim yöntemlerini kullanarak, özniteliklerin yararlılıklarına ilişkin bireysel öznitelik sıralama listeleri oluştur,
2. Her bir kromozom *n* uzunluğunda bir permütasyon içerecek ve her bir elemanın konumu ilgili elemanın sıralamasını belirtecek şekilde listeleri temsil et,
3. Toplumu, *n*, sıralama işlemi uygulanacak nesne sayısını temsil edecek şekilde *S=k.n (k=20)* bireyden oluştur,
4. Spearman uzaklığı kullanarak, toplumdaki her bir bireyi değerlendir,
5. Turnuva seçim yöntemi (*k*=2) kullanarak, toplumdan rastgele olarak *k* tane birey seç,
6. Konum tabanlı çaprazlama uygula,
7. Ekleme mutasyon operatörü uygula,
8. Belirlenen iterasyon sayısına ulaşılıncaya dek, genetik algoritma sürecini devam ettir,
9. Sıra birleştirme ile elde edilmiş birleştirilmiş öznitelik sıralamasını döndür.

4.2 Geliştirilen Sınıflandırıcı Topluluklarında Çok Amaçlı Eniyilemeye Dayalı Ağırlıklı Oylama Yöntemi

Bu bölümde, tez çalışması kapsamında geliştirilen çok amaçlı eniyilemeye dayalı ağırlıklı oylama yöntemi sunulmaktadır (Onan et al., 2016). Geliştirilen çok amaçlı eniyilemeye dayalı ağırlıklı oylama yönteminde, sınıflandırıcı topluluğunda yer alan temel sınıflandırma algoritmalarının ağırlıkları, çok amaçlı sezgisel eniyileme yöntemleri kullanılarak belirlenmektedir. *N*, sınıflandırıcı topluluğunda temel sınıflandırıcı olarak kullanılan temel öğrenme algoritmalarının sayısı olmak üzere C_1, C_2, …, C_N ile temsil edilmektedir. *M*, çıktı sınıfı sayısını temsil etmektedir. Çok amaçlı eniyilemeye dayalı sınıflandırıcı topluluğu tasarımı, sınıflandırıcıların ağırlık değerlerinin, sınıflandırıcıların farklı çıktı sınıflarındaki performanslarına ve amaç fonksiyonlarına dayalı olarak en uygun olarak atanmasını gerektirmektedir. Sınıflandırıcılara ağırlık değerlerinin atanmasında, ilgili çıktı sınıfı için amaç fonksiyonu bakımından yüksek başarıma sahip sınıflandırıcılara daha yüksek ağırlık değerleri verilmesi amaçlanmaktadır. Her bir sınıflandırıcıya, sınıflandırıcının belirli bir çıktı sınıfındaki başarımını temsil etmek üzere [0,1] aralığında uniform ağırlık değerleri atanmaktadır. Bu şekilde, her bir sınıflandırıcı ve çıktı sınıfı çifti için ağırlık değerleri tutulmakta ve bu değerler, çok amaçlı eniyileme algoritması aracılığıyla iyileştirilmektedir.

Her bir sınıflandırıcı ve her bir çıktı sınıfı için, *M*x*N* uzunluğunda bir dizi temsili kullanılmaktadır. Bu temsilde, ağırlık değerleri, her bir sınıflandırıcının, belirli bir çıktı sınıfı için ağırlık derecesini belirtmektedir. Şekil 4.2'de beş sınıflandırıcıdan oluşan ve iki sınıflı bir sınıflandırma problemini çözmek amacıyla geliştirilen bir sınıflandırıcı topluluğuna ilişkin temsil örneği sunulmaktadır.



Şekil 4.2. Beş sınıflandırıcıdan oluşan bir probleme ilişkin örnek temsil

Şekil 4.2'de verilen temsilde, beş sınıflandırıcıdan oluşan ve iki sınıflı bir sınıflandırma problemini çözmek amacıyla geliştirilen bir sınıflandırıcı topluluğunda, ağırlık değerleri 10 uzunluğunda bir dizi ile temsil edilmektedir. Dizide yer alan ilk iki ağırlık değeri, ilk temel sınıflandırıcının sırasıyla birinci ve

ikinci çıktı sınıfı için belirlenen ağırlık değerlerini temsil etmektedir. Dizi temsilinde gerçek kodlama kullanılmış, dizi ilk değer ataması rastgele olarak yapılmıştır. Bu temsilde, dizilere [0-1] aralığında uniform değerler, Eşitlik 4.4'e göre atanmaktadır:

$$s = \frac{rand()}{RAND\_MAX + 1} \tag{4.4}$$

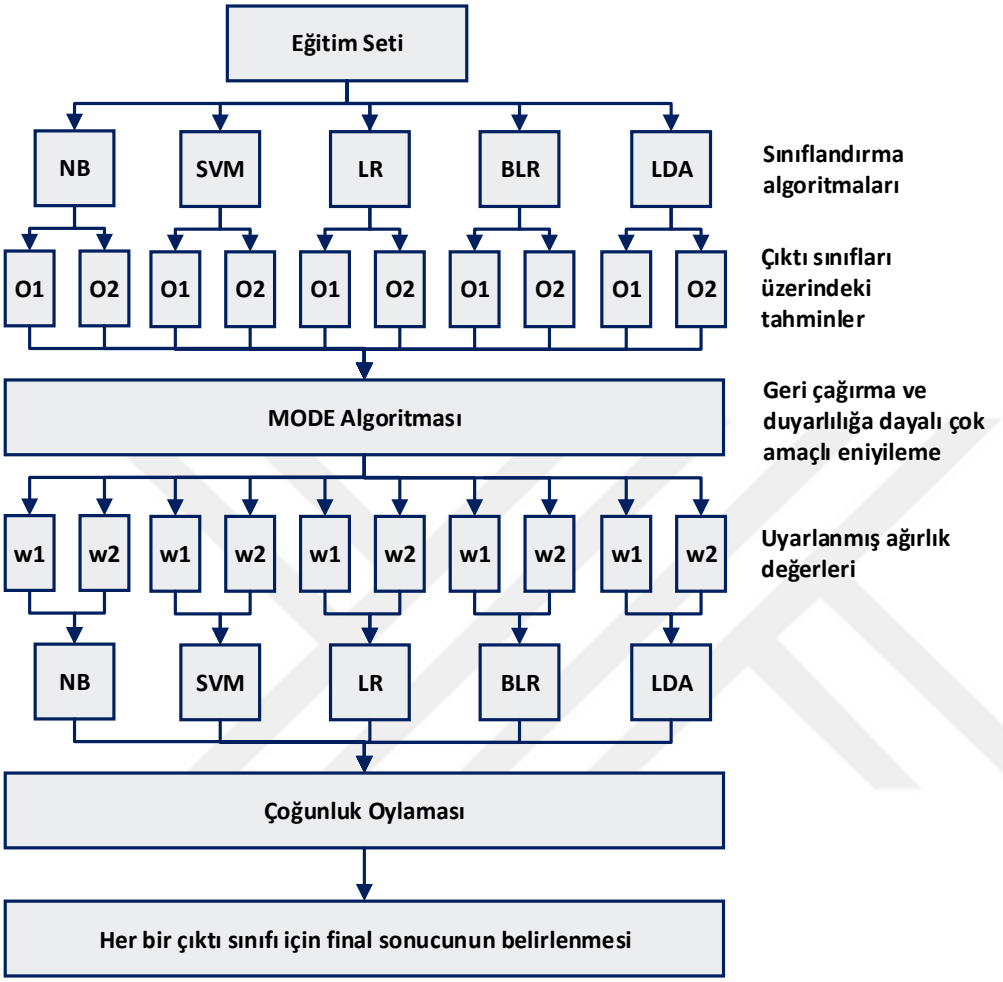
Sınıflandırma algoritmalarının başarımlarının değerlendirilmesinde kullanılan başlıca ölçütler arasında, doğru sınıflandırma oranı, hata oranı (*error rate*), geri çağırma (*recall*), duyarlılık (*precision*) ve F-ölçütü (*F*-measure) gibi değerlendirme ölçütleri yer almaktadır (Costa et al., 2007). Geliştirilen çok amaçlı eniyilemeye dayalı ağırlıklı oylama yönteminde, amaç fonksiyonu olarak kullanılacak ölçütlerin belirlenebilmesi için birkaç farklı ölçüt dikkate alınarak deneysel analizler yapılmıştır. Temel olarak üç farklı çok amaçlı eniyileme fonksiyonu incelenmiştir. Birinci amaç fonksiyonu, sınıflandırmadaki tüm geri çağırma ve duyarlılık değerlerinin dikkate alınmasıdır. İkinci amaç fonksiyonu, her bir bireysel çıktı sınıfındaki F-ölçütü değerinin dikkate alınması ve üçüncü amaç fonksiyonu, her bir bireysel çıktı sınıfının geri çağırma ve duyarlılık değerlerinin dikkate alınmasıdır. Değinilen amaç fonksiyonları ile deneysel çalışmalarda kullanılan veri setleri üzerinde gerçekleştirilen analizlerde, en yüksek başarımın, birinci amaç fonksiyonu ile elde edildiği gözlenmiştir. Bu nedenle, çalışmada, sınıflandırmadaki tüm geri çağırma ve duyarlılık değerlerinin dikkate alan amaç fonksiyonu kullanılmıştır. Farklı ölçütler ile yapılan deneysel çalışmalar sonucunda, duyarlık ve geri çağırma amaç fonksiyonları olarak alınmıştır. Çalışmada, sınıflandırıcı ağırlıkları, çok amaçlı eniyileme algoritmaları aracılığıyla iyileştirildiğinden, her bir çıktı sınıfına ilişkin amaç fonksiyonu değerleri hesaplanmıştır.

Makine öğrenmesi sınıflandırıcıları, metin madenciliği uygulamalarında başarı ile uygulanmıştır. Metin madenciliğinde sıklıkla kullanılan sınıflandırıcılar arasında, istatistiksel sınıflandırıcılar (Naive Bayes algoritması), karar ağacı ve kural tabanlı sınıflandırıcılar (ID3 ve C4.5 algoritması gibi), regresyon yöntemleri (lojistik regresyon gibi), yapay sinir ağları, destek vektör makineleri ve örnek tabanlı sınıflandırma algoritmaları (K-en yakın komşu algoritması) yer almaktadır (Sebastiani, 2002). Sınıflandırıcı topluluklarının başarımlarındaki önemli noktalardan birisi, bir araya getirilecek temel sınıflandırma algoritmalarının belirlenmesidir. Sınıflandırıcı seçimi (*classifier selection*), sınıflandırıcı kümesi içerisinden, birlikte etkin bir biçimde başarım gösterebilecek uygun bir

sınıflandırma altkümesinin belirlenmesini gerektirir. Sınıflandırıcı seçimi yöntemleri, temel olarak, statik ve dinamik sınıflandırıcı seçimi yöntemleri olmak üzere iki sınıf altında incelenmektedir (Ranawana and Palade, 2006). Statik sınıflandırıcı seçiminde, sınıflandırıcı topluluğunda yer alacak temel sınıflandırma algoritmaları, sınıflandırıcıların eğitim setindeki başarımlarına göre belirlenir (Burduk, 2015). Buna karşılık olarak, dinamik sınıflandırıcı seçimi yöntemlerinde, her bir örneklem için özel bir öznitelik altkümesinin belirlenmesi söz konusudur (Burduk, 2015; Ruta and Gabrys, 2005). Az sayıda sınıflandırıcı içeren durumlarda bile sınıflandırıcı topluluğunda yer alacak sınıflandırıcıların belirlenmesi, oldukça fazla olası durumun göz önünde bulundurulmasını gerektirmektedir. Bu nedenle, sınıflandırıcı seçiminde, metasezgisel yöntemler ve değerlendirme ölçütleri sıklıkla kullanılmaktadır. Bu bölümde, sunulan çok amaçlı eniyilemeye dayalı ağırlıklı oylama yönteminde kullanılacak sınıflandırma algoritmalarının belirlenmesinde, çoğunluk oylaması hatası (*majority voting error*) ve ileri arama algoritmasına (*forward search*) dayalı statik sınıflandırıcı seçimi yöntemi kullanılmıştır (Ruta and Gabrys, 2005). Sınıflandırıcı topluluğunda hangi sınıflandırıcıların yer alacağının belirlenmesi için, WEKA yazılımında gerçekleştirimi yapılmış olan on beş farklı sınıflandırıcı algoritması ile deneysel çalışmalarda kullanılan veri setleri üzerinde analizler gerçekleştirilmiştir. İncelenen sınıflandırma algoritmaları arasında, saklı Markov modeli (*hidden Markov model*), Naive Bayes algoritması, seyrek üretimsel model (*sparse generative model*), destek vektör makineleri, lojistik regresyon, Bayesci lojistik regresyon (*Bayesian logistic regression*), lineer diskriminant analizi (*linear discrimant analysis*), K-yıldız algoritması (*K-star algorithm*), K-en yakın komşu algoritması (*K-nearest neighbour algorithm*), C4.5 algoritması, lojistik model ağaçları, rastgele ağaç algoritması (*Random Forest algorithm*), FURIA algoritması, yapay sinir ağları ve radyal tabanlı fonksiyon ağları (*radial basis function networks*) yer almaktadır.

Statik seçim yöntemi ile yapılan analizler sonucunda, sınıflandırıcı topluluğunda, Naive Bayes algoritması (NB), destek vektör makineleri (SVM), lojistik regresyon (LR), Bayesci lojistik regresyon (BLR) ve lineer diskriminant analizi (LDA) yer almaktadır. Bu sınıflandırıcıların ağırlıklarına, çok amaçlı diferansiyel gelişim algoritması (MODE algoritması) aracılığıyla uygun değerler atandıktan sonra, sınıflandırıcı çıktıları, çoğunluk oylaması ile birleştirilmekte ve sınıflandırıcı topluluğunun çıktısı oluşturulmaktadır. Geliştirilen çok amaçlı eniyilemeye dayalı ağırlıklı oylama yönteminin, beş sınıflandırıcıdan oluşan ve iki çıktı sınıfına sahip bir sınıflandırıcı topluluğuna ilişkin mimarisi Şekil 4.3'te

sunulmuştur. Burada, sınıflandırma probleminin çıktı sınıfları O1 ve O2 ile, her bir çıktı sınıfına karşılık sınıflandırıcılara atanan ağırlık değerleri ise w1 ve w2 ile temsil edilmektedir.



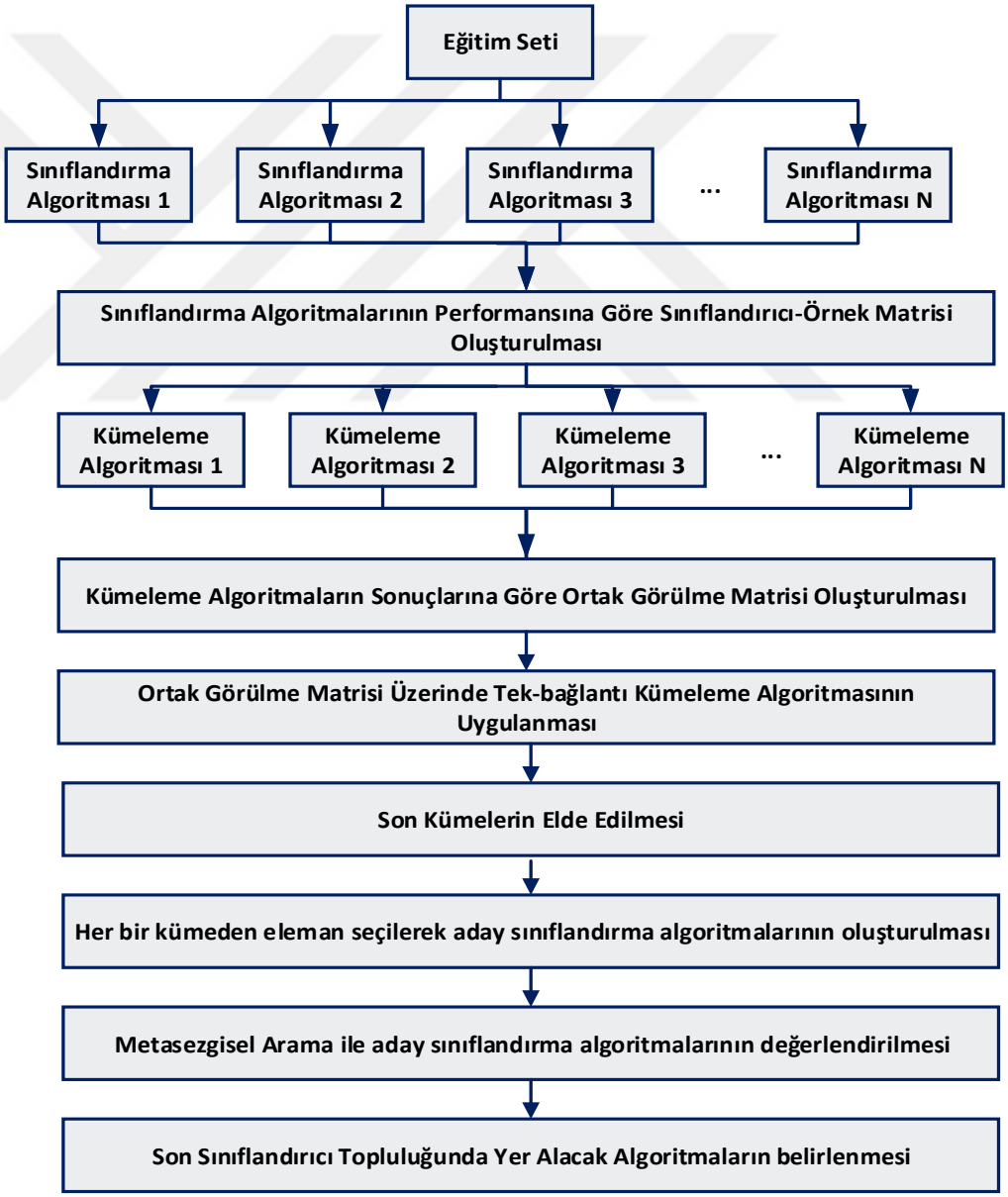
Şekil 4.3. Çok amaçlı eniyilemeye dayalı ağırlıklı oylama yöntemi mimarisi

4.3 Geliştirilen Sınıflandırıcı Topluluğu Budama Yöntemi

Geliştirilen sınıflandırıcı topluluğu budama yöntemi, rastsal topluluk budama yöntemleri (metasezgisel arama algoritmaları aracılığıyla model kütüphanesinden uygun bir altkümenin seçilmesi) ile kümelemeye dayalı topluluk budama yöntemlerini bir arada kullanan melez bir yöntemdir. Geliştirilen yöntemde, farklı temel öğrenme (sınıflandırma) algoritmaları ve bu algoritmaların farklı parametre değerleri alınarak, model kütüphanesi oluşturulmaktadır. Ardından, model kütüphanesinde yer alan her bir temel sınıflandırma algoritması, eğitim seti üzerinde uygulanarak, algoritmaların başarımları elde edilmektedir. N, eğitim setinde yer alan örnek sayısını ve M, model kütüphanesinde yer alan temel

sınıflandırma algoritması sayısını temsil etmek üzere *M*x*N* boyutlu bir sınıflandırıcı-örnek matrisi elde edilmektedir. Eğitim setindeki herhangi bir örneğin, belirli bir sınıflandırma algoritması tarafından doğru sınıflandırılması durumunda, sınıflandırıcı-örnek matrisinin karşılık gelen konumu bir ile doldurulmaktadır. Aksi takdirde, sınıflandırıcı-örnek matrisinin karşılık gelen konumu sıfır ile doldurulmaktadır. Böylelikle, sıfır ve bir değerleri içeren ve farklı sınıflandırma algoritmalarının, eğitim setindeki örnekleri sınıflandırmadaki başarım özelliklerini özetleyen bir matris elde edilmektedir.

Geliştirilen ortak kümelemeye ve metasezgisel aramaya dayalı sınıflandırıcı topluluğu budama mimarisinin genel yapısı Şekil 4.4'te sunulmaktadır.



Şekil 4.4. Geliştirilen budama mimarisinin genel yapısı

Algoritma 4.2 Ortak kümelemeye ve metasezgisel aramaya dayalı sınıflandırıcı topluluğu budama yöntemi

1. Model kütüphanesinde yer alan her bir temel sınıflandırma algoritmasını, eğitim seti üzerinde uygula ve sınıflandırma algoritmalarının doğru sınıflandırma başarımlarını elde et,
2. Temel sınıflandırma algoritmalarının, eğitim setindeki her bir örneği, doğru ya da yanlış sınıflandırmasına göre, sınıflandırıcı-örnek matrisi oluşturarak, belirli bir sınıflandırma algoritması ile doğru sınıflandırılan örnekleri 1 ve yanlış sınıflandırılan örnekleri matriste 0 ile temsil et,
3. Sınıflandırıcı-örnek matrisi üzerinde, K-ortalama++, beklenti maksimizasyonu ve öz-düzenlemeli harita kümeleme algoritmalarının her birini uygula,
4. Kümeleme algoritmalarının sonuçlarına dayalı olarak, nesnelerin birlikte görülme durumlarını temsil edecek bir ortak görülme matrisi oluştur,
5. Ortak görülme matrisi üzerinde, tek bağlantılı kümeleme algoritması uygulayarak küme topluluğunun son kümelemesini elde et,
6. Küme topluluğu ile elde edilen kümelerdeki her bir sınıflandırma algoritmasının birbirleriyle ikili *Q*-istatistiği değerlerini hesapla,
7. Her bir kümeden, *Q*-istatistiğine göre birbirleriyle en farklı olan (en düşük ölçüt değerine sahip) iki sınıflandırma algoritmasını aday sınıflandırma algoritması olarak seç,
8. Her bir kümeden seçilen sınıflandırma algoritmalarını, aday sınıflandırma algoritması havuzuna ekle,
9. Aday sınıflandırma algoritması havuzundan hangi algoritmaların yer aldığı altkümenin seçileceğini ENORA algoritması kullanılarak belirle.
10. ENORA algoritması ile elde edilen altkümeyi son sınıflandırıcı topluluğu olarak döndür.

Algoritma 4.2'de ise geliştirilen sınıflandırıcı topluluğu budama yönteminin temel adımları özetlenmektedir.

Geliştirilen sınıflandırıcı topluluğu budama yönteminde, elde edilen sınıflandırıcı-örnek matrisi üzerinde, kümeleme algoritması uygulanarak, eğitim setindeki örnekleri sınıflandırmada, benzer özellikler gösteren sınıflandırma algoritmalarının aynı kümelerde yer alacak şekilde gruplara atanması amaçlanmaktadır. Sınıflandırıcı topluluğu tasarımındaki en önemli noktalardan birisi çeşitlilik (*diversity*) sağlanmasıdır (Kuncheva and Whitaker, 2003). Kümeleme analizi ile benzer başarıma sahip sınıflandırıcıların aynı kümelerde gruplandırılması ve her bir kümeden bir ya da birkaç temsilci sınıflandırma algoritmasının aday sınıflandırma algoritması olarak seçilmesi ile etkin bir sınıflandırıcı topluluğu tasarımı için gerekli çeşitliliğin sağlanması hedeflenmektedir. Bu doğrultuda, sınıflandırıcı-örnek matrisi üzerinde kümeleme algoritması uygulanarak, örnekler üzerindeki sınıflandırma davranışlarına göre, sınıflandırma algoritmaları kümelere atanır.

Geliştirilen yöntemde, her bir kümeden seçilen sınıflandırma algoritmaları, sınıflandırıcı topluluğunda yer alacak aday sınıflandırıcılar olarak ele alınmaktadır. Aday sınıflandırıcılar içerisinden hangi sınıflandırıcıların son sınıflandırıcı topluluğunda yer alacağının belirlenmesi, *n* aday sınıflandırıcısı

sayısını temsil etmek üzere, 2^n olası altkümenin değerlendirilmesini gerektirmektedir. Bu nedenle, aday sınıflandırıcılardan, son sınıflandırıcı topluluğunda yer alacak yöntemlerin belirlenmesinde arama uzayının metasezgisel yöntemler aracılığıyla değerlendirilmesi uygun düşmektedir.

Geliştirilen sınıflandırıcı topluluğu budama yöntemi, temel olarak kümeleme ve metasezgisel aramaya dayalıdır. Yöntemden, etkin bir sınıflandırıcı topluluğu elde edilebilmesi için, kümeleme aşamasında hangi algoritmanın ya da algoritmaların kullanılacağının belirlenmesi, kümeleme algoritması ile elde edilen kümelerden sınıflandırma algoritmalarının hangi ölçüte ya da kurala göre seçileceğinin belirlenmesi ve arama uzayının hangi metasezgisel arama algoritması kullanılarak inceleneceğinin belirlenmesi gerekmektedir.

Yöntemin, kümeleme aşamasında, K-ortalama, K-ortalama++, beklenti maksimizasyonu ve öz-düzenlemeli harita algoritmaları temel kümeleme algoritmaları olarak kullanılmıştır. Farklı kümeleme algoritmaları ile aynı veri setinde elde edilen bölümlemeler birbirlerinden oldukça farklı olabilmektedir. Ortak kümeleme, kümeleme sonucunun, birden çok kümeleme algoritması ile elde edilen küme topluluğuna dayandırılması ile daha yüksek kümeleme kalitesine sahip bölümlemeler elde edilmesini amaçlamaktadır (Vega-Pons and Ruiz-Shulcloper, 2011). Bu doğrultuda, geliştirilen sınıflandırıcı topluluğu budama yönteminin kümeleme aşamasında, ortak kümeleme yaklaşımının etkinliği değerlendirilmiştir. Değinilen K-ortalama, K-ortalama++, beklenti maksimizasyonu ve öz-düzenlemeli harita algoritmalarının birbirleriyle bir araya getirildikleri tüm olası birleşimlerde, geliştirilen sınıflandırıcı topluluğu budama mimarisinin performansı değerlendirilerek, en yüksek başarımın öz-düzenlemeli harita, beklenti maksimizasyonu ve K-means++ algoritmalarını bir araya getiren küme topluluğu ile elde edildiği görülmüştür. Bu nedenle, geliştirilen yöntemin kümeleme aşamasında, değinilen öz-düzenlemeli harita, beklenti maksimizasyonu ve K-means++ algoritmalarına dayalı ortak kümeleme yöntemi kullanılmıştır.

Ortak kümelemeye dayalı küme topluluğu tasarımında, farklı küme bölümlemeleri bir ortak görüş fonksiyonu aracılığıyla birleştirilmektedir. Geliştirilen yöntemin, kümeleme aşamasında, nesnelerin birlikte görülmesine dayalı bir ortak görüş fonksiyonu kullanılmıştır. Burada, kümeleme algoritmaları ile elde edilen farklı bölümlemelerde nesnelerin birlikte görülme (aynı küme etiketine sahip olma) durumlarını temsil edecek şekilde bir ortak görülme matrisi

oluşturulur. Ortak görülme matrisinin her bir hücresinin değeri, Eşitlik 4.5'e göre hesaplanır (Vega-Pons and Ruiz-Shulcloper, 2011):

$$CA_{ij} = \frac{1}{m}\sum_{t=1}^{m}\delta(P_t(x_i), P_t(x_j)) \tag{4.5}$$

Burada, $P_t(x_i)$, herhangi bir x_i nesnesinin P_t bölümlemesindeki küme etiketi değerini temsil etmektedir. Karşılaştırılmakta olan x_i ve x_j nesnelerinin, P_t bölümlemesindeki küme etiketi değerleri aynı olduğu takdirde $\delta(a,b)=1$ olarak, aksi takdirde ise 0 olarak alınmaktadır. Ortak görülme matrisi, x_i ve x_j nesnelerinin P'deki tüm bölümlemelerde kaç kez aynı kümede yer aldıklarını veren bir temsildir. Bu nedenle, nesneler arası benzerliklere ilişkin bilgi vermektedir. Ardından, ortak görülme matrisi, nesneler arasında bir uzaklık ölçütü olarak kullanılarak, tek bağlantı kümeleme algoritmasının ortak görülme matrisi üzerinde uygulanması ile kümeleme algoritmalarının sonuçlarını birleştiren son bölümleme elde edilmektedir (Fred and Jain, 2005).

Ortak kümelemeye dayalı küme topluluğu ile son bölümlemenin elde edilmesinin ardından, her bir kümeden aday sınıflandırma algoritmalarının seçilmesi için yedi farklı seçim kuralı denenmiştir. İlk incelenen iki kural, kümeden eleman (sınıflandırma algoritması) seçiminin rastgele yapılması ve her bir kümeden bir ya da iki elemanın alınmasıdır. Üçüncü eleman seçim kuralında, her bir kümeden, eğitim setindeki doğru sınıflandırma başarımlarına göre, en yüksek başarıma sahip ikişer eleman alınmıştır. Dördüncü eleman seçim kuralı, kümeden elemanların, ölçekli ağırlığa göre seçilmesidir. Burada, öncelikle, kümede yer alan her bir sınıflandırma algoritmasının, doğru sınıflandırma başarımlarına göre, sınıflandırma algoritmalarına [0-1] aralığında uniform ağırlık değerleri atanır. Ardından, yüksek sınıflandırma başarımına sahip olan sınıflandırma algoritmaları, daha yüksek seçilme şansına sahip olacak şekilde seçim gerçekleştirilir.

Geriye kalan seçim kurallarında, üç farklı ikili çeşitlilik ölçütü (Q-istatistiği, uyuşmazlık ölçütü ve çift hata ölçütü) aracılığıyla kümeden eleman seçimi gerçekleştirilmiştir. Öncelikle belirtilen çeşitlilik ölçütleri kullanılarak, her bir küme için, yer alan kümeleme algoritmaları arasındaki ikili çeşitlilik değerleri hesaplanmış ve her bir kümeden birbirinden en farklı başarıma sahip iki sınıflandırma algoritması aday sınıflandırma algoritmaları olarak seçilmiştir.

Q-istatistiği (*Q-statistics*), herhangi iki sınıflandırma algoritması D_i ve D_k arasındaki çeşitliliği, Eşitlik 4.6'ya göre hesaplar (Kuncheva and Whitaker, 2003):

$$Q_{i,k} = \frac{n^{11}n^{00} - n^{01}n^{10}}{n^{11}n^{00} + n^{01}n^{10}} \tag{4.6}$$

Burada, n^{11}, her iki sınıflandırıcının da doğru sınıflandırdığı nesne sayısını, n^{00}, her iki sınıflandırıcının da yanlış sınıflandırdığı nesne sayısını, n^{10}, D_i sınıflandırıcısının doğru sınıflandırıp diğer sınıflandırıcının yanlış sınıflandırdığı nesne sayısını ve n^{01}, D_k sınıflandırıcısının doğru sınıflandırıp diğer sınıflandırıcının yanlış sınıflandırdığı nesne sayısını temsil etmektedir. *Q*-istatistiği değeri, [-1, 1] aralığında değerler almaktadır ve negatif değerler alması, karşılaştırılmakta olan sınıflandırıcıların, farklı nesneleri sınıflandırmada hata yaptıklarını göstermektedir.

Uyuşmazlık ölçütü (*disagreement measure*, *Dis*), karşılaştırılmakta olan sınıflandırıcılardan birisinin doğru, diğerinin yanlış sınıflandırdığı nesne sayısının, toplam nesne sayısına bölünmesi ile Eşitlik 4.7'ye göre hesaplanmaktadır (Kuncheva and Whitaker, 2003):

$$Dis_{i,k} = \frac{n^{01} + n^{10}}{n^{11} + n^{10} + n^{01} + n^{00}} \tag{4.7}$$

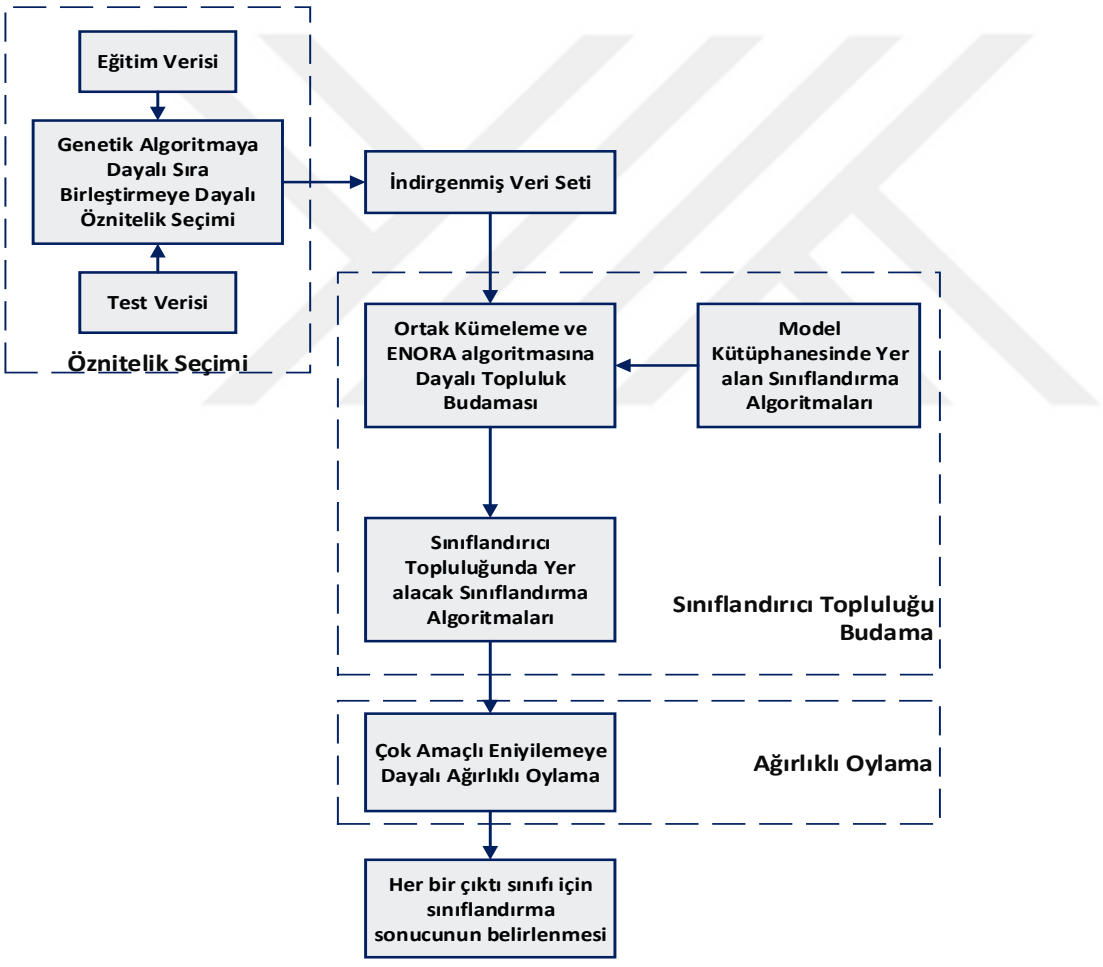
Çift hata ölçütü (*double-fault measure*, *DF*), her iki sınıflandırıcının da yanlış sınıflandırdığı nesne sayısının, toplam nesne sayısına bölünmesi ile elde edilmektedir. Eşitlik 4.8'de çift hata ölçütü verilmiştir (Kuncheva and Whitaker, 2003):

$$DF_{i,k} = \frac{n^{00}}{n^{11} + n^{10} + n^{01} + n^{00}} \tag{4.8}$$

Geliştirilen sınıflandırıcı topluluğu budama yönteminin, son aşamasında, kümeleme algoritmaları ile elde edilen aday sınıflandırma algoritmaları ile elde edilen arama uzayı, metasezgisel arama algoritmaları aracılığıyla değerlendirilmektedir. Geliştirilen yöntemde, hangi arama algoritmasının kullanılması gerektiğine yönelik deneysel çalışmalarda, en iyi önce arama algoritması, genetik algoritma, açgözlü arama algoritması, parçacık sürü eniyilemesi algoritması ve ENORA algoritması değerlendirilmiştir. Yapılan deneysel çalışmalarda, en yüksek başarım, ENORA algoritması ile elde edildiği için, geliştirilen yöntemde bu algoritma kullanılmıştır. ENORA algoritmasında, çoğunluk oylaması hatası (*majority voting error*) ve *Q*-istatistiği, amaç fonksiyonu olarak kullanılmıştır.

4.4 Geliştirilen Görüş Sınıflandırma Mimarisi

Bu bölümde, tez kapsamında geliştirilen yöntemleri bütünleştiren, görüş sınıflandırma mimarisinin genel yapısı sunulmaktadır. Geliştirilen sınıflandırma mimarisi, genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçim yöntemi, geliştirilen çok amaçlı diferansiyel gelişim algoritmasına dayalı sınıflandırıcı topluluğu birleştirme kuralı ve geliştirilen sınıflandırıcı topluluğu budama yöntemini birarada kullanarak, görüş sınıflandırma alanında, yüksek başarıma sahip etkin bir sınıflandırma modeli elde edilmesini amaçlamaktadır. Geliştirilen sınıflandırıcı topluluklarına dayalı mimarinin genel yapısı Şekil 4.5'te sunulmaktadır.



Şekil 4.5. Geliştirilen görüş sınıflandırma mimarisinin genel yapısı

Geliştirilen görüş sınıflandırma mimarisinde, öncelikle, veri setinin uygun bir öznitelik altkümesi ile temsil edilebilmesine yönelik olarak, öznitelik seçimi işlemi uygulanmaktadır. Öznitelik aşamasında, yedi farklı bireysel filtre tabanlı

öznitelik seçim yöntemi (bilgi kazancı, ki-kare ölçütü, kazanç oranı ölçütü, simetrik belirsizlik katsayısı, Pearson korelasyon katsayısı, ReliefF algoritması ve olasılıklı önem ölçütü), daha yüksek başarıma sahip bir öznitelik altkümesi elde etmek amacıyla birleştirilmiştir. Farklı öznitelik seçim yöntemleri ile elde edilen sıralamaların birleştirilmesinde genetik algoritma kullanılmıştır. Genetik algoritma ile sıra birleştirme aşamasına ilişkin ayrıntılı bilgi Bölüm 4.1'de sunulmuştur. Öznitelik seçimi aşamasının sonunda, veri setinde yüksek bilgi vericiliğe sahip öznitelikler tutulurken, diğer öznitelikler ortadan kaldırılarak, indirgenmiş veri seti elde edilmektedir. Ardından, sınıflandırıcı topluluğunda yer alacak temel öğrenme algoritmalarının belirlenmesi amacıyla, indirgenmiş veri seti üzerinde, melez sınıflandırıcı topluluğu budama aşaması uygulanmaktadır. Topluluk budama aşamasında, model kütüphanesinde yer alan temel öğrenme algoritmalarından hangilerinin, sınıflandırıcı topluluğu altkümesinde yer alacağının belirlenmesi için öncelikle model kütüphanesinde bulunan her bir öğrenme algoritması, eğitim seti üzerinde uygulanmaktadır. Böylelikle, her bir öğrenme algoritmasının, eğitim setindeki örnekleri sınıflandırmalarına ilişkin temel karakteristikleri elde edilmektedir. Ardından, sınıflandırma algoritmalarından elde edilen başarım matrisi üzerinde, ortak kümelemeye dayalı olarak sınıflandırma algoritmaları, benzer başarım gösteren algoritmalar aynı kümede yer alacak şekilde gruplandırılmaktadır. Kümelerden aday sınıflandırma algoritmaları seçilmekte ve aday sınıflandırma algoritmaları ENORA algoritması ile değerlendirilerek, sınıflandırıcı topluluğunda yer alacak sınıflandırma algoritmaları belirlenmektedir. Sınıflandırıcı topluluğu budama aşamasına ilişkin ayrıntılı bilgi Bölüm 4.3'te aktarılmaktadır.

Geliştirilen görüş sınıflandırma mimarisinin ağırlıklı oylama aşamasında ise, sınıflandırıcı topluluğunda yer alan temel öğrenme algoritmalarının, sınıflandırıcı topluluğunun çıktısında, başarımları oranında etkili olabilmeleri amacıyla, çok amaçlı diferansiyel gelişim algoritmasına dayalı ağırlıklı oylama birleştirme kuralı kullanılarak, ağırlık değerleri eniyilenmektedir. Ağırlıklı oylama aşamasına ilişkin ayrıntılı bilgi Bölüm 4.2'de sunulmaktadır.

5. DENEYSEL SONUÇLAR

Bu bölümde, tez çalışması kapsamında geliştirilen yöntemler ile elde edilen deneysel sonuçlar sunulmaktadır.

5.1 Veri Setleri

Görüş sınıflandırmaya ilişkin yöntemlerin sınanması için dokuz farklı alana ait veri setleri kullanılmıştır. Bu veri setleri, "Camera", "Camp", "Doctor", "Drug", "Laptop", "Lawyer", "Music", "Radio" ve "TV" veri setleridir. Bu veri setleri, ilk olarak Whitehead and Yaeger (2009) tarafından geliştirdikleri görüş sınıflandırma yönteminin farklı alanlardaki başarımlarını incelemek amacıyla kullanılmıştır. Veri setleri, ilgili çalışmanın başlıca yazarına ilişkin web sitesinden indirilebilmektedir (Whitehead, 2014). "Camera" veri seti, Amazon.com sitesinden elde edilen dijital kamera değerlendirmelerini içermektedir. Veri seti oluşturulurken, çok sayıda değerlendirme içerenler seçilmiştir. "Camp" veri seti, CampRatingz.com sitesinden elde edilen yaz kampı değerlendirmelerinden oluşmaktadır. Değerlendirmelerin önemli bir bölümü, yaz kampına katılan gençlerin değerlendirmelerinden oluşmaktadır. "Doctor" veri seti, RateMDs.com sitesinden elde edilen doktor değerlendirmelerini içermektedir. Çalışmada kullanılan "Doctor" veri seti ile "Lawyer" veri seti, insan değerlendirmeleri örnekleridir. "Drug" veri seti, DrugrRantingz.com sitesinden elde edilen ilaç değerlendirmelerini, "Laptop" veri seti ise Amazon.com sitesinden elde edilen farklı üreticilere ait diz üstü bilgisayar değerlendirmelerini içermektedir. "Lawyer" veri seti, LawyerRatingz.com sitesinden alınan avukat değerlendirmelerini, "Music" veri seti, Amazon.com sitesinden elde edilen müzik CD'si değerlendirmelerini, "Radio" veri seti, RadioRatingz.com sitesinden elde edilen radyo programlarına ilişkin kısa değerlendirmeleri, "TV" veri seti ise TVRatingz.com sitesinden elde edilen TV programı değerlendirmelerini içermektedir. Veri setleri arasında en kısa ve ayrıntısız değerlendirmeleri içeren "TV" veri setidir. Çizelge 5.1'de çalışmada kullanılan veri setlerinin pozitif ve negatif görüş kutbu değerleri ve veri setleri 1-gram ve 2-gram ile temsil edildiğinde elde edilen öznitelik sayıları özetlenmiştir. Deneysel sonuçlar ile veri setlerinde en yüksek başarım, terim sıklığı ve 1-gram temsili ile elde edildiği için, çalışmada veri setleri değinilen öznitelik temsili aracılığıyla temsil edilmiştir (Onan and Korukoğlu, 2015b).

Çizelge 5.1. Görüş sınıflandırma veri setleri (Whitehead, 2014)

Veri Seti	Pozitif Görüş Kutbu Sayısı	Negatif Görüş Kutbu Sayısı	Öznitelik Sayısı (1-gram)	Öznitelik Sayısı (2-gram)
Camera	250	248	1352	1704
Camp	402	402	2045	1735
Doctor	739	739	1578	1679
Drug	401	401	1438	1662
Laptop	88	88	2010	3136
Lawyer	110	110	2474	7734
Music	291	291	1398	1705
Radio	502	502	1923	3054
TV	235	235	2834	9195

5.2 Değerlendirme Ölçütleri

Sınıflandırma algoritmalarının başarımlarının ölçümlenmesi için sınıflandırıcıların değerlendirilmesinde sıklıkla kullanılan doğru sınıflandırma oranı kullanılmıştır. Doğru sınıflandırma oranı (ACC), belirli bir ikili sınıflandırma yönteminin bir şartı hangi oranda doğru olarak saptadığını belirleyen istatistiksel bir ölçüttür. Doğru sınıflandırma oranı, doğru pozitifler ve doğru negatiflerin toplamının, doğru pozitif, yanlış pozitif, yanlış negatif ve doğru negatifler toplamına oranlanması ile Eşitlik 5.1'e göre hesaplanmaktadır:

$$ACC = \frac{TN + TP}{TP + FP + FN + TN} \tag{5.1}$$

Burada, *TN, TP, FP* ve *FN* sırası ile doğru negatif, doğru pozitif, yanlış pozitif ve yanlış negatif sayılarını temsil etmektedir.

5.3 Öznitelik Seçimi Deneysel Sonuçları

Bu bölümde, geliştirilen genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçim yöntemi ve temel filtre tabanlı öznitelik seçim yöntemlerinin sınanmasında kullanılan görüş madenciliği veri setleri, izlenilen deneysel süreç, kullanılan değerlendirme ölçütleri ile elde edilen deneysel sonuçlar sunulmaktadır.

5.3.1 Öznitelik seçimi deneysel süreci

Bu bölümde sunulan deneysel çalışmada, görüş sınıflandırma veri setlerinin makine öğrenmesi yöntemleri üzerinde başarımlarının incelenmesinde, 10-kat çapraz geçerleme yöntemi kullanılmıştır. 10-kat çapraz geçerleme yöntemi ile

orijinal veri seti rastgele olarak 10 eşit parçaya ayrılarak, her bir yinelemede bu parçalardan biri modeli sınamak için kullanılırken geriye kalan dokuz parça eğitim amacıyla kullanılmıştır. Süreç, her bir parça birer kez sınama verisi olarak kullanılacak biçimde on kez yinelenmiştir. Deneysel sonuçlar bölümünde listelenen sonuçlar, 10-kat çapraz geçerleme sürecinden elde edilen ortalama değerleri belirtmektedir. Deneylerde, filtre tabanlı öznitelik seçim yöntemlerinin ve sınıflandırma algoritması sonuçlarının alınmasında, WEKA (Waikato Environment for Knowledge Analysis) 3.7.11 kullanılmıştır (Witten et al., 2011). Geliştirilen genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçim yönteminin gerçekleştirimi, Java dilinde yapılmıştır. Geliştirilen sıra birleştirmeye dayalı öznitelik seçim yönteminin, diğer sıra birleştirme yöntemleri ile karşılaştırılabilmesi için, daha önceki çalışmalarda sıklıkla kullanılan dokuz temel sıra birleştirme yöntemleri kullanılmıştır (Dittman et al., 2013):

- **Ortalama sıra birleştirme (*mean, MA*):** Özniteliğin, birleştirilen listelerdeki sıralaması göz önünde bulundurularak, ortalama sıra değeri hesaplanır ve ortalama değeri, ilgili özniteliğin sıralaması olarak kullanılır.

- **Medyan sıra birleştirme (*median, MEA*):** Özniteliğin, birleştirilen listelerdeki değerlerinin medyanı, ilgili özniteliğin sıralaması olarak kullanılır.

- **En yüksek sıra birleştirme (*highest rank, HRA*):** Özniteliğin, birleştirilen listeler içerisindeki en iyi sırası (en düşük sıra değeri), ilgili özniteliğin sıralaması olarak alınır.

- **En düşük sıra birleştirme (*lowest rank, LRA*):** Özniteliğin, birleştirilen listeler içerisindeki en kötü sırası (en yüksek sıra değeri), ilgili özniteliğin sıralaması olarak alınır.

- **Kararlılık seçimi (*stability selection, SSA*):** Kararlılık seçimi yönteminde, bir eşik değeri seçilerek, ilgili özniteliğin, her bir birleştirilen listedeki sıralamasının eşik değeri geçip geçmediği belirlenir. Buna göre, her bir özniteliğe sıralama değeri atanır (Haury et al., 2011).

- **Üstel ağırlıklı sıra birleştirme (*exponential weighting, EWA*):** Kararlılık seçimi yöntemine benzer. Burada, özniteliklere sıralama atanması, $e^{-r/s}$ üstel formülüne göre belirlenir. Üstel formülde, r, ilgili özniteliğin sıralaması ve s, eşik değerini temsil etmektedir.

- **İyileştirilmiş Borda sıra birleştirme (*enhanced Borda, EBA*):** Her bir özniteliğin sıralaması, ilgili özniteliğin Borda sayımı (*Borda count*) değerinin, ilgili özniteliğin kararlılık seçimi ile elde edilen değeri ile çarpımı ile belirlenir (Haury et al., 2011).

- **Round Robin sıra birleştirme (*Round Robin, RRA*):** Bu yöntemde, öncelikle birleştirilecek olan listelere rastgele sıra değerleri atanır. Ardından ilk listenin birinci sırasında yer alan öznitelikten başlanarak, her bir seferinde bir sonraki sıraya sahip listenin birinci sırasında yer alan öznitelik birleştirilmiş listeye eklenir. Süreç, ilk listeden başlanarak, tüm listelerin diğer sıralardaki öznitelikleri için son listeye eklenecek öznitelik kalmayıncaya dek yinelenir.

- **Gürbüz sıra birleştirme (*robust rank aggregation, RORA*):** Gürbüz sıra birleştirme yönteminde, özniteliklere istatistiksel önem derecelerine karşılık olarak birer değer atanarak, birleştirilmiş listede istatistiksel olarak birbirleriyle ilişkili özniteliklerin tutulması amaçlanır (Haury et al., 2011).

Geliştirilen genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçim yönteminin (GA) başarımı, daha önce değinilen temel filtre tabanlı öznitelik seçim yöntemleri (bilgi kazancı, ki-kare ölçütü, kazanç oranı ölçütü, simetrik belirsizlik katsayısı, Pearson korelasyon katsayısı, Relief-F algoritması ve olasılıklı önem ölçütü) ve sıra birleştirme yöntemleri (ortalama sıra birleştirme, medyan sıra birleştirme, en yüksek sıra birleştirme, en düşük sıra birleştirme, kararlılık seçimi, üstel ağırlıklı sıra birleştirme, iyileştirilmiş Borda sıra birleştirme ve Round Robin sıra birleştirme) ile karşılaştırmalı olarak sunulmaktadır. Sıra birleştirme yöntemlerinin her birinde yedi filtre tabanlı öznitelik seçim yöntemi ile elde edilen listeler birleştirilmiştir. Yöntemlerin değerlendirilmesinde Naive Bayes (NB) ve K-en yakın komşu (KNN) sınıflandırıcıları kullanılmıştır. Öznitelik sıralama yöntemlerinde en yüksek ölçüt değeri elde eden kaç tane özniteliğin öznitelik altkümesi olarak kabul edileceğinin belirlenmesi gerekmektedir. Bu konuda en çok kullanılan yöntemlerden birisi öznitelik altkümesi eleman sayısının, toplam öznitelik sayısına oransal olarak belirlenmesidir. Bu nedenle, deneysel çalışmada, toplam öznitelik sayılarına ilişkin farklı oransal değerler (%10-%90 aralığı) için öznitelikler değerlendirilmiş, en yüksek başarım özniteliklerin %60'ı tutulduğunda elde edilmiş ve bu sonuçlar listelenmiştir.

5.3.2 Öznitelik seçimi deneysel sonuçları ve tartışma

Deneysel sonuçlar, görüş madenciliğinde sıklıkla kullanılan dokuz veri seti üzerinde alınmıştır. Çizelge 5.2'de temel filtre tabanlı öznitelik seçim yöntemleri ve sıra birleştirmeye dayalı öznitelik seçim yöntemleri ile Naive Bayes algoritması (NB) kullanıldığında elde edilen doğru sınıflandırma oran ortalamaları ve ortalama standart hataları sunulmuştur. Çizelge 5.3'te ise öznitelik seçim yöntemleri ile K-en yakın komşu algoritması (KNN) ile elde edilen doğru sınıflandırma oran ortalamaları ve ortalama standart hataları listelenmektedir. Çizelgelerde, belirli bir sınıflandırma algoritmasında elde edilen en yüksek değer koyu ve altı çizili, ikinci en yüksek değer ise koyu ve italik kullanılarak belirtilmiştir. Bu bölümde, öznitelik altkümesi eleman sayısı olarak, toplam öznitelik sayısının, en yüksek bilgi vericiliğe sahip %60'ı veri setinde tutulmuştur.

Çizelge 5.2 ve Çizelge 5.3'te sunulan doğru sınıflandırma oranı değerlerine göre, geliştirilen genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçim yöntemi (GA), temel filtre tabanlı ve sıra birleştirmeye dayalı öznitelik seçim yöntemlerinden daha yüksek başarım göstermektedir. Naive Bayes algoritması ile en yüksek doğru sınıflandırma oranı (%97.39), "Laptop" veri seti üzerinde, genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçim yöntemi kullanıldığında elde edilmektedir. "Camp", "Laptop", "Lawyer" ve "TV" veri setleri için, Naive Bayes algoritması ile en yüksek ikinci doğru sınıflandırma oranları, en yüksek sıra birleştirme yöntemi (HRA) ile elde edilmiştir. "Camera" veri seti için, Naive Bayes algoritması ile en yüksek ikinci doğru sınıflandırma oranı, en düşük sıra birleştirme yöntemi (LRA) ile elde edilmiştir. "Music" veri seti için, Naive Bayes algoritması ile en yüksek ikinci doğru sınıflandırma oranı, ortalama sıra birleştirme yöntemi (MA) ile alınmıştır. "Drug" ve "Radio" veri setlerinde ise en yüksek ikinci doğru sınıflandırma oranları, temel filtre tabanlı öznitelik seçim yöntemi olan, Pearson korelasyon katsayısına dayalı öznitelik seçimi ile elde edilmiştir. "Doctor" veri setinde ise en yüksek ikinci doğru sınıflandırma oranı, temel filtre tabanlı öznitelik seçim yöntemi olan bilgi kazancına dayalı öznitelik seçimi ile elde edilmiştir.

Çizelge 5.2. Naive Bayes algoritması ile elde edilen doğru sınıflandırma oran ortalamaları ve standart hataları

Yöntem	Camera	Camp	Doctor	Drug	Laptop	Lawyer	Music	Radio	TV	Ortalama
IG	86.78±0.72	90.90±0.63	***91.24±0.92***	78.16±0.98	90.43±0.45	84.74±0.50	81.61±0.67	85.04±0.59	77.44±0.57	85.15±0.57
CHI	86.54±0.56	90.20±0.69	89.99±0.71	79.52±0.81	90.06±0.66	86.17±0.47	82.99±0.65	84.42±0.69	78.07±0.57	85.33±0.50
PCC	88.97±0.51	88.57±0.71	89.32±0.83	***90.20±0.73***	89.01±0.91	89.39±0.57	90.33±0.67	***90.76±0.58***	88.46±0.82	89.45±0.69
GAIN	85.58±0.49	91.62±0.87	88.78±0.47	77.31±0.88	89.82±0.47	86.33±0.84	81.92±0.37	83.14±0.54	77.87±0.52	84.71±0.90
RL	79.83±0.65	85.24±0.71	87.29±0.74	71.83±0.56	85.36±0.67	82.55±0.50	77.72±0.70	78.47±0.55	71.10±0.56	79.93±0.61
PS	86.53±0.61	92.49±0.40	89.08±0.58	78.83±0.63	89.72±0.80	86.72±0.39	82.25±0.63	83.63±0.67	77.81±0.52	85.23±0.53
GA	**93.23±0.52**	**95.41±0.57**	**94.12±0.66**	**91.21±0.49**	**97.38±0.10**	**97.39±0.26**	**90.62±0.54**	**91.57±0.37**	**96.47±0.47**	**94.16±0.52**
SU	87.40±0.64	92.37±0.71	90.20±0.79	78.83±0.64	89.52±0.79	84.97±0.44	82.25±0.59	83.52±0.59	79.98±0.64	85.45±0.33
MA	90.11±0.69	90.16±0.62	90.16±0.57	89.72±0.53	90.02±0.62	89.92±0.55	***90.41±0.73***	89.93±0.49	90.22±0.62	90.07±0.66
MEA	88.65±0.66	89.03±0.43	88.88±0.82	88.03±0.46	88.28±0.62	89.37±0.56	89.79±0.76	89.06±0.51	89.43±0.78	88.95±0.59
HRA	87.94±0.57	***94.44±0.34***	90.40±0.53	83.55±0.65	***96.84±0.19***	***96.92±0.30***	86.24±0.50	87.92±0.75	***91.02±0.49***	***90.59±0.66***
LRA	***90.49±0.63***	89.44±0.61	89.69±0.39	89.94±0.53	89.39±0.96	90.99±0.66	89.65±0.91	90.55±0.93	90.27±0.61	90.05±0.61
SSA	89.61±0.71	88.88±0.47	89.16±0.72	87.92±0.45	88.75±1.01	89.68±0.59	88.43±0.72	88.25±0.73	90.12±0.85	88.98±0.62
EWA	90.19±0.40	90.75±0.50	90.78±0.78	89.37±0.64	89.42±0.95	90.90±0.47	89.77±0.81	89.89±0.89	89.38±0.66	90.05±0.64
EBA	89.78±0.64	88.34±0.84	91.11±0.48	89.87±0.61	89.72±0.72	87.90±0.56	89.13±0.74	89.44±0.79	89.56±0.58	89.43±0.64
RRA	88.23±0.57	88.59±0.75	88.18±0.72	88.73±0.83	88.70±0.72	89.17±0.52	88.33±0.52	89.20±0.64	88.81±0.70	88.66±0.61
RORA	87.33±0.53	87.84±0.71	87.61±0.43	87.48±0.64	87.56±0.53	86.97±0.57	87.83±0.38	87.23±0.56	87.83±0.67	87.52±0.63

Çizelge 5.3. KNN ile elde edilen doğru sınıflandırma oran ortalamaları ve standart hataları

Yöntem	Camera	Camp	Doctor	Drug	Laptop	Lawyer	Music	Radio	TV	Ortalama
IG	67.26±0.60	70.91±0.45	68.37±0.42	55.61±0.72	54.12±0.99	61.23±0.68	54.70±0.52	64.76±0.60	61.46±0.68	62.05±0.66
CHI	67.52±0.58	68.62±0.78	67.85±0.44	54.77±0.63	55.86±0.57	61.74±0.69	53.36±0.73	64.25±0.73	62.75±0.48	61.86±0.62
PCC	60.60±0.49	69.63±0.72	66.54±0.51	55.73±0.83	57.42±0.49	72.36±0.60	53.55±0.62	59.76±0.74	64.84±0.62	62.27±1.44
GAIN	67.24±0.86	69.17±0.51	68.56±0.61	54.54±0.49	53.98±0.51	60.59±0.60	53.03±0.52	63.24±0.69	61.86±0.68	61.36±1.35
RL	60.72±0.63	59.73±0.59	68.08±0.63	53.84±0.64	55.01±0.77	59.31±0.86	56.72±0.34	59.40±0.57	58.08±0.54	58.99±0.46
PS	67.55±0.43	70.28±0.64	67.11±0.69	55.52±0.59	55.09±0.35	62.19±0.53	53.34±0.41	64.41±0.59	61.06±0.54	61.84±0.63
GA	**91.59±0.88**	**92.20±0.48**	**89.98±0.47**	**87.57±0.42**	**87.86±0.47**	**97.39±0.26**	**90.69±1.02**	**90.43±0.43**	**94.27±0.63**	**91.33±0.63**
SU	66.05±0.24	69.64±0.52	67.42±0.54	54.86±0.35	54.14±0.44	59.75±0.80	53.57±0.66	64.13±0.42	62.57±0.69	61.35±0.31
MA	79.04±0.65	77.21±0.77	77.75±0.35	78.18±0.67	76.93±0.54	77.01±0.44	77.70±0.83	78.29±0.68	78.09±0.62	77.80±0.92
MEA	76.23±0.76	77.02±0.60	77.10±0.50	76.63±0.62	77.32±0.79	76.49±0.91	77.14±0.75	76.73±0.71	76.98±0.76	76.85±0.94
HRA	***86.70±0.55***	85.74±0.75	***86.39±0.82***	86.39±0.60	86.44±0.29	***86.80±0.88***	85.66±0.70	85.47±0.49	85.13±0.61	86.08±1.14
LRA	76.43±0.63	77.39±0.61	74.74±0.43	77.90±0.82	76.68±0.58	76.88±0.73	76.44±0.49	77.69±0.43	76.14±0.56	76.70±0.93
SSA	77.14±0.82	77.66±0.61	78.76±0.73	76.68±0.69	77.96±0.49	76.89±0.62	78.04±0.49	78.84±0.55	78.01±0.74	77.78±0.95
EWA	75.51±0.54	76.90±0.49	76.50±0.27	76.35±0.73	77.20±0.58	76.66±0.79	76.18±0.70	77.19±0.71	76.28±0.21	76.53±1.01
EBA	77.52±0.39	76.17±0.78	76.84±0.59	76.46±0.54	75.62±0.90	76.13±0.57	77.21±0.66	76.14±0.53	76.62±0.50	76.52±0.97
RRA	61.56±0.49	62.00±0.63	61.25±0.54	61.27±0.54	61.25±0.76	59.99±0.54	60.76±0.80	61.63±0.77	61.16±0.22	61.21±0.93
RORA	86.29±0.72	***85.85±0.42***	84.89±0.66	***86.46±0.62***	***87.35±0.31***	86.26±0.64	***86.30±0.77***	***85.82±0.64***	***85.95±0.57***	***86.13±0.95***

K-en yakın komşu algoritması ile en yüksek doğru sınıflandırma oranı (%97.39), “Lawyer” veri seti üzerinde, geliştirilen yöntem (GA) ile alınmaktadır. En yüksek ikinci doğru sınıflandırma oranları ise, “Camera”, “Doctor”, “Lawyer” veri setlerinde en yüksek sıra birleştirme yöntemi (HRA) ile, “Camp”, “Drug”, “Laptop”, “Music”, “Radio” ve “TV” veri setlerinde, gürbüz sıra birleştirme yöntemi (RORA) ile alınmıştır. K-en yakın komşu algoritması ile elde edilen sonuçlar incelendiğinde, sıra birleştirmeye dayalı öznitelik seçim yöntemlerinin, temel filtre tabanlı öznitelik seçim yöntemlerine kıyasla daha yüksek başarım gösterdikleri görülmektedir.

Geliştirilen genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçimi, temel filtre tabanlı öznitelik seçim yöntemleri ve korelasyon tabanlı öznitelik seçim yöntemi (CBS) (öznitelik altkümeleri en iyi önce arama algoritması ile değerlendirildiğinde) ile öznitelik seçim sürecinde geçen süre bakımından karşılaştırılarak, sonuçlar Çizelge 5.4’te saniye olarak verilmiştir.

Çizelge 5.4. Öznitelik seçim yöntemlerinin çalışma süresi bakımından karşılaştırılması

Yöntem	Camera	Camp	Doctor	Drug	Laptop	Lawyer	Music	Radio	TV
IG	0.40	0.89	***1.39***	***0.64***	***0.22***	***0.25***	***0.50***	1.10	0.76
CHI	***0.35***	***0.73***	1.43	0.66	***0.22***	***0.25***	0.53	***0.97***	***0.58***
PCC	**0.05**	**0.10**	**0.16**	**0.09**	**0.03**	**0.03**	**0.07**	**0.11**	**0.07**
GAIN	0.48	1.10	1.83	0.80	0.26	0.30	0.63	1.17	0.66
RL	2.20	4.81	20.52	9.05	1.00	0.97	7.35	7.80	2.47
PS	1.39	2.87	5.59	2.02	0.56	0.89	1.73	2.75	1.82
SU	1.65	3.70	5.85	3.37	0.67	1.21	2.46	3.96	2.35
CBS	73.78	64.88	90.98	70.46	59.11	59.46	68.14	72.73	64.56
GA	54.53	62.17	84.96	64.56	50.11	51.24	61.32	66.41	56.90

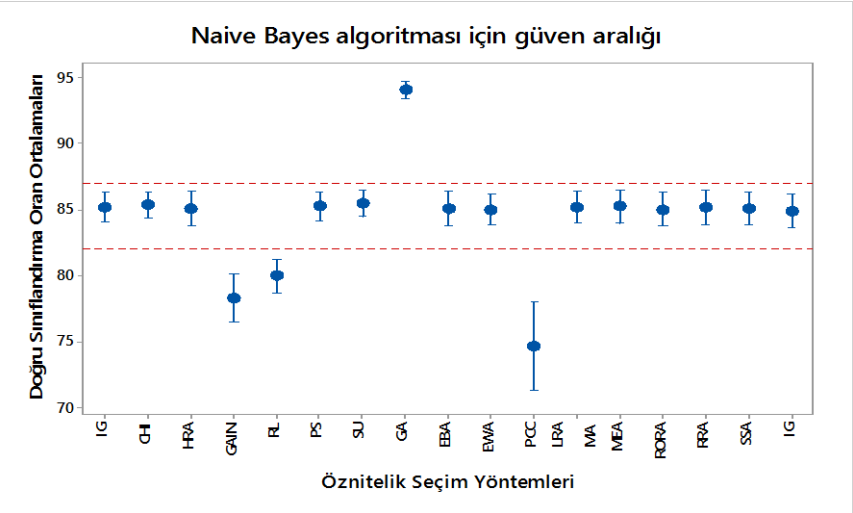
Çizelge 5.4’te sunulan sonuçlar incelendiğinde, çalışma süresi bakımından en etkin sonuçların Pearson korelasyon katsayısı (PCC) yöntemi ile, ikinci en hızlı sonuçların ki-kare ölçütü (CHI) ya da bilgi kazancı ölçütü (IG) ile elde edildiği görülmektedir. Bireysel filtre tabanlı öznitelik seçim yöntemleri, çalışma süreleri bakımından, tutarlılık tabanlı öznitelik seçim yöntemi (CBS) ve geliştirilen genetik algoritma (GA) ile sıra birleştirmeye dayalı öznitelik seçim yöntemine kıyasla daha etkindir. Öznitelik seçiminde, tutarlılık tabanlı öznitelik seçim yöntemi, korelasyon tabanlı öznitelik seçimi gibi filtre tabanlı grup öznitelik seçim yöntemleri sıklıkla kullanılmakla birlikte, öznitelik altkümesinin yararlılığının değerlendirilmesinde belirli bir sezgisel algoritmanın çalışmasını gerektirdiğinden, daha yavaş yöntemlerdir. Benzer şekilde, geliştirilen öznitelik seçim yöntemi, öncelikle bireysel filtre tabanlı öznitelik seçim yöntemleri ile

bireysel sıralamaların elde edilmesini ve bireysel sıralamaların genetik algoritmaya dayalı sıra birleştirme yöntemi ile birleştirilmesini gerektirdiğinden, çalışma süresi bakımından daha yavaş bir yöntemdir. Geliştirilen yöntem, filtre tabanlı grup öznitelik seçim yönteminden (CBS) daha düşük çalışma süresine sahip ve elde edilen öznitelik altkümesinin tahminleme başarımı bakımından daha etkin bir yöntemdir.

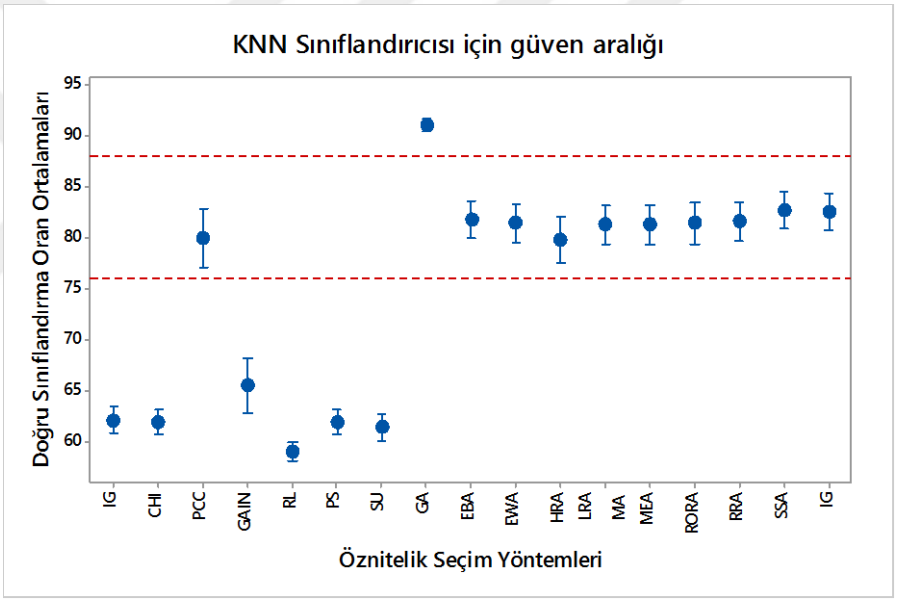
Bu bölümde sunulan öznitelik seçim yöntemlerine ilişkin sonuçların istatistiksel analizi, Minitab istatistik paket programında gerçekleştirilmiştir. Deneysel sonuçlar, iki-yönlü varyans analizi (*two-way ANOVA test*) kullanılarak incelenmiştir. Çizelge 5.5'te öznitelik seçim yöntemleri ile Naive Bayes ve K-en yakın komşu sınıflandırıcıları ile elde edilen sonuçlara ilişkin analiz sonuçları sunulmaktadır. Burada, DF, SS, MS, F ve P, sırasıyla, serbestlik derecesi (*degrees of freedom*), uyarlanmış kareler toplamı (*adjusted sum of squares*), uyarlanmış kareler ortalaması (*adjusted mean square*), F-değeri (*F-value*) ve olasılık değerini temsil etmektedir. Çizelge 5.5'te sunulan iki-yönlü varyans analizi sonuçlarına göre, karşılaştırılan öznitelik seçim yöntemlerinin başarımları arasında, istatistiksel olarak anlamlı bir farklılık bulunmaktadır ($p<0.0001$).

Çizelge 5.5. Öznitelik seçimi iki-yönlü varyans analizi sonuçları

Naive Bayes algoritması					
Kaynak	**DF**	**SS**	**MS**	**F**	**P**
Veri Seti	8	20708.1	2588.51	215.58	0.000
Öznitelik Seçim Yöntemi	16	22816.9	1426.06	118.77	0.000
Veri Seti*Öznitelik Seçim Yöntemi	128	33531.6	261.97	21.82	0.000
Hata	1377	16534.1	12.01		
Toplam	1529	93590.7			
K-en yakın komşu algoritması					
Kaynak	**DF**	**SS**	**MS**	**F**	**P**
Veri Seti	8	38324	4790.46	296.11	0.000
Öznitelik Seçim Yöntemi	16	153663	9603.93	593.65	0.000
Veri Seti*Öznitelik Seçim Yöntemi	128	54284	424.10	26.21	0.000
Hata	1377	22277	16.18		
Toplam	1529	268548			



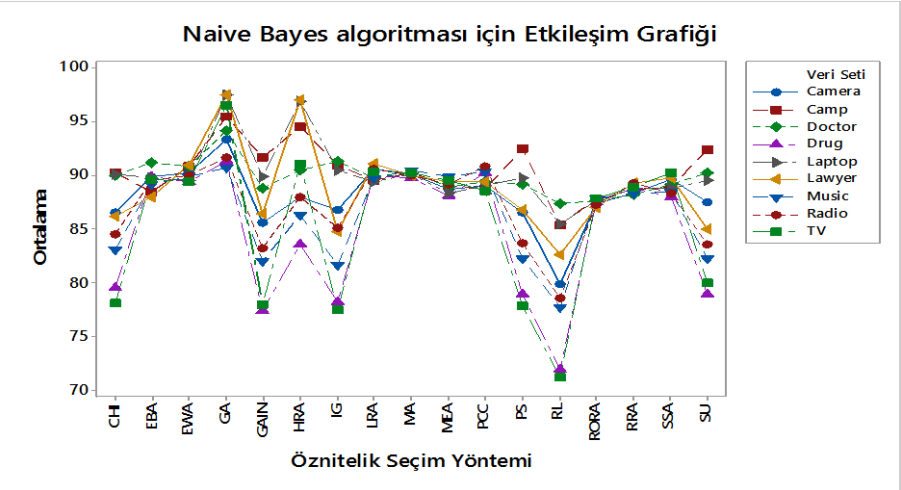
Şekil 5.1. Naive Bayes algoritmasına ilişkin güven aralığı



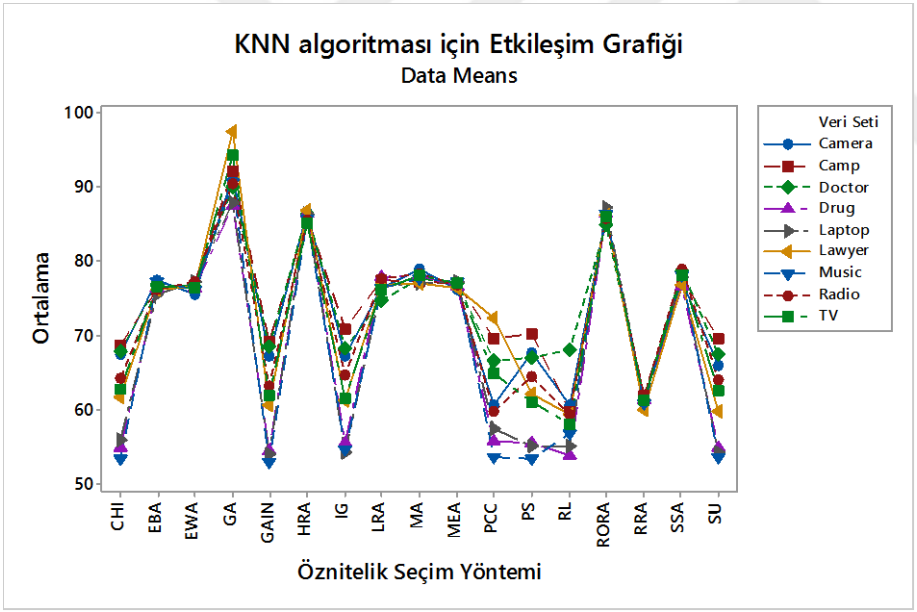
Şekil 5.2. K-en yakın komşu algoritmasına ilişkin güven aralığı

Şekil 5.1'de karşılaştırılan öznitelik seçim yöntemleri ile Naive Bayes algoritmasında elde edilen sonuçlara ilişkin güven aralığı grafiği verilmiştir. Şekil 5.2'de ise öznitelik seçim yöntemleri ile K-en yakın komşu algoritmasında elde edilen sonuçlara ilişkin güven aralığı grafiği sunulmuştur. Şekil 5.1 ve Şekil 5.2'de verilen güven aralığı grafiklerinden görülebildiği gibi, geliştirilen genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçim yönteminin, temel filtre tabanlı ve sıra birleştirmeye dayalı öznitelik seçim yöntemlerine kıyasla elde ettiği daha yüksek başarım, istatistiksel olarak anlamlıdır. Bunun yanı sıra, diğer sıra

birleştirmeye dayalı öznitelik seçim yöntemlerinin doğru sınıflandırma oran ortalamaları arasında istatistiksel olarak anlam bulunmamaktadır.



Şekil 5.3. Naive Bayes algoritmasına ilişkin etkileşim grafiği



Şekil 5.4. K-en yakın komşu algoritmasına ilişkin etkileşim grafiği

Şekil 5.3'te farklı veri setleri üzerinde Naive Bayes sınıflandırıcısı ile elde edilen doğru sınıflandırma oran ortalamalarına ilişkin etkileşim grafiği sunulmuştur. Şekil 5.4'te ise farklı veri setlerinde K-en yakın komşu algoritması ile elde edilen doğru sınıflandırma oran ortalamalarının değişimi özetlenmektedir.

5.4 Ağırlıklı Oylama Deneysel Sonuçları

Bu bölümde, geliştirilen çok amaçlı eniyilemeye dayalı ağırlıklı oylama yönteminin başarımı, dördüncü bölümde değinilen temel sınıflandırıcı topluluğu yöntemleri ve sınıflandırıcı topluluğu birleştirme kuralları ile karşılaştırmalı olarak sunulmaktadır. Bu bölümde, deneysel analizlerde izlenilen deneysel süreç, değerlendirme ölçütü, sonuçlar ve tartışmalara yer verilmektedir.

5.4.1 Ağırlıklı oylama deneysel süreci

Bu bölümde, geliştirilen çok amaçlı eniyilemeye dayalı ağırlık oylama yönteminin başarımının değerlendirilmesinde izlenen deneysel süreç sunulmaktadır. Geliştirilen yöntemin sınanmasında, ayrıntıları daha önce verilen görüş sınıflandırmaya ilişkin dokuz farklı alana ait veri setleri (“Camera”, “Camp”, “Doctor”, “Drug”, “Laptop”, “Lawyer”, “Music”, “Radio” ve “TV” veri setleri) kullanılmıştır. Yöntemlerin, başarımlarının incelenmesinde 10-kat çapraz geçerleme yöntemi kullanılmıştır.

Deneysel çalışmalarda kullanılan sınıflandırma algoritmaları ve sınıflandırıcı topluluğu yöntemleri, WEKA (Waikato Environment for Knowledge Analysis) 3.7.11 aracılığıyla gerçekleştirilmiştir (Witten et al., 2011). Önerilen ağırlıklı oylamaya dayalı sınıflandırıcı topluluğu yaklaşımının gerçekleştirimi Java dilinde yapılmıştır. Geliştirilen sınıflandırıcı topluluğu yönteminin değerlendirilmesinde, beş temel sınıflandırma algoritması, altı sınıflandırıcı topluluğu yöntemi ile ağırlıklı ve ağırlıksız oylama yöntemleri kullanılmıştır. AdaBoost, Bagging, Dagging ve Rastgele alt-uzay sınıflandırıcı toplulukları, her bir temel öğrenme algoritması temel sınıflandırıcı olarak kullanılarak her bir algoritma için beşer farklı sınıflandırıcı topluluğu oluşturulmuştur. Yığılmış genelleme ve oylamaya dayalı yöntemler ise beş temel öğrenme algoritmasını (Naive Bayes, destek vektör makineleri, lojistik regresyon, Bayesci lojistik regresyon ve lineer diskriminant analizi) birleştirerek elde edilmiştir. Bunun yanı sıra, geliştirilen ağırlıklı oylamaya dayalı sınıflandırıcı topluluğunun performansı genetik algoritmaya dayalı (Ekbal and Saha, 2011b) ve çok amaçlı benzetilmiş tavlamaya dayalı (Ekbal and Saha, 2013) metasezgisel sınıflandırıcı toplulukları ile karşılaştırılmıştır. MODE algoritmasının ve diğer metasezgisel yöntemlerin parametre değerleri olarak, ilgili çalışmalarda belirtilen temel parametre değerleri kullanılmıştır. Çizelge 5.6’da arama algoritmalarının temel parametre değerleri özetlenmiştir.

Çizelge 5.6. Arama algoritmalarının parametre değerleri

Algoritma	Parametre
Genetik Algoritma	Mutasyon oranı = 0.9, Çaprazlama oranı = 0.1, Toplum büyüklüğü = 100, Jenerasyon sayısı = 50
Benzetilmiş tavlama	Maksimum sıcaklık (T_{max}) = 100, Minimum sıcaklık (T_{min}) = 0.00001, Esnek limit (SL) = 200, Sıkı limit (HL) = 100, Soğutma oranı (a) = 0.8, İterasyon sayısı = 100
DE	Çaprazlama oranı = 0.5, Mutasyon oranı = 0.5, Toplum büyüklüğü = 100, Jenerasyon sayısı = 50
CMDPSO	Hareketsizlik katsayısı (w) = 0.7298, Hızlanma sabitleri ($c1$,$c2$) = 1.49618, Maksimum hız (V_{max}) = 6.0, Toplum büyüklüğü = 30, İterasyon sayısı = 100
MODE	Ölçekleme faktörü (b) = 0.5, Toplum büyüklüğü = 100, Jenerasyon sayısı = 50

5.4.2 Ağırlıklı oylama deneysel sonuçları ve tartışma

Çizelge 5.7'de temel öğrenme algoritmaları ve temel sınıflandırıcı topluluğu yöntemleri (AdaBoost, Bagging, Dagging ve rastgele alt-uzay) ile görüş sınıflandırma veri setleri üzerinde elde edilen sonuçlar sunulmuştur. Çizelge 5.7'de "Camera", "Drug" ve "Lawyer" veri setleri için en yüksek doğru sınıflandırma oranı Naive Bayes algoritmasının Bagging yöntemi ile bir arada kullanılması ile elde edilmektedir. "Camp" veri seti için en yüksek doğru sınıflandırma oranı, Bayesci lojistik regresyon ile, "Doctor" veri seti için en yüksek doğru sınıflandırma oranı, lineer diskriminant analizi ile elde edilmektedir. "Laptop" veri seti için en yüksek doğru sınıflandırma oranı Naive Bayes algoritmasının AdaBoost yöntemi ile bir arada kullanılması ile elde edilmektedir. "Music" veri seti için en yüksek doğru sınıflandırma oranı, lojistik regresyon Bagging yöntemi ile bir arada kullanılınca elde edilmektedir. "Radio" veri seti için en yüksek doğru sınıflandırma oranı, lineer diskriminant analizi rastgele alt uzay yöntemi ile bir arada kullanıldığında elde edilmektedir. "TV" veri seti için en yüksek doğru sınıflandırma oranı Naive Bayes algoritması ile elde edilmektedir. Çizelge 5.7'deki sonuçlar genel olarak değerlendirildiğinde sınıflandırıcı topluluğu yöntemlerinin, temel sınıflandırma algoritmaları üzerinde önemli bir performans iyileştirmesine neden olmadıkları gözlemlenmektedir.

Çizelge 5.7. Temel sınıflandırıcıların ve sınıflandırıcı topluluklarının doğru sınıflandırma oran ortalamaları ve standart hataları

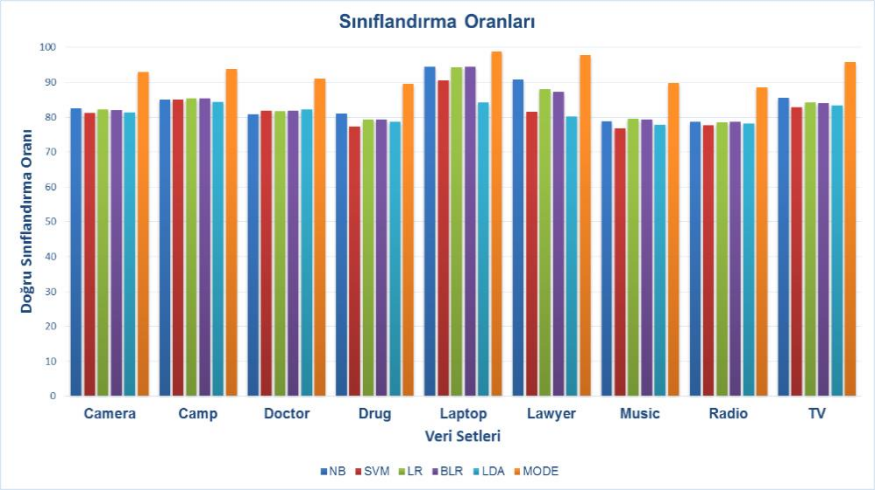
Yöntem	Camera	Camp	Doctor	Drug	Laptop	Lawyer	Music	Radio	TV	Ortalama
NB	***82.51±0.63***	85.06±0.42	80.90±0.66	***80.97±0.77***	***94.50±0.38***	***90.77±0.63***	78.81±0.54	78.68±0.48	**85.58±0.81**	**84.20±0.59**
SVM	81.15±0.75	84.98±0.62	81.85±0.55	77.26±0.57	90.66±0.48	81.53±0.51	76.84±0.64	77.68±0.71	82.83±0.59	81.64±0.53
LR	82.21±0.72	***85.39±0.44***	81.77±0.79	79.28±0.62	94.33±0.79	88.09±0.58	79.57±0.87	78.47±0.68	84.15±0.53	83.70±0.56
BLR	82.03±0.75	**85.42±0.54**	81.79±0.80	79.42±0.64	***94.50±0.55***	87.29±0.55	79.27±0.67	78.68±0.46	84.08±0.68	83.61±0.55
LDA	81.38±0.78	84.40±0.72	**82.25±0.57**	78.69±0.67	84.15±0.46	80.20±0.62	77.80±0.66	78.12±0.46	83.39±0.44	81.15±0.36
AdaBoost+NB	81.21±0.60	85.31±0.79	81.41±0.57	79.65±0.82	**95.00±0.72**	87.82±0.64	78.26±0.56	77.15±0.72	84.65±0.65	83.38±0.60
AdaBoost+SVM	78.61±0.73	83.78±0.87	80.54±0.25	75.50±0.38	90.82±0.67	82.59±0.49	75.68±0.74	76.25±0.40	81.33±0.65	80.57±0.53
AdaBoost+LR	80.20±0.52	83.96±0.60	81.33±0.78	78.29±0.66	92.16±0.70	85.81±0.63	78.41±0.64	76.63±0.71	83.64±0.41	82.27±0.50
AdaBoost+BLR	79.97±0.57	83.85±0.51	81.15±0.39	77.70±0.60	92.16±0.61	84.60±0.64	78.26±0.62	76.54±0.37	82.71±0.55	81.88±0.53
AdaBoost+LDA	80.14±0.70	83.70±0.68	80.90±0.57	76.64±0.45	82.32±0.70	75.36±0.58	73.76±0.58	76.57±0.57	82.27±0.78	79.07±0.41
Bagging+NB	**82.68±0.65**	83.12±0.57	81.06±0.46	**81.26±0.25**	94.32±0.80	**91.31±0.45**	79.27±0.74	***78.68±0.71***	83.21±0.58	***83.88±0.55***
Bagging+SVM	80.79±0.55	84.55±0.57	81.79±0.51	76.93±0.48	91.50±0.88	81.54±0.75	77.05±0.46	76.30±0.56	81.45±0.67	81.32±0.52
Bagging+LR	82.27±0.38	84.91±0.76	81.89±0.50	78.73±0.71	94.50±0.53	86.49±0.54	**79.98±0.47**	77.89±0.49	83.64±0.64	83.37±0.56
Bagging+BLR	82.27±0.71	84.95±0.53	81.83±0.61	78.66±0.80	94.50±0.53	85.82±0.73	***79.93±0.54***	77.80±0.58	83.58±0.70	83.26±0.51
Bagging+LDA	80.73±0.72	84.18±0.57	***81.99±0.70***	78.00±0.70	71.31±0.53	74.71±0.60	76.29±0.94	77.30±0.61	82.83±0.41	78.59±0.47
Dagging+NB	81.15±0.70	82.76±0.53	80.72±0.65	76.16±0.57	90.99±0.55	86.75±0.56	72.85±0.59	72.64±0.86	80.94±0.90	80.55±0.68
Dagging+SVM	79.37±0.57	81.44±0.55	79.70±0.50	70.56±0.79	88.82±0.50	78.31±0.45	72.70±0.61	71.35±0.81	73.93±0.42	77.35±0.62
Dagging+LR	82.15±0.34	82.68±0.64	81.35±0.81	75.17±0.77	89.32±0.78	82.60±0.83	77.91±0.52	73.32±0.51	77.44±0.31	80.22±0.52
Dagging+BLR	82.09±0.52	82.50±0.64	81.23±0.63	74.48±0.73	89.82±0.34	83.41±0.66	77.80±0.90	73.67±0.52	76.94±0.76	80.22±0.54
Dagging+LDA	74.88±0.46	81.48±0.29	79.32±0.61	67.55±0.44	89.49±0.59	79.51±0.60	68.35±0.42	67.69±0.55	76.88±0.51	76.13±0.73
Rastgele Altuzay+NB	82.27±0.46	84.07±0.82	80.46±0.69	80.42±0.42	94.16±0.65	90.37±0.56	77.75±0.56	78.15±0.38	***85.14±0.50***	83.64±0.61
Rastgele Altuzay+SVM	82.04±0.68	79.80±0.63	79.78±0.39	77.63±0.69	89.49±0.65	85.67±0.88	78.81±0.68	77.24±0.46	79.07±0.77	81.06±0.46
Rastgele Altuzay+LR	82.21±0.94	79.94±0.64	79.46±0.66	78.11±0.96	90.99±0.77	86.62±0.59	79.47±0.55	77.36±0.49	79.57±0.61	81.53±0.52
Rastgele Altuzay+BLR	82.15±0.55	80.02±0.53	79.50±0.56	78.18±0.42	91.16±0.44	86.62±0.96	79.17±0.59	77.45±0.69	79.57±0.47	81.54±0.51
Rastgele Altuzay+LDA	82.21±0.62	83.41±0.39	81.43±0.45	80.45±0.42	92.16±0.72	88.62±0.87	79.62±0.62	**79.41±0.62**	84.65±0.52	83.55±0.47

Çizelge 5.8. Sınıflandırıcı topluluğu birleştirme kurallarının doğru sınıflandırma oran ortalamaları ve standart hataları

Yöntem	Camera	Camp	Doctor	Drug	Laptop	Lawyer	Music	Radio	TV	Ortalama
Yığılmış Genelleme	83.45±0.69	85.46±0.61	82.13±0.51	82.29±0.71	94.66±0.88	90.10±0.70	82.56±0.58	79.99±0.50	86.78±0.82	85.27±0.50
Yığılmış Genelleme (StackingC)	83.10±0.65	85.82±0.59	81.91±0.47	82.76±0.99	94.99±0.70	90.24±0.75	82.66±0.45	80.67±0.85	86.78±0.38	85.44±0.51
Olasılıklar Ortalaması	87.67±0.66	88.47±0.77	86.92±0.44	84.95±0.67	93.94±0.45	92.73±0.71	82.82±0.79	84.13±0.62	90.21±0.81	87.98±0.44
Olasılıklar Çarpımı	83.65±0.61	84.83±0.72	85.61±0.49	79.80±0.68	90.15±0.65	88.18±0.55	75.03±0.75	78.55±0.65	84.82±0.61	83.40±0.52
Maksimum Olasılık Oylama	84.99±0.49	86.48±0.63	86.56±0.93	81.71±0.98	90.53±0.53	88.18±0.71	79.95±0.50	81.34±0.72	85.96±0.57	85.08±0.42
Minimum Olasılık Oylama	83.65±0.63	84.83±0.96	85.61±0.55	79.80±0.51	90.15±0.73	88.18±0.56	75.03±0.89	78.55±0.39	84.82±0.50	83.40±0.54
Çoğunluk Oylaması	87.40±0.42	88.56±0.61	86.87±0.86	85.20±0.45	94.32±0.90	93.03±0.53	83.51±0.41	84.26±0.90	90.35±0.84	88.17±0.46
Basit Ağırlıklı Oylama	88.47±0.72	89.63±0.87	86.60±0.73	85.29±0.73	94.82±0.75	93.06±0.56	83.27±0.66	84.37±0.57	91.53±0.92	88.56±0.49
Ölçeklenmiş Ağırlıklı Oylama	88.97±0.78	90.13±0.57	86.36±0.65	85.79±0.41	95.32±0.43	93.56±0.50	82.93±0.46	84.87±0.59	91.95±0.39	88.88±0.45
En-iyi en kötü Ağırlıklı Oylama	89.40±0.90	90.48±0.65	86.87±0.76	85.51±0.71	95.84±0.75	94.29±0.79	84.36±0.63	85.60±0.70	92.32±0.84	89.41±0.47
İkinci Dereceden en iyi-en kötü ağırlıklı oylama	88.27±0.83	89.43±0.29	87.93±0.40	85.01±0.41	94.63±0.59	92.90±0.67	82.36±0.65	84.21±0.86	90.50±0.50	88.36±0.43
Genetik Algoritma	89.43±0.61	90.77±0.47	87.39±0.42	86.11±0.51	96.02±0.61	94.35±0.71	84.70±0.53	85.49±0.75	92.37±0.75	89.63±0.47
Benzetilmiş tavlama	***91.51±0.32***	***92.99±0.73***	***89.92±0.71***	***88.32±0.65***	***97.32±0.49***	***96.57±0.95***	***88.19±0.90***	***87.64±0.37***	***94.31±0.85***	***91.86±0.44***
DE	90.53±0.65	91.47±0.36	89.27±0.67	86.98±0.60	95.72±0.77	95.80±0.61	87.92±0.48	87.69±0.55	93.59±0.57	91.00±0.38
CMDPSO	91.08±0.61	92.36±0.88	89.23±0.57	87.84±0.57	96.65±0.57	96.08±0.80	86.41±0.52	87.24±0.91	93.93±0.62	91.20±0.50
MODE	**92.87±0.53**	**93.74±0.37**	**91.05±0.46**	**89.62±0.62**	**98.86±0.65**	**97.87±0.33**	**89.82±0.63**	**88.60±0.63**	**95.74±0.71**	**93.13±0.43**

Çizelge 5.8'de ağırlıksız oylama yöntemleri, ağırlıklı oylama yöntemleri ve metasezgisel ağırlıklı oylama yöntemleri ile elde edilen sonuçlar sunulmuştur. Çizelge 5.8'de sunulan sonuçlardan görüldüğü gibi, en yüksek doğru sınıflandırma oranı önerilen çok amaçlı diferansiyel gelişim algoritması (MODE) ile elde edilmektedir. Her bir veri seti için en yüksek ikinci doğru sınıflandırma oranları, benzetilmiş tavlamaya dayalı ağırlıklı oylama ile elde edilmektedir.

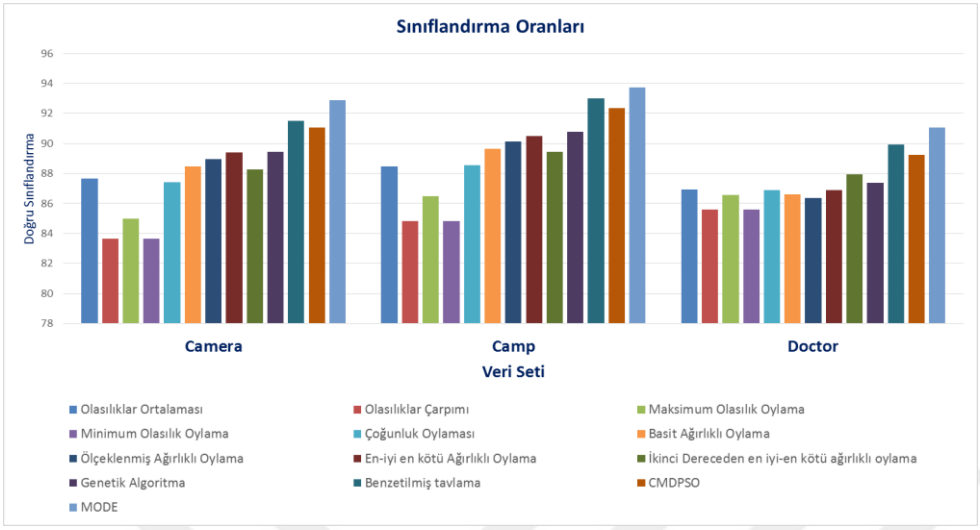
Çizelge 5.8'de sunulan doğru sınıflandırma oranlarından açıkça görülebildiği gibi, ağırlıksız oylama yöntemleri, temel ağırlıklı oylama yöntemleri ve metasezgisel algoritmalara dayalı ağırlıklı oylama yöntemleri, çalışmada kullanılan beş temel öğrenme algoritmasına kıyasla daha iyi sonuçlar elde etmektedir. Bunun yanı sıra, yığılmış genelleme algoritmasının, temel öğrenme algoritmaları üzerindeki performans iyileştirmesi, oylama yöntemlerine kıyasla daha azdır. Ağırlıksız oylama yöntemleri içerisinde en yüksek başarım genellikle çoğunluk oylaması yöntemi ile alınmıştır. Temel ağırlıklı oylama yöntemleri içerisinde en yüksek başarım genellikle en iyi-en kötü ağırlıklı oylama yöntemi ile elde edilmiştir.



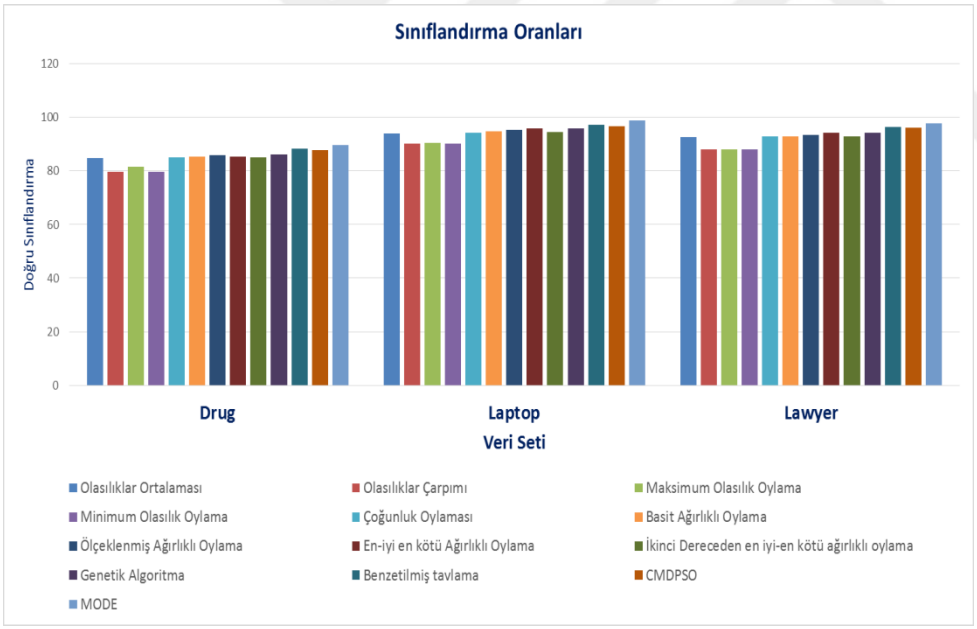
Şekil 5.5. Temel öğrenme algoritmalarının MODE ile doğru sınıflandırma oranı bakımından karşılaştırılması

Şekil 5.5'te önerilen ağırlıklı oylama yönteminin temel öğrenme algoritmaları ile performans karşılaştırması sunulmuştur. Şekil 5.6-5.8'de ise ağırlıklı oylama yöntemlerinin, ağırlıksız oylama yöntemlerinin ve metasezgisel ağırlıklı oylama yöntemlerinin doğru sınıflandırma oranı bakımından karşılaştırmaları sunulmuştur. Şekillerden açıkça görülebildiği gibi, MODE

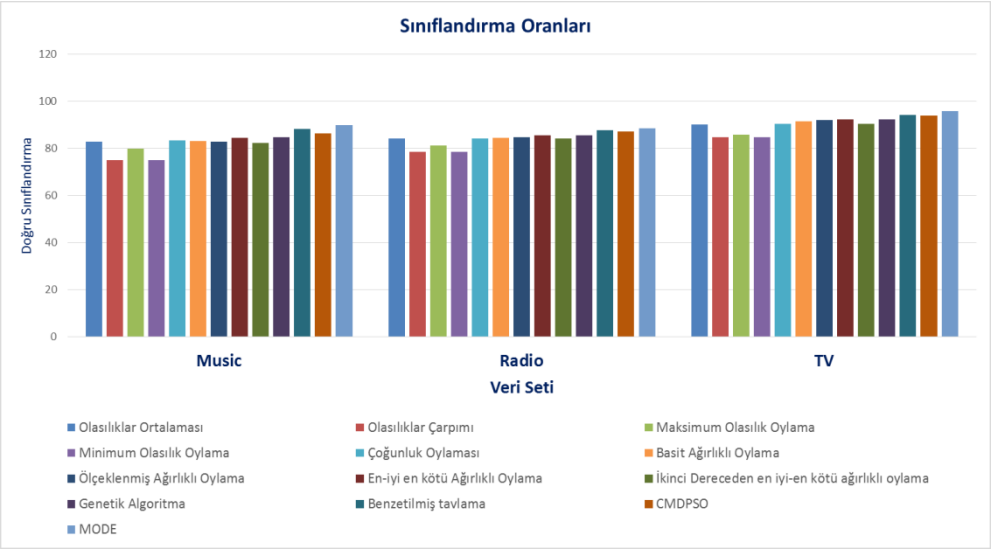
ağırlıklı oylama yöntemi, diğer oylama yöntemlerine kıyasla önemli ölçüde performans iyileşmesine neden olmaktadır.



Şekil 5.6. Camera, Camp ve Doctor veri setleri için sınıflandırma oranları



Şekil 5.7. Drug, Laptop ve Lawyer veri setleri için sınıflandırma oranları



Şekil 5.8. Music, Radio ve TV veri setleri için sınıflandırma oranları

Çizelge 5.9. Topluluk birleştirme yöntemlerinin çalışma süresi bakımından karşılaştırılması

Yöntem	Camera	Camp	Doctor	Drug	Laptop	Lawyer	Music	Radio	TV
Çoğunluk Oylaması	**69.39**	**68.72**	***293.22***	***109.96***	***100.79***	***73.30***	**109.96**	***82.47***	**45.82**
En-iyi En-kötü Ağırlıklı Oylama	***73.30***	***98.04***	**247.40**	**100.79**	**73.30**	**45.82**	***128.28***	**73.30**	***54.98***
Benzetilmiş Tavlama	316.81	336.30	1209.60	439.85	320.72	183.27	549.82	302.40	192.46
MODE	300.18	278.74	1072.07	387.59	258.40	129.20	516.79	258.40	172.28

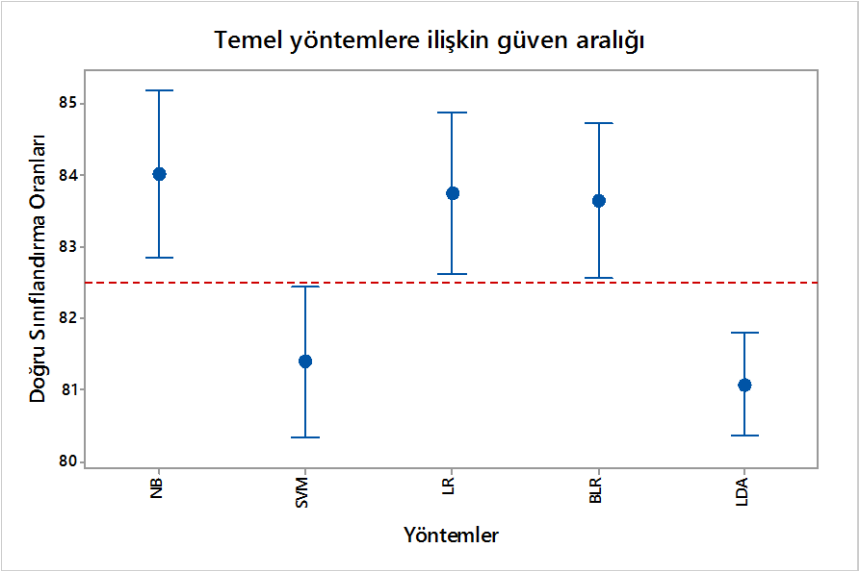
Çizelge 5.9'da sınıflandırıcı topluluğu birleştirme kurallarının çalışma süresi bakımından karşılaştırmaları saniye olarak sunulmuştur. Burada, her bir sınıflandırıcı topluluğu birleştirme kuralı sınıfından, en yüksek doğru sınıflandırma başarımına sahip olanlara yer verilmiştir. Çalışma süreleri bakımından, en hızlı sonuçlar ağırlıksız oylama yöntemi (çoğunluk oylaması) ve ağırlıklı oylama yöntemi (en-iyi en kötü ağırlıklı oylama) ile elde edilmektedir. Metasezgisel ağırlıklı oylamaya dayalı yöntemler (benzetilmiş tavlama ve çok amaçlı diferansiyel gelişim algoritması), yüksek doğru sınıflandırma başarımlarına karşın, çalışma süreleri bakımından, temel ağırlıksız ve ağırlıklı oylama yöntemlerinden daha yavaş yöntemlerdir. Metasezgisel yöntemler, arama uzayının incelenmesinde, birden fazla çözümün yinelemeli olarak, daha iyi bir çözüme ulaşılıncaya dek sürdürülmesini gerektirdiğinden, temel yöntemlere kıyasla daha uzun çalışma süreleri gerektirmektedir.

Çizelge 5.10. Sınıflandırıcı topluluğu iki-yönlü varyans analizi sonuçları

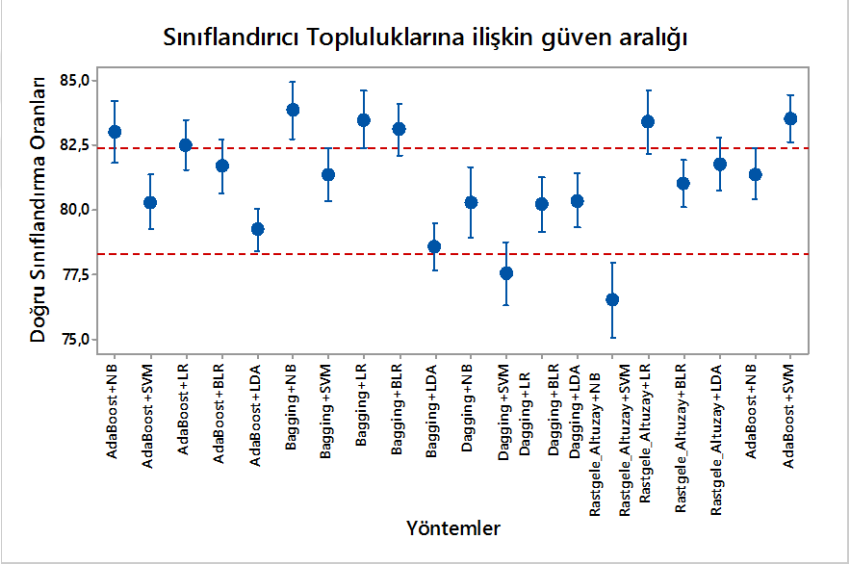
Kaynak	DF	SS	MS	F	P
Veri seti	8	57989.8	7248.7	1762.02	0.000
Algoritmalar	40	60099.6	1502.5	365.22	0.000
Veri seti*Algoritmalar	320	14843.4	46.4	11.28	0.000
Hata	3321	13662.2	4.1		
Toplam	3689				

Bu bölümde sunulan deneysel sonuçların istatistiksel olarak analiz edilmesi için Minitab istatistik paket programı kullanılmış ve iki-yönlü varyans analizi (*two-way ANOVA test*) uygulanmıştır. Çizelge 5.10'da iki-yönlü varyans analizine ilişkin analiz sonuçları sunulmuştur. Burada DF, SS, MS, F ve P, sırasıyla, serbestlik derecesi (*degrees of freedom*), uyarlanmış kareler toplamı (*adjusted sum of squares*), uyarlanmış kareler ortalaması (*adjusted mean square*), F-değeri (*F-value*) ve olasılık değerini temsil etmektedir. Çizelgede sunulan iki-yönlü varyans analizi sonuçlarına göre, karşılaştırılan sınıflandırma algoritmalarının doğru sınıflandırma başarımları arasında, istatistiksel olarak anlamlı bir farklılık bulunmaktadır ($p<0.0001$). Karşılaştırılan dokuz veri seti ile elde edilen sonuçlar arasında da, istatistiksel olarak anlamlı anlamlı bir farklılık bulunmaktadır ($p<0.0001$). Bunun yanı sıra, karşılaştırılan veri setleri ve algoritmalar arasındaki etkileşim istatistiksel olarak önemlidir.

Şekil 5.9'da karşılaştırılan temel sınıflandırma algoritmalarına ilişkin güven aralığı grafiği (%95 güven aralığı için) sunulmuştur. Karşılaştırılan algoritmalar arasındaki istatistiksel farklılıklara dayalı olarak güven aralığı grafiği, kırmızı kesikli çizgiler ile temsil edilen iki temel bölüme ayrılmıştır. Dolayısıyla, Naive Bayes, lojistik regresyon ve Bayesci lojistik regresyon sınıflandırıcıları ile elde edilen daha yüksek doğru sınıflandırma oranları istatistiksel olarak anlamlıdır.



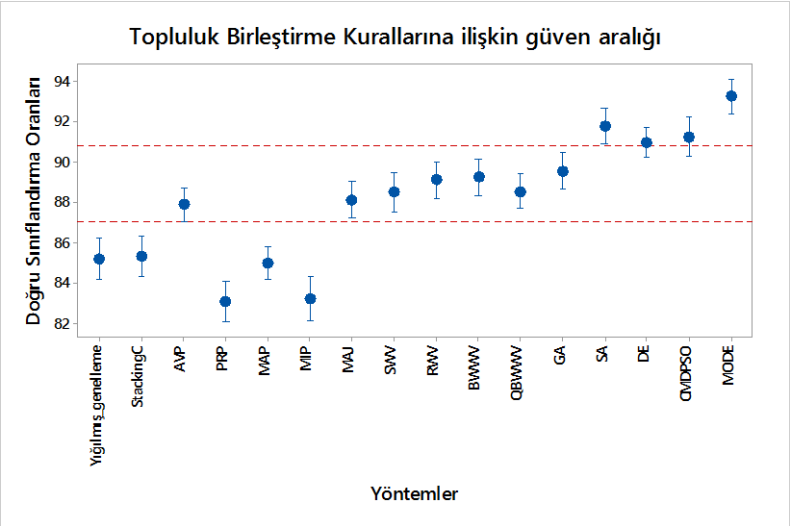
Şekil 5.9. Temel öğrenme algoritmalarına ilişkin güven aralığı grafiği



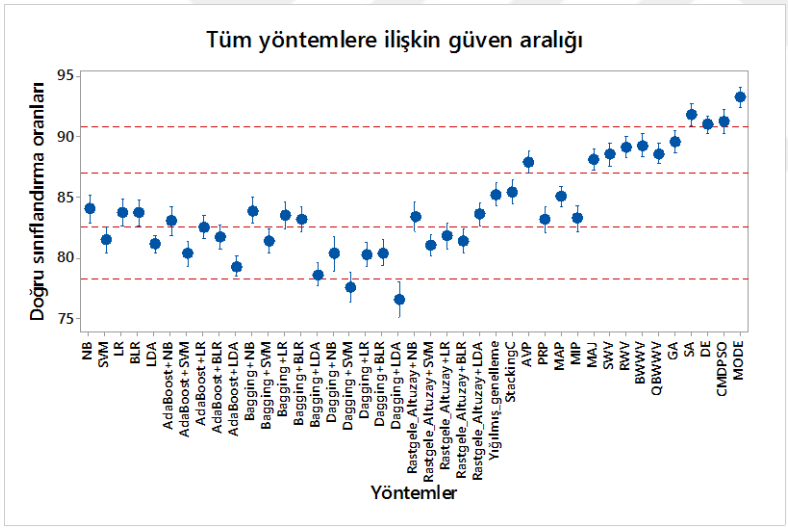
Şekil 5.10. Sınıflandırıcı topluluklarına ilişkin güven aralığı grafiği

Şekil 5.10'da karşılaştırılan temel sınıflandırıcı topluluğu algoritmalarına ilişkin güven aralığı grafiği (%95 güven aralığı için) verilmektedir. Karşılaştırılan algoritmalar arasındaki istatistiksel farklılıklara dayalı olarak güven aralığı grafiği, kırmızı kesikli çizgiler ile temsil edilen üç temel bölüme ayrılmıştır. Şekil 5.10'da Bagging algoritmasının Naive Bayes ile oluşturduğu sınıflandırıcı topluluğu, Bagging algoritmasının lojistik regresyon ile oluşturduğu sınıflandırıcı topluluğu ve AdaBoost algoritmasının destek vektör makineleri ile oluşturduğu sınıflandırıcı topluluğu, diğer sınıflandırıcı topluluklarına kıyasla istatistiksel olarak anlamlı, daha yüksek doğru sınıflandırma başarımı göstermektedir. Şekil

5.11'de ise Çizelge 5.8'de sunulan topluluk birleştirme kurallarına ilişkin güven aralıkları (%95 güven aralığı için) sunulmaktadır.



Şekil 5.11. Topluluk birleştirme kurallarına ilişkin güven aralığı grafiği



Şekil 5.12. Karşılaştırılan sınıflandırma algoritmaları ve sınıflandırıcı topluluklarına ilişkin güven aralığı grafiği

Şekil 5.12'de ise karşılaştırılan tüm algoritmaların güven aralıkları (%95 güven aralığı için) sunulmuştur. Karşılaştırılan algoritmalar arasındaki istatistiksel farklılıklara dayalı olarak güven aralığı grafiği, kırmızı kesikli çizgiler ile temsil edilen bölümlere ayrılmıştır. Geliştirilen çok amaçlı diferansiyel gelişim algoritmasına ağırlıklı oylama yöntemi (MODE) ve benzetilmiş tavlamaya dayalı ağırlıklı oylama yöntemleri, doğru sınıflandırma oranları bakımından, grafiğin bir başka bölgesinde konuşlanmaktadır. Bu nedenle, bu iki algoritma ile elde edilen

yüksek doğru sınıflandırma oranı değerlerinin, diğer karşılaştırılan yöntemlerden istatistiksel olarak anlamlı bir farklılık oluşturduğu görülmektedir.

5.5 Topluluk Budaması Deneysel Sonuçları

Bu bölümde, geliştirilen ortak kümeleme ve metasezgisel aramaya dayalı sınıflandırıcı topluluğu budama yönteminin başarımı, üç temel sınıflandırıcı topluluğu budama yöntemi (Model kütüphanesinden topluluk seçimi algoritması, Bagging sınıflandırıcı seçimi algoritması ve LibD3C algoritması) ile karşılaştırmalı olarak sunulmaktadır. Bu bölümde, deneysel analizlerde izlenilen deneysel süreç ve sonuçlar aktarılmaktadır.

5.5.1 Topluluk budaması deneysel süreci

Geliştirilen sınıflandırıcı topluluğu budama yönteminin başarımının değerlendirilmesinde, daha önce değinilen görüş sınıflandırmaya ilişkin dokuz farklı alana ait veri setleri (“Camera”, “Camp”, “Doctor”, “Drug”, “Laptop”, “Lawyer”, “Music”, “Radio” ve “TV” veri setleri) kullanılmıştır. Deneysel çalışmalarda, 10-kat çapraz geçerleme yöntemi kullanılmıştır. Sınıflandırma algoritmaları ve karşılaştırılan sınıflandırıcı topluluğu budama yöntemleri (Model kütüphanesinden topluluk seçimi algoritması, Bagging sınıflandırıcı seçimi algoritması ve LibD3C algoritması) WEKA (Waikato Environment for Knowledge Analysis) 3.7.11 kullanılarak gerçekleştirilmiştir (Hall et al., 2009). Önerilen sınıflandırıcı topluluğu budama yönteminin gerçekleştirimi Java dilinde yapılmıştır. Çizelge 5.11’de deneysel çalışmalarda model kütüphanesi olarak kullanılan temel öğrenme algoritmaları (sınıflandırma algoritmaları) aktarılmaktadır. ESM ve LibD3C algoritmaları ile geliştirilen sınıflandırıcı topluluğu budama yönteminde, Çizelge 5.11’de değinilen model kütüphanesi kullanılmıştır. Deneysel analizlerde, ESM, BES ve LibD3C algoritmaları ile elde edilen en iyi başarımların belirlenebilmesi için değinilen yöntemler, farklı parametre değerleri ile incelenmiştir. ESM algoritmasında arama algoritması olarak, ileri arama, geri arama, ileri-geri arama ve en iyi model yöntemleri kullanılmıştır. Değerlendirme ölçütü olarak ise ortalama karesel hata (*root mean squared error*), doğru sınıflandırma ölçütü (*accuracy*), duyarlılık, geri çağırma, ROC eğrisinin altında kalan alan ve F-ölçütü kullanılmıştır. Bu şekilde, ESM algoritması için 24 farklı durum değerlendirilmiştir. Benzer şekilde, BES algoritması için, ortalama karesel hata, doğru sınıflandırma ölçütü, ROC eğrisinin altında kalan alan, duyarlılık, geri çağırma, F-ölçütü gibi farklı değerlendirme

ölçütleri ve parametre değerleri için 80 farklı durum değerlendirilmiştir. LibD3C algoritması için ise beş farklı sınıflandırıcı topluluğu birleştirme kuralının (olasılıklar ortalaması, olasılıklar çarpımı, çoğunluk oylaması, en az olasılık ve en çok olasılık) performansı değerlendirilmiştir. Deneysel sonuçlar bölümünde, değinilen sınıflandırıcı topluluğu budama algoritmalarının tüm durumları içerisindeki en iyi başarımları listelenmiştir.

Çizelge 5.11. Model kütüphanesindeki temel öğrenme algoritmaları

Sınıflandırıcı Grubu	Temel Öğrenme Algoritmaları
Bayes tabanlı sınıflandırıcılar (5)	Bayesci lojistik regresyon (Norm tabanlı hiper parametre seçimi), Bayesci lojistik regresyon (Çapraz geçerleme tabanlı hiper parametre seçimi), Bayesci lojistik regresyon (Özel değer tabanlı hiper parametre seçimi), Naive Bayes, Naive Bayes (katlıterim)
Fonksiyon tabanlı sınıflandırıcılar (14)	Lineer diskriminant analizi, Kernel lojistik regresyon (Çoklu çekirdek), Kernel lojistik regresyon (Normalize edilmiş çoklu çekirdek), LibLINEAR (L2-düzenlenmiş lojistik regresyon), LibSVM (radyal tabanlı fonksiyon), LibSVM (doğrusal çekirdek), LibSVM (Çok terimli çekirdek), LibSVM (Sigmoid çekirdek), Çok katmanlı algılayıcı ağlar, radyal tabanlı fonksiyon ağları, lojistik regresyon, Gauss radyal tabanlı fonksiyon ağları
Örnek tabanlı sınıflandırıcıları (10)	K-en yakın komşu algoritması (K:1, K:2, K:3, K:4, K:5, K:6, K: 7, K: 8, K: 9, K: 10)
Kural tabanlı sınıflandırıcılar (3)	FURIA algoritması (T-norm), FURIA (minimum T-norm), RIPPER
Karar ağacı sınıflandırıcıları (8)	BFTree (Budanmamış), BFTree (Sonradan budama), BFTree (Önceden Budama), C4.5 (J48), NBTree, Random Forest, Random Tree

Çizelge 5.12. Arama algoritması parametre değerleri

Arama Algoritması	Parametre Değerleri
En iyi önce arama algoritması	Arama yönü: İleriye, Aramada değerlendirilen alt grupların maksimum önbellek boyutu: 1, İzin verilen ardışık iyileşmeyen çözüm sayısı: 5
Açgözlü arama algoritması	Tutulan öznitelik sayısı: Tüm, Yürütme yuvası sayısı: 1
Parçacık sürü eniyilemesi algoritması	Bireysel ağırlık: 0.34, Hareketsizlik ağırlığı: 0.33, Yineleme sayısı: 20, Mutasyon olasılığı: 0.01, Mutasyon çeşidi: bit-çevirme, Toplum büyüklüğü: 20, Çekirdek: 1, Sosyal ağırlık: 0.33
Genetik algoritma	Çaprazlama olasılığı: 0.6, Değerlendirilecek kuşak sayısı: 20, Mutasyon olasılığı: 0.033, Toplum büyüklüğü: 20, Çekirdek: 1
Diferansiyel gelişim algoritması	Ölçeklendirme faktörü:0.5, Toplum büyüklüğü:100, Değerlendirilecek kuşak sayısı: 50
ENORA Algoritması	Değerlendirilecek kuşak sayısı: 10, Toplum büyüklüğü: 100, Çekirdek: 1

Geliştirilen/önerilen sınıflandırıcı topluluğu budama algoritması, kümeleme ve rastsal topluluk budaması yöntemlerine dayalı melez bir yöntemdir. Yöntemin, kümeleme aşamasında, farklı kümeleme algoritmalarının başarımları (K-ortalama, K-ortalama++, beklenti maksimizasyonu ve öz-düzenlemeli harita algoritması ve

bu yöntemlerin tüm olası alt kümeleri ile elde edilen küme toplulukları) değerlendirilmiştir. Kümelerden aday sınıflandırma algoritmalarının seçilmesinde daha önce değinilen 7 farklı kural incelenmiştir. Aday sınıflandırma algoritmalarına ilişkin arama uzayının değerlendirilmesinde ise, parametre değerleri Çizelge 5.12'de sunulan arama algoritmaları kullanılmıştır.

Bunun yanı sıra, geliştirilen sınıflandırıcı topluluğu budama yönteminin başarımı, AdaBoost, Bagging ve Rastgele alt uzay algoritması gibi temel sınıflandırıcı topluluğu yöntemleri ile karşılaştırılmıştır. Değinilen yöntemlerde, Naive Bayes algoritması temel sınıflandırma algoritması olarak kullanılmıştır.

5.5.2 Topluluk budaması deneysel sonuçları ve tartışma

Geliştirilen modelin değerlendirilmesine ilişkin deneysel analizlerde, farklı kümeleme algoritmaları (K-ortalama, K-ortalama++, beklenti maksimizasyonu, öz-düzenlemeli harita algoritması ve bu algoritmaların altkümeleri) kümeleme toplulukları oluşturmak üzere değerlendirilmiştir. Bunun yanı sıra, her bir kümeden aday sınıflandırma algoritmalarının seçiminde yedi farklı kural (RAN1, RAN2, BEST, WS, Q-istatistiği, Dis ve DF) incelenmiştir. Aday sınıflandırıcılara ilişkin arama uzayının incelenmesinde ise altı arama algoritmasının (en iyi önce arama algoritması, açgözlü arama algoritması, parçacık sürü eniyilemesi algoritması, diferansiyel gelişim algoritması ve ENORA algoritması) başarımı değerlendirilmiştir. Çizelgelerde belirli bir veri seti için elde edilen en yüksek doğru sınıflandırma başarımları koyu ve altı çizili olarak, ikinci en yüksek değerler ise koyu ve italik olarak belirtilmiştir.

Geliştirilen yönteme ilişkin tasarım konularından birincisi, kümeleme algoritması kullanılarak sınıflandırma algoritmalarının gruplara ayrılmasının ardından her bir kümeden aday sınıflandırma algoritmalarının hangi kurala göre seçileceğinin belirlenmesidir. Çizelge 5.13'te kümeden aday sınıflandırma algoritması seçiminde kullanılan yedi farklı kuralın ortalama doğru sınıflandırma oranları, standart hata ile birlikte verilmiştir. Çizelge 5.13'te verilen sonuçlardan da görülebildiği gibi, değerlendirilen kurallar içerisinde en yüksek başarım Q-istatistiğine dayalı eleman seçimi ile ikinci en yüksek sonuçlar uyuşmazlık ölçütüne dayalı eleman seçimi ile ve en kötü başarım, her bir kümeden rastgele birer eleman seçilmesi ile elde edilmiştir.

Çizelge 5.13. Kümeden eleman seçim kurallarının karşılaştırılması

Seçim Kuralı	Q-istatistiği	Dis	DF	WS	BEST	RAN2	RAN1
Camera	**92.91±0.09**	***91.81±0.06***	91.27±0.07	90.71±0.07	90.15±0.07	89.58±0.10	88.20±0.11
Camp	**94.13±0.09**	***92.94±0.06***	92.37±0.07	91.79±0.07	91.19±0.07	90.53±0.10	89.33±0.10
Doctor	**92.99±0.09**	***91.77±0.06***	91.19±0.07	90.65±0.07	90.11±0.07	89.58±0.10	88.31±0.11
Drug	**91.94±0.08**	***90.85±0.06***	90.25±0.07	89.71±0.07	89.20±0.07	88.68±0.10	87.37±0.10
Laptop	**96.34±0.08**	***95.49±0.06***	95.11±0.07	94.75±0.07	94.34±0.07	93.93±0.10	93.09±0.09
Lawyer	**94.62±0.09**	***93.49±0.06***	92.92±0.07	92.41±0.07	91.91±0.07	91.31±0.10	89.91±0.11
Music	**92.17±0.08**	***91.06±0.07***	90.45±0.07	89.72±0.07	89.00±0.07	88.28±0.11	86.64±0.12
Radio	**91.10±0.09**	***89.92±0.06***	89.38±0.07	88.85±0.07	88.34±0.07	87.80±0.10	86.61±0.09
TV	**94.04±0.08**	***93.12±0.06***	92.59±0.07	92.09±0.07	91.60±0.07	91.03±0.09	89.89±0.11
Ortalama	**93.36±0.06**	***92.27±0.06***	91.72±0.06	91.18±0.06	90.64±0.06	90.08±0.07	88.82±0.07

Çizelge 5.14. Farklı arama algoritmalarının başarımlarının karşılaştırılması

Arama Algoritması	ENORA	DE	GA	PSO	BFS	GS
Camera	**91.12±0.16**	***90.85±0.15***	90.75±0.15	90.60±0.15	90.43±0.16	90.21±0.17
Camp	**92.23±0.17**	***91.95±0.15***	91.84±0.15	91.68±0.15	91.52±0.16	91.31±0.17
Doctor	**91.13±0.16**	***90.86±0.15***	90.74±0.15	90.59±0.15	90.43±0.15	90.20±0.17
Drug	**90.17±0.16**	***89.91±0.14***	89.80±0.14	89.65±0.15	89.49±0.15	89.20±0.16
Laptop	**95.07±0.13**	***94.85±0.11***	94.77±0.11	94.67±0.11	94.55±0.12	94.40±0.12
Lawyer	**92.83±0.16**	***92.57±0.14***	92.45±0.15	92.30±0.15	92.13±0.15	91.91±0.17
Music	**90.14±0.18**	***89.85±0.17***	89.73±0.17	89.54±0.18	89.35±0.19	89.08±0.21
Radio	**89.30±0.16**	***89.03±0.14***	88.93±0.14	88.77±0.15	88.63±0.15	88.47±0.16
TV	**92.43±0.14**	***92.22±0.13***	92.14±0.14	92.01±0.14	91.88±0.14	91.63±0.16
Ortalama	**91.60±0.07**	***91.34±0.07***	91.24±0.07	91.09±0.07	90.93±0.07	90.72±0.08

Geliştirilen yönteme ilişkin tasarım konularından ikincisi, aday sınıflandırıcılar arasından hangilerinin son budanan sınıflandırıcı topluluğunda yer alacağının belirlenmesine yönelik olarak kullanılacak arama algoritmasının belirlenmesidir. Çizelge 5.14'te farklı arama algoritmaları ile elde edilen doğru sınıflandırma oranları ve standart hata sunulmaktadır. Farklı arama algoritmaları arasında en yüksek doğru sınıflandırma başarımı, ENORA algoritması ile elde edilmiştir. İkinci en yüksek başarım, diferansiyel gelişim algoritması ile en kötü performans ise açgözlü arama algoritması ile alınmıştır. Geliştirilen yönteme ilişkin tasarım konularından üçüncüsü, kümeleme aşamasında kullanılacak algoritmanın belirlenmesidir. Bu doğrultuda, on beş farklı kümeleme algoritması göz önünde bulundurulmuştur.

Çizelge 5.15. Farklı kümeleme algoritmalarının karşılaştırılması

Yöntem	Camera	Camp	Doctor	Drug	Lawyer	Laptop	Music	Radio	TV	Ortalama
KM	90.26±0.24	91.36±0.25	90.25±0.24	89.33±0.24	91.97±0.24	94.36±0.17	89.20±0.29	88.48±0.22	91.61±0.25	90.76±0.12
KM++	90.28±0.24	91.39±0.24	90.28±0.24	89.34±0.23	91.99±0.24	94.35±0.16	89.25±0.29	88.49±0.22	91.65±0.24	90.78±0.11
EM	90.30±0.24	91.40±0.24	90.30±0.24	89.36±0.23	92.02±0.23	94.37±0.16	89.27±0.28	88.52±0.22	91.71±0.21	90.81±0.12
SOM	90.31±0.24	91.41±0.24	90.31±0.23	89.38±0.23	92.02±0.23	94.38±0.16	89.26±0.29	88.51±0.22	91.73±0.21	90.81±0.12
SOM+EM	90.33±0.23	91.43±0.24	90.33±0.23	89.39±0.23	92.03±0.23	94.40±0.16	89.29±0.28	88.53±0.22	91.73±0.21	90.83±0.11
SOM+KM++	90.34±0.23	91.43±0.24	90.33±0.23	89.40±0.22	92.05±0.23	94.40±0.16	89.30±0.28	88.53±0.22	91.75±0.21	90.84±0.11
EM+KM++	90.39±0.23	91.48±0.24	90.40±0.23	89.43±0.22	92.10±0.23	94.43±0.16	89.36±0.27	88.59±0.22	91.78±0.20	90.88±0.12
KM+EM	90.38±0.23	91.47±0.24	90.38±0.23	89.44±0.23	92.09±0.23	94.40±0.16	89.34±0.28	88.57±0.23	91.76±0.20	90.87±0.13
KM+KM++	90.37±0.23	91.46±0.24	90.37±0.23	89.41±0.22	92.08±0.23	94.43±0.16	89.31±0.28	88.57±0.22	91.75±0.20	90.86±0.12
SOM+KM	90.35±0.23	91.44±0.24	90.35±0.23	89.41±0.23	92.05±0.23	94.44±0.16	89.29±0.28	88.54±0.22	91.76±0.20	90.85±0.13
SOM+EM+KM++	**91.95±0.23**	**93.03±0.24**	**91.95±0.23**	**90.99±0.22**	**93.66±0.23**	**96.00±0.17**	**90.90±0.27**	**90.12±0.22**	**93.34±0.20**	**92.44±0.11**
KM+EM+KM++	90.40±0.23	91.50±0.24	90.39±0.23	89.46±0.23	92.11±0.23	94.46±0.16	89.36±0.28	88.58±0.22	91.81±0.20	90.90±0.11
SOM+KM+EM	90.41±0.23	91.51±0.24	90.41±0.23	89.46±0.22	92.11±0.23	94.47±0.16	89.37±0.27	88.60±0.22	91.81±0.20	90.91±0.12
SOM+KM+KM++	***91.92±0.23***	***93.01±0.23***	***91.92±0.23***	***90.97±0.22***	***93.62±0.22***	***95.99±0.16***	***90.88±0.27***	***90.11±0.22***	***93.30±0.19***	***92.41±0.12***
SOM+EM+KM+KM++	91.91±0.23	93.00±0.23	91.90±0.22	90.96±0.22	93.61±0.22	95.95±0.16	90.86±0.27	90.09±0.22	93.29±0.20	92.40±0.11

Çizelge 5.15'te değerlendirilen kümeleme algoritmaları ile veri setlerinde elde edilen doğru sınıflandırma oranları sunulmuştur. Değerlendirilen yöntemler arasında en yüksek başarım, öz-düzenlemeli harita algoritması, beklenti maksimizasyonu ve K-ortalama++ algoritmalarını içeren kümeleme topluluğu ile elde edilmiştir. İkinci en yüksek başarım ise, K-ortalama, beklenti maksimizasyonu ve K-ortalama++ algoritmalarını içeren kümeleme topluluğu ile alınmıştır. Çizelge 5.15'te, ortak kümelemeye dayalı yöntemler genellikle kümeleme algoritmalarının tek başlarına kullanılmalarına kıyasla daha yüksek başarım elde etmektedir.

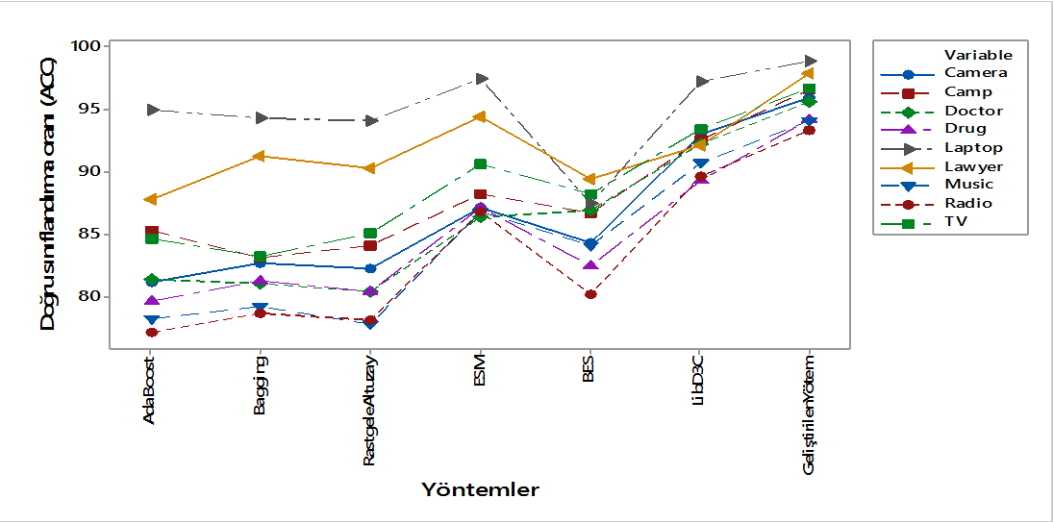
Çizelge 5.16. Geliştirilen yöntemin temel sınıflandırıcı topluluğu yöntemleri ile kıyaslanması

Veri Seti	AdaBoost	Bagging	Rastgele alt-uzay	ESM	BES	LibD3C	Geliştirilen yöntem
Camera	81.21±0.60	82.68±0.65	82.27±0.46	87.11±0.74	84.38±0.62	***93.04±0.61***	**95.92±0.55**
Camp	85.31±0.79	83.12±0.57	84.07±0.82	88.23±0.51	86.76±0.79	***92.6±0.53***	**96.58±0.67**
Doctor	81.41±0.57	81.06±0.46	80.46±0.69	86.38±0.48	86.93±0.74	***92.23±0.47***	**95.65±0.65**
Drug	79.65±0.82	81.26±0.25	80.42±0.42	87.17±0.54	82.5±0.83	***89.33±0.38***	**94.27±0.68**
Laptop	95.00±0.72	94.32±0.80	94.16±0.65	***97.49±0.39***	87.48±0.69	97.32±0.59	**98.92±0.44**
Lawyer	87.82±0.64	91.31±0.45	90.37±0.56	***94.41±0.54***	89.43±0.58	92.16±0.45	**97.90±0.47**
Music	78.26±0.56	79.27±0.74	77.75±0.56	86.85±0.72	84.11±0.47	***90.77±0.54***	**94.16±0.61**
Radio	77.15±0.72	78.68±0.71	78.15±0.38	86.83±0.25	80.16±0.34	***89.67±0.55***	**93.37±0.47**
TV	84.65±0.65	83.21±0.58	85.14±0.50	90.65±0.59	88.21±0.38	***93.46±0.68***	**96.73±0.63**
Ort.	83.38±0.60	83.88±0.55	83.64±0.61	89.46±1.32	85.55±0.99	***92.29±0.79***	**95.94±0.61**

Deneysel analizler sonucunda değerlendirilen tüm farklı yöntemler içerisinde en yüksek başarım, kümeleme algoritması olarak öz-düzenlemeli harita algoritması, beklenti maksimizasyonu ve K-ortalama++ algoritmalarını içeren kümeleme topluluğu, kümeden eleman seçimi yöntemi olarak *Q*-istatistiği ve arama algoritması olarak ENORA algoritması kullanıldığında elde edilmiştir. Çizelge 5.16'da bu yöntemin başarımı, üç temel sınıflandırıcı topluluğu yöntemi (AdaBoost, Bagging ve Rastgele alt-uzay) ve üç temel sınıflandırıcı topluluğu budama yöntemi (ESM, BES ve LibD3C) ile karşılaştırılmıştır. "Camera", "Drug", "Lawyer", "Music" ve "Radio" veri setleri için en yüksek başarımın Bagging yöntemi ile "Camp", "Doctor" ve "Laptop" veri setleri için en yüksek başarımın AdaBoost algoritması ile ve "TV" veri seti için en yüksek başarımın Rastgele alt-uzay yöntemi ile elde edildiği görülmektedir.

Çizelge 5.16'da sunulan temel sınıflandırıcı topluluğu budama yöntemlerinin başarımları değerlendirildiğinde, melez bir budama algoritması olan LibD3C algoritmasının, ESM ve BES yöntemlerine kıyasla genellikle daha

yüksek başarım elde ettiği görülmektedir. Bunun yanı sıra, geliştirilen yöntem, tüm veri setleri için en yüksek doğru sınıflandırma oranını vermektedir. Şekil 5.13'te karşılaştırılmakta olan sınıflandırıcı topluluğu algoritmalarının farklı veri setleri üzerindeki başarımları özetlenmektedir.



Şekil 5.13. Sınıflandırıcı topluluğu algoritmalarının farklı veri setlerinde karşılaştırılması

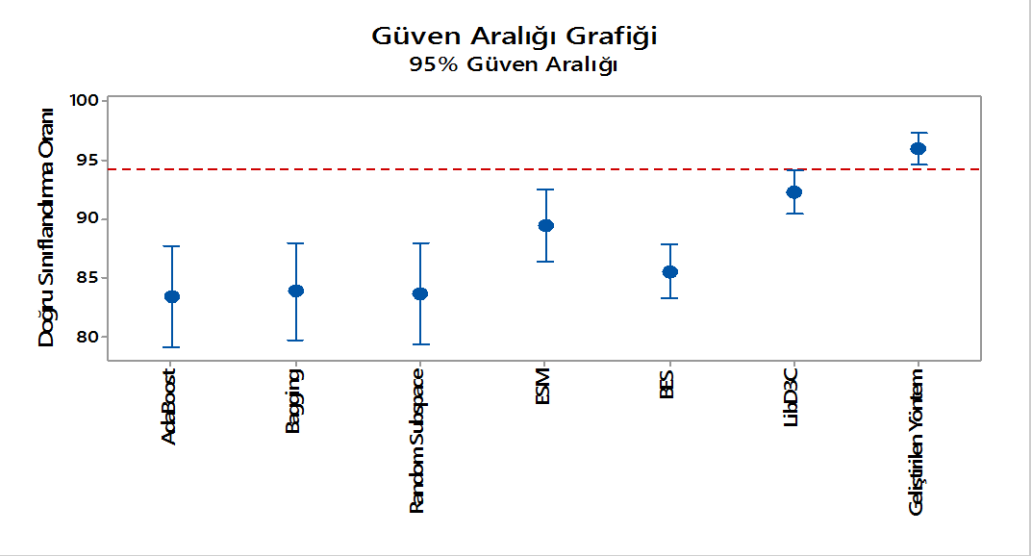
Çizelge 5.17. Topluluk budaması yöntemlerinin çalışma süreleri karşılaştırılması

Yöntem	Camera	Camp	Doctor	Drug	Laptop	Lawyer	Music	Radio	TV
ESM	***92.90***	***121.34***	441.14	***350.29***	**48.35**	**46.29**	232.42	***465.15***	**114.08**
BES	**80.69**	**100.90**	**180.17**	**120.01**	***50.69***	***60.48***	**100.27**	**160.27**	150.64
LibD3C	118.82	148.05	***406.53***	406.85	57.99	76.63	***229.42***	483.48	161.00
Geliştirilen Yöntem	106.33	149.31	434.50	488.06	75.52	97.96	299.57	630.89	209.19

Çizelge 5.17'de topluluk budaması yöntemlerinin çalışma süreleri bakımından karşılaştırılmaları saniye olarak sunulmaktadır. Topluluk budaması yöntemlerine ilişkin sonuçlar incelendiğinde, en iyi (en düşük) çalışma süresi sonuçlarının genellikle, model kütüphanesinden sınıflandırıcı seçimi (ESM) ve Bagging sınıflandırıcı seçimi (BES) yöntemleri ile elde edildiği görülmektedir. LibD3C ve tez çalışması kapsamında geliştirilen yöntem, melez sınıflandırıcı topluluğu budama yöntemleridir. Melez sınıflandırıcı topluluğu budama yöntemleri, birden fazla yöntemin birarada çalıştırılması sonucu sınıflandırıcı topluğunda yer alacak temel sınıflandırma algoritmalarını belirlediğinden dolayı, daha yavaş yaklaşımlardır.

Çizelge 5.18. Topluluk budaması iki-yönlü varyans analizi sonuçları

Kaynak	DF	SS	MS	F	P
Veri seti	8	779.25	97.407	22.02	0.000
Algoritmalar	6	1317.5	219.583	49.64	0.000
Hata	48	212.32	4.423		
Toplam	62	2309.07			



Şekil 5.14. Topluluk budaması yöntemlerine ilişkin güven aralığı grafiği

Temel sınıflandırıcı topluluğu algoritmaları (AdaBoost, Bagging ve rastgele alt-uzay), temel sınıflandırıcı topluluğu budama yöntemleri (ESM, BES ve LibD3C) ile geliştirilen yöntemin başarımı, istatistiksel olarak, Minitab istatistik paket programında yer alan, iki-yönlü varyans analizi (two-way ANOVA test) kullanılarak değerlendirilmiştir. Çizelge 5.18'de iki-yönlü varyans analizine ilişkin sonuçlar aktarılmaktadır. Burada DF, SS, MS, F ve P, sırasıyla, serbestlik derecesi (*degrees of freedom*), uyarlanmış kareler toplamı (*adjusted sum of squares*), uyarlanmış kareler ortalaması (*adjusted mean square*), F-değeri (*F-value*) ve olasılık değerini temsil etmektedir. Çizelgede sunulan iki-yönü varyans analizi sonuçlarına göre, karşılaştırılan algoritmalar ile elde edilen doğru sınıflandırma başarımları arasında istatistiksel olarak anlamlı bir farklılık bulunmaktadır ($p<0.0001$). Şekil 5.14'te karşılaştırılan algoritmaların ortalama doğru sınıflandırma oranlarına ilişkin güven aralığı grafiği (%95 güven aralığı için) sunulmuştur. Karşılaştırılan algoritmalar arasındaki istatistiksel farklılıklara göre, güven aralığı grafiği kırmızı kesikli çizgi ile iki temel bölüme ayrılmıştır. Geliştirilen sınıflandırıcı topluluğu budama yöntemi ile elde edilen güven aralığı, diğer güven aralıkları ile kesişmemektedir. Dolayısıyla, geliştirilen sınıflandırıcı

topluluğu budama yöntemi ile elde edilen yüksek doğru sınıflandırma oranı değerlerinin, diğer karşılaştırılan yöntemlerden istatistiksel olarak anlamlı bir farklılık oluşturduğu görülmektedir.

5.6 Görüş Sınıflandırma Mimarisi Deneysel Sonuçları

Bu bölümde, geliştirilen görüş sınıflandırma mimarisi ile elde edilen deneysel sonuçlar verilmektedir.

5.6.1 Görüş sınıflandırma mimarisi deneysel süreci

Geliştirilen görüş sınıflandırma mimarisinin başarımının değerlendirilmesi amacıyla, görüş sınıflandırmaya ilişkin dokuz farklı alana ait veri setleri (“Camera”, “Camp”, “Doctor”, “Drug”, “Laptop”, “Lawyer”, “Music”, “Radio” ve “TV” veri setleri) kullanılmıştır. Deneysel çalışmalarda, 10-kat çapraz geçerleme yöntemi kullanılmıştır. Geliştirilen görüş sınıflandırma mimarisi, genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçim yöntemi, çok amaçlı diferansiyel gelişim algoritmasına dayalı ağırlıklı oylama yöntemi ve ortak kümeleme ve ENORA algoritmasına dayalı sınıflandırıcı topluluğu budama yöntemini birlikte kullanmaktadır. Bu nedenle, tüm bileşenlerinin Şekil 4.5’teki gibi birleştirildiği yapının yanı sıra, bileşenlerin farklı şekillerde bir araya getirildikleri durumlarda elde edilen başarımları değerlendirmek maksadıyla aşağıda belirtilen konfigürasyonlar dikkate alınmıştır:

- **Öznitelik seçimi + MODE Ağırlıklı oylama:** Burada, öznitelik seçimi aşamasında, genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçimi yöntemi kullanılmıştır. Ardından, indirgenmiş veri seti, Şekil 4.3’te sunulan sınıflandırıcı topluluğu ile birleştirilmektedir. Çok amaçlı ağırlık eniyileme aşamasında, MODE algoritması kullanılmaktadır.

- **Öznitelik seçimi + Topluluk Budaması:** Burada, öznitelik seçimi aşamasında, genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçimi yöntemi kullanılmıştır. Ardından, indirgenmiş veri seti, Bölüm 4.3’te sunulan topluluk budama yöntemi ile birleştirilmektedir.

- **MODE Ağırlıklı Oylama + Topluluk Budaması:** Burada, öznitelik seçimi uygulanmamaktadır. Bölüm 4.3’te sunulan topluluk budama yöntemi, MODE ağırlıklı oylama algoritması kullanılarak birleştirilmektedir.

5.6.2 Görüş sınıflandırma mimarisi deneysel sonuçları

Çizelge 5.19'da görüş sınıflandırma mimarisi ile elde edilen deneysel sonuçlar sunulmaktadır. Değinilen bölümlerde tartışıldığı gibi, geliştirilen öznitelik seçim yöntemi, sınıflandırıcı topluluğu budama yöntemi ve ağırlıklı oylamaya dayalı birleştirme kuralı, ilgili alandaki başlıca yöntemler ile karşılaştırılarak, daha iyi sonuçlar elde edildiği gözlenmiştir. Bu bölümde, tez kapsamında geliştirilen öznitelik seçim yöntemi, sınıflandırıcı topluluğu budama yöntemi ve çok amaçlı diferansiyel gelişim algoritmasının bir arada kullanılması ile elde edilen sonuçlar sunulmaktadır. Çizelge 5.19'da sunulan sonuçlar incelendiğinde, geliştirilen temel yöntemler (öznitelik seçimi, ağırlıklı oylama ve topluluk budaması) içerisinde, en yüksek başarımın, topluluk budaması yöntemi ile elde edilmektedir. İkinci en yüksek başarımın genellikle öznitelik seçim yöntemi ile ve üçüncü en yüksek başarımın genellikle ağırlıklı oylama ile elde edilmektedir. Makine öğrenmesinde, veri setinin uygun bir öznitelik altkümesi ile temsil edilmesi performans açısından oldukça büyük öneme sahiptir. Bu doğrultuda, ağırlıklı oylama ve topluluk budaması yöntemleri, öznitelik seçiminden sonra kullanıldığında, her iki yöntemin de doğru sınıflandırma başarımlarının arttığı gözlenmektedir. Bölüm 4.4'te sunulan görüş sınıflandırma mimarisi, karşılaştırılan yöntemler arasında en yüksek başarımı elde etmektedir.

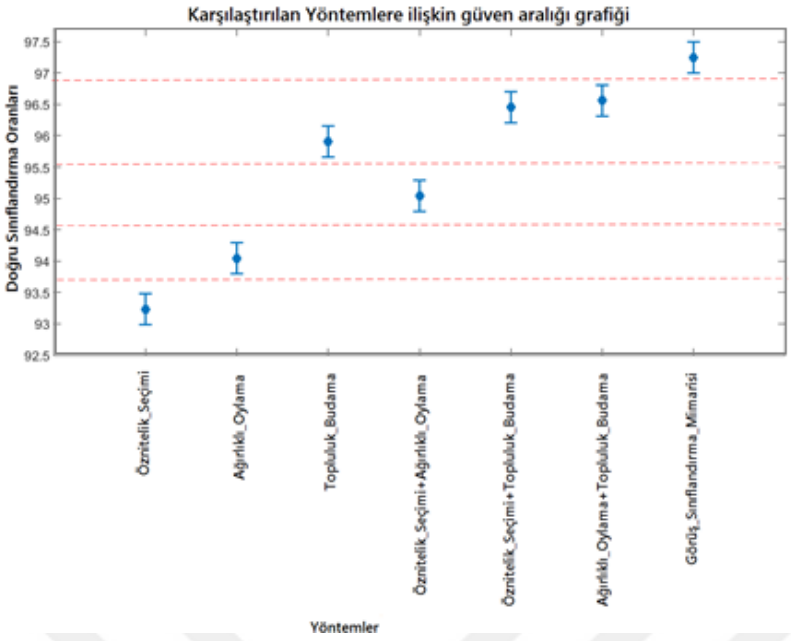
Çizelge 5.19'da karşılaştırılan sonuçlar arasında ikinci en yüksek başarım, "Camera", "Laptop", "Radio" ve "TV" veri setleri için, genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçimi yöntemi, topluluk budama yöntemi ile birlikte kullanıldığında elde edilmektedir. "Camp", "Doctor", "Drug", "Lawyer" ve "Music" veri setleri için ikinci en yüksek başarım, topluluk budama yöntemi, MODE ağırlıklı oylama algoritması kullanılarak birleştirildiğinde elde edilmektedir. Çizelge 5.20'de iki-yönlü varyans analizine ilişkin sonuçlar aktarılmaktadır. Burada DF, SS, MS, F ve P, sırasıyla, serbestlik derecesi (*degrees of freedom*), uyarlanmış kareler toplamı (*adjusted sum of squares*), uyarlanmış kareler ortalaması (*adjusted mean square*), F-değeri (*F-value*) ve olasılık değerini temsil etmektedir. Çizelgede sunulan iki-yönü varyans analizi sonuçlarına göre, karşılaştırılan algoritmalar ile elde edilen doğru sınıflandırma başarımları arasında istatistiksel olarak anlamlı bir farklılık bulunmaktadır ($p<0.0001$).

Çizelge 5.19. Görüş sınıflandırma mimarisinin geliştirilen temel yöntemler ile kıyaslanması

Yöntem	Camera	Camp	Doctor	Drug	Laptop	Lawyer	Music	Radio	TV	Ortalama
Öznitelik Seçimi	93.23±0.52	95.41±0.57	94.12±0.66	91.21±0.49	97.38±0.10	97.39±0.26	90.62±0.54	91.57±0.37	96.47±0.47	94.16±0.52
Ağırlıklı Oylama	92.87±0.53	93.74±0.37	91.05±0.46	89.62±0.62	98.86±0.65	97.87±0.33	89.82±0.63	88.60±0.63	95.74±0.71	93.13±0.43
Topluluk Budama	95.92±0.55	96.58±0.67	95.65±0.65	94.27±0.68	98.92±0.44	97.90±0.47	94.16±0.61	93.37±0.47	96.73±0.63	95.94±0.61
Öznitelik Seçimi + Ağırlıklı Oylama	93.86±0.27	96.57±0.36	95.03±0.26	92.39±0.19	98.97±0.15	97.97±0.27	91.64±0.35	92.33±0.27	96.62±0.30	95.04±0.28
Öznitelik Seçimi + Topluluk Budama	***96.41±0.27***	96.92±0.43	96.16±0.31	94.61±0.33	***99.07±0.29***	98.08±0.27	94.95±0.25	***94.33±0.26***	***97.57±0.29***	96.45±0.19
Ağırlıklı Oylama + Topluluk Budama	96.21±0.23	***96.96±0.29***	***96.51±0.35***	***95.11±0.25***	99.02±0.24	***98.56±0.25***	***95.00±0.18***	94.31±0.20	97.35±0.22	***96.55±0.18***
Görüş Sınıflandırma Mimarisi	**96.72±0.31**	**97.59±0.30**	**97.44±0.28**	**95.67±0.24**	**99.35±0.16**	**99.12±0.19**	**95.27±0.24**	**95.28±0.46**	**98.81±0.17**	**97.25±0.18**

Çizelge 5.20. Görüş sınıflandırma mimarisi iki-yönlü varyans analizi sonuçları

Kaynak	DF	SS	MS	F	P
Veri seti	8	2817.22	352.153	245.98	0.000
Algoritmalar	6	1144.07	190.678	133.19	0.000
Veri seti*Algoritmalar	48	473.72	9.869	6.89	0.000
Hata	567	811.74	1.432		
Toplam	629	5246.75			



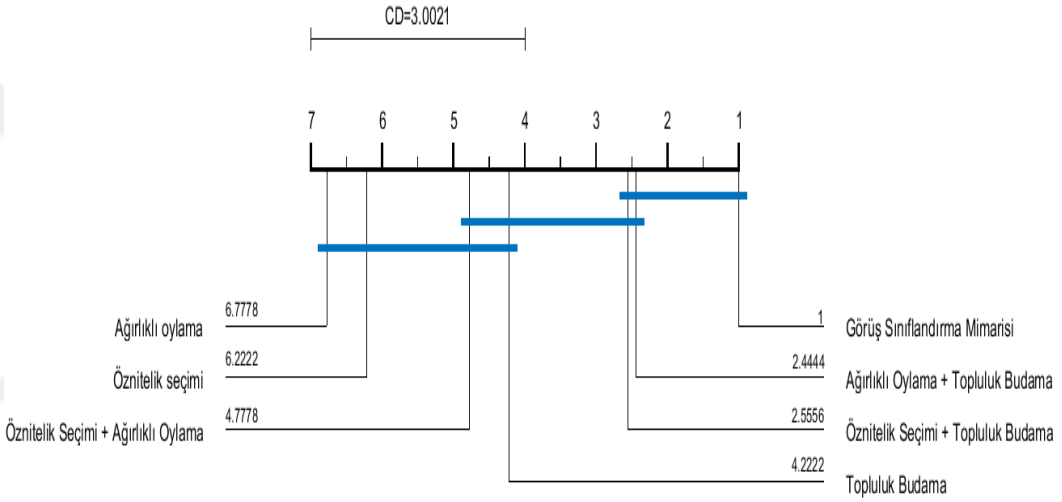
Şekil 5.15. Görüş sınıflandırma mimarisine ilişkin güven aralığı grafiği

Şekil 5.15'te karşılaştırılan algoritmaların ortalama doğru sınıflandırma oranlarına ilişkin güven aralığı grafiği (%95 güven aralığı için) sunulmuştur. Güven aralığı grafiğinden görüldüğü gibi, geliştirilen görüş sınıflandırma mimarisi ile elde edilen yüksek doğru sınıflandırma oranları, öznitelik seçimi, ağırlıklı oylama ve topluluk budama gibi geliştirilen temel yöntemlere ilişkin güven aralıkları ile ve temel yöntemlerin bir arada kullanıldığı diğer birleşimler ile kesişmemektedir.

Çizelge 5.20'de parametrik bir istatistiksel test olan iki yönlü varyans analizine ilişkin sonuçlar sunulmuştur. Buna ek olarak, bu bölümde sunulan yöntemler arasındaki başarı sıralamalarının değerlendirilmesi amacıyla parametrik olmayan bir istatistiksel test olan Friedman testi uygulanmıştır. Friedman testi, farklı sınıflandırma algoritmalarının farklı veri setleri üzerindeki başarımlarının anlamlılığını değerlendirmeye yönelik bir testtir. Friedman testi sıfır hipotezine göre, karşılaştırılan yöntemler arasında anlamlı bir istatistiksel farklılık bulunmamaktadır. Çizelge 5.21'de verilen sonuçlardan görülebildiği gibi, çalışmamızda karşılaştırılan yöntemler için $p<0.0001$ elde edilmiştir. Dolayısı ile sıfır hipotezi reddedilmiştir ve Friedman testine göre karşılaştırılan yöntemlerin başarımları arasında anlamlı bir farklılık bulunmaktadır. Şekil 5.16'da ise karşılaştırılan yöntemlere ilişkin kritik fark diyagramı (CD=3.0021 için) sunulmuştur.

Çizelge 5.21. Görüş sınıflandırma mimarisi Friedman testi sonuçları

Yöntemler	N	Medyan	Sıralamaların Toplamı	Sıralamaların ortalamalası
Öznitelik Seçimi	9	94.10	56	6.2222
Ağırlıklı Oylama	9	93.10	61	6.7778
Topluluk Budama	9	95.85	38	4.2222
Öznitelik Seçimi + Ağırlıklı Oylama	9	95.03	43	4.7778
Öznitelik Seçimi + Topluluk Budama	9	96.33	23	2.5556
Ağırlıklı Oylama + Topluluk Budama	9	96.49	22	2.4444
Görüş Sınıflandırma Mimarisi	9	97.20	9	1
S=51.71, DF=6, p=0.000				



Şekil 5.16. Karşılaştırılan yöntemlere ilişkin Nemenyi kritik fark diyagramı

Şekil 5.16'da verilen Nemenyi kritik fark diyagramı, Friedman testine göre sınıflandırma algoritmalarının başarımları arasında anlamlı farklılık elde edilmesinden sonra uygulanmıştır. Nemenyi testi sonuçlarına göre, karşılaştırmada kullanılan sınıflandırma yöntemleri, üç temel gruba ayrılmaktadır. Nemenyi testine göre, herhangi iki sınıflandırıcının istatistiksel olarak birbirlerinden farklı olabilmesi için, sıralama dereceleri farkının en az kritik değer (CD) kadar olması gerekmektedir. Burada, görüş sınıflandırma mimarisi, ağırlıklı oylama ve topluluk budaması, öznitelik seçimi ve topluluk budaması aynı grupta yer almaktadır. Benzer şekilde, ağırlıklı oylama, öznitelik seçimi ile öznitelik seçimi ve ağırlıklı oylama yöntemleri, bir başka grupta yer almaktadır. Geliştirilen görüş sınıflandırma mimarisi, tüm karşılaştırmalar içerisinde en iyi sıra derecesine sahiptir.

6. SONUÇ VE TARTIŞMA

Bu tez çalışmasında, görüş madenciliği, bir metin sınıflandırma problemi olarak ele alınmış, etkin bir görüş sınıflandırma mimarisi geliştirmek için makine öğrenmesi düzeyinde yöntemler geliştirilerek, başarımın artırılması sağlanmıştır.

Makine öğrenmesi algoritmalarının etkin ve hızlı bir biçimde çalışabilmesi için gerekli en önemli unsurlardan biri, verinin uygun bir biçimde temsil edilmesi ve özniteliklerin seçilmesidir. Metin sınıflandırma alanının en önemli problemlerinden biri, verinin yüksek boyutlu ve seyrek olmasıdır. Bu bağlamda, tez çalışmasının temel hedeflerinden biri, veri setinin etkin bir öznitelik seçim yöntemi kullanılarak, uygun bir öznitelik altkümesi ile temsil edilmesi olmuştur. Bu doğrultuda, filtre tabanlı bireysel öznitelik seçim yöntemleri (bilgi kazancı, ki-kare ölçütü, kazanç oranı ölçütü, simetrik belirsizlik katsayısı, Pearson korelasyon katsayısı, ReliefF algoritması, olasılıklı önem ölçütü gibi), filtre tabanlı grup öznitelik seçim yöntemleri (korelasyon tabanlı öznitelik seçimi, tutarlılık tabanlı öznitelik seçim yöntemi gibi) ve sarmalama tabanlı öznitelik seçim yöntemleri, metin sınıflandırma veri setleri üzerinde değerlendirilmiştir. Filtre tabanlı grup öznitelik seçim yöntemlerinde ve sarmalama tabanlı öznitelik seçim yöntemlerinde, en iyi önce arama algoritması, genetik algoritma, açgözlü arama algoritması, doğrusal ileri seçim algoritması, parçacık sürü eniyilemesi, sıra arama algoritması gibi farklı sezgisel arama yöntemlerinin başarımları incelenmiştir. Deneysel sonuçlar incelendiğinde, metin sınıflandırmada, filtre tabanlı bireysel öznitelik seçim yöntemleri ile elde edilen öznitelik altkümelerinin, filtre tabanlı grup öznitelik seçim yöntemlerine ve sarmalama tabanlı öznitelik seçim yöntemlerine kıyasla daha yüksek doğru sınıflandırma başarımına sahip olduğu gözlenmiştir. Filtre tabanlı bireysel öznitelik seçim yöntemlerinde, öznitelik altkümesi, tüm özniteliklerin belirli bir ölçüte göre sıralanması ve eşik değeri aşan özniteliklerin seçilmesi ile oluşturulur. Filtre tabanlı grup öznitelik seçim yöntemlerinde ise aday öznitelik altkümeleri, öznitelik değerlendirme ölçütü ve sezgisel arama algoritmaları aracılığıyla değerlendirilir. Sarmalama tabanlı öznitelik seçim yöntemlerinde ise öznitelik seçimi, belirli bir öğrenme algoritmasının başarımına dayalı olarak gerçekleştirilir. Dolayısıyla, filtre tabanlı bireysel öznitelik seçim yöntemleri, hesaplama etkinliği bakımından, filtre tabanlı grup öznitelik seçim yöntemlerine ve sarmalama tabanlı öznitelik seçim yöntemlerine kıyasla daha etkin yöntemlerdir. Bu nedenle, tez çalışması kapsamında geliştirilen öznitelik seçim yönteminde, filtre tabanlı bireysel öznitelik seçim yöntemleri, temel yöntemler olarak alınmıştır. Topluluk

öğrenmesi, sınıflandırma ve regresyon başta olmak üzere, temel öğrenme algoritmalarının çıktılarının birleştirilmesi ile daha güçlü kararlar verilmesini amaçlayan bir makine öğrenmesi disiplinidir. Tez kapsamında geliştirilen çalışmada, öznitelik seçimi problemi, topluluk öğrenmesi ilkeleri bağlamında değerlendirilerek, farklı öznitelik seçim yöntemleri ile elde edilen öznitelik altkümelerinin, daha uygun, daha yüksek başarıma sahip bir öznitelik altkümesi elde edilmek üzere, bir araya getirilmesi amaçlanmıştır. Bu doğrultuda, farklı öznitelik seçim yöntemleri ile elde edilen özniteliklere ilişkin sıralamalar, sıra birleştirme problemi aracılığıyla birleştirilmiştir. Sıra birleştirme problemi, farklı bireysel sıralama listelerine olabildiğince yakın yeni bir sıralama elde edilmesini amaçlayan bir eniyileme problemidir. Tez çalışması kapsamında geliştirilen öznitelik seçim yönteminde, genetik algoritmaya dayalı sıra birleştirme yöntemi ile yedi farklı bireysel öznitelik ölçütüyle (bilgi kazancı, ki-kare ölçütü, kazanç oranı ölçütü, simetrik belirsizlik katsayısı, Pearson korelasyon katsayısı, ReliefF algoritması, olasılıklı önem ölçütü) elde edilen sıralamalar birleştirilmiştir. Ortalama sıra birleştirme, medyan sıra birleştirme, en yüksek sıra birleştirme, en düşük sıra birleştirme gibi farklı sıra birleştirme yöntemleri ve geliştirilen sıra birleştirme yöntemi ile metin sınıflandırma veri setleri üzerinde deneysel analizler gerçekleştirilmiştir. Bu deneysel analizlerde, sıra birleştirme yöntemlerinin, bireysel öznitelik ölçütlerinin başarımlarını iyileştirdiği, en yüksek başarımın ise geliştirilen genetik algoritmaya dayalı sıra birleştirme yöntemi ile elde edildiği gözlenmiştir.

Sınıflandırıcı toplulukları, sınıflandırma işleminin, tek bir sınıflandırma algoritması yerine, birden çok öğrenme algoritmasının ortak kararına göre yapılması ile daha etkin sınıflandırma modelleri oluşturulmasına yönelik bir araştırma alanıdır. Makine öğrenmesi alanında, sınıflandırma algoritmalarının başarımlarının artırımında sıklıkla uygulanan sınıflandırıcı toplulukları, görüş sınıflandırma alanındaki makine öğrenmesine dayalı çalışmalarda da başarıyla uygulanmaktadır (Prabowo and Thelwall, 2009; Xia et al., 2011; Wang et al., 2014). Topluluk öğrenmesi süreci, topluluk oluşturma, topluluk budama ve topluluk birleştirme olmak üzere üç temel aşamadan oluşmaktadır. Tez kapsamında etkin bir görüş sınıflandırma mimarisi geliştirilebilmesi için, topluluk öğrenmesi sürecinin topluluk budama ve topluluk birleştirme aşamalarında yöntemler geliştirilmiştir.

Topluluk öğrenmesi sürecinin topluluk birleştirme aşamasında, sınıflandırıcı topluluğunda yer alan temel öğrenme algoritmaları, belirli bir birleştirme kuralı

kullanılarak birleştirilmekte ve sınıflandırıcı topluluğunun çıktısı elde edilmektedir. Etkin bir sınıflandırıcı topluluğu mimarisi tasarımı için, sınıflandırıcıların birleştirilmesinde kullanılan sınıflandırıcı topluluğu birleştirme kuralı büyük önem taşımaktadır. Bu kapsamda, metin sınıflandırma veri setleri üzerinde bağımlı sınıflandırıcı topluluğu yöntemleri, bağımsız sınıflandırıcı topluluğu yöntemleri, temel ağırlıksız oylama yöntemleri, temel ağırlıklı oylama yöntemleri ve yığılmış genelleme algoritması değerlendirilmiştir. Deneysel analizler sonucunda, geliştirilen sınıflandırıcı topluluğu mimarisinde yer alacak sınıflandırma algoritmalarının birleştirilmesinde ağırlıklı oylamaya dayalı bir birleştirme kuralı kullanılmasının daha etkin olabileceği gözlenmiştir. Bu doğrultuda, sınıflandırıcı topluluğunda yer alan sınıflandırma algoritmalarına uygun ağırlık değerleri atanması problemi bir eniyileme problemi olarak ele alınmış ve farklı eniyileme algoritmalarının (genetik algoritma, çok amaçlı benzetilmiş tavlama, çok amaçlı diferansiyel gelişim algoritması gibi) başarımları değerlendirilmiştir. Deneysel analizler sonucunda, geliştirilen mimaride çok amaçlı diferansiyel gelişim algoritması kullanılmasına karar verilmiştir. Burada, çok amaçlı diferansiyel gelişim algoritması kullanılarak, sınıflandırıcıların farklı çıktı sınıflarındaki performanslarına ve amaç fonksiyonlarına dayalı olarak (duyarlık ve geri çağırmaya dayalı amaç fonksiyonu), en uygun ağırlık değerlerinin belirlenmesi amaçlanmaktadır. Deneysel analizler, ağırlıksız oylama yöntemlerinin, temel ağırlıklı oylama yöntemlerinin ve metasezgisel algoritmalara dayalı ağırlıklı oylama yöntemlerinin, temel öğrenme algoritmalarına kıyasla daha yüksek başarım elde ettiğini göstermektedir. Ağırlıklı oylama yöntemleri ise ağırlıksız oylama yöntemlerine kıyasla daha yüksek başarım elde etmektedir.

Topluluk öğrenme sürecinin topluluk budama aşamasında, sınıflandırıcı topluluğunda yer alan öğrenme (sınıflandırma) algoritmalarından bir bölümünün budanması ile daha yüksek başarıma ve daha düşük hesaplama maliyetine sahip bir sınıflandırıcı topluluğu elde edilmesi amaçlanır. Topluluk budama yöntemleri, üstel, rastsal, ardışık, sıralı ve kümelemeye dayalı olmak üzere beş temel ilke doğrultusunda tasarlanmaktadır. Tez çalışması kapsamında, etkin bir sınıflandırıcı topluluğu budama yöntemi geliştirilebilmesi için, temel sınıflandırıcı topluluğu budama algoritmalarından, model kütüphanesinden topluluk seçimi, Bagging sınıflandırıcı seçimi ve LibD3C algoritmalarının başarımları incelenmiştir. Deneysel analizler sonucunda, k-ortalama, dinamik seçim ve ardışık aramaya dayalı melez bir sınıflandırıcı topluluğu budama yöntemi olan LibD3C algoritmasının, metin sınıflandırmada diğer sınıflandırıcı topluluğu budama yöntemlerine kıyasla daha yüksek başarım elde ettiği gözlenmiştir. Bu doğrultuda,

tez kapsamında, metasezgisel arama algoritmaları ile kümelemeye dayalı topluluk budama yöntemlerini bir araya getiren melez bir sınıflandırıcı topluluğu budama yöntemi geliştirilmiştir. Geliştirilen sınıflandırıcı topluluğu budama yönteminde, öncelikle temel sınıflandırma algoritmaları farklı parametre değerleri ile kullanılarak model kütüphanesi oluşturulmaktadır. Ardından, model kütüphanesinde yer alan her bir öğrenme algoritması, eğitim seti üzerinde uygulanarak başarımları kaydedilmektedir. Öğrenme algoritmaları, eğitim setindeki örnekleri doğru/yanlış sınıflandırma karakteristiklerine dayalı olarak kümeleme algoritması kullanılarak kümelere atanmaktadır. Geliştirilen mimaride, başarımın tek bir kümeleme algoritmasının performansını ortadan kaldırmak amacıyla, ortak kümeleme yaklaşımına dayalı bir yöntem benimsenmiştir. Bu yöntemde, sınıflandırma algoritmaları, K-ortalama++, beklenti maksimizasyonu ve öz-düzenlemeli harita kümeleme algoritmaları kullanılarak kümelere atanmaktadır. Ardından, farklı kümeleme algoritmaları ile elde edilen kümeler, nesnelerin farklı bölümlemelerde birlikte görülme durumlarına dayalı olarak birleştirilmektedir. Daha sonra, her bir kümeden, çeşitlilik ölçütüne dayalı olarak aday sınıflandırma algoritmaları seçilmektedir. Aday sınıflandırma algoritması havuzundan hangi algoritmaların, budanmış sınıflandırıcı topluluğunda yer alacağının belirlenmesi ise metasezgisel aramaya (ENORA algoritması) dayalı olarak yapılmaktadır. Tez çalışması kapsamında yapılan analizler sonucunda, geliştirilen melez sınıflandırıcı topluluğu budama yönteminin kümeleme aşamasında, ortak kümelemenin kullanılmasının başarımı artırdığı görülmektedir. Kümeleme algoritmaları sonucu elde edilen bölümlemelerden aday sınıflandırma algoritmalarının seçilmesinde, *Q*-istatistiğinin kullanılmasının başarımı artırdığı görülmektedir. Bunun yanı sıra, aday sınıflandırma algoritması havuzunun ENORA algoritması aracılığıyla değerlendirilmesi, diğer metasezgisel arama algoritmalarına kıyasla daha yüksek başarım sağlamaktadır. Geliştirilen sınıflandırıcı topluluğu budama yönteminin performansı, temel sınıflandırıcı topluluğu budama yöntemleri (AdaBoost, Bagging ve rastgele alt-uzay) ve temel sınıflandırıcı topluluğu budama yöntemleri (model kütüphanesinden sınıflandırıcı seçimi, Bagging sınıflandırıcı seçimi ve LibD3C algoritması) ile karşılaştırıldığında, geliştirilen yöntem ile metin sınıflandırmada daha etkin sonuçlara ulaşıldığı görülmektedir.

Tez kapsamında geliştirilen genetik algoritma ile sıra birleştirmeye dayalı öznitelik seçim yöntemi, sınıflandırıcı topluluğu budama yöntemi ve çok amaçlı eniyilemeye dayalı ağırlıklı oylama birleştirme kuralı, görüş sınıflandırma mimarisi adı verilen bir mimari şeklinde bir araya getirilmiştir. Metin

sınıflandırma alanındaki deneysel çalışmalarda, geliştirilen görüş sınıflandırma mimarisinin etkin sonuçlar verdiği gözlenmiştir.

Tez kapsamında geliştirilen yöntemlerin katkıları şöyle özetlenebilir:

- Metin sınıflandırmada öznitelik seçiminin bir sıra birleştirme problemi olarak ele alınması ve farklı öznitelik seçim yöntemlerinin etkin bir biçimde bir araya getiren, genetik algoritmaya dayalı sıra birleştirme yönteminin geliştirilmesi,

- Sınıflandırma algoritmalarına uygun ağırlık değerleri atanarak, sınıflandırıcı topluluğunun etkin bir şekilde birleştirilmesi probleminin, bir eniyileme problemi olarak ele alınması ve çok amaçlı diferansiyel gelişim algoritmasına dayalı sınıflandırıcı topluluğu birleştirme kuralı geliştirilmesi,

- Metin sınıflandırma ve görüş sınıflandırma alanında, farklı ağırlıksız ve ağırlıklı oylama yöntemlerine ilişkin kapsamlı deneysel analizlerin gerçekleştirilmesi,

- Çok amaçlı diferansiyel gelişim algoritmasının ilk kez sınıflandırıcı topluluğu ağırlık ataması probleminin eniyilenmesinde kullanılması,

- Görüş sınıflandırma alanındaki çalışmalarda ilk kez ağırlıklı oylama düzeyinde başarım iyileştirmesi sağlanması,

- Sınıflandırıcı topluluğu budama için ortak kümeleme ve ENORA algoritmasına dayalı etkin, melez bir yöntem geliştirilmesi,

- Ortak kümeleme yönteminin ilk kez sınıflandırıcı topluluğu budama problemi içerisinde kullanılması ve ENORA algoritmasının ilk kez sınıflandırıcı topluluğu altkümelerini aramak amacıyla uygulanması şeklindedir.

Tez çalışması kapsamında yapılan çalışmaların devamı olarak gerçekleştirilebilecek gelecek çalışmalar şu şekilde özetlenebilir:

- Geliştirilen öznitelik seçimi, ağırlıklı oylama ve sınıflandırıcı topluluğu budama mimarisi, görüş sınıflandırma veri setlerinde test edilmiş ve etkinlikleri değerlendirilmiştir. Geliştirilen yöntemler, doğru sınıflandırma oran ortalamaları bakımından, standart yöntemlere kıyasla daha etkin sonuçlar vermektedir. Makine öğrenmesi alanındaki birçok problem, sınıflandırma işlemi uygulanmasını gerektirmektedir. Bu nedenle, geliştirilen yöntemler, metin sınıflandırma başta olmak üzere, farklı örüntü sınıflandırma problemlerinde uygulanarak etkinlikleri değerlendirilebilir.

- Geliştirilen öznitelik seçim yönteminde, farklı filtre-tabanlı öznitelik seçim yöntemleri ile elde edilen sıralamaların birleştirilmesinde, genetik algoritmaya dayalı sıra birleştirme yöntemi kullanılmıştır. Genetik algoritmanın yanı sıra, diğer metasezgisel arama yöntemlerinin, öznitelik seçimi sıra birleştirmedeki etkinlikleri, metin sınıflandırma ve diğer sınıflandırma problemlerinde incelenerek, daha kapsamlı analizler gerçekleştirilebilir.

- Geliştirilen ağırlıklı oylama sınıflandırıcı topluluğu birleştirme kuralında, çok amaçlı diferansiyel gelişim algoritmasının yanı sıra, farklı metasezgisel yöntemlerin başarımları değerlendirilebilir. Geliştirilen birleştirme kuralı, metin sınıflandırma ve doğal dil işleme alanındaki farklı problemlere uyarlanabilir.

- Geliştirilen sınıflandırıcı topluluğu budama mimarisinin kümeleme aşamasında, farklı kümeleme algoritmaları kullanılarak başarımları incelenebilir. Ortak kümeleme sürecinin, ortak görüş fonksiyonu aşamasında, farklı medyan bölümleme ve nesnelerin birlikte görülmesine dayalı yöntemlerin başarımları değerlendirilebilir.

- Geliştirilen sınıflandırıcı topluluğu budama mimarisinin, kümeden eleman seçim aşamasında, farklı sınıflandırıcılar arası çeşitlilik ölçütlerinin uygunluğu değerlendirilebilir.

- Geliştirilen sınıflandırıcı topluluğu budama mimarisi, başka metin sınıflandırma ve örüntü sınıflandırma problemlerine uyarlanabilir.

KAYNAKLAR DİZİNİ

Abe, S., 2010, Support vector machines for pattern classification, Springer, London, 467p.

Aburomman, A.A. and Reaz, M.B.I., 2016, A novel SVM-kNN-PSO ensemble method for intrusion detection system, *Applied Soft Computing*, 38: 360-372.

Aggarwal, C.C. and Zhai, C.X., 2012, A survey of text classification algorithms, 77-128, Mining Text Data, Aggarwal, C.C. and Zhai, C.X. (Eds.), Springer Verlag, Berlin, 524p.

Aha, D. W., Kibler, D. and Albert, M. K., 1991, Instance based learning algorithms, *Machine Learning*, 6: 37-66.

Ahmad, A. and Dey, L., 2005, A feature selection technique for classificatory analysis, *Pattern Recognition Letters*, 26:43-56.

Aksela, M., 2003, Comparison of classifier selection methods for improving committee performance, *Lecture Notes in Computer Science*, 2709: 84-93.

Aledo, J. A., Gamez, J. A., Molina, D., 2013, Tackling the rank aggregation problem with evolutionary algorithms, *Applied Mathematics and Computation*, 222:632-644.

Arthur, D. and Vassilvitskii, S., 2007, K-means++: the advantage of careful seeding, *Proceedings of the Eighteenth Annual Symposium on Discrete Algorithms*, 1027-1035.

Augustyniak, L., Szymanski, P., Kajdanowicz, T. and Tuliglowicz, W., 2016, Comprehensive study on lexicon-based ensemble classification sentiment analysis, *Entropy*, 18(1): 4.

Bashir, S., Qamar, U. and Khan, F.H., 2015a, Heterogeneous classifiers fusion for dynamic breast cancer diagnosis using weighted vote based ensemble, *Quality and Quantity*, 49: 2061-2076.

Bashir, S., Qamar, U. and Khan, F.H., 2015b, BagMOOV: a novel ensemble for heart disease prediction bootstrap aggregation with multi-objective optimized voting, *Australas Phys Eng Sci Med*, 38: 305-323.

Bhatia, M.P.S. and Khalid, A.K., 2008, Information retrieval and machine learning: supporting technologies for web mining research and practice, *Webology*, 5(2).

KAYNAKLAR DİZİNİ (devam)

Birant, D., 2011, Comparison of decision tree algorithms for predicting potential air pollutant emissions with data mining models, *Journal of Environmental Informatics,* 17(1):46-53.

Bors, A. G., 2001, Introduction of the Radial Basis Function (RBF) Networks, *Online Symposium for Electronics Engineers,* 1(1):1-7.

Bouaguel, W., Brahim, A.B. and Limam, M., 2013a, Feature selection by rank aggregation and genetic algorithms, *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval and the International Conference on Knowledge Management and Information Sharing,* 74-81.

Bouaguel, W., Mufti, G.B. and Limam, M., 2013b, Rank aggregation for filter feature selection in credit scoring, *Lecture Notes in Computer Science,* 8284:7-15.

Breiman, L., 1996, Bagging Predictors, *Machine Learning,* 24: 123-140

Broomhead, D.S., and Lowe, D., 1988, Multivariate function interpolation and adaptive networks, *Complex Systems,* 22:321-355.

Burduk, R., 2015, Method of static classifiers selection using the weights of based classifiers, 85-94, Soft Computing in Computer and Information Science, Wilinski, A., Fray, I.E., Pejas, J. (Eds.), Springer Verlag, Berlin, 446p.

Cao, J., Kwong, S., Wang, R., Li, X., Li, K. and Kong, X., 2015, Class-specific soft voting based multiple extreme learning machines ensemble, *Neurocomputing,* 149: 275-284.

Caruana, R., Niculescu-Mizil, A., Crew, G. and Ksikes, A., 2004, Ensemble selection from libraries of models, *Proceedings of ICML 04,* 18-18.

Chandrashekar, G. and Sahin, F., 2014, A survey on feature selection methods, *Computers and Electrical Engineering,* 40:16-28.

Chen, J., Huang, H., Tian, S. and Qu, Y., 2009, Feature selection for text classification with Naïve Bayes, *Expert Systems with Applications,* 36:5432-5435.

Costa, E.P., Lorena, A.C., Carvalho, A.C.P.L.F. and Freitas, A.A., 2007, A review of performance evaluation measures for hierarchical classifiers, *Proceedings of the AAAI-2007 Workshop,* 182-196.

KAYNAKLAR DİZİNİ (devam)

Da Silva, N.F.F., Hruschka, E.R. and Hruschka, E.R., 2014, Tweet sentiment analysis with classifier ensembles, *Decision Support Systems*, 66:170-179.

Dai, Q., 2013, A novel ensemble pruning algorithm based on randomized greedy selective strategy and ballot, *Neurocomputing*, 122: 258-265.

Dash, M. and Liu, H., 1997, Feature selection for classification, *Intelligent Data Analysis*, 1:131-156.

Deb, K., Pratap, A., Agarwal, S. and Meyarivan, T., 2002, A fast and eltist multiobjective genetic algorithm: NSGA-II, *IEEE Transactions on Evolutionary Computation*, 6(2):182-197.

Dehzangi, A. and Karamizadeh, S., 2011, Solving protein fold prediction problem using fusion of heterogeneous classifiers, *Information*, 14(11):3611-3621.

Dempster, A.P., Laird, N.M. and Rubin, D.B., 1977, Maximum likelihood from incomplete data via the em algorithm, *Journal of the Royal Statistical Society*, 39:1-38.

Diao, R., 2014, Feature selection with harmony search and its applications, PhD Thesis, Aberystwyth University, 231p (unpublished).

Dietterich, T.G., 2000, Ensemble methods in machine learning, 1-15, Multiple Classifier Systems, Kittler, J. (Ed.), Springer-Verlag, Berlin.

Dittman, D.J., Khoshgoftaar, T.M., Wald, R. and Napolitano, A., 2013, Classification performance of rank aggregation techniques for ensemble gene selection, *Proceedings of the Twenty-Sixth International FLAIRS Conference*, 420-425.

Dwork, C., Kumar, R., Naor, M. and Sivakumar, D., 2001, Rank aggregation methods for the Web, *Proceedings of the 10th International World Wide Web Conference*, 613-622.

Ekbal, A. and Saha, S., 2011a, A multi-objective simulated annealing approach for classifier ensemble: named entity recognition in Indian languages as case studies, *Expert Systems with Applications*, 38: 14760-14772.

Ekbal, A. and Saha, S., 2011b, Weighted vote-based classifier ensemble for named entity recognition: a genetic algorithm-based approach, *ACM Transactions on Asian Language Information Processing*, 10(2):9-37.

KAYNAKLAR DİZİNİ (devam)

Ekbal, A. and Saha, S., 2013, Combining feature selection and classifier ensemble using a multi-objective simulated annealing approach: application to named entity recognition, *Soft Computing*, 17:1-16.

Engelbrecht, A.P., 2007, Computational intelligence: an introduction, John Wiley & Sons, New York, 628p.

Fersini, E., Messina, E. and Pozzi, F.A., 2014, Sentiment analysis: Bayesian ensemble learning, *Decision Support Systems*, 68:26-38.

Fogel, D., 1988, An evolutionary approach to the traveling salesman problem, *Biological Cybernetics*, 60(2):139-144.

Fowlkes, E.B. and Mallows, J.A., 1983, A method for comparing two hierarchical clustering, *Journal of American Statistical Association*, 78: 553-569.

Fred, A.L.N. and Jain, A.K., 2005, Combining multiple clustering using evidence accumulation, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27: 835-850.

Freund, Y., and Schapire, R.E., 1996, Experiments with a new boosting algorithm, *Machine Learning: Proceedings of the Thirteenth International Conference*, 325-332.

Garcia-Gutierrez, J.G., Mateos-Garcia, D., Garcia, M. and Riquelme-Santos, J.C., 2015, An evolutionary-weighted majority voting and support vector machines applied to contextual classification of Lidar and imagery data fusion, *Neurocomputing*, 163:17-24.

Garcia-Laencina, P.J., Sancho-Gomez, J-L., and Figueiras-Vidal, A.R., 2010, Pattern Classification with missing data: a review, *Neural Computing Applications*, 19(2): 263-282.

Gashler, M., Giraud-Carrier, C. and Martinez, T., 2008, Decision tree ensemble: small heterogeneous is better than large homogeneous, *Proceedings of ICMLA '08,* 900-905.

Gehrke, J., 2003, Decision trees, 3-24, The Handbook of Data Mining, N. Ye (Ed.), Lawrence Erlbaum Associates Publishers, London.

Genkin, A., Lewis, D.D. and Madigan, D., 2007, Large-scale Bayesian logistic regression for text classification, *American Statistical Association and the American Society for Quality Technometrics*, 49(3):291-304.

KAYNAKLAR DİZİNİ (devam)

Ghaemi, R., Sulaiman, M.N., Ibrahim, H. and Mustapha, N., 2009, A survey: clustering ensemble techniques, *World Academy of Science, Engineering and Technology*, 50: 636-645.

Ghosh, J. and Acharya, A., 2011, Cluster ensembles, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(4): 305-315.

Goldberg, D.E. and Richardson, J., 1987, Genetic algorithms with sharing for multimodal function optimization, *Proceedings of the Second International Conference on Genetic Algorithms*, 41-49.

Gunal, S., 2012, Hybrid feature selection for text classification, *Turkish Journal of Electrical Engineering and Computer Science*, 20(S2):1296-1311.

Gütlein, M., 2006, Large scale attribute selection using wrappers, Diploma Thesis, University of Freiburg, 135p (unpublished).

Guyon, I. and Elisseeff, A., 2003, An introduction to variable and feature selection, *Journal of Machine Learning Research*, 3:1157-1182.

Hall, M. A. and Holmes, G., 2003, Benchmarking attribute selection techniques for discrete class data mining, *IEEE Transactions on Knowledge and Data Engineering*, 15(6):1437-1447.

Hall, M. A. and Smith, L.A., 1998, Practical feature subset selection for machine learning, *Proceedings of the 21st Australasian Computer Science Conference*, 181-191.

Hall, M. A., 1999, Correlation-based feature selection for machine learning, PhD Thesis, University of Waikato, 198p (unpublished).

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. and Witten, I.H., 2009, The WEKA Data mining software: an update, *SIGKDD Explorations*, 11(1):10-18.

Han, J. and Kamber, M., 2006, Data Mining Concepts and Techniques, Morgan Kaufmann Publishers, San Francisco, 743p.

Haury, A.C., Gestraud, P. and Vert J.P., 2011, The influence of feature selection methods on accuracy, stability and interpretability of molecular signatures, *Plos One*, 6(12):e28210.

KAYNAKLAR DİZİNİ (devam)

Ho, T.K., 1998, The random subspace method for constructing decision forests, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8):832-844.

Jain, A.K., 2010, Data clustering: 50 years beyond k-means, *Pattern Recognition Letters*, 31: 651-666.

Javed, K., Maruf, S. and Babri, H.A., 2015, A two-stage Markov blanket based feature selection algorithm for text classification, *Neurocomputing*, 157:91-104.

Jimenez, F., Sanchez, G. and Juarez, J.M., 2014, Multi-objective evolutionary algorithms for fuzzy classification in survival prediction. *Artificial Intelligence in Medicine*, 60: 197-219.

Jin, X. and Han, J., 2010, Expectation maximization clustering, 382-383, Encylopedia of Machine Learning, Sammut, C. and Webb G, I. (Eds.), Springer Verlag, Berlin.

Jong, K., Mary, J., Cornuejols, A., Marchiori, E. and Sebag, M., 2004, Ensemble feature ranking, *Lecture Notes in Computer Science*, 3202:267-278.

Kim, H., Kim, H., Moon, H. and Ahn, H., 2011, A weight-adjusted voting algorithm for ensembles of classifiers, *Journal of the Korean Statistical Society*, 40(4):437-449.

King, M.A., Abrahams, A.S. and Ragsdale, C.T., 2015, Ensemble learning methods for pay-per-click campaign management, *Expert Systems with Applications*, 42:4818-4829.

Kira, K. and Rendell, L. A., 1992a, The feature selection problem: traditional methods and a new algorithm, *Proceedings of Ninth National Conference on Artificial Intelligence*, 129-134.

Kira, K., and Rendell, L. A., 1992b, A practical approach to feature selection, *Proceedings of International Workshop on Machine Learning*, 249-256.

Kohonen, T., 2001, Self-organizing maps, Springer-Verlag, Berlin.

Koller, D. and Sahami, M., 1996, Toward optimal feature selection, *Proceedings of International Conference on Machine Learning*, 284-292.

KAYNAKLAR DİZİNİ (devam)

Kotsiantis, S.B. and Pintelas, P.E., 2005, Selective averaging of regression models. Annals of Mathematics, *Computing & Teleinformatics*, 1(3): 65-74.

Kuncheva, L., 2014, Combining Pattern Classifiers: Methods and Algorithms, John Wiley & Sons Publishers, New Jersey, 376p.

Kuncheva, L.I. and Whitaker, C.J., 2003, Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy, *Machine Learning*, 51: 181-207.

Langley, P., 1994, Selection of relevant features in machine learning, *Proceedings of the AAAI Fall Symposium on Relevance*, 1-5.

Larranaga, P., Kuijpers, C.M.H., Murga, R.H., Inza, I. and Dizdarevic, S., 1999, Genetic algorithms for the travelling salesman problem: a review of representations and operators, *Artificial Intelligence Review*, 13:129-170.

Lima, A.C.E.S., de Castro, L.N. and Corchado, J.M., 2015, A polarity analysis framework for Twitter messages, *Applied Mathematics and Computation*, 270:756-767.

Lin, C., 2011, Probabilistic topic models for sentiment analysis on the web, PhD Thesis, University of Exeter, 135p (unpublished).

Lin, C., Chen, W., Qiu, C., Wu, Y., Krishnan, S. and Zhou, Q., 2014, LibD3C: ensemble classifiers with a clustering and dynamic selection strategy, *Neurocomputing*, 123: 424-435.

Lin, Y., Wang, X., Li, Y. and Zhou, A., 2015, Integrating the optimal classifier set for sentiment analysis, *Social Network Analysis and Mining*, 5:1-13.

Liu, B., 2012, Sentiment analysis and opinion mining, Morgan & Claypool, New York, 168p.

Margineantu, D.D. and Dietterich, T.G., 1997, Pruning adaptive boosting, *Proceedings of the Fourteenth International Conference on Machine Learning*, 211-218.

Martinez-Munoz, G. and Suarez, A., 2006, Pruning in ordered bagging ensembles, *Proceedings of the 23rd International Conference on Machine Learning*, 609-616.

KAYNAKLAR DİZİNİ (devam)

Medhat, W., Hassan, A. and Korashy, H., 2014, Sentiment analysis algorithms and applications: a survey, *Ain Shams Engineering Journal,* 5(4):1093-1113.

Mendes-Moreira, J., Soares, C., Jorge, A.M. and De Sousa, J.F., 2012, Ensemble approaches for regression: a survey, *ACM Computing Surveys*, 45(1): 10-39.

Mendialdua, I., Arruti, A., Jauregi, E., Lazkano, E. and Sierra, B., 2015, Classifier subset selection to construct multi-classifiers by means of estimation of distribution algorithms, *Neurocomputing,* 157: 46-60.

Mesleh, A.M., 2011, Feature sub-set selection metrics for Arabic text classification, *Pattern Recognition Letters,* 32(14):1922-1929.

Mezura-Montes, E., Reyes-Sierra, M. and Coello Coello, C.A., 2008, Multi-objective optimization using differential evolution: a survey of the state of the art, 173-196, Advances in Differential Evolution, Chakraborty, U.K. (Ed.), Springer-Verlag, Berlin.

Mirkin, B., 2001, Reinterpreting the category utility function, *Machine Learning*, 45(2): 219-228.

Mitchell, M., 1999, An Introduction to Genetic Algorithms, MIT Press, London.

Moreno-Seco, F., Inesta, J.M., Leon, P.J. and Mico, L., 2006, Comparison of classifier fusion methods for classification in pattern recognition tasks, *Lecture Notes in Computer Science*, 4109:705-713.

Narendra, P. M. and Fukunaga, K., 1977, A branch and bound algorithm for feature selection, *IEEE Transactions on Computers,* 26(9):917-922.

Niuniu, X., and Yuxun, L., 2010, Review of decision trees, *Proceedings of the Third IEEE International Conference on Computer Science and Information Technology,* 105-109.

Obitko, M., 1998, "Introduction to genetic algorithms", http://www.obitko.com/tutorials/ (Erişim Tarihi: Mayıs 2016)

Onan, A. and Korukoğlu S., 2015a, A feature selection model based on genetic rank aggregation for text sentiment classification, *Journal of Information Science*, doi:10.1177/0165551515613226 (in press).

KAYNAKLAR DİZİNİ (devam)

Onan, A. and Korukoğlu, S., 2015b, Ensemble methods for opinion mining, *Proceedings of the 23th IEEE Signal Processing and Communications Applications Conference (SIU),* 212-215.

Onan, A., Korukoğlu, S. and Bulut, H., 2016, A multiobjective weighted voting ensemble classifier based on differential evolution algorithm for text sentiment classification, *Expert Systems with Applications*, 62: 1-16.

Orr, M.J.L., 1996, Introduction to Radial Basis Function Networks, Technical Report, University of Edinburg, Scotland.

Pang, B. and Lee, L., 2004, A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts, *Proceedings of the Annual Meeting on Association for Computational Linguistics (ACL),* 271-278.

Pang, B., Lee, L. and Vaithyanathan, S., 2002, Thumbs up?: sentiment classification using machine learning techniques, *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP),* 79–86.

Panov, P., and Džeroski, S., 2007, Combining bagging and random subspaces to create better ensembles, *Lecture Notes in Computer Science*, 4723:118-129.

Partalas, I., Tsoumakas, G. and Vlahavas, I., 2012, A study on greedy algorithms for ensemble pruning, Technical Report, Aristotle University of Thessaloniki, 37p (unpublished)

Partalas, I., Tsoumakas, G., Katakis, I. and Vlahavas, I., 2006, Ensemble pruning via reinforcement learning, *Proceedings of the 4th Hellenic Conference on Artificial Intelligence*, 301-310.

Pihur, V., Datta, S. and Datta, S., 2007, Weighted rank aggregation of cluster validation measures: a Monte Carlo cross-entropy approach, *Bioinformatics*, 23(13):1607-1615.

Pinto, F., 2013, Metalearning for dynamic integration in ensemble methods, Thesis Proposal, University of Porto (unpublished).

Polikar, R., 2006, Ensemble based systems in decision making, *IEEE Circuits and Systems Magazine*, 6(3):21-45.

Prabowo, R., and Thelwall, M., 2009, Sentiment analysis: a combined approach, *Journal of Informetrics*, 3:143-157.

KAYNAKLAR DİZİNİ (devam)

Prati, R.C., 2012, Combining feature ranking algorithms through rank aggregation, *Proceedings of the International Joint Conference on Neural Networks*, 1-8.

Price, K.V., Storn, R.M. and Lampinen J.A., 2005, Differential evolution: a practical approach to global optimization, Springer, Berlin, 539p.

Provost, F.J., and Kolluri, V., 1997, A survey of methods for scaling up inductive learning algorithms, *Proceedings of 3rd International Conference on Knowledge Discovery and Data Mining*, 131-169.

Psychas, I.D., Delimpasi, E. and Marinakis, Y., 2015, Hybrid evolutionary algorithms for the multi-objective traveling salesman problem, *Expert Systems with Applications*, 44(22):8956-8970.

Ranawana, R. and Palade, V., 2006, Multi-classifier systems: review and a roadmap for developers, *International Journal of Hybrid Intelligent Systems*, 3(1):35-61.

Rich, E. and Knight, K., 1991, Artificial Intelligence, McGraw-Hill, New York.

Robnik-Sikonja, M. and Kononenko, I., 2003, Theoretical and empirical analysis of ReliefF and RReliefF, *Machine Learning*, 53:23-69.

Rokach, L. and Maimon, O., 2010, Classification Trees, 149-175, Data Mining and Knowledge Discovery Handbook, Maimon, O. and Rokach, L. (Eds.), Springer Verlag, New York.

Rokach, L., 2010, Ensemble-based classifiers, *Artificial Intelligence Review*, 33:1-39.

Roli, F., Giacinto, G. and Vernazza, G., 2001, Methods for designing multiple classifier systems, *Lecture Notes in Computer Science*, 2096: 78-87.

Ruta, D. and Gabrys, B., 2001, Application of the evolutionary algorithms for classifier selection in multiple classifier systems with majority voting, Proceedings *of the Second International Workshop on Multiple Classifier Systems*, 399-408.

Ruta, D. and Gabrys, B., 2005, Classifier selection for majority voting, *Information Fusion*, 6:63-81.

KAYNAKLAR DİZİNİ (devam)

Saha, S. and Ekbal, A., 2013, Combining multiple classifiers using vote based classifier ensemble technique for named entity recognition, *Data & Knowledge Engineering*, 85: 15-39.

Sarkar, C., Cooley, S. and Srivastava, J., 2014, Robust feature selection technique using rank aggregation, *Applied Artificial Intelligence*, 28(3):243-257.

Schwenker, F., Kestler, H.A., and Palm, G., 2001, Three learning phases for radial-basis-function networks, *Neural Networks*, 14:439-458.

Sebastiani, F., 2002, Machine learning in automated text categorization, *ACM Computing Surveys*, 34(1):1-47.

Selim, S.Z. and Ismail, M.A., 1984, K-means-type algorithms: a generalized convergence theorem and characterization of local optimality, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6(1): 81-87.

Shatkay, H. and Craven, M., 2012, Mining the Biomedical Literature, MIT Press, London, 150p.

Sheen, S. and Sirisha, A.P., 2013, Malware detection by pruning of parallel ensemble using harmony search, *Pattern Recognition Letters*, 34: 1679-1686.

Sheydaei, N., Saraee, M. and Shahgholian, A., 2015, A novel feature selection method for text classification using association rules and clustering, *Journal of Information Science*, 41(1):3-15.

Shmueli, G., Patel, N. R. and Bruce, P. C., 2010, Data Mining for Business Intelligence: Concepts, Tecniques and Applications in Microsoft Office Excel with XLMiner, John Wiley & Sons Publishers, New Jersey.

Storn, R. and Price, K., 1997, Differential evolution simple and efficient heuristic for global optimization over continuous spaces, *Journal of Global Optimization*, 11(4):341-359.

Strehl, A. and Ghosh, J., 2003, Cluster ensembles - a knowledge reuse framework for combining multiple partitions, *Journal of Machine Learning Research*, 3: 583-617.

Sun, Q. and Pfahringer, B., 2011, Bagging ensemble selection, *Proceedings of the 24th Australasian Joint Conference on Artificial Intelligence*, 251-260.

KAYNAKLAR DİZİNİ (devam)

Sun, Q., 2014, Meta-learning and the full model selection problem, PhD Thesis, University of Waikato (unpublished).

Swiderski, B., Osowski, S., Kruk, M. and Barhoumi, W., 2016, Aggregation of classifiers ensemble using local discriminatory power and quantiles, *Expert Systems with Applications*, 46: 316-323.

Syswerda, G., 1991, Schedule optimization using genetic algorithms, 332-349, Handbook of Genetic Algorithms, Davis, L. (Ed.), Van Nostrand Reinhold, New York.

Taboada, M., Brooke, J., Tofiloski, M., Voll, K. and Stede M., 2011, Lexicon-based methods for sentiment analysis, *Computational Linguistics*, 37(2):267-307.

Tamon, C. and Xiang, J., 2000, On the boosting pruning problem, *Lecture Notes in Computer Science*, 1810: 404-412.

Tan, F., 2007, Improving feature selection techniques for machine learning, PhD Thesis, Georgia State University, 111p (unpublished).

Tan, P.N., Steinbach, M. and Kumar, V., 2005, Introduction to Data Mining, Addison-Wesley, Boston.

Tan, S. and Zhang, J., 2008, An empirical study of sentiment analysis for Chinese documents, *Expert Systems with Applications*, 34:2622-2629.

Theodoridis, S. and Koutroumbas, K., 1999, Pattern Recognition, Academic Press, New York.

Ting K.M., and Witten, I.H., 1997, Stacking bagged and dagged models, *Proceedings of 14th International Conference on Machine Learning*, 367-375.

Tinos, R., and Murta, L. O. J., 2009, Use of the q-Gaussian Function in Radial Basis Function Networks, *Foundations of Computational Intelligence*, 5:127-145.

Tsoumakas, G., Partalas, I. and Vlahavas, I., 2008, A taxonomy and short review of ensemble selection, *Proceedings of ECAI 08 Workshop on Supervised and Unsupervised Ensemble Methods and Their Applications*, 1-6.

KAYNAKLAR DİZİNİ (devam)

Uysal, A.K. and Gunal, S., 2012, A novel probabilistic feature selection method for text classification, *Knowledge-based Systems*, 36:226-235.

Vapnik, V., 1995, The Nature of Statistical Learning Theory, Springer-Verlag, New York.

Vega-Pons, S. and Ruiz-Shulcloper, J., 2011, A survey of clustering ensemble algorithms, *International Journal of Pattern Recognition and Artificial Intelligence*, 25(3): 337-372.

Wald, R., Khoshgoftaar, T.M. and Wald, R., 2012, Ensemble gene selection versus single gene selection: which is better?, *Proceedings of the Twenty-Sixth FLAIRS Conference*, 350-355.

Wang, G., Sun, J., Ma, J., Xu, K. and Gu, J., 2014, Sentiment classification: the contribution of ensemble learning, *Decision Support Systems*, 57:77-93.

Wang, G., Zhang, Z., Sun, J., Yang, S. and Larson, C.A., 2015, POS-RS: A random subspace method for sentiment classification based on part-of-speech analysis, *Information Processing and Management*, 51:458-479.

Wang, H., Qian, G. and Feng, X.Q., 2013, Predicting consumer sentiment using online sequential extreme learning machine and intuitionistic fuzzy sets, *Neural Computing and Applications*, 22(3-4):479-489.

Wang, S., Li, D., Song, X., Wei, Y. and Li, H., 2011, A feature selection method based on improved fisher's discriminant ratio for text sentiment classification, *Expert Systems with Applications*, 38:8696-8702.

Whitehead, M, 2014, "Sentiment Analysis Datasets", http://personalwebs.coloradocollege.edu/~mwhitehead/files/datasets/ (Erişim Tarihi: Kasım 2014)

Whitehead, M. and Yaeger, L., 2009, Building a general purpose cross-domain sentiment mining model, *Proceedings of WRI World Congress on Computer Science and Information Engineering*, 472-476.

Witten, I.H., Frank, E., and Hall, M.A., 2011, Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann Publishers, Burlington.

Wolpert, D.H., 1992, Stacked generalization, 1992, Stacked generalization, *Neural Networks*, 5(2):241-259.

KAYNAKLAR DİZİNİ (devam)

Xia, H., Xia, Z. and Wang, Y., 2016, Ensemble classification based on supervised clustering for credit scoring, *Applied Soft Computing Journal*, 43:73-86.

Xia, R., Zong, C. and Li, S., 2011, Ensemble of feature sets and classification algorithms, *Information Sciences,*181:1138-1152.

Xue, B., Zhang, M. and Browne, W.N., 2013, Particle swarm optimization for feature selection in classification: a multi-objective approach, *IEEE Transactions on Cybernetics*, 43(6): 1656-1671.

Yang, Y. and Pedersen, J. O., 1997, A comparative study on feature selection in text categorization, *Proceedings of the Fourteenth International Conference on Machine Learning*, 412-420.

Zhang, C., Zeng, D., Li, J., Wang, F.Y. and Zuo, W., 2009, Sentiment analysis of Chinese documents: from sentence to document level, *Journal of the American Society for Information Science and Technology*, 60(12):2474-2487.

Zhang, D., Xu, H., Su, Z. and Xu, Y., 2015, Chinese comment sentiment classification based on word2vec and SVMperf, *Expert Systems with Applications*, 42:1857-1863.

Zhang, H. and Cao, L., 2014, A spectral clustering based ensemble pruning approach, *Neurocomputing*, 139: 289-297.

Zhang, P. and He, Z., 2015, Using data-driven feature enrichment of text representation and ensemble technique for sentence-level polarity classification, *Journal of Information Science*, 41(4): 531-549.

Zhang, Y., Zhang, H., Cai, J. and Yang, B., 2014, A weighted voting classifier based on differential evolution, *Abstract and Applied Analysis*, 2014:1-6.

Zhao, Y., and Zhang, Y., 2008, Comparison of Decision Tree Methods for Finding Active Objects, *Advances in Space Research*, 41(12):1955-1959.

Zheng, N. and Xue, J., 2009, Statistical learning and pattern analysis for image and video processing, Springer, Berlin, 365p.

Zhou, Z-H. and Tang, W., 2003, Selective ensemble of decision trees, *Lecture Notes in Computer Science*, 2639:476-483.

KAYNAKLAR DİZİNİ (devam)

Zhou, Z-H., 2012, Ensemble methods: foundations and algorithms, CRC Press, New York, 236p.

Zhou, Z-H., Wu, J. and Tang, W., 2002, Ensembling neural networks: many could be better than all, *Artificial Intelligence*, 137: 239-263.

Zhu, Z., Ong, Y-S. and Dash, M., 2007, Wrapper-filter feature selection algorithm using a memetic framework, *IEEE Transactions on Systems, Man, Cybernetics*, 37(1):70-76.

ÖZGEÇMİŞ

AYTUĞ ONAN

aytugonan@gmail.com

EĞİTİM

Lisans, İzmir Ekonomi Üniversitesi, Bilgisayar Mühendisliği Bölümü, 2010.
Yüksek Lisans, Ege Üniversitesi, Bilgisayar Mühendisliği Bölümü, 2013.
Doktora, Ege Üniversitesi, Bilgisayar Mühendisliği Bölümü, 2013-devam ediyor.

AKADEMİK DENEYİM

Araştırma Görevlisi, Ege Üniversitesi, Bilgisayar Mühendisliği Bölümü, 2010-2013.
Araştırma Görevlisi, Celal Bayar Üniversitesi, Yönetim Bilişim Sistemleri Bölümü, 2013-2014.
Araştırma Görevlisi, Celal Bayar Üniversitesi, Bilgisayar Mühendisliği Bölümü, 2014-devam ediyor.

BAŞARILAR, ÖDÜLLER VE BURSLAR

TÜBİTAK BİDEB 2211- Yurt İçi Doktora Bursu, 2013-2016.
TÜBİTAK BİDEB 2210- Yurt İçi Yüksek Lisans Bursu, 2010-2012.
TÜBİTAK Uluslararası Bilimsel Yayınları Teşvik Programı, 2015, 2016.
İzmir Ekonomi Üniversitesi Başarı Bursu, 2007-2010.
İzmir Ekonomi Üniversitesi Yüksek Şeref Öğrencisi, 2006-2010.

SCI/SCI-E KAPSAMINDAKİ DERGİLERDE YAYINLANAN MAKALELER

- Onan A, Korukoğlu S, Bulut H. (2016). Ensemble of keyword extraction methods and classifiers in text classification. *Expert Systems with Applications,* 57: 232-247.
- Onan A, Korukoğlu S, Bulut H. (2016). A multiobjective weighted voting ensemble classifier based on differential evolution algorithm for text sentiment classification. *Expert Systems with Application,* 62: 1-16.
- Onan A, Korukoğlu S. (2015). A feature selection model based on genetic rank aggregation for text sentiment classification. *Journal of Information Science,* doi: 10.1177/0165551515613226.

DİĞER HAKEMLİ DERGİLERDE YAYINLANAN MAKALELER

- Onan A, Korukoğlu S, Bulut H. (2016). LDA-based topic modelling in text sentiment classification: an empirical analysis. *International Journal of Computational Linguistics and Applications.*
- Onan A, Korukoğlu S. (2016). Makine öğrenmesi yöntemlerinin görüş madenciliğinde kullanılması üzerine bir literatür araştırması. *Pamukkale Üniversitesi Mühendislik Bilimleri Dergisi,* 22(2): 111-122.

ULUSLARARASI/ULUSAL BİLİMSEL KONFERANSLARDA SUNULAN BİLDİRİLER

- Onan A, Korukoğlu S. (2016). Exploring performance of instance selection methods in text sentiment classification. *Artificial Intelligence Perspectives in Intelligent Systems,* Springer Switzerland, pp. 167-179.
- Onan A, Korukoğlu S. (2016). Metin sınıflandırmada öznitelik seçim yöntemleri, Akademik Bilişim Konferansı (AB 2016), 3-5 Şubat 2016, Aydın-Türkiye.
- Onan A, Korukoğlu S. (2015). Görüş madenciliğinde sınıflandırıcı toplulukları, IEEE 23. Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SİU 2015), 16-19 Mayıs 2015, Malatya-Türkiye.

EKLER

Ek 1 Camera Veri Setinden Bir Kesit

We bought this camera and have been more than happy with it's performance. We are not professional photographers, we just needed something easy, cheap, and reliable. This camera is all those things! The battery issue we heard about does not seem to be a problem, the pictures come out crisp, and it could not be easier to learn how to use. We are very pleased with this product.
1.0
I have owned several high end digital SLR cameras including the Canon XT and XTi. I have also owned several point and shoot cameras including the Canon G7 and the Panasonic TZ3. This camera is a blend that combines the performance of the DSLRs and the ease of use of the point and shoot.The camera features the ultrasonic focusing motor of the Canon DSLR lens which makes it focus faster than most point and shoot cameras. Along with that speed comes a focusing accuracy similar to a DSLR. The 12x zoom range has an amazing reach, equivalent to 36mm to 432mm (in a 35mm film camera).The controls on the camera are well laid out. It is evident that Canon has learned from all previous models and they finally have things about right. With the additional adapters it is possible to add ND filters or a tele-converter to give a top end of 950mm or more (Raynox 2020).For me the DSLR equipment load was just too much. If your camera and lens cost so much you can't take them on vacation, they are worthless no matter how good the pictures might be. With the S5 IS you get great pictures AND a camera you can afford to replace if it gets stolen or damaged.
1.0
I'm considering returning this camera.Functions are not so bad considering the price. But, battery life is really short. It's maybe because this camera does not work on the voltage which is a little lower than full.I could use the battery battered from this camera for the other camera.
-1.0
This camera is amazing! The 7.1 megapixels make extremely clear pictures and the size is great for a point-and-shoot camera. It is working out great for business and personal!
1.0
I've not been a fan of Olympus' most recent compact digital cameras, but the Stylus 710 is winning me over. First, it's just a gorgeous camera. It's unusual shape makes it appear very small once it's in your hands. Its metal finish looks smart yet rugged. Controls on such a small camera are always less than ideal, but they're done well here.The menu is very good, especially in the photo example/text description it gives you in the scene modes. Don't have your manual? Just turn the selection dial to the "guide" mode, and it'll get you there!The weather proof ability sets this apart from most of the competition. You may not plan on using your camera in the rain, but the rain may not plan on your taking pictures!As I've stated before, I'm not a big fan of the xD cards. SD is simply dominate right now, easier to find, more capatible with more things, and usually easier to find a bargin. But the new "H" series cards are of equal quality to the best SD cards. But xD card and charger with a 3' cord (v. flipout plug), drops this winner one star.
1.0

Ek 2 Camp Veri Setinden Bir Kesit

This was my daughter's 2nd year at the camp and she just loved it. Next year I will send her for 3 or 4 weeks. The overall experience was a good one for her. I find YMCA Camping Services to be a warm, nurturing, safe environment for my daughter and I would recommend this camp to friends, relatives and assosiates.
1.0
I loved my time as a staffer here!
1.0
Hiawatha Youth Camp is the best place ever. Everyone in the WORLD should go there. This is the place where I found Christ and so did many others. I have only went once but I am definitely going again this year. I would like to see the people I met last year (like Guthrie :D) and my old councelors. I get to go in two months and I can't WAIT!!!
1.0
It simply doesn't get any better than Becket!
1.0
I was a program counselor at this camp, and from looking at other camps, this one had great facilities (eg, most camps dont have bathrooms in each cabin). I have some great memories, and still hear from some of my campers, and friends from around the world! However, I thought that there was MORE drama w/ the counselors than w/ the campers! It felt like middle school, all the gossip of who is hooking up with whom...and some counselors putting their personal life over being there for their campers.
1.0
Best camp in the world. Its hard to explain how much this camp has changed my life over the last 6 summers. As a 10 year old going with my brother and his frind, I was some what scared, but the counselors and other campers at The Nest welcomed me into the community and made me crave coming back after each three week session. Each camper learns a new meaning of freindship, cummunity, and family when they attend camp that can only be defined by expirience. If there is any question of whether or not to go to this camp, just go, and if you dont like it then it is not for you but I will find that really hard to believe with the kind of camp Eagles Nest is. Each summer I have matured as a member of society and as a human being. This once again is the best camp that you will find.
1.0
KENNOLYN is the BEST! even though i have been going for only 3 years, it's the highlight of my summer..i have met some of the coolest and nicest friends and we will always keep in touch. the activities are SO fun and i always feel like part of a family when i go to kennolyn. Kennolyn is so beautiful and i love the atmostphere of the staff, campers, and environment! my memories at kennolyn will be with me forever
1.0
FWF if my LIFE!!! i am sooo excited to go back this summer!!! i go for 2 sessions and i NEVER want to go home!! it is my summer home and i could never be anywhere else..IF YOU ARE LOOKING FOR AN UNFORGETTABLE SUMMER...COME TO FWF!!!!! you make your best friends...and it is so much fun! i will always treasure my times at FWF...i miss my campiess <3
1.0

Ek 3 Doctor Veri Setinden Bir Kesit

bad
-1.0
I really like Dr. Greaney. However, his office receptionists are rude (on many occasions). I am thinking of leaving his practice for this very reason.
1.0
He is nice but seems to make up his opinion before you're done talking. Also, he seems uncomfortable doing breast exams and taking sexual histories.
-1.0
Very friendly and listens to her patients.
1.0
Wonderful Doctor. I cant say enough good things about him.
1.0
Although Dr. Shoukri is polite, and did explain things..he would only tell patient the very basics, and leave important details out, which I found out later by getting a second opinion. He had the tendency to be condesending, and failed to write down changes in my medication, which caused problems with my pharmacy. Also, he was never on time. I felt like my physical well-being was of no concern to him. I would Not recommend this Dr.
-1.0
Takes time to listen to issues and explain all options. Very professional, personable and knows how to put children at ease.
1.0
Dr. Gianturco is a wonderful doctor. Very lively personality and genuinely concerned with your well being and getting you answers.
1.0
Very kind and considerate. Willing to listen and takes lots of notes.
1.0
I do like Wanner. She has always fixed whatever I came in for or either set me up with a referrel. My insurance as well as her office staff drive me nuts because while I know Dr. Wanner is flexible with writing me a referrel, getting that referrel out of the office staff is like pulling teeth. Expect to wait. I've never been taken on time. A normal wait time for me was 1hr - 1hr 1/2 although I have waited up to 2 hrs before being seen by the Dr. She has always taken her time with me and the one time she was rushed and I asked her to slow down for me she did. So, thats why I've kept going to her.
1.0
I would like anyone who is considering seeing Dr.Poliakin that you are making the BEST choice for yourself and your baby. I had 3 babies with another OB before I found Dr Poliakin!!It was night and day! He delivered my last 2 children and he is the BEST!!! If you have to wait in his office, it is because YOU are important to him and he takes his time with each of his patients. He never makes you feel rushed, no matter how busy he may be. Babies come at all hours so if he has a delivery to do then it is out of his hands! I knew that while I was pregnant my baby and I had the best possible care and I continue to have that trust in him today. Dr. Poliakin also has a WONDERFUL, LOVING and CARING staff! I agree with everyone who said they were sad after their babies were born because they missed the office. It truly is a great feeling to know you are in such TRUSTED hands.
1.0

Ek 4 Drug Veri Setinden Bir Kesit

I WAS ON YAZ FOR 6 WEEKS AND THOSE WERE THE WORST 6 WEEKS OF MY LIFE. MY MUSCLES TWITCHED AND JERKED, LOWER BACK HURT AND BREAST SO TENDER.I QUIT 2 WEEKS INTO 2ND PACK. NEVER AGAIN WILL I TAKE IT
-1.0
I was put on Cymbalta for nerve pain which it has not helped. I was not depressed at the time. I started on 60mg, had severe nausea and headaches, then was moved to 30mg, the same thing, so then 20mg and from there went back up to 60mg. It has been 4 months and I want to get off of it. The Dr. gave me 30mg for a week and then is going to give me 20mg. I hope this works! I haven't been sleeping because of it! Be careful before you start taking it, it is expensive and horrible to get off of! Not to mention all the drug interactions!
-1.0
I spend $56.05 on this medication per month it is killing me, I can not afford it! I have been taking this medication since 11/06 I was just fine, but the past 2 months I have had sugary urine, bad mood swings and almost no period at all. I have gained 10 lbs i am miserable. On the 13th of July I am going to my Doctor to have blood work done and be taken off this pill. I have been on so many contaceptives that I no longer know what to take. I have been looking at the Yaz, but I just hope it isn't as much as I pay per month and the side effects, please no more severe side effects.
-1.0
So, I've been taking Ortho Tri Cyclen Lo for about a month & 1/2 now. I'm the last set of blue pills in the second month. Yesterday, I had a HORRIBLE panic attack out of nowhere. I've gone through panic attacks from time to time in my life, but nothing like this one. Has anyone else experienced panic attacks from this drug or any other birth control pill?
-1.0
i would have to wake up and vomit at 2 AM every morning while taking this pill. didnt have any other side effects, but the vomiting was just too much. once i threw up 5 times in one night.
-1.0
I have been on Lutera for a little over a month. It is effective, but I feel very bloated and my period never came (cramps did, but not my period) and I have had a lot of breast tenderness. I am going to stick with it for a while to see if things get better.... But over all it is not bad, but it is the first bc pill that I have taken so I have nothing to compare it to.... but I have not had any nausea or anything....well see.
-1.0
I want to begin by saying I'm very appreciative for all of your honest reviews! They made me realize that some fatigue and pain during sex may have been from the Ring. I've been very tired lately and have periodic pain during sex that I'd never had before, also some loss of sex drive that I attributed to chronic fatigue! Anyway I will definitely reconsider this BC method, even though I had awesome light and short periods.. I Still sometimes forgot to put it in and take it out on the right day. And so my search continues for the right contraceptive! Good luck ladies.
1.0

Ek 5 Laptop Veri Setinden Bir Kesit

i bought this laptop because a had to change my vaio, and I think that toshiba is an excellent device, but to be honest i want to change vista.
1.0
I have a Toshiba Satellite A215 and experienced the blank screen pretty much every day. Wasted many hours on the phone with Toshiba techs reinstalling Vista, updating the Video Driver and BIOS which DOESN'T WORK. So I thought what the hey I'll just go without a computer for a week and have Toshiba fix it. Sent it in (which was a major hassle to go through UPS) and about a week later I got it back. Whoo Hoo!! Finally, a working computer right?! WRONG!! About 2-3 hours after running it BLANK SCREEN STRIKES AGAIN!!!! (All they did was update BIOS) So for those of you thinking about sending your A215 in to Toshiba DON'T DO IT!! It will come back still defective. My story however I believe at this point ends happily because of a suggestion someone posted on this site a couple weeks ago. This was the user's suggestion "If you are running VISTA, try going to the Power Options, advanced options, processor power management, and change the value of the minimum processor state to 100%." (THANKS RICARDO!!!) It's been a week now and still no blank screen. My computer NEVER has gone that long without blanking out so I'm extremely hopeful the problem is solved. Hope this works for everyone..... -Trevor
-1.0
I just adore this laptop! I am a recent convert to Mac and treated myself to the 17 inch MacBook Pro when I first opened my small business earlier this year. I liked it a lot so bought myself this small 13 inch to use at home and to carry on business trips. Well, I soon found myself completely attached to this smaller machine. It is lightweight and has a battery that lasts an impressively long time. I can carry it anywhere and be set up to go in a flash, wherever there is a wireless connection. And the speed of the wireless is terrific! I also purchased an Airport Extreme and am now running my home network through its new, faster n standard. I have an old digital camera that was a huge hassle to use with Windows. I had to download software and there were always glitches when I tried to upload photos from the camera card. Now, with the MacBook's iPhoto, no software is necessary. I just plug the camera into the laptop via a USB connector and the software does the rest. I had pretty much given up on digital photography, but now find that I am using the camera a lot and easily sharing pictures with friends over the Web. The only drawback to this laptop is that it gets quite hot when it's in use for awhile. (I've noticed that happens with my MacBook Pro too, as well as with other laptops I've had. But the Pro doesn't present a problem because I don't use it on my lap because it's a bit too large for comfort). So I purchased a lightweight, insulated lap pad and now I have no problem keeping the MacBook on my lap for hours.I am incredibly enthused about the 13 inch MacBook and highly recommend it to everyone. It is a great value for the price. I only wish I had converted to Mac sooner.
1.0

Ek 6 Lawyer Veri Setinden Bir Kesit

I do not know anything about politics but this judge is not fair. If he favors a law firm, he will try and give assistance at trial to the favored firm by letting the other side ask any question they want and by letting all their irrelevant evidence in and he will not give the same lattitude to the not favored other side. People have to know what really goes on and contact your political party leaders to ask questions. They are allowing the bottom of the barrel to get on the Judiciary Bench in NYS. Apparently this Judge does not care about the children of litigants and the importance of children's well-being to become productive members of this society. This is a shame and he should be held accountable for not caring about children. In addition, this Judge does not care about the proposition live within your means and that this should be taught to children. He favors the non working and non productive rich against the hard working person with good morals and ethicss. Disgrace!
-1.0
Very smart and aggressive divorce lawyer. He was great in protecting me and my kids.
1.0
Thomas Mesereau: Very professional. He ROCKS! Gotta love the hair *wink* :D Go T-Mez!
1.0
Outstanding Knowledge of the law. Very good trail lawyer. Very satisified!
1.0
He is very professional and competent. He will do his homework and follow through. He's very knowledgable and was of great assistance. He is wonderful about communicating with you about your case.
1.0
Marina is the best attorney I know. She's help me and many members of my family.
1.0
She's a real asset to the deparment and county of L.A.
1.0
An extroidinary lawyer with enviable skills, high integrity and even higher principles.
1.0
Mr. Mesereau is a great man. He is the best lawyer in the world. He did a wonderful job on the Michael Jackson case.I can't thank him enough his work on the MJ case.
1.0
This woman is a monster - she illegally garnishes the wages of men who have always paid their child support. The Los Angeles Times (July 2006) recently reported how she trivialized the fact that her department kept an abused child was hidden from her father for more than a decade - the father and the daughter are now suing Cruz and the Los Angeles County Child Support Services Department (Do a Google Search for Dad, daughter separated by negligence) and Cruz fought a California Supreme Court Ruling that would require that her corrupt agency stop collecting bogus child support from men who were NOT the fathers of the children that Cruz was stealing from (again, do a Google search for the Washington Times Article entitled, Court asked to 'depublish' child-support ruling)
-1.0

Ek 7 Music Veri Setinden Bir Kesit

The Boss gives best preformance since Born in the USA.Little bit sad but very consistent rock.Fans would love it
1.0
I always believed that Carrie wouldn't fail in make a good second album, even with the pressure of the awards that she won, especially the Grammy's.The album is more country sound than "Some Hearts" and that's not a bad thing, because even sounding more country she will keep her fans and probably will make new ones.I was surprised how Carrie sounded good singing the uptempo songs, she improved a lot and sounds more honest than she sounded in "Some Hearts"My favorite's songs are "You Won't Find This", "The More Boys I Meet" and "So Small".I think Carrie made an album that will make her fans happy, her label happy and herself too, just the haters that won't.
1.0
I have been a Britney fan for a long time, but after seeing the VMA's I was skeptical about how "Blackout" would turn out. But, after reading critic reviews about it actually being good, I thought that I would try it out... It is AWESOME!! I absolutely love this record!! She has done it again. It has many club-worthy beats and songs to workout to (which I love!!). Go buy it Go buy it!! You will not be sorry!!!
1.0
I was excited about getting this compact disc about Wicked, the hottest ticket on Broadway. I liked the beginning about how the witch was born green. I think the compact disc does serve a service in providing us a closer view of the musical itself. I didn't care for the story involving Glinda, the good witch, and Elfaba, the bad witch who is green. I didn't like the story itself. While I am sure that the musical is much better than just listening to it but I was uncomfortable with Elfaba's portrayal as being born green and bad while Glinda was both popular and pretty. I think I was just more sympathetic towards Elfaba because she was the outcast and I always find myself among the outcasts so I felt sorry for Elfaba. I don't find much sympathy for Glinda who has spent her entire life being both popular and gifted so of course, she doesn't change. She gets rewarded while poor Elfaba gets treated as bad.
-1.0
To my friends Chris, Duffy, Doug, Christy, Cathy C, Josie, Cathy G, Cindy, Tom and the whole AU Crew. We rocked on "E" street when it all began. They were the best of times...with the aurora rising behind us. This album is fun and at times reminds me of the early days. Especially the songs 'Girls in their summer Clothes' and "I'll work for your love'. These are classic Springsteen tunes that really remind me of the fun we had together. Now I am Fat and old...and rich (thanks Howie). Who can ever forget those early days in McDowell Hall. Remember when Bruce, Clarence and the crew came to visit? Where was I? Drunk probably. Well you guys really rock and I miss the good ol days. What ever happened to Crazy Dave's cat? Thanks for the memories. Thanks Bruce, Little Stevie and the "E" street band...and, of course...the Big Man..The General
1.0

Ek 8 Radio Veri Setinden Bir Kesit

He was syndicated in my city on regular radio and I listened for awhile, but I finally got bored. Tell the truth, he just sounded kind of tired. Maybe the new format will fire him up, but how challenging and hungry can you be if you're worth . . . a half billion dollars!! I think young Howard would make fun of old Howard. Besides, Sirius equipment is junk. I went with XM and I find OnA to be more than a worthy replacement, they're very funny. Sorry, Howard.
-1.0
I like it better on the old station format, it seems that they are trying to do to much now. I love this guys, it is a complete entertainment
1.0
Ugh. Is this show even on anymore I thought it was canned in April due to (sigh) poor ratings News to me if he made it.
-1.0
This guy sucks. Get a true progressive. Get Sam back. Lionel is so lame, I dont even listen anymore.
-1.0
THE GREATEST GUY SHOW IN THE COUNTRY.
1.0
Awful show. Unoriginal/ripped off material. Pathetic.
-1.0
ALL ANTI-HUSKERS MUST LEAVE THIS STATE.
-1.0
Most either really like the guys or hate them. Expect all ones and fives for ratings.
1.0
JV & Elvis are the new blood to NY radio.Current FM radio in NY was getting old & boring until The Dog House hit the scene.
1.0
Great show, very funny
1.0
i think they should let the interviews flow without stepping on the guests yelling BORING gets boring
1.0
This show is ok. Kinda strange though with only 2 guys and 1 of them is doing all the voices of the other characters that dont really exist lol.
-1.0
So not funny. Like two retarded high school kids.
-1.0
Bring them back they are the bomb
1.0
I grew up listening to him. He was great until his damned enormous ego got in the way. Now he is just a shell of what he was.
-1.0
I was a diehard fan. I wish he would do the show again. He takes so much time off it's not worth the subscription. SAD!
-1.0
hoo hoo who
1.0

Ek 9 TV Veri Setinden Bir Kesit

While the networks continue to produce mindless drivel, like America's Got Talent, or The OC, intelligent shows like Scrubs continue to get the cold shoulder. It's a shame. Scrubs is by far the smartest and funniest show on television right now.
1.0
It had some good moments but its over, let it go.
-1.0
Really enjoy this show, but it seems to never be on. Could ABC just run it 4 weeks in a row It loses it's must see status when they do reruns every few weeks.
1.0
This show is pretty good I only watch 3 times though.
1.0
RAY ROMANO IS NOT FUNNY, PLEASE DONT ENCOURAGE HIM
-1.0
This and the office makes the best hour on tv.
1.0
slow beginning, but not later on. Warrants another viewing. good and spooky
1.0
one of the best shows this season how ever i never know when its on fox changed the times
1.0
ADORE HUGH LAURIE!! I look forward to every new episode just to see what Hugh will do next. Did I say that I ADORE HUGH LAURIE Good luck at the Emmys Hugh. I will have my fingers crossed for you and if you win, you will hear me clapping and yelling!!!
1.0
I wish he would go back to doing more interviews with representatives.
1.0
David Carusso's character is kind of weird, however I still watch the show. It's good.
1.0
This show has some of the best characters and plots and it keeps getting better.
1.0
James Spader's and WIlliam Shatner's charachters must have been separated at birth. They MAKE this show ... bth as screwed up as the other, and they feed on each other. Great chemistry between the actors. Wacky show ... original writing.
1.0
Ilove this Show! This show makes me want to be a doctor because it show the cast being a famliy!
1.0
abo****ely hysterical from the start
1.0
this show sucks... and you're stupid if you actually like it
-1.0
great show. very entertaining!
1.0
It makes me want to go back to Vegas
1.0
Terrible show. Just isn't that funny or informative. Colbert should have stayed on the Daily Show where he was an actual asset to television.
-1.0

Ek 10 Türkçe-İngilizce Terimler Sözlüğü

Açgözlü tepe tırmanma	*Greedy hill climbing*
Adaptif destekleme	*AdaBoost*
Ağırlıklı oylama	*Weighted voting*
Ağırlıksız oylama	*Unweighted voting*
Alıcı işletim karakteristiği eğrisi	*ROC curve*
Ardışık budama yöntemleri	*Sequential pruning methods*
Artırımlı toplu öğrenme	*Incremental batch learning*
Aşırı öğrenme makinesi	*Extreme learning machine*
Ayrık örneklem	*Disjoint samples*
Bağımlı sınıflandırıcı toplulukları	*Dependent methods*
Basit ağırlıklı oylama	*Simple weighted voting*
Basit rastgele yerine koyarak örnekleme	*Simple random sampling with replacement*
Bayes model ortalaması	*Bayesian model averaging*
Bayesci lojistik regresyon	*Bayesian logistic regression*
Beklenti maksimizasyonu	*Expectation maximization*
Belge-seviyesi görüş madenciliği	*Document-level opinion mining*
Benzetilmiş tavlama	*Simulated annealing*
Bilgi kazancı	*Information gain*
Birlikte görülme matrisi	*Co-association matrix*
Çaprazlama	*Crossover*
Çeşitlilik	*Diversity*
Çift hata ölçütü	*Double fault measure*
Çizge tabanlı	*Graph based*
Çoğunluk oylaması	*Majority voting*
Çoğunluk oylaması hatası	*Majority voting error*
Çok amaçlı eniyileme	*Multiobjective optimization*
Değerlendiriciler arası uyuşma	*Inter-rater agreement*
Destek vektör makineleri	*Support vector machines*
Destekleme	*Boosting*
Diferansiyel gelişim algoritması	*Differential evolution algorithm*
Doğru sınıflandırma oranı	*Accuracy rate*
Duyarlılık	*Precision*
Duygu analizi	*Sentiment analysis, opinion mining*
En düşük sıra birleştirme	*Lowest rank aggregation*
En iyi önce arama	*Best first search*
En iyi-en kötü ağırlıklı oylama	*Best-worst weighted voting*
En yüksek sıra birleştirme	*Highest rank aggregation*
Evrimsel programlama	*Evolutionary programming*
Fayda fonksiyonu	*Utility function*
Filtre-tabanlı	*Filter-based*
Fisher diskriminant oranı	*Fisher's discriminant ratio*
Fowlkes-Mallows indeksi	*Fowlkes-Mallows index*
F-ölçütü	*F-measure*

Ek 10 Türkçe-İngilizce Terimler Sözlüğü (devam)

Genetik algoritmalar	*Genetic algorithms*
Genetik çeşitlenme	*Recombination*
Genetik programlama	*Genetic programming*
Geri çağırma	*Recall*
Gizli konu	*Latent topic*
Görüş kutbu	*Opinion polarity*
Görüş madenciliği	*Opinion mining, Sentiment analysis*
Gürbüz sıra birleştirme	*Robust rank aggregation*
Hata oranı	*Error rate*
İkili turnuva seçimi	*Binary tournament selection*
İkinci dereceden en iyi-en kötü ağırlıklı oylama	*Quadratic best-worst weighted voting*
İleri arama	*Forward search*
İyileştirilmiş Borda sıra birleştirme	*Enhanced Borda rank aggregation*
Kalabalık mesafesi	*Crowding distance*
Karar ağacı	*Decision tree*
Kararlılık seçimi birleştirme	*Stability selection aggregation*
Kararsız	*Unstable*
Karesel ortalama hata	*Root mean squared error*
Karınca kolonisi eniyilemesi	*Ant colony optimization*
Kat değişim oranı	*Fold change ratio*
Kazanç oranı	*Gain ratio*
K-en yakın komşu algoritması	*K-nearest neighbour algorithm*
Kendall'ın tau uzaklığı	*Kendall's tau distance*
Ki-kare ölçütü	*Chi-squared metric*
Konum tabanlı çaprazlama	*Position based crossover*
K-ortalama	*K-means*
K-ortalama++	*K-means++*
Koşullu rastgele alan	*Conditional random field*
Küme oluşturma	*Cluster generation*
Küme toplulukları	*Cluster ensembles*
Kümeleme analizi	*Cluster analysis*
Kümelemeye dayalı budama yöntemleri	*Clustering based pruning methods*
Lineer diskriminant analizi	*Linear discriminant analysis*
Lojistik regresyon	*Logistic regression*
Maksimum olasılık	*Maximum probability*
Medyan bölümleme	*Median partition*
Medyan sıra birleştirme	*Median rank aggregation*
Metasezgisel yöntemler	*Metaheuristic methods*
Minimum olasılık	*Minimum probability*
Model güdümlü örneklem seçimi	*Model guided instance selection*
Mutasyon	*Mutation*
Nesnelerin birlikte görülmesi	*Object co-occurence*
Normalize edilmiş karşılıklı bilgi	*Normalized mutual information*
NP-tam	*NP-complete*

Ek 10 Türkçe-İngilizce Terimler Sözlüğü (devam)

Olasılıklar çarpımı	*Product of probability*
Olasılıklar ortalaması	*Average of probabilities*
Olasılıklı önem ölçütü	*Probabilistic significance measure*
On-kat çapraz geçerleme	*Ten-fold cross validation*
Ortak görüş fonksiyonu	*Consensus function*
Ortak kümeleme	*Consensus clustering*
Ortalama sıra birleştirme	*Mean rank aggregation*
Öğreticili öğrenme	*Supervised learning*
Öğreticisiz öğrenme	*Unsupervised learning*
Öklit uzaklığı	*Euclidean distance*
Ölçeklenmiş ağırlıklı oylama	*Rescaled weighted voting*
Öz-düzenlemeli haritalar	*Self-organizing maps*
Öznitelik seçimi	*Feature selection*
Parçacık sürüsü eniyilemesi	*Particle swarm optimization*
Pearson korelasyon katsayısı	*Pearson correlation coefficient*
Q-istatistiği	*Q-statistics*
Radyal tabanlı fonksiyon ağları	*Radial basis function networks*
Rastgele alt-uzay	*Random Subspace*
Rastgele orman	*Random Forest*
Rastsal budama yöntemleri	*Randomized pruning methods*
Rehberli yerel arama	*Guided local search*
ROC eğrisi altındaki alan	*Area under ROC curve*
Sabit kural sınıflandırıcı topluluğu birleştirme kuralı	*Fixed-rule output combination*
Saflık	*Purity*
Saklı Markov model	*Hidden Markov model*
Sapma	*Deviance*
Sarmalama-tabanlı	*Wrapper-based*
Sınıflandırıcı seçimi	*Classifier selection*
Sınıflandırıcı toplulukları	*Classifier ensemble, ensemble of classifiers*
Sıra birleştirme	*Rank aggregation*
Sıralı budama yöntemleri	*Ranked pruning methods*
Simetrik belirsizlik katsayısı	*Symetrical uncertainty coefficient*
Sinyal gürültü oranı	*Signal to noise ratio*
Sözcük torbası	*Bag of words*
Spearman uzaklığı	*Spearman footrule distance*
Stokastik	*Stochastic*
Tam arama	*Exhaustive search*
Terim sıklığı	*Term frequency*
Terim varlığı	*Term presence*
Topluluk birleştirme	*Ensemble integration*
Topluluk budama	*Ensemble pruning*
Topluluk oluşturma	*Ensemble generation*
Topluluk öğrenmesi	*Ensemble learning*

Ek 10 Türkçe-İngilizce Terimler Sözlüğü (devam)

Tümcenin ögeleri	*Part of speech*
Tümce-seviyesi görüş madenciliği	*Sentence-level opinion mining*
Uygunluk paylaşımı	*Fitness sharing*
Uyuşmazlık ölçütü	*Disagreement measure*
Üstel ağırlıklı sıra birleştirme	*Exponential weighting rank aggregation*
Üstel budama yöntemleri	*Exponential pruning methods*
Varlık ismi çıkarımı	*Named entity recognition*
Varlık/özellik seviyesi görüş madenciliği	*Entity/aspect level opinion mining*
Yarı-öğreticili öğrenme	*Semi-supervised learning*
Yeniden etiketleme	*Relabelling*
Yerel arama	*Local search*
Yığılmış genelleme	*Stacking*
Yinelemeli yerel arama	*Iterated local search*
Yol temsili	*Path representation*
Yüksek boyutluluk	*High dimensionality*