Journal of the Faculty of Engineering and Architecture of Gazi University 33:1 (2018) 75-87

Mühendislik Mimarlık Fakültesi Dergisi
Journal of The Faculty of Engineering and Architecture of Gazi University

Elektronik / Online ISSN : 1304 - 4915
Basılı / Printed ISSN : 1300 - 1884

Saldırı tespit sistemlerinde genetik algoritma kullanarak nitelik seçimi ve çoklu sınıflandırıcı füzyonu

Atilla Özgür[1]*, Hamit Erdem[2]

[1]Jacobs University, Mathematics & Logistics, Logistics, Campus Ring 1, Bremen, 28759, Germany
[2]Başkent Üniversitesi, Mühendislik Fakültesi, Elektrik Elektronik Mühendisliği Bölümü, Bağlıca Kampüsü, Ankara, 06530, Türkiye

Ö N E Ç I K A N L A R

- 8 farklı sınıflandırıcı kullanılarak sınıflandırıcı füzyonu yapılmıştır
- Sınıflandırıcı füzyonu ile birlikte nitelik seçme yapılmıştır
- Nitelik seçme ve sınıflandırıcı füzyonu için genetik algoritmalar kullanılmıştır

Makale Bilgileri
Geliş: 31.05.2016
Kabul: 26.09.2017

DOI:
10.17341/gazimmfd.406781

Anahtar Kelimeler:
Nitelik seçme, sınıflandırıcı füzyonu, genetik algoritma, saldırı tespit sistemleri, makine öğrenmesi

ÖZET

Bilişim sistemlerinin gelişmesiyle, saldırı tespit sistemlerinin (STS) kullanımı önem kazanmıştır. Bu sistemlerin çalışması, genellikle sınıflandırma problemi çerçevesinde değerlendirilebilir. Sınıflandırıcı uygulamalarının en önemli aşamalardan birisi nitelik seçme aşamasıdır. Günümüzde, sınıflandırıcı başarısını artırmak için, tek sınıflandırıcıların yerine sınıflandırıcı füzyonu kullanımı önerilmektedir. Bu çalışmada; saldırı tespit sınıflandırma uygulamalarında, nitelik seçme ve sınıflandırıcı füzyonu ağırlık belirleme işlemlerinin, genetik algoritma (GA) kullanılarak yapılması önerilmektedir. Bu sisteme, Genetik Algoritma tabanlı Nitelik Seçme ve Ağırlık Bulma (GA-NS-AB) adı verilmiştir. GA-NS-AB, saldırı tespit sistemi NSL-KDD veri kümesi üzerinde uygulanmıştır. Çoklu sınıflandırıcı füzyonunda sınıflandırıcı sayısının 2 ile 8 arasında olduğu doğrusal ağırlıklı birleştirme yöntemi kullanılmıştır. Kullanılan sınıflandırıcılar şunlardır: Adaboost, Karar Ağacı, Lojistik Regresyon, Saf Bayes, Rastgele Orman, Gradient Boosting, En yakın K komşu ve Yapay Sinir Ağları (Çok Katmanlı Perseptron). Önerilen yöntem, GA-NS-AB, diğer füzyon yöntemleri (basit ve olasılık oy) ve tek sınıflandırıcı sonuçları ile karşılaştırılmıştır. Daha önce yayınlanan diğer çalışmaları ile karşılaştırıldığında, GA-NS-AB'nin daha başarılı olduğu görülmektedir. GA-NS-AB ile eğitim ve test süresi azaltılarak, doğruluk oranı değerleri daha yüksek bir sınıflandırıcı füzyonu elde edilmiştir.

Feature selection and multiple classifier fusion using genetic algorithms in intrusion detection systems

H I G H L I G H T S

- 8 different classifier is used for classifier fusion
- Classifier fusion and feature selection is solved together
- Genetic Algorithms are used for feature selection and classifier fusion

Article Info
Received: 31.05.2016
Accepted: 26.09.2017

DOI:
10.17341/gazimmfd.406781

Keywords:
Feature selection, classifier fusion, genetic algorithms, intrusion detection systems, machine learning

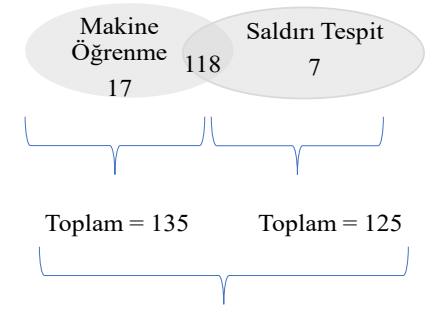
ABSTRACT

With the improvements in information systems, intrusion detection systems (IDS) become more important. IDS can be thought as a classification problem. An important step of classification applications is feature selection step. Nowadays, to improve accuracy of classifiers, it is recommended to use classifier fusion instead of single classifiers. This study proposes to use genetic algorithms for both feature selection and weight selection for classifier fusion in IDS. This proposed system called GA-NS-AB, has been applied to NSL-KDD dataset. Number of classifiers used in fusion changes between 2 and 8. Following classifiers have been used: Adaboost, Decision Tree, Logistic Regression, Naive Bayes, Random Forests, Gradient Boosting, K-Nearest Neighbor, and Neural Networks Multi-Layer Perceptron. The results of the proposed method have been compared with simple voting and probability voting fusion methods and single classifiers. In addition, GA-NS-AB is also compared with previous results. GA-NS-AB is a high accuracy classifier fusion that reduces test and training time.

*Sorumlu Yazar/Corresponding Author: / herdem@baskent.edu.tr / Tel: +90 312 246 6666 / 1350

1. GİRİŞ (INTRODUCTION)

Günümüzde bilgisayar sistemleri günlük hayatın bir parçası oldukları kadar, birçok hizmetin de temelinde bulunmaktadırlar. Milyonlarca kullanıcı, bilgisayar tabanlı sistemler üzerinden internete bağlanmaktadır. Ama bu geniş ağ yapısı saldırılara maruz kalmaktadır. Yapılan saldırıların karmaşıklığı ve sıklığı ise giderek artmaktadır. Bu saldırılara karşı korumak için oluşturulan yazılımlara, saldırı tespit sistemleri (STS) (Intrusion Detection Systems) adı verilmektedir [1]. Saldırı tespit sistemleri, tespit mekanizmalarına göre ikiye, imza tabanlı ve anomali tabanlı, ayrılmaktadır [1]. İmza tabanlı STSler, saldırı tespiti için bir imza veri tabanı tutmaktadır. Sisteme gelen yeni istekler, bu veri tabanı ile karşılaştırılır. Bu karşılaştırmaya göre saldırı veya normal olduklarına karar verilir. Anomali tabanlı sistemlerde ise sistem tarafından Normal istek tanımı yapılmaktadır. Bu tanım genellikle istatiksel ve sınıflandırıcı tabanlıdır. Gelen isteklerin bu tanıma uygunluklarına göre saldırı (anomali) veya normal olduklarına karar verilir [2]. Saldırı tespit sistemleri ve bu alandaki makine öğrenmesi uygulamaları, tarama makalelerinde detaylı değerlendirilmiştir [2, 3]. Yazarların daha önceki tarama çalışmasına [4]'e göre, STS ile ilgili, 2010 yılından sonra SCI indeksli dergilerde yayınlanan 142 çalışmanın, 118 tanesinde STS ve Makine Öğrenme teknikleri birlikte kullanılmıştır, Şekil 1.



Şekil 1. STS ve makine öğrenmesi çalışma sayısı
(IDS and machine learning study counts)

Nitelik seçme (feature selection) aşaması, çok boyutlu veri kümelerinin sınıflandırmasını etkileyen işlemlerden birisidir. Nitelik seçme işlemi, NP-Zor (NP-hard) bir problem olduğu için farklı yöntemler kullanılmıştır [5]. Örneğin Yıldız vd. [6] tarafından meme kanseri sınıflandırılmasında nitelik seçme aşamasında genetik algoritma uygulanmıştır. Benzer şekilde, evrimsel algoritmalar nitelik seçme ve niteliklerin ağırlıklarının bulunmasında kullanılmıştır [7]. STS veri kümeleri nitelik sayısı ve örneklem sayısı açısından büyük olduklarından, makine öğrenme algoritmalarının eğitilmesi ve testi uzun zamanlar alabilmektedir. Nitelik Seçme, bu işlemleri hızlandırması ve diğer yararları (2.3 Nitelik Seçme) nedeniyle STS uygulamalarında sıklıkla uygulanır [2, 4].

Yapılan STS çalışmalarında, tek sınıflandırıcı makine öğrenme algoritmaları sıkça kullanılmıştır. Örneğin, Yapay Sinir Ağları, KDD99 veri kümesi üzerinde denenmiştir [8, 9]. STS üzerinde yapılan birden fazla makine öğrenme algoritması birleştirme çalışması Bass [10] tarafından yapılmıştır. Bu konuda diğer çalışmalarda [11, 12] mevcuttur. Yazarların daha önceki tarama çalışmasına [4] göre, 142 STS ve Makine Öğrenmesi çalışmasından, 14 tanesinde sınıflandırıcı füzyonu yapıldığı gözlemlenmiştir. Füzyon sırasında, sınıflandırıcılar nihai sonuç için tek oy kullanabilir (basit oy) veya olasılık değerleri olarak oy kullanabilir. Bazı sınıflandırıcıların etkisi ağırlıklı doğrusal birleştirmeyle artırılabilir. Ağırlıkların belirlenme yöntemi, sınıflandırıcı füzyon işleminin başarısını etkileyebilmektedir. Ağırlıkların belirleme işlemi genellikle saha bilgisi ve deneysel çalışma ile yapılırken, bu işlem için sezgisel eniyileme (heuristic optimization) algoritmaları da tercih edilmiştir: Diferansiyel Evrim [13] ve Genetik Algoritmalar [14, 15].

Makine öğrenme ve STS sistemlerinde son yıllarda karma (hybrid) yapılar öne çıkmaktadır. 2010-2015 yıllarında yapılan 142 yayın içinde 50 karma çalışma bulunmaktadır [4]. Karma yapılar farklı algoritmaların bir arada kullanılmasıdır. Genellikle, sezgisel eniyileme (optimizasyon) algoritmalarının, diğer makine öğrenme teknikleri ile birleştirilmesi çok kullanılmaktadır. Sezgisel eniyileme algoritmaları arasında en çok kullanılanlardan birisi genetik algoritmalardır. GA, küresel bir eniyileme algoritmasıdır ve NP-Zor problemlerin çözümünde sıkça kullanılır. Örneğin, GA çok bilinen kısa yol probleminin eniyilemesinde kullanılmıştır [16]. GA benzer şekilde, diğer mühendislik problemlerinde, Üstündağ vd. [17], radar sinyallerinde gürültü temizleme, Gürsü [18] aşırı akım tahmini gibi konularda uygulanmıştır. Ayrıca Yıldız vd. [6] tarafından, Meme kanseri konusunda nitelik seçme ve veri füzyonu işleminde kullanılmıştır.

Tablo 1. STS ve makine öğrenmede 142 makalede kullanılan yöntem sayıları [4]
(IDS and machine learning 142 articles used method counts)

Teknik	Makale Sayısı
STS	*125*
Makine Öğrenmesi	135
Karma(Hybrid)	50
Nitelik Seçme	34
Füzyon	14
Genetik Algoritma	16

Sylvester ve Chawla [14] sınıflandırıcıların ağırlıklarını GA ile belirleyen, EVEN adını verdikleri, bir yöntem önermişlerdir. Maghsoudi vd. [15], GA ile sınıflandırıcı ağırlıklarını bularak hiper spektral görüntüleri sınıflandırmışlardır. Görüldüğü gibi, GA nitelik seçme ve sınıflandırıcı füzyonunda ağırlık bulma konusunda daha önceki çalışmalarda da kullanılmıştır. Fakat bu iki işlemi bir arada yapan bir çalışma, yazarların bildiği kadarı ile bulunmamaktadır.

Önerilen GA-NS-AB (Genetik Algoritma Tabanlı Nitelik Seçme ve Ağırlık Bulma) adı verilen çalışmada, nitelik seçimi ve heterojen çoklu sınıflandırıcı füzyonunun tek adımda yapılmasını hedeflenmiştir. Bu iki işleminin tek aşamada yapılması ile ilk adımda yapılan hataların, diğer adıma olan etkisi azaltılmıştır. GA-NS-AB, GA kullanarak, eş zamanlı nitelik seçimi ve sınıflandırıcıların doğrusal katılım ağırlıkları belirlemektedir. Her iki aşamanın nihai sınıflandırıcı üzerindeki etkisi, önceki çalışmalarda verilmiştir. Böylece, nitelik seçimi ile sınıflandırıcı başarısı (doğruluk oranı) düşürülmeden, eğitim ve test işlemleri hızlandırılırken, tek sınıflandırıcılar farklı oranlarda birleştirilebilir. Eğitim ve test işlemlerinin hızlı yapılması STSler için kritik bir adımdır [19]. Önerilen ağırlıklı sınıflandırıcı füzyonunda, kullanılan sınıflandırıcı sayısının etkisini görmek için, füzyon işleminde 2 ile 8 arası farklı sınıflandırıcı kullanılmıştır. Önceki, STS çalışmaları gözden geçirilerek, Adaboost, Karar Ağacı, Lojistik Regresyon, Saf Bayes, Rastgele Orman, Gradient Boosting, En Yakın K-Komşu ve Yapay Sinir Ağları (Çok Katmanlı Perseptron) sınıflandırıcıları füzyon işleminde kullanılmıştır. Önerilen yapının sınıflandırma açısından başarısı istatistiksel test yöntemlerinden ANOVA ve t-test yöntemleriyle analiz edilmiştir. Özet olarak, GA-NS-AB yönteminin katkıları aşağıdaki gibi sıralanabilir. (1) Nitelik seçme ve sınıflandırıcının birleştirilmesindeki ağırlıkların eşzamanlı olarak belirlenmesi. (2) Veri kümesinin büyüklüğü ve problemin NP-Zor olması dikkate alınarak, sezgisel eniyileme genetik algoritmanın kullanılması (Bu uygulamada, NSL-KDD veri kümesi kullanılmıştır.). (3) Bu iki işlemin aynı anda yapılması ile eğitim ve test süresinin kısaltılması. (4) Çoklu sınıflandırıcının birleştirilmesindeki sınıflandırıcı sayısının etkisinin belirlenmesi (bu çalışmada 2-8 sınıflandırıcı denenmiştir). (5) Çoklu sınıflandırıcı birleştirilmesinde doğrusal ağırlıklı, basit oy ve olasılık birleştirme yöntemlerinin karşılaştırılması. (6) Daha önceden aynı veri seti(NSL-KDD) üzerinde yapılan çalışmalar ile karşılaştırılmış ve daha başarılı olduğu görülmüştür. (7) Uygulanan yöntemin sınıflandırma başarısı, istatistiksel ANOVA ve t-test yöntemleri ile de gösterilmiştir.

2. MATERYAL VE METOD
(MATERIAL AND METHOD)

2.1. NSL-KDD ve KDD99 Veri Kümeleri
(NSL-KDD and KDD99 Datasets)

Günümüzde hala STS araştırmalarında, görece eski DARPA, KDD99 ve NSL-KDD veri kümeleri kullanılmaktadır [20,21]. MIT Lincoln laboratuvarı tarafından 1998 yılında ilk DARPA STS veri kümesi oluşturulmuştur [22]. Bu veri kümesinin nitelik çıkarılmış bir hali [23], KDD 99 yarışmasında kullanılmıştır. KDD99 üzerinde makine öğrenme algoritmalarının daha iyi çalışması için, iyileştirme yapılarak-mükerrer kayıtlar silinmiş, veri boyutu azaltılmış - NSL-KDD veri kümesi oluşturulmuştur [24]. NSL-KDD veri tabanı [25] adresinden indirilebilir. Bu veri kümelerinin eksik yönleri bilinmesine [26, 27] rağmen, hala STS araştırmalarında en çok kullanılan veri kümeleridirler [20, 4]. Bu veri kümelerinin büyüklükleri ve genel özellikleri, Tablo 2'de verilmiştir. Bu çalışmada, NSL-KDD veri kümesi [25], eğitim, onaylama ve test için kullanılmıştır, Tablo 3. Eğitim veri seti kullanılarak füzyon sırasında kullanılan sınıflandırıcılar eğitilmiştir. Onaylama veri seti ile füzyon ağırlıkları bulunmuştur. Önceki iki işlemde hiç kullanılmayan test veri seti ile önerilen metodun sonuçları bulunmuştur.

2.2. Sınıflandırıcı Başarısı (Classifier Performance)

Sınıflandırıcının başarısı farklı yöntemlerle ölçülebilir. Örneğin STS sınıflandırmasında, çıkış iki sınıftan oluşur. (Normal, Saldırı). Sınıflandırıcı, birçok örneklemi doğru sınıflandırırken, bazı saldırıları normal ve bazı normal durumları saldırı gibi işaretleyebilir. Bu dört durum: (1) Normal sınıflandırılan normal giriş verisi: Doğru Pozitif (DP); (2) Saldırı olduğu halde, normal olarak işaretlenen veri: Yanlış Pozitif (YP); (3) Normal olduğu halde saldırı olarak işaretlenen: Yanlış Negatif (YN); (4)Bir saldırının, saldırı olarak işaretlenmesi: Doğru Negatif (DN)'tir. Bu dört durum dikkate alınarak, Tablo 4 Hata Matrisi (Confusion Matrix) oluşturulmuştur. Sınıflandırıcının başarısı için

Tablo 2. Veri Kümeleri [4] (Datasets)

Adı	Eğitim Boyutu	Test Boyutu	Not
DARPA 99	6.2GB	3.67GB	Asıl veri kümesi. TCP/IP dosyaları
KDD99	4898431 örneklem	311029 örneklem	Nitelik çıkarılmış ve ön işleme yapılmış
NSL-KDD	125973 örneklem	22544 örneklem	Mükerrer kayıtlar silinmiş. Boyut azaltılmış

Tablo 3. Deneylerde kullanılan veri kümesi (NSL-KDD)
(Dataset that is used in experiments NSL-KDD)

	Eğitim	Onaylama	Test
%	100	10	90
Adet	125973	2254	20290

Tablo 4. Hata matrisi (Confusion matrix)

Tahmin Edilen Sınıf	Gerçek Sınıf: Saldırı	Gerçek Sınıf: Normal	Denklem 1 Doğruluk Oranı $\frac{DP + DN}{DP + DN + YN + YP}$
Saldırı	Doğru Pozitif (DP)	Yanlış Pozitif(YP)	Denklem 2 F1-Değeri $\frac{DP}{2 * DP + YN + +YP}$
Normal	Yanlış Negatif(YN)	Doğru Negatif(DN)	

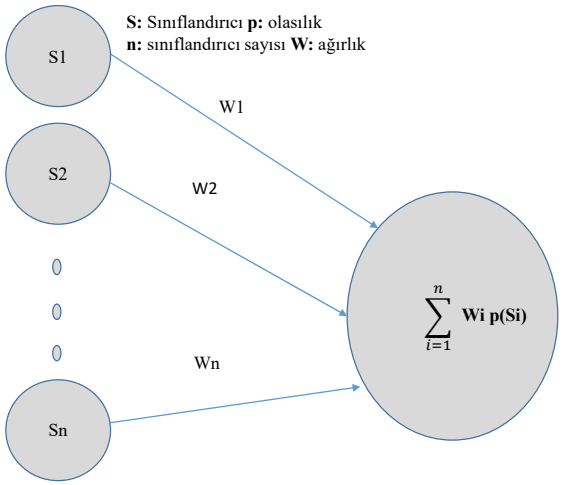
Denklem 1 Doğruluk Oranı ve Denklem 2 F1-Değeri adı verilen sayısal ölçütler tanımlanmıştır.

2.3. Nitelik Seçme (Feature Selection)

Büyük veri kümeleri üzerinde sınıflandırma çalışması yaparken, en önemli aşamalardan birisi nitelik seçme aşamasıdır. Bir veri kümesinde çok sayıda nitelik bulunması, makine öğrenme algoritmalarının eğitim ve test süresini artırır. Ayrıca bazı makine öğrenme algoritmaları, fazla (redundant) niteliklere karşı duyarlıdır ve bu tür algoritmalarda performans düşüşleri ortaya çıkabilir [5]. Bundan dolayı sınıflandırıcı sistemlerinde nitelik seçme çok kullanılan bir ön işlemdir [2, 3]. Bu önişleme çerçevesinde, bir taraftan nitelik sayısı azaltılarak öğrenme algoritması hızlandırılır; diğer taraftan, sınıflandırıcının başarısının düşmemesine özen gösterilir. KDD99 ve NSL-KDD veri kümelerinde 41 nitelik ve bir hedef sınıf bulunmaktadır. Veri örneklem sayısının çok olması eğitim ve test işlemlerinin uzun sürmesine neden olmaktadır. Bu veri kümesinde nitelik sayısı fazla ve örneklem sayısı çok olduğundan nitelik seçme aşaması sıkça uygulanır [4]. Tablo 1'e göre 142 çalışmanın 34 tanesinde nitelik seçme aşaması uygulanmıştır. Nitelik seçme işlemi genel olarak iki şekilde yapılmaktadır: Filtreleme ve Sarmalama Yöntemleri [5]. Filtreleme yöntemlerinde nitelikler seçilen bir kıstasa göre (örneğin: Bilgi Kuramı entropi) sıralanmakta ve sıralamada en üst seviyede olanlar seçilmektedir. Sarmalama yöntemlerinde ise sınıflandırıcı bir kara kutu olarak kullanılır. Seçilen nitelik kümelerinin sınıflandırıcı performans değerleri kontrol edilmekte ve başarı seviyesi yüksek nitelik kümeleri seçilmektedir. Bu makalede sarmalama nitelik seçimi yöntemi kullanılmıştır. Çalışmada kullanılan GA yöntemi iki parçalı (nitelik ve ağırlık) genom kullanmaktadır. Nitelik genom parçası üzerindeki yapılan mutasyon ve çaprazlama işlemleri ile farkı nitelik kümeleri elde edilmektedir. GA ile oluşturulan nitelik kümeleri ile sınıflandırıcı füzyonu eğitilmekte ve onaylama seti üzerindeki doğruluk oranı başarısına göre, nitelik kümesinin başarısına karar verilmektedir.

2.4. Sınıflandırıcı Füzyonu (Classifier Fusion)

Sınıflandırma çalışmalarında, başarıyı artırmak için, birden fazla makine öğrenme algoritması birleştirilebilir. Bu işlem farklı araştırma alanlarında, farklı isimlerle anılmaktadır. Bunlardan birkaçı; Sınıflandırıcı Füzyonu (Classifier Fusion), Sınıflandırıcı Kümesi (Ensemble), Kombination (Classifier Combination) [11]. Füzyon sınıflandırıcı çalışmaları heterojen (farklı algoritmalar) veya homojen (aynı algoritma) olabilir [28]. Sınıflandırıcıların birleştirme aşamasında, genellikle 3 farklı yöntem kullanılır [28]. Bu yöntemler: basit oy (simple or majority voting), olasılık oy (probability voting) ve ağırlıklı (weighted) birleştirme olarak adlandırılır. Sınıflandırıcıların basit oy yönteminde her sınıflandırıcı tek oy kullanır ve en çok oy alan çıktıya nihai sonuç olarak karar verilir. Olasılık oy yönteminde sonuç, her sınıflandırıcının olasılık oy değerleri toplanarak bulunur. Olasılık oy yönteminde, Şekil 2'deki ağırlıklar bir (1) değeri alıyor olarak düşünülebilir. Ağırlıklı birleştirmede, her sınıflandırıcının olasılık değeri bir ağırlık ile çarpılarak toplanır, Şekil 2.



Şekil 2. Çoklu sınıflandırıcı füzyonu (Multiple classifier fusion)

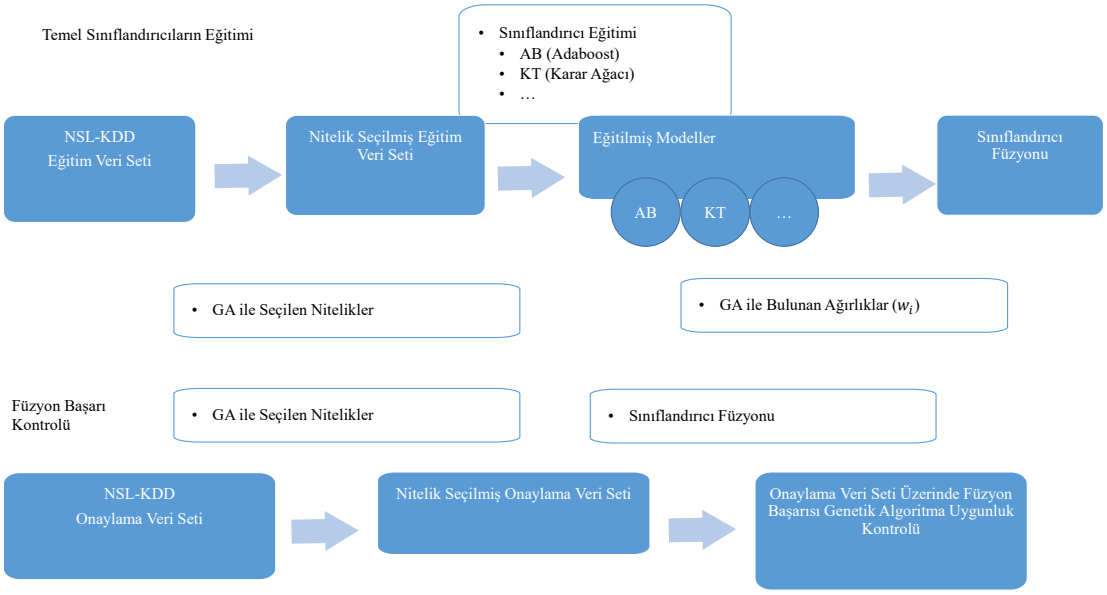
2.5. Genetik Algoritma (Genetic Algorithms)

Genetik algoritma (GA) doğal seçim teorisinden esinlenen bir eniyileme algoritmasıdır. Genetik algoritmada problem bir gen yapısında kodlanır. İkilik sistemde kodlanmış gen yapısı yaygın olarak kullanılır. Problemin incelenmesinde kullanılan tüm genler toplumu oluşturmaktadır. Toplum içindeki genlerin, nesiller boyunca değişime uğratılması ile çözüm uzayının farklı yerlerinde aramalar yapılmaktadır. Genler üzerinde değişim genellikle çaprazlama ve mutasyon ile yapılmaktadır. Çaprazlama toplumdaki 2 bireyin bir sonraki nesile genlerini aktarması işlemidir. Çaprazlanacak genlerin seçilmesinde kullanılan farklı yöntemler vardır. Bu çalışmada en başarılı genlerin seçilme olasılığını artıran Rulet Tekeri kullanılmaktadır. Mutasyon, genlerin bir kısmının belirli bir olasılık ile değişmesi işlemidir. Çaprazlama başarılı sonuçların daha sonraki nesillere aktarılması sağlarken, mutasyon çözüm uzayında farklı

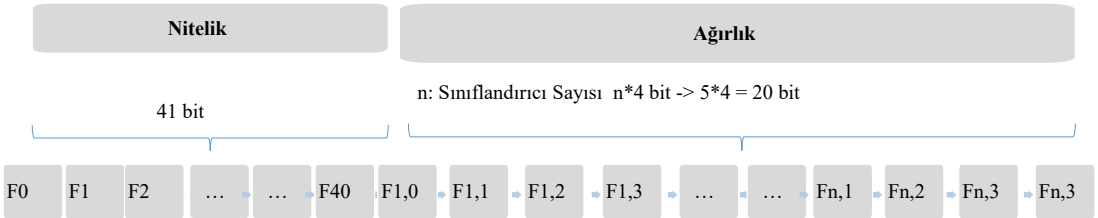
bölgelerin aranmasını sağlayarak, yerel minimuma takılma olasılığını azaltmaktadır [29]. Sadece çaprazlama kullanıldığında, toplumdaki en başarılı genler kaybedilmektedir. Bundan dolayı GA'larda elitizim prensibi, en başarılı genlerin bir kısmının değişime uğramadan bir sonraki nesile aktarılmasını önermektedir.

2.6. Önerilen Yöntem (Proposed Method)

Bu çalışmada, GA kullanılarak, STS uygulamalarında, nitelik seçme ve çoklu sınıflandırıcı ağırlık bulma işlemi yapılmaktadır. Önerilen yönteme GA-NS-AB (Genetik Algoritma tabanlı Nitelik Seçme ve Ağırlık Bulma) adı verilmiştir. GA-NS-AB için önerilen genel blok yapı Şekil 3'te verilmiştir. Şekilde görüldüğü gibi, nitelikleri seçmek ve ağırlıkları belirlemek için genetik algoritma kullanılmıştır. GA kullanılarak nitelik seçme ve ağırlık bulma işlemleri yapılmaktadır. GA tarafından seçilen nitelikler ile sınıflandırıcılar eğitim veri seti üzerinde eğitilmektedir. Eğitilen sınıflandırıcılar GA tarafından bulunan ağırlıklar kullanılarak birleştirilir. Bu iki işlemin başarısı, GA uygunluk fonksiyonu çıktısı olan onaylama veri seti üzerindeki doğruluk oranı ile ölçülmektedir. Nihai sistemin başarısı GA tarafından hiç görülmeyen test veri tabanı üzerinde sınanmaktadır. Genetik algoritmada en önemli seçim problemin gen yapısı olarak kodlanmasıdır. Bu çalışmada, ikilik sistemde kodlanmış nitelikler + ağırlıklar adı verilen iki parçalı bir genom kodlaması kullanılmıştır, Şekil 4. Nitelikler genom parçası kullanılan veri kümesi (NSL-KDD) niteliklerine göre 41 bitten oluşmaktadır. Eğer ilgili bit 0 ise o nitelik seçilmemiş, eğer ilgili bit 1 ise o nitelik seçilmiştir. Örneğin "1100.." 41 bit uzunluktaki bir genomda 0. (duration) ve 1. (protocol_type) nitelik seçilmiştir. Ağırlıklar genom parçasının boyutu, sınıflandırıcı birleştirme işlemi sırasında kullanılan sınıflandırıcı sayısına göre değişmektedir. Her sınıflandırıcı için ağırlık olarak 4 bit verilmiştir. Buradaki değer sınıflandırıcının son karara etkisini belirlemektedir. Örneğin 1010 bit tekeri, x10 olarak etki etmektedir. Eğer 5 sınıflandırıcı kombine ediliyorsa 5 * 4 = 20 bit, genomun ağırlıklar parçasını oluşturmaktadır. Buna göre 5 sınıflandırıcı birleştirilen genom, nitelikler (41 bit) ve ağırlıklar (5 sınıflandırıcı * 4bit = 20 bit), toplam olarak 61 bit uzunluktadır. Kullanılan diğer GA parametreleri Tablo 5'te verilmiştir. Önerilen çalışma GA tabanlı eniyileme problemi olduğundan, eniyilemenin başarısını test etmek için, füzyon sınıflandırıcının onaylama veri kümesi üzerindeki doğruluk oranı başarısı uygunluk fonksiyonu olarak belirlenmiştir. Uygunluk fonksiyonuna girdi olarak gelen genom, nitelik ve ağırlık olarak 2 parçaya ayrılmaktadır. Nitelik parçası kullanılarak, eğitim seti ve onaylama veri kümesi üzerinde nitelik seçimi yapılmaktadır. Daha sonra eğitim seti kullanılarak küme sınıflandırıcıları eğitilmektedir. Eğitilen sınıflandırıcılar nihai karara ağırlık gen parçasından gelen değer kadar etki etmektedirler. Füzyon sınıflandırıcı daha sonra onaylama veri kümesi



Şekil 3. Önerilen yöntemin (GA-NS-AB) genel yapısı (Proposed method (GA-NS-AB) general structure)



Şekil 4. Problemin kodlaması ve ilgili genom ikilik yapısı (Genomic coding of problem and 2-part genome)

Tablo 5. Genetik algoritma üstün parametreler (Genetic algorithm hyper parameters)

Adı Türkçe	Name English	Değeri – Value
Genom tipi	Genome type	İkilik-binary
Genom Uzunluk	Genome Length	53-69 bit
Toplum	Population	80
Çaprazlama Oranı	Crossover Rate	0,9
Mutasyon oranı	Mutation rate	0,02
Nesil Sayısı	Generation Number	100
Elitizim	Elitism	1 gen
Seçim	Selection	Rulet Tekeri

üzerinde çalıştırılmaktadır. Onaylama veri kümesi üzerindeki doğruluk oranı (Denklem 1) başarısı, uygunluk fonksiyonu tarafından uygunluk skoru olarak döndürülmektedir. Uygunluk fonksiyonu için doğruluk oranı hesaplaması akışı Şekil 3 verilmiştir. Ayrıca tüm sistem için gerçekleştirme yalancı kodu Şekil 5'te verilmiştir. Önerilen sistemin gerçekleştirilmesi için python, scikit-learn, matplotlib ve pyevolve kullanılmıştır. Bu çalışmada GA yapısında kullanılan üstün (hyper) parametreler, Back [30]'ın tavsiyeleri kullanılarak seçilmiş ve Tablo 5'te verilmiştir.

```
ESIK_DEGERI <- 0.90
MAKSIMUM_ITERASYON <- 100

toplum <- Toplumu rastgele olarak oluştur.
egitim_ds_tum_nitelik <- eğitim veri kümesini oku
eğitim_sınıf <- eğitim dataset sınıflarını oku
validasyon_ds_tum_nitelik <- validasyon veri kümesini oku
validasyon_sınıf <- validasyon dataset sınıfları oku
sınıflandırıcılar <- kulllanılan sınıflandırıcı listesi oluştur
en_iyi_gen <- null

for i=1 to MAKSIMUM_ITERASYON
  for j=0 to length(toplum)
    gen = toplum(j)
    ( gen_nitelikler, gen_agirliklar) = parçala(gen)
    egitim_ds = nitelik_sec(gen_nitelikler,egitim_ds_tum_nitelik)
    validasyon_ds = nitelik_sec(gen_nitelikler,validasyon_ds_tum_nitelik)
    sınıflandırıcılar = train_sınıfifiers(egitim_ds)
    kombine = Kombinasyon(sınıflandırıcılar,gen_agirliklar)
    dogruluk_oranı_val = kombine.score(validasyon_ds,validasyon_sınıf)
    gen.skor = dogruluk_oranı_val
    if (dogruluk_oranı_val > ESIK_DEGERI)
      en_iyi_gen = gen
      goto son
    end if
  end for
  sıralanmış_toplum <- Toplumu skor değerlerine göre sırala.
  yeni_toplum <- elitizme göre gen ata (sıralanmış_toplum)
  en_iyi_gen <- en_iyi_gen_bul(sıralanmış_toplum)
  yeni_toplum <- çaprazlama ve mutasyon (sıralanmış_toplum)
  toplum <- yeni_toplum
end for
son:
  return en_iyi_gen
```

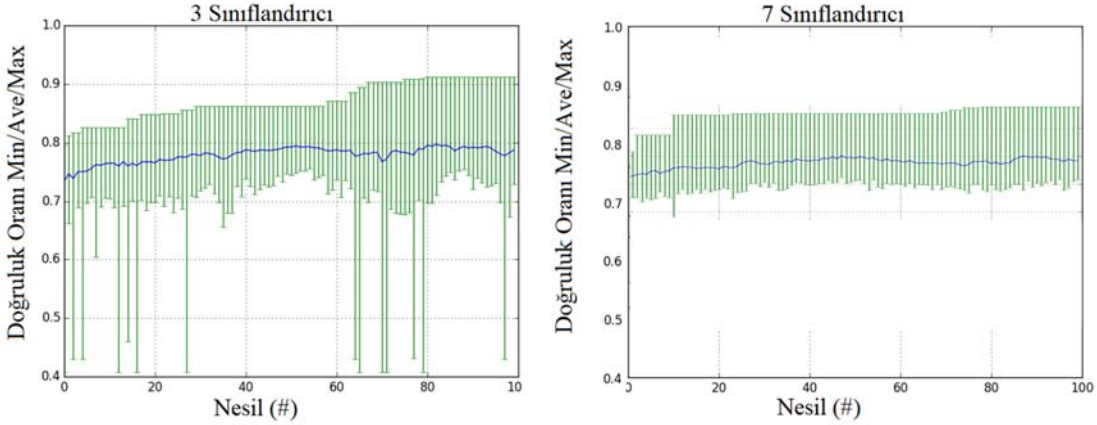
Şekil 5. Yöntemin (GA-NS-AB) yalancı kod ile gösterimi (Pseudo code of GA-NS-AB)

3. BENZETİM ÇALIŞMASI VE SONUÇLAR (SIMULATION STUDY AND RESULTS)

GA-NS-AB, veri kümesi NSL-KDD kullanılarak test edilmiştir. (1) Adaboost, (2) Karar Ağacı, (3) Lojistik Regresyon, (4) Saf Bayes, (5) Rastgele Orman, (6) Gradient Boosting, (7) En yakın K komşu ve (8) Yapay Sinir Ağları (Çok Katmanlı Perseptron) sınıflandırıcıları kullanılarak sınıflandırıcı füzyonu yapılmıştır. Füzyon sırasında 2-8 arası tek sınıflandırıcı kullanılarak, basit oy, olasılık oy ve ağırlıklı birleştirme işlemleri uygulanmıştır. Deneyler sırasında 8 temel sınıflandırıcı için, boş küme ve tek kümeler hariç, tüm alt küme kombinasyonları, 8'in 2'li + 8'in 3'lü 8'in 8'li kombinasyonları toplam 255 olmak üzere denenmiştir. Farklı 6 rastgele tohum (random seed) kullanarak deney tekrarlanmıştır. Toplamda 1530 farklı genome bulunmuştur. Sınıflandırıcı füzyonu için, uygunluk fonksiyonunun nesillere göre değişimi Şekil 6'da gösterilmiştir. Şekil 6'da görüldüğü gibi sistem başarısı nesiller boyunca düzenli olarak artmaktadır. Ama 3 sınıflandırıcı birleştirme Onaylama seti üzerinde 0,9 doğruluk oranının üstüne çıkmayı başarabilirken, 7 sınıflandırıcı ile birleştirme bu değerin altında kalmaktadır.

3.1. Füzyon Performans Değerleri – Doğruluk Oranı
(Fusion Performance Values – Accuracy)

Yapılan sınıflandırıcı füzyon benzetim çalışmasındaki en yüksek 10 doğruluk oranı değeri bulunan füzyon sınıflandırıcılar ve kullandıkları temel sınıflandırıcılar Tablo 6'da verilmiştir. Füzyon benzetim çalışmasındaki maksimum, ortalama, standart sapma ve minimum doğruluk oranı değerleri Tablo 7'de verilmiştir. Aynı şekilde yapılan sınıflandırıcı füzyon benzetim çalışmasındaki ortalama doğruluk oranı performans değerleri Şekil 7'de verilmiştir. Verilen şekilde sınıflandırıcı birleştirmesinde uygulanan üç yöntemin performansı gösterilmiştir (Basit oy, Olasılık oy, Ağırlıklı birleştirme).Şekil 7'de, kullanılan birleştirme yönteminden bağımsız olarak, sınıflandırıcı sayısının artması ile test seti üzerindeki başarı değerlerinin düştüğü açık bir şekilde görülmektedir. Bağlanım (Regression) Doğruları, sınıflandırıcı sayısı artıkça doğruluk oranlarındaki azalmayı göstermektedir. En hızlı azalan ağırlıklı füzyondur. Şekil 7, Tablo 6 ve Tablo 7 sonuçlarına göre, ağırlıklı birleştirme yöntemi, kullanılan sınıflandırıcı sayısının az olduğu (3-4 sınıflandırıcı) durumlarda, basit oy ve olasılık oy birleştirme yöntemlerinden daha iyi sonuçlar

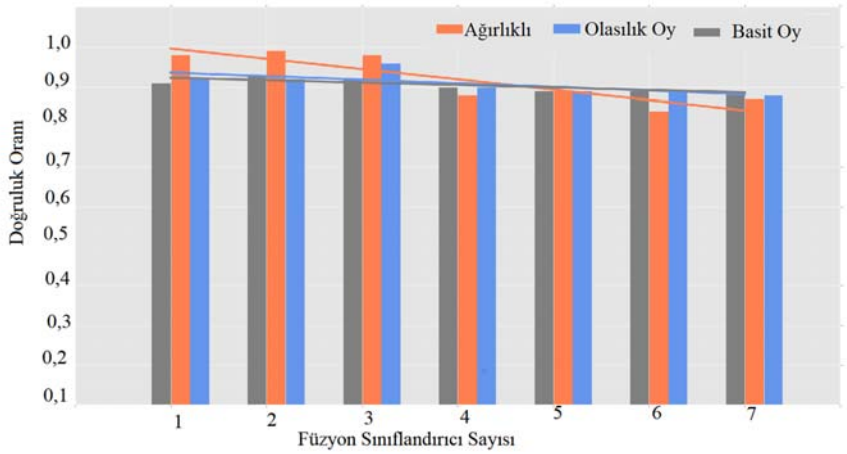


Şekil 6. Örnek GA koşuları -3 ve 7 sınıflandırıcı kombinasyonu onaylama veri seti
(Example ga runs for 3 and 7 classifier combinations on validation dataset)

Tablo 6. En başarılı 10 füzyon doğruluk oranı ve kullanılan sınıflandırıcılar (Best 10 fusion accuracies and used classifiers)

Sıra	FüzyonSınıflandırıcı Sayısı	Doğruluk Oranı	AB	KA	LR	SB	RO	GB	KNN	ÇKP
1	4	0,9088	√	√	√				√	
2	3	0,9075	√	√	√					
3	4	0,9059	√	√	√				√	
4	3	0,9041	√	√	√					
5	4	0,9028	√	√	√				√	
6	4	0,9027	√	√	√				√	
7	3	0,9022	√	√	√					
8	4	0,9013	√	√	√				√	
9	4	0,9006	√	√	√				√	
10	3	0,9003	√	√	√					

AB: Adaboost, KA: Karar Ağacı LR: Lojistik Regresyon SB: Saf Bayes RO: Rastgele Orman GB: Gradient Boosting KNN: En yakın K komşu ÇKP: Yapay Sinir Ağları (Çok Katmanlı Perseptron)



Şekil 7. Füzyon yapılan sınıflandırıcı sayısına göre performans değerlendirmesi – test veri seti
(Performance comparison according to fused classier counts)

vermektedir. Sonuçlara göre, ortalamada 3 sınıflandırıcı birleşimi diğerlerine göre daha iyi sonuç vermiştir. Ama en başarılı 10 tane sonuç arasında hem 3 sınıflandırıcı birleşimi ve 4 sınıflandırıcı birleşimi bir arada bulunmaktadır. Bu durumda kaynak tüketimi açısından 3 sınıflandırıcı birleşimin kullanılması önerilmektedir. Benzer şekilde en başarılı sınıflandırıcı füzyonlarında en çok Adaboost, Karar Ağacı, Lojistik Regresyon ve en yakın komşu temel sınıflandırıcıları kullanılmıştır. Genel olarak, füzyon yapılan sınıflandırıcı sayısı artıkça, başarı değerleri düşmektedir. Bu durum nedeni olarak, sınıflandırıcı füzyonunda mümkün olduğunca farklı sınıflandırıcıların kullanılması gerektiği

gösterilmiştir [11, 32]. Sınıflandırıcı farklılığının (classifier diversity) kabul edilmiş bir tanımı yoktur [31]. Yine de sınıflandırıcı farklılığının füzyon sonucuna etkisi kabul edilen bir gerçektir. Kullanılan sınıflandırıcı sayısının artması ile sınıflandırıcılar arasındaki farklılıklar azaldığı için başarı değerleri düşmektedir [29, 32]. Ağırlıklı birleştirmenin diğer birleştirmelere göre daha hızlı düşmektedir. Bu durumun sınıflandırıcı farklılığı (diversity) yüzünden olduğu değerlendirilmektedir. Ağırlıklı birleştirmede sınıflandırıcı farklılığı sayesinde elde edilen kazançlar ağırlıklı birleştirme daha fazla hissedilmektedir. Ama sınıflandırıcı sayısı artıkça bu kazanç azalmaktadır. Azalan bu kazanç ağırlıklar ile çarpıldığında daha hızlı düşüşe yol açmaktadır.

3.2. Füzyon Performans Değerleri – ROC ve F1-Değeri
(Fusion Performance Values ROC and F1 Value)

Benzer şekilde ROC eğrisinin altında kalan alan ve F1-Değerleri Tablo 8'de verilmiştir. Tablo 7 ve Tablo 8'deki sonuçlar bir birine paralellik göstermektedir. Görüldüğü gibi en başarılı ortalama değerler 3 ve 4 sınıflandırıcının füzyon yapıldığı deneylerdir. Ama sınıflandırıcı sayısı artıkça, bu değerler (doğruluk oranı ve F1-Değeri) düşmektedir. Ayrıca 2 deneyin ROC Eğrileri Şekil 8 verilmiştir. Şekil 8 üzerinde de sınıflandırıcı sayısı artıkça başarının azaldığı görülmektedir.

3.3. Füzyon Performans Değerleri – Nitelik Seçme
(Fusion Performance Values – Feature Selection)

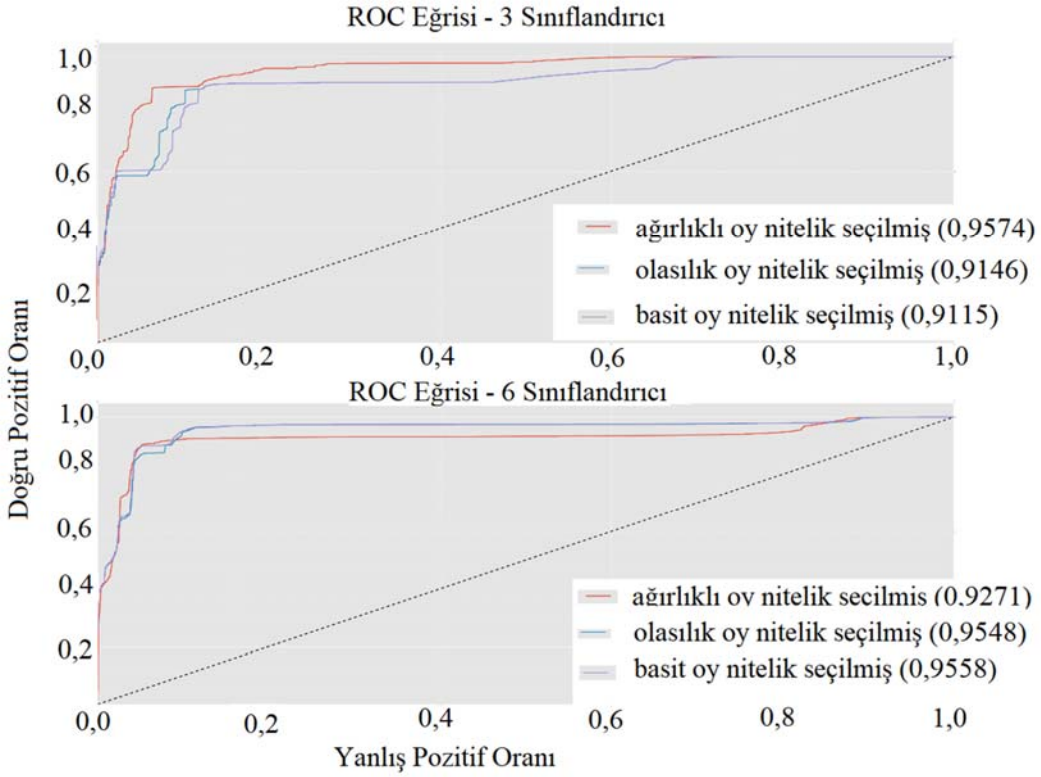
Yapılan deneylerde 41 nitelikten, ortalama 20,78, minimum 11 ve maksimum 28 tanesi seçilmiştir. Nitelik *service* deneylerde %91 seçilmiş olarak en ayrıştırıcı niteliktir. Nitelikler ve seçilme sayıları Tablo 9'da verilmiştir. Önerilen çalışmada GA-tabanlı nitelik seçmenin etkisini göstermek için, veri kümesinin tüm nitelikleri kullanılarak, sınıflandırıcı füzyonu yapılmıştır. Aynı koşullarda, GA ile seçilen nitelikler kullanılarak, tekrar sınıflandırıcı birleştirmesi yapılmıştır. Bu iki durumun karşılaştırılması Tablo 10'da verilmiştir. Elde edilen sonuçlara göre, GA-tabanlı nitelik seçmenin sınıflandırıcı başarısı üzerindeki pozitif etkisi olduğu görülmektedir. Füzyon işleminde kullanılan sınıflandırıcıların tek başlarına uygulandıklarında, test seti üzerindeki başarıları Tablo 11'de görülmektedir. Nitelik seçme işlemi, hem füzyon hem de tek sınıflandırıcı doğruluk oranının yükselmesinde etkilidir. En başarılı tek sınıflandırıcılar, 0,83 doğruluk oranı ile karar ağacı ve gradient boosting olmuştur. Füzyon yöntemleri tek sınıflandırıcılardan daha iyi sonuçlar, 0,9 doğruluk oranı, vermiştir, Tablo 10 ve Tablo 11, İlginç bir şekilde gradient boosting en başarılı füzyon sınıflandırılmasında kullanılmamıştır, Tablo 6. Nitelik seçmenin eğitim ve test süresine etkisi Tablo 12 ile verilmiştir. GA tabanlı nitelik seçme, eğitim süresini %30-%34 ve test süresini ise %23-%25 oranında azaltmıştır.

Tablo 7. Çoklu sınıflandırıcı füzyonunda doğruluk oranı performans değerleri – test veri seti
(Multiple classifier fusion accuracy performance values on test dataset)

Füzyon yöntemi	Sınıflandırıcı Sayısı Maksimum, Ortalama, Standart Sapma ve Minumum Değerler						
	2	3	4	5	6	7	8
	mak-orta std-min	mak-orta std-min	mak-orta std-min	mak-orta std-min	mak-orta std-min	mak –orta std-min	mak-orta std-min
Basit Oy	0,86 0,81	0,89 0,84	0,85 0,82	0,83 0,80	0,81 0,79	0,83 0,79	0,82 0,79
Olasılık Oy	0,03 0,76	0,03 0,69	0,02 0,78	0,02 0,84	0,02 0,74	0,02 0,76	0,12 0,75
	0,87 0,82	0,90 0,82	0,90 0,86	0,80 0,02	0,86 0,79	0,83 0,79	0,81 0,78
Ağırlıklı	0,03 0,77	0,03 0,70	0,03 0,82	0,91 0,78	0,03 0,75	0,02 0,75	0,11 0,73
	0,89 0,88	0,91 0,89	0,91 0,88	0,06 0,76	0,90 0,79	0,81 0,74	0,83 0,77
	0,02 0,77	0,03 0,72	0,03 0,80	0,76 0,71	0,12 0,45	0,05 0,57	0,14 0,75

Tablo 8. Nitelik seçimi ile füzyon yöntemlerinin sınıflandırıcı sayısına göre ek performans değerleri (ortalama değerler)
(Performance values for fusion with feature selection according to classifier counts (mean values))

Füzyon yöntemi	Sınıflandırıcı Sayısı	ROC Alanı	F1-Değeri	Sınıflandırıcı Sayısı	ROC Alanı	F1-Değeri
Basit Oy	2	0,9072	0,5606	6	0,9504	0,7403
Olasılık Oy	2	0,9083	0,7255	6	0,9500	0,7666
Ağırlıklı	2	0,9102	0,6874	6	0,9479	0,7778
Basit Oy	3	0,9315	0,7639	7	0,9474	0,7867
Olasılık Oy	3	0,9270	0,7508	7	0,9488	0,7834
Ağırlıklı	3	0,9321	0,7902	7	0,9499	0,7892
Basit Oy	4	0,9294	0,7219	8	0,9517	0,7763
Olasılık Oy	4	0,9292	0,7132	8	0,9534	0,7889
Ağırlıklı	4	0,9377	0, 7590	8	0,9506	0,8067
Basit Oy	5	0,94023	0,7685			
Olasılık Oy	5	0,9403	0,7715			
Ağırlıklı	5	0,9398	0,7593			



Şekil 8. Örnek ROC eğrileri (Example ROC curves)

Tablo 9. NSL-KDD nitelik listesi ve yapılan deneylerde seçilme yüzdeleri
(Features and percentage of their selection in experiments)

Nitelik Adı	Seçim Yüzdesi	Nitelik Adı	Seçim Yüzdesi
Service	91,58	count	49,00
dst_host_rerror_rate	73,64	su_attempted	48,74
num_outbound_cmds	68,75	duration	48,68
Urgent	61,27	num_failed_logins	48,09
dst_host_srv_rerror_rate	59,92	srv_count	47,28
logged_in	59,60	land	46,58
Flag	59,55	src_bytes	46,42
diff_srv_rate	59,49	hot	45,88
num_compromised	58,74	is_host_login	45,67
rerror_rate	58,36	same_srv_rate	45,67
dst_bytes	57,45	srv_serror_rate	45,02
dst_host_srv_diff_host_rate	56,70	dst_host_same_srv_rate	43,36
serror_rate	56,37	dst_host_serror_rate	40,67
dst_host_srv_serror_rate	55,68	wrong_fragment	38,57
num_shells	55,51	srv_diff_host_rate	38,35
srv_rerror_rate	55,24	is_guest_login	38,03
num_access_files	52,39	protocol_type	35,45

**Ortalamada 20,78 özellik seçilmiştir.

3.4. Daha Önce Yapılan Çalışmalar ile Karşılaştırma (Comparison With Literature)

Elde edilen sonuçları daha önce NSL-KDD ver i tabanı üzerinde yapılan 10 çalışma ile karşılaştırılmıştır, Tablo 13. NSL-KDD üzerinde yapılan birçok çalışmada, test veri seti olarak, NSL-KDD test veri yerine eğitim veri seti kullanılmaktadır. Aynı veri seti üzerinde eğitim ve test yapılması nedeni ile doğruluk oranları çok iyi çıkmaktadır. Tablo 13 üzerindeki Kang ve Kim [32] (0,9693), Pereira vd. [33] (0,9661) sonuçları eğitim seti üzerinde sınıflandırıcıların ezberlediği sonuçlardır. Önerilen yöntem test veri tabanı kullanan diğer yöntemler ile karşılaştırıldığında daha iyi sonuç elde ettiği görülmektedir.

Tablo 10. Füzyon yöntemlerinin nitelik sayısına göre sınıflandırma başarıların karşılaştırılması test veri seti (doğruluk oranı maksimum değerler)
(Comparison of classifier accuracy of fusion methods according to feature selection on test dataset (maximum accuracy values given)

Füzyon yöntemi/Nitelik sayısı	41 nitelik (Tümü)	GA ile seçim
Basit Oy	0,76	0,89
Olasılık Oy	0,79	0,91
Ağırlıklı	0,79	0,91

Ortalamada 20,78 nitelik seçilmiştir

Tablo 11. Tekil sınıflandırıcıların, nitelik sayısına göre karşılaştırılması, doğruluk oranları ortalama ve standart sapma değerleri – test veri seti
(Comparision of single classifier according to selected feature count accuracy values and standard deviation – test dataset)

Sınıflandırıcı → Nitelik sayısı↓	Tek sınıflandırıcı Ortalama(Standart Sapma)							
	AdaBoost	Karar Ağacı	Gradient Boosting	KNN	Lojistik Regresyon	Saf Bayes	Rastgele Orman	Çok Katmanlı Perceptron
41 (Tümü)	0,77/0	0,78/0,01	0,83/0	0,77/0	0,67/0	0,45/0	0,77/0	0,75/0,02
GA ile Seçilmiş (20,78)	0,78/0,02	0,83/0,03	0,83/0,03	0,79/0,02	0,70/0,06	0,50/0,12	0,79/0,02	0,76/0,02

Tablo 12. Nitelik seçmenin eğitim ve test sürelerine olan etkisi
(Effect of feature selection to training and testing time)

	Eğitim Süresi (Sn) (Eğitim Veri Seti)			Test Süresi (Sn) (Test Veri Seti)		
	Tüm Nitelikler	Nitelik Seçilmiş (Ortalama 20,78)	Azalma %	Tüm Nitelikler	Nitelik Seçilmiş (Ortalama 20,78)	Azalma %
Basit Oy	60,21	40,70	32,40	6,43	4,91	23,63
Olasılık Oy	60,23	39,73	34,57	5,81	4,41	24,09
Ağırlıklı	60,32	41,91	30,05	5,84	4,33	25,85

Tablo 13. NSL-KDD daha önce yapılan çalışmalar ile karşılaştırma
(NSL-KDD comparison with literature)

Çalışma	Yıl	Nitelik Seçme	Test Veri Tabanı	Doğruluk Oranı
Rastgeri vd. [34]	2015	Evet	Eğitim	0,7800
Kang ve Kim [32]	2016	Evet	Eğitim	0,9693
Pereira vd. [33]	2012	Evet	Eğitim	0,9661
Seresht ve Azmi [35]	2014	Hayır	Eğitim	0,8831
Farid vd. [36]	2014	Hayır	Eğitim	0,8344
Singh vd. [37]	2015	Evet	Eğitim	0,9867
Bhattacharya vd. [38]	2015	Evet	Test	0,8314
Mohammadi vd. [39]	2012	Evet	Test	0,8014
Liu vd. [40]	2016	Evet	Test	0,7460
Hoz vd. [41]	2015	Evet	Test	0,8800
Önerilen Çalışma (3 Sınıflandırıcı Füzyon)	2017	Evet	Test	0,9088
Önerilen Çalışma (4 Sınıflandırıcı Füzyon)	2017	Evet	Test	0,9075

Tablo 14. ANOVA ve T-test istatiksel test sonuçları (ANOVA and t-test statistical tests results)

	p *değeri* * <0,05 önemli, **< 0,001daha önemli			
Sınıflandırıcı sayısı	ANOVA	t-test Ağırlıklı-Basit oy	t-test Ağırlıklı-Olasılık oy	t-test Basit Oy-Olasılık Oy
2	*	*	*	0,1015
3	**	**	**	0,201
4	**	**	**	**
5	*	0,036*	0,16	0,07
6	0,626	0,42	0,76	0,15
7	**	**	0,001*	0,006*
8	**	**	0,002*	0,011*

3.5. İstatiksel Testler (Statistical Tests)

Elde edilen sonuçları detaylı değerlendirmek için, füzyon sınıflandırıcı sonuçları istatiksel olarak karşılaştırılmıştır, Tablo 14. ANOVA istatiksel testi, gruplar arasındaki farkların istatiksel olarak önemli olup olmadığını ölçmektedir. Küçük p-değeri (*<0,05) gruptaki üyelerin birbirinden farklı olduğunu göstermektedir. Daha küçük p-değerleri (**) istatiksel olarak farkın daha güçlü olduğunu göstermektedir. Tablo 14 ANOVA testine göre, füzyon sonucu elde edilen doğruluk oranı değerleri: 3, 4, 7, 8 sınıflandırıcı birleşimi için istatiksel olarak birbirinden farklıdır. Diğer sınıflandırıcı birleştirmelerinde (2, 5 ve 6), hem ANOVA hem t-test sonuçlarına göre bu farklılık azalmakta veya ortadan kalkmaktadır. Tablo 14 t-test sonuçlarına göre, ağırlıklı birleşim, basit ve olasılık oy birleşimden istatiksel olarak farklıdır. Bu durumda, 3-4 sınıflandırıcı füzyonundaki, yüksek doğruluk oranları istatiksel olarak ta anlamlıdır. Füzyon sırasında, 7 sınıflandırıcı kullanımında ise ağırlıklı birleştirme daha az değerler almaktadır. Bu sonuçlara göre 3 ve 4 sınıflandırıcı kullanılarak yapılan ağırlıklı füzyon tavsiye edilebilir.

4. SONUÇLAR VE TARTIŞMALAR (RESULTS AND DISCUSSIONS)

Bu çalışma, saldırı tespit sınıflandırma uygulamalarında, genetik algoritma kullanarak, nitelik seçme ve sınıflandırıcı füzyonu ağırlık belirleme işlemleri yapıldı. Önerilen GA-NS-AB yönteminde, NSL-KDD veri kümesi kullanılarak, sonuçlar analiz edildi. Sınıflandırıcı birleştirmesinde üç yöntem test edildi. Alınan sonuçlara göre, (1) STS de, çoklu sınıflandırıcı füzyonunun, tek sınıflandırıcılara göre daha başarılı olduğu görüldü. (2) Çoklu sınıflandırıcı füzyonunda, sınıflandırıcı sayısıyla, nihai sınıflandırma başarısı arasındaki ilişki analiz edildi. Kullanılan veri kümesinde, 3-4 sınıflandırıcının birleştirilmesinin yeterli olacağı görüldü. (3) GA tabanlı nitelik seçme ve sınıflandırıcı ağırlık birleştirme katsayılarının eşzamanlı belirlenmesinin, eğitim ve test süresini azalttığı ve sınıflandırma başarısını artırdığı görüldü. (4) Ağırlıklı sınıflandırıcı füzyonunun, genel olarak basit oy ve olasılık oy yöntemlerinden daha iyi sonuç verdiği görüldü. (5) Daha önceden aynı veri seti(NSL-KDD) üzerinde yapılan çalışmalar ile karşılaştırılarak daha başarılı olduğu görüldü. (5) Bu sonuçların istatiksel olarak geçerli olduğu gösterildi. Burada önerilen ve uygulanan yöntem, GA-NS-AB, diğer büyük veri kümelerinde kullanılabilir veya başka eniyileme yöntemi seçilerek, eğitim ve sınıflandırıcı başarısı açısından test edilebilir. Tüm kombinasyonların denenmesi durumunda arama uzayı çok büyümektedir. Sınıflandırıcı sayısı 8 olarak seçildiğinde tüm füzyon kombinasyonları 255 tane olmaktadır. Bu durum deney sayısını ve süresini çok artırmaktadır. Sınıflandırıcı sayılarının sabit değil değişimli olduğu genetik algoritma çözümü daha zor bir problemdir. Örneğin değişen uzunlukta genome yapısında genetik algoritma kaç sınıflandırıcının ve hangi kombinasyonlarda bu sınıflandırıcıların kullanılacağına karar verebilmektedir. İlerideki bir çalışmada genetik algoritma ve diğer meta-heuristik metotların değişen uzunlukta versiyonlarının aynı probleme uygulanması planlanmaktadır.

KAYNAKLAR (REFERENCES)

1. Scarfone K. ve Mell P. , Guide to intrusion detection and prevention systems (IDPS), NIST, ABD, 2007.
2. Ganapathy S., Kulothungan K., Muthurajkumar S., Vijayalakshmi M., Yogesh P. ve Kannan A., Intelligent feature selection and classification techniques for intrusion detection in networks: a survey, EURASIP Journal on Wireless Communications and Networking, 2013 (1), 273-289, 2013.
3. Kolias C., Kambourakis G. ve Maragoudakis M., Swarm Intelligence in Intrusion Detection: A Survey, Computers and Security, 30 (8), 625-642, 2011.
4. Özgür A. ve Erdem H., A Review of KDD99 Dataset Usage in Intrusion Detection and Machine Learning between 2010 and 2015, PeerJ Preprints 4:e1954v1, 2016.
5. Guyon I. ve Elisseeff A., An introduction to variable and feature selection, Journal of Machine Learning Research, 3, 1157-1182, 2003.
6. Yıldız O., Tez M., Bilge H.Ş., Akcayol M.A., Güler İ., Gene selection for breast cancer classification based on data fusion and genetic algorithm, Journal of the Faculty of Engineering and Architecture of Gazi University, 27 (3), 659-668, 2012.
7. Pérez-Rodríguez J., Arroyo-Peña A. G. ve García-Pedrajas N., Simultaneous instance and feature selection and weighting using evolutionary computation: Proposal and study, Applied Soft Computing, 37, 416-443, 2015.

8. Sağıroğlu Ş., Yolaçan E.N., Yavanoğlu U., Designing and developing an intelligent intrusion detection system, Journal of the Faculty of Engineering and Architecture of Gazi University, 26 (2), 325-340, 2011.

9. Tuncer T., Tatar Y., Implementation of the FPGA based programmable embedded intrusion detection system, Journal of the Faculty of Engineering and Architecture of Gazi University, 27 (1), 59-69, 2012.

10. Bass T., Intrusion Detection Systems and Multisensor Data Fusion, Commun. ACM, 43 (4), 99-105, 2000.

11. Kuncheva L.I., Bezdek J.C. ve Duin R.P., Decision templates for multiple classifier fusion: an experimental comparison, Pattern Recognition, 34 (2), 299-314, 2001.

12. Wang Y., Yang H., Wang X. ve Zhang R., Distributed intrusion detection system based on data fusion method, Fifth World Congress on Intelligent Control and Automation, 2004.

13. Zhang Y., Zhang H., Cai J. ve Yang B., A Weighted Voting Classifier Based on Differential Evolution, Abstract and Applied Analysis, 2014, 6, 2014.

14. Sylvester J. ve Chawla N. V., Evolutionary Ensemble Creation and Thinning, The 2006 IEEE International Joint Conference on Neural Network Proceedings, 2006.

15. Maghsoudi Y.A.A., Zoej M.V. ve Mojaradi B., Weighted Combination Of Multiple Classifiers For The classification Of Hyperspectral Images Using A Genetic algorithm, ISPRS Commission I Symposium, From Sensors to Imagery, 2006.

16. Dener M., Akcayol M.A. , Toklu S., Bay Ö., Genetic algorithm based a new algorithm for time dynamic shortest path problem, Journal of the Faculty of Engineering and Architecture of Gazi University, 26 (4), 915-928, 2011.

17. Üstündağ M., Avcı E., Gökbulut M., Ata F., Denoising of weak radar signals using wavelet packet transform and genetic algorithm, Journal of the Faculty of Engineering and Architecture of Gazi University, 29 (2), 375-383, 2014.

18. Gürsu B., Optimum overcurrent relay coordination via genetic algorithm method stopped by penalty function in substations, Journal of the Faculty of Engineering and Architecture of Gazi University, 29 (4), 665-676, 2014.

19. Kruegel C., Kruegel F., Vigna G. ve Kemmerer R., Stateful intrusion detection for high-speed network's, Security and Privacy, 2002. Proceedings. 2002 IEEE Symposium on, 2002.

20. Catania C.A. ve Garino C.G., Automatic network intrusion detection: Current techniques and open issues, Computers & Electrical Engineering, 38 (5), 1062-1072, 2012.

21. Hubballi N. ve Suryanarayanan V., False alarm minimization techniques in signature-based intrusion detection systems: A survey, Computer Communications,49, 1-17, 2014.

22. Cunningham R.K., Lippmann R.P., Fried D.J., Garfinkel S.L., Graf I. , Kendall K., Wyschogrod D. ve Zissman M.A., Evaluating intrusion detection systems without attacking your friends: The 1998 DARPA intrusion detection evaluation, 1999.

23. Lee W. ve Stolfo S. J., A framework for constructing features and models for intrusion detection systems, ACM Transactions on Information and System Security, 3, 227-261, 2000.

24. Tavallaee M., Bagheri E., Lu W. ve Ghorbani A.A., A detailed analysis of the KDD CUP 99 data set, Proceedings of the Second IEEE international conference on Computational intelligence for security and defense applications, Piscataway, NJ, USA, 2009.

25. NSL-KDD, Download Link of NSL-KDD in Github, 2016. https://github.com/ati-ozgur/NSL_KDD. Yayın Tarihi Ocak 17, 2017. Erişim tarihi Ocak 15, 2018.

26. Sommer R. ve Paxson V., Outside the Closed World: On Using Machine Learning for Network Intrusion Detection, Proceedings of the 2010 IEEE Symposium on Security and Privacy, Washington, DC, USA, 2010.

27. Brugger S., KDD Cup 99 dataset (Network Intrusion) considered harmful, https://www.Kdnuggets.com/ news /2007/n18/4i.html, Yayın Tarihi: 15 Eylül 2007, Erişim tarihi Ocak 15, 2018.

28. Kuncheva L.I., Combining Pattern Classifiers: Methods and Algorithms, Wiley-Interscience, 2004.

29. Kalınlı A., Aksu Ö., Genetic algorithm model based on dominant gene selection operator, Journal of the Faculty of Engineering and Architecture of Gazi University, 26 (4), 869-875, 2011.

30. Bäck T., Optimal Mutation Rates in Genetic Search, Proceedings of the 5th International Conference on Genetic Algorithms, San Francisco, CA, USA, 1993.

31. Kuncheva J., Measures of Diversity in Classifier Ensembles and Their Relationship with the Ensemble Accuracy, Machine Learning, 51 (2), 181-207, 2003.

32. Kang S.H. ve Kim K.J., A feature selection approach to find optimal feature subsets for the network intrusion detection system, Cluster Computing, 19, 325-333, 2016.

33. Pereira C.R., Nakamura R.Y.M., K., Costa A.P. ve Papa J.P., An Optimum-Path Forest framework for intrusion detection in computer networks, Engineering Applications of Artificial Intelligence, 25, 1226-1234, 2012.

34. Rastegari S., Hingston P. ve Lam C.P., Evolving statistical rulesets for network intrusion detection, Applied Soft Computing, 33, 348-359, 2015.

35. Seresht N.A. ve Azmi R., MAIS-IDS: A distributed intrusion detection system using multi-agent AIS approach, Engineering Applications of Artificial Intelligence, 35, 286-298, 2014.

36. Farid D.M., Zhang L., Rahman C.M., Hossain M.A. ve Strachan R., Hybrid decision tree and naïve Bayes classifiers for multi-class classification tasks, Expert Systems with Applications, 41, 1937-1946, 2014.

37. Singh R., Kumar H. ve Singla R. K., An intrusion detection system using network traffic profiling and online sequential extreme learning machine, Expert Systems with Applications, 42, 8609-8624, 2015.

38. Bhattacharya S. ve Selvakumar S., LAWRA: a layered wrapper feature selection approach for network attack detection, Security and Communication Networks, 8, 3459-3468, 2015.

39. Mohammadi M., Raahemi B., Akbari A. ve Nassersharif B., New class-dependent feature transformation for intrusion detection systems, Security and Communication Networks, 5, 1296-1311, 2012.

40. Liu Q., Yin J., Leung V.C.M., Zhai J.H., Cai Z. ve Lin J., Applying a new localized generalization error model to design neural networks trained with extreme learning machine, Neural Computing and Applications, 27, 59-66, 2016.

41. Hoz L.E.D., Ortiz A., Ortega J. ve Prieto B., PCA filtering and probabilistic SOM for network intrusion detection, Neurocomputing, 164, 71-81, 2015.