



BILGISAYAR BİLİMLERİ VE TEKNOLOJİLERİ DERGİSİ

JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGIES

ARALIK (DECEMBER) 2020

CİLT (VOLUME) : 1

SAYI (ISSUE) : 2

EDİTÖR: DOÇ. DR. ERDİNÇ AVAROĞLU

ISSN 2717-8579





Bilgisayar Bilimleri ve Teknolojileri Dergisi

BİLGİSAYAR BİLİMLERİ VE TEKNOLOJİLERİ DERGİSİ

CİLT 1, SAYI 2

ISSN: 2717-8579

ARALIK 2020



Bilgisayar Bilimleri ve Teknolojileri Dergisi

Dergi Hakkında

Bilgisayar Bilimleri ve Teknolojileri Dergisi bilim ve teknolojideki gelişmelere paralel olarak bilgisayar bilimleri ve teknolojileri alanında yeni gelişmelerle ilgili yapılan çalışmaları yayınlayan bir dergidir.

Amaç & Kapsam

BIBTED Dergisi,

✚ Bilgisayar Bilimleri ve Teknolojileri Dergisinin amacı bilgisayar alanında yapılan özgün çalışmaları yayınlamaktır. Yazım kurallarına uygun olarak hazırlanan eser, dergi editörlüğüne değerlendirme için hakemlere gönderilir. Bilgisayar Bilimleri ve Teknolojileri Dergisinde **KÖR HAKEMLİK** uygulaması mevcuttur. Yayımlanmasına, hakemlerin görüşü doğrultusunda Dergi Editör ve Yayın Kurulu karar verir. Gönderilen makaleler yayımlansın veya yayımlanmasın iade edilmez. Dergimizde yayınlanan yazıların her türlü sorumluluğu (bilimsel, mesleki, hukuki, etik vb.) yazarlara aittir. Yayınlanan yazıların telif hakkı dergiye aittir ve referans gösterilmeden aktarılamaz. Araştırmacılar arasındaki bilimsel iletişimi oluşturmak amacıyla aşağıda nitelikleri açıklanan, başka bir yerde yayımlanmamış makaleler Türkçe ve İngilizce olarak kabul edilmekte ancak Türkçe Kabul edilen makalenin özetinin İngilizce de basılması zorunluluğu vardır.

Aşağıdaki türlerdeki makaleler dergide yayına kabul edilmektedir:

- ✚ **Araştırma makalesi:** Özgün bir araştırmayı sonuçlarıyla birlikte sunan makale,
- ✚ **Derleme makale:** Bilgisayar Mühendisliği alanında belli bir konuda yeterli sayıda bilimsel makaleyi tarayıp, özetleyen, değerlendirme yapan ve bulguları yorumlayan makale,
- ✚ **Endüstriyel makale:** Bu alanda endüstride yapılan araştırma ve geliştirilen yeni ürün veya teknolojilerin açıklandığı makale,
- ✚ **Tez çalışması:** Lisansüstü düzeyde yapılan özgün bir tez çalışmasının genişletilmiş özetini içeren yazı,
- ✚ **Kitap yorumu:** Bilgisayar mühendisliği alanında yayınlanmış yeni bir kitabın tanıtılması ve değerlendirilmesi.
- ✚ **Kısa Bildiri:** Yapılan bir araştırmanın önemli bulgularını açıklayan yeni bir yöntem veya teknik tanımlayan yazılar.

Bütün yazıların Telif Hakkı Devri, yazarlarına bir form gönderilmek suretiyle alınır. Telif Hakkı Devir Formu göndermeyen yazarların yayınları işleme konmaz. Yayımlanmasına karar verilen yazılar üzerine yazarlarınca hiçbir eklenti yapılamaz.

Her yazı konusu ile ilgili en az iki hakeme gönderilerek şekil ve içerik bakımından incelenir. Dergide yayımlanabilecek nitelikteki yazılar dizgisi yapıldıktan sonra, yazarlarına gönderilerek baskı öncesi gözden geçirilir. Makale içinde, dergide basıldığı haliyle gözükken hataların sorumluluğu yazarlarına aittir. Hata, editörlük ofisinden kaynaklandığı takdirde düzeltme yayımlanabilir.

Derginin Kapsamı;

Bilgisayar Bilimleri ve Teknolojileri Dergisinin kapsamı, akıllı sistemler, algoritmalar, benzetim, bilgisayar ağları, bilgisayar grafiği, bilgisayarla görme, bilgisayar mimarisi, bilgiye erişim, bilimsel hesaplama, bilişim güvenliği, biyoenformatik, kriptografi, paralel işleme, doğal dil işleme donanım, görüntü işleme, hesaplama kuramı, işaret işleme, işletim sistemleri, makine öğrenmesi, mobil sistemler, modelleme, tıbbi bilişim, veri madenciliği, veri tabanı sistemleri, yazılım mühendisliği, siber güvenlik, yapay zeka dahil olmak üzere bilgisayar bilimleri ve teknolojilerin tüm alanları içerir.

Yayımlanma Sıklığı

Yılda 2 sayı

ISSN

2717-8579

WEB

<https://dergipark.org.tr/tr/pub/bibted>

İletişim

eavaroglu@mersin.edu.tr / ttuncer@firat.edu.tr / kemaladem@gmail.com



Bilgisayar Bilimleri ve Teknolojileri Dergisi

EDİTÖR

Doç. Dr. Erdiç AVAROĞLU

Mersin Üniversitesi, Mühendislik Fakültesi / Bilgisayar Mühendisliği, Mersin

EDİTÖR YARDIMCILARI

Doç. Dr. Taner TUNCER

Fırat Üniversitesi, Mühendislik Fakültesi / Bilgisayar Mühendisliği, Elâzığ

Dr. Öğr. Üyesi. Kemal ADEM

Aksaray Üniversitesi, İktisadi ve İdari Bilimler Fakültesi / Yönetim Bilişim Sistemleri, Aksaray

EDİTÖR KURULU

- **Prof. Dr. Zeki YETKİN, MERSİN ÜNİVERSİTESİ**
- **Doç. Dr. İsmail KOYUNCU, AYFON KOCATEPE ÜNİVERSİTESİ**
- **Dr. Öğr. Üyesi Murat TUNA, KIRKLARELİ ÜNİVERSİTESİ**
- **Dr. Öğr. Üyesi Abdullah ELEWİ, MERSİN ÜNİVERSİTESİ**
- **Dr. Öğr. Üyesi Abdullah Erhan AKKAYA, İNÖNÜ ÜNİVERSİTESİ**
- **Dr. Öğr. Üyesi Lutfiye KUŞAK, MERSİN ÜNİVERSİTESİ**
- **Dr. Öğr. Üyesi Fatma Bünyal ÜNEL, MERSİN ÜNİVERSİTESİ**
- **Dr. Öğr. Üyesi Çiğdem ACI, MERSİN ÜNİVERSİTESİ**
- **Dr. Öğr. Üyesi Soner KIZILOLUK, TURGUT ÖZAL ÜNİVERSİTESİ**
- **Dr. Öğr. Üyesi Selman YAKUT, TURGUT ÖZAL ÜNİVERSİTESİ**

DANIŞMA KURULU

- **Prof. Dr. Ahmet Bedri ÖZER, FIRAT ÜNİVERSİTESİ**
- **Prof. Dr. Murat YAKAR, MERSİN ÜNİVERSİTESİ**
- **Doç. Dr. Fatih ÖZKAYNAK, FIRAT ÜNİVERSİTESİ**
- **Dr. Öğr. Üyesi Mehmet ACI, MERSİN ÜNİVERSİTESİ**
- **Dr. Öğr. Üyesi Murat TUNA, KIRKLARELİ ÜNİVERSİTESİ**
- **Doç. Dr. İsmail KOYUNCU, AFYON KOCATEPE ÜNİVERSİTESİ**

DİL EDİTÖRLERİ

- **Dr. Öğr. Üyesi Abdullah ELEWİ, MERSİN ÜNİVERSİTESİ**
- **Dr. Öğr. Üyesi Abdullah Erhan AKKAYA, İNÖNÜ ÜNİVERSİTESİ**
- **Arş. Gör. Dr. Dilek SABANCI, GAZİOSMANPAŞA ÜNİVERSİTESİ**

MİZANPAJ

- **Arş. Gör. Semih KAHVECİ, MERSİN ÜNİVERSİTESİ**



Bilgisayar Bilimleri ve Teknolojileri Dergisi

İçindekiler

Contents

ARAŞTIRMA MAKALELERİ; RESEARCH ARTICLES;

S.No

- 42-50 *Çok İşlemcili Gerçek Zamanlı Sistemlerde Zamanlama Algoritmaları*
Scheduling Algorithms in Multiprocessor Real-Time Systems
Muhammet Kürşat Yıldırım, Lütfi Süder, Ahmed Abd Alkader, Abdullah Elewi
- 51-57 *Sosyal Medya Kullanıcılarının Cumhuriyet İttifakı Hakkındaki Görüşlerinin Makine Öğrenmesi*
Teknikleri ile Sınıflandırılması
Classification of Social Media Users' Opinions on the Cumhuriyet Alliance with Machine Learning Techniques
Yaşar Safalı
- 66-73 *Son İşlem Algoritmalarının İncelenmesi*
Examination of Post Processing Algorithms
Didem Yosunlu, Erdinç AVAROĞLU
- 74-81 *A Distributed Computer System Architecture Acting As A High Performance Computing System For Big Data Analytics*
Büyük Veri Analitiği İçin Yüksek Performans Hesaplama Sistemi Gibi Davranan Bir Dağıtık Bilgisayar Sistemi Mimarisi
Hamza EROL, Timuçin KORKMAZ

DERLEME MAKALELERİ; REVIEW ARTICLES;

S.No

- 58-65 *Yeni Nesil Web Paradigması: Web 4.0*
New Generation Web Paradigm: Web 4.0
Betül Ersöz



Araştırma Makalesi

Çok İşlemcili Gerçek Zamanlı Sistemlerde Zamanlama Algoritmaları

Muhammet Kürşat Yıldırım*¹, Lütfi Süder¹, Ahmed Abd Alkader², Abdullah Elewi¹

¹ Mersin Üniversitesi, Bilgisayar Mühendisliği Bölümü, Mersin, Türkiye

² Hatay Mustafa Kemal Üniversitesi, Hassa Meslek Yüksekokulu, Hatay, Türkiye

Anahtar Kelimeler:

Gerçek zamanlı sistemler
Çok işlemci
Zamanlama
MCRTsim

ÖZ

Gerçek zamanlı sistemler hayatımızın birçok alanında çok önemli bir rol oynar. Bu sistemlerin düzenli çalışabilmesi ve performans bakımından daha verimli olabilmesi için kullanılan bazı algoritmalar vardır. Bu çalışmada çok işlemcili gerçek zamanlı sistemler üzerinde çalışan zamanlama algoritmalarından olan, Evrensel Erken Biten Önce (Global Earliest Deadline First-GEDF), Bölmeli Erken Biten Önce (Partitioned Earliest Deadline First-PEDF), Bölmeli Monoton Oran (Partitioned Rate Monotonic-PRM) ve Evrensel Monoton Oran (Global Rate Monotonic-GRM) algoritmaları ele alınmıştır. Bu algoritmalar işlemciye gelen görevlerin hangi sıra ile işleme konulacağını belirlemektedir. Bu algoritmalarla ilgili bazı örnekler, açık kaynaklı MCRTsim uygulamasında çalıştırılmıştır ve alınan karşılaştırmalı sonuçları verilmiştir.

Scheduling Algorithms in Multiprocessor Real-Time Systems

Keywords:

Real-time systems
Multiprocessor
Scheduling
MCRTsim

ABSTRACT

Real-time systems play a crucial role in many aspects of our lives. There are some algorithms used for these systems to work regularly and to be more efficient in terms of performance. In this study, the scheduling algorithms for multiprocessor real-time systems are discussed. These algorithms include: Global Earliest Deadline First (GEDF), Partitioned Earliest Deadline First (PEDF), Partitioned Rate Monotonic (PRM) and Global Rate Monotonic (GRM) algorithms. These algorithms determine the order in which the tasks coming to the processor will be processed. Comparative results of these algorithms using open-source MCRTsim application have been given.

*Sorumlu Yazar

(m.kursatyildirim33@gmail.com) ORCID ID 0000-0001-7346-8868
(lutfisuder1992@gmail.com) ORCID ID 0000-0002-6949-0824
(akader@mku.edu.tr) ORCID ID 0000-0002-0538-7924
(elewi@mersin.edu.tr) ORCID ID 0000-0001-9774-5292

1. GİRİŞ

Gerçek zamanlı sistemler kullanımı oldukça yaygın olan ve her geçen gün daha da yaygınlaşan sistemlerdir. Verilen görevlerin tam zamanında yerine getirilmesi için gerçek zamanlı sistemlere ihtiyaç vardır. Çünkü bir işin tam zamanında yapılması gerekiyorsa bu iş, zaman açısından kritik bir iştir. Başka bir deyişle, gerçek zamanlı sistemler, sistemin gerçek zamana tabi olduğu anlamına gelir, yani yanıt, belirli bir zamanlama kısıtlaması dahilinde garanti edilmeli veya sistem belirtilen zaman sınırını karşılamalıdır. Gündelik yaşantımızda güvenlik (radar sistemleri), ulaşım (otomobillerdeki fren sistemleri), haberleşme (cep telefonları) gibi birçok alanda gerçek zamanlı sistemler kullanılmaktadır.

Günümüzde işlemcilerin artan maliyetlerinden dolayı bir gömülü sistemde birden fazla görev ve iş eş zamanlı olarak çalıştırılmaya başlanmıştır. Bu görevlerin doğru, eksiksiz ve güvenilir bir şekilde çalıştırılması ve kaynakların doğru kullanılması çok önemlidir.

Çok işlemcili gömülü sistemler en az iki veya daha fazla işlemciden oluşur. Çok işlemcili zamanlamada, çeşitli işlemlerin aynı anda çalışması için yükü paylaşan birden fazla işlemci (CPU) vardır.

Çok işlemcili bir sistemde ayarlanan gerçek zamanlı bir görevin son tarihlerini karşılamak için bir zamanlama algoritması gerekir. Bu zamanlama algoritması, sistemdeki her görevin dağıtımın uygun bir biçimde yapılmasından (dağıtma problemi) ve ne zaman ve hangi sırayla bu görevlerin çalıştırılmasının (zamanlama problemi) belirlenmesinden sorumludur. Yaşanılan genel problemler şunlardır:

- Tek bir işlemci için bazı araştırma sonuçları her zaman birden fazla işlemci için uygulanamaz (Dhall ve Liu, 1978; Garey ve Johnson, 1979).
- Birden fazla işlemcide farklı zamanlama anormallikleri ortaya çıkabilir (Baker, 2003; Andersson, 2003; Ha ve Liu 1994).
- Dağıtma problemine çözüm olarak yüksek hesaplama karmaşıklığına sahip algoritmalar kullanmak gerekir.

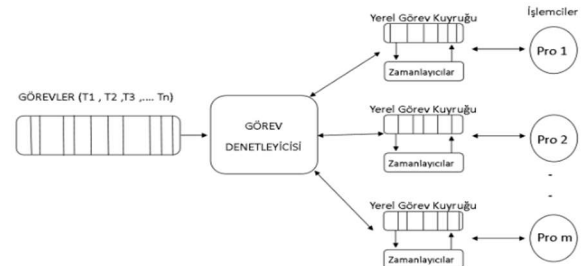
Yukarıdaki problemlerin giderilebilmesi için literatürde bulunan çeşitli algoritmalar kullanılabilir. Bu çalışmada en sık kullanılan algoritmalar olan GEDF ve GRM algoritmalarından bahsedilmiş olup, MCRTsim uygulaması üzerinde çalıştırılan simülasyon sonuçları ve karşılaştırmaları verilmiştir. MCRTsim uygulaması, Java programlama dili ile geliştirilmiş ve açık kaynaklı bir uygulamadır. Tek işlemcili, çok işlemcili ve çok çekirdekli işlemcilere sahip gerçek zamanlı sistemler için açık kaynaklı bir görev zamanlama simülatörü sunar (Wu ve Huang, 2017). Yaptığımız çalışmada MCRTsim uygulamasının

yeniden geliştirilip derlenmiş olan versiyonu kullanılmıştır (URL-1).

Bu çalışmada, ikinci bölümde zamanlama algoritmalarının (GRM, GEDF, PEDF, PRM) yapısı ve çalışma mantıkları anlatılmıştır. Gerekli matematiksel denklemler verilmiştir. Üçüncü bölümde ise bu algoritmalarla ilgili örnekler MCRTsim uygulaması üzerinde simüle edilmiştir. Sonuç bölümünde bu simülasyon sonuçları değerlendirilmiş ve çıkarımlarda bulunulmuştur.

2. ZAMANLAMA ALGORİTMALARI

Zamanlama algoritmaları sistemlerin kritik seviyedeki görevleri en doğru şekilde işlemelerini sağlamak için oluşturulan algoritmalarlardır. Bu algoritmalar evrensel (global) ve bölmeli (partitioned) olarak iki şekilde işlenmektedir (Davis ve Burns, 2011). Bölmeli algoritmalar verilecek bir görevi hangi işlemci üstlenmişse o görevi sadece o işlemci icra etmektedir. Bölmeli zamanlama algoritmalarının genel yapısı Şekil 1.(a)'da gösterilmiştir. Evrensel algoritmada ise bir görev farklı farklı işlemciler tarafından icra edilebilmektedir (Zapata ve Alvarez, 2005; Guan, 2016). Evrensel zamanlama algoritmalarının genel yapısı Şekil 1.(b)'de gösterilmiştir. Burada dikkat edilmesi gereken nokta ise bir görevin farklı parçaları aynı anda farklı farklı işlemcilerde işlenemez.



Şekil 1.(a) Bölmeli zamanlama algoritmalarının genel yapısı

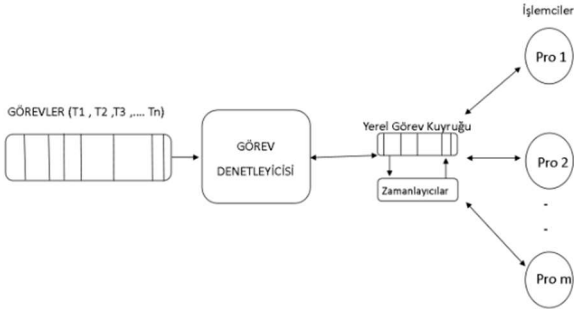
Global zamanlamada, bir görev hazır kuyruğuna konur ve tüm işlemciler tarafından paylaşılır ve her an m yüksek öncelikli görev, yürütme için seçilen m işlemci tarafından işleme konur.

Global zamanlama algoritmalarının tasarım düzeni şu şekildedir:

- Görevlerin(tasks) işlemcilere aktarılması sisteme yüksek yük getirir.
- Bir işlemciden diğerine aktarılmasına izin verildiğinden, işlemciler yüksek olma eğilimindedir. Bu iletişim kanalları paylaşılan iletişim hafızasının kullanılmasını sağlar.
- Bu tasarım düzeninde Dhall Etkisi oluşabilir ve görevler küçük kullanımlı

şekilde planlanamayabilir (Dhall ve Liu, 1978).

- Bu tasarım düzeninde bazı programlama anomalileri oluşabilir (Lauzac ve ark., 1998; Andersson ve Jonsson, 2000; Andersson, 2003).



Şekil 1.(b) Evrensel zamanlama algoritmalarının genel yapısı

Bölmeli algoritmaları gelen görevlerin işlemciye hangi öncelik sırasına göre yerleştirileceğini belirlemede kullanılan algoritmalarıdır. Bu algoritmaların doğru çalışması için görevlerin işlemciye aynı anda gelmesi gerekmektedir. Literatürde birçok bölme algoritması bulunmaktadır. Bu yazıda daha çok ilk bulunan yere yerleştirme (First Fit-FF), en iyi yerleştirme (Best Fit-BF), en kötü yerleştirme (Worst Fit-WF) algoritmaları (Zapata ve Alvarez, 2005; Elewi ve ark., 2014) ve bunların türevleri incelenecek ve MCRTsim uygulaması üzerinden analiz edilecektir. Bu algoritmaların en büyük göreve öncelik vererek yerleştirmesine de FFD (FF Decreasing), BFD (BF Decreasing) ve WFD (WF Decreasing) denmektedir.

İlk Bulunana Azalan Yerleştirme (FFD): FFD algoritmasında, öğeler ilk önce azalan boyutta sıralanır ve sonra $C(B_i) + s(a_i) \leq l$ denklemine göre sığacağı ilk yere (en düşük indeks) yerleştirilir. Aksi takdirde, ilk elemanı a_i olan yeni bir boş alan başlatılır (Guochuan ve Enyu, 1998).

En İyi Azalan Yerleştirme (BFD): BFD algoritmasında, öğeler ilk önce azalan boyutta sıralanır ve çalıştırılır.

En Kötü Azalan Yerleştirme (WFD): WFD algoritmasında, öğeler ilk önce azalan sırada sıralanır ve daha sonra boşluğu en büyük olan işlemciye yerleştirilir ve çalıştırılır.

Bölme algoritmaları zamanlama algoritmaları ile uygun şekilde çalışabilmektedirler. Özellikle Erken Biten Önce (Earliest Deadline First-EDF) zamanlama algoritması şu şekilde grup oluşturabilmektedir (Beitollahi ve Deconinck, 2006):

$$A = \{NF, BF, FF, \dots\}x(EDF \text{ Algoritması})$$

2.1. Evrensel Monoton Oran (Global Rate Monotonic-GRM)

Evrensel Monoton Oran (GRM) zamanlama algoritması en sık kullanılan algoritmalarından bir tanesidir. Bu algoritmada ilk işlenecek görev en küçük periyotlu görevdir. İstisnai bir durum olarak, görevin işlemciye varış zamanı öncelik durumunu değiştirmektedir. GRM algoritmasında işlemci bir görevi icra ederken daha yüksek öncelikli bir görev gelirse o göreve geçiş yapan (preemptive) bir algoritmadır. Gerekli durumlarda bir görevden diğer göreve geçişi olmayacak şekilde (nonpreemptive) de ayarlanabilir. RM algoritmasının hatasız çalışması için 1973 yılında Liu ve Layland tarafından ortaya çıkarılan denklem 1 (U işlemci kullanım yüzdesi, n görev sayısı, C hesaplanma süresi ve P periyot olmak üzere) deki şartı sağlaması gerekmektedir. Bu şart tek işlemcili durumlarda geçerlidir (Liu ve Layland, 1973).

$$U = \sum_{i=1}^n \left(\frac{C_i}{P_i} \right) \leq n(\sqrt{2} - 1) \quad (1)$$

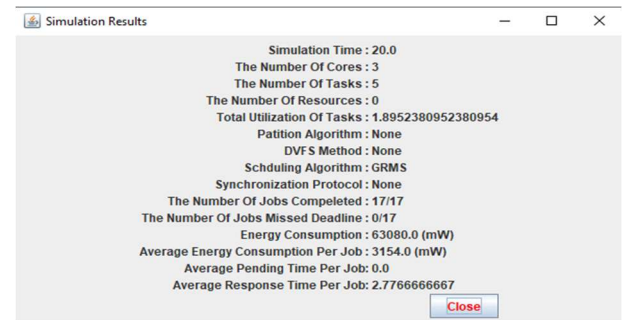
Aşağıda bulunan denklem 2'ye göre, son teslim tarihleri olan bir dizi periyodik görev, m işlemciyle, $m \geq 2$ şartını sağlayacak şekilde preemptive global monoton oran (GRM) algoritması kullanılarak programlanırsa görevlerin başarıyla tamamlanması garantidir (Baker, 2003).

$$\sum_{i=1}^n \frac{C_i}{P_i} \leq \frac{m}{2} (1 - \alpha) + \alpha \quad (2)$$

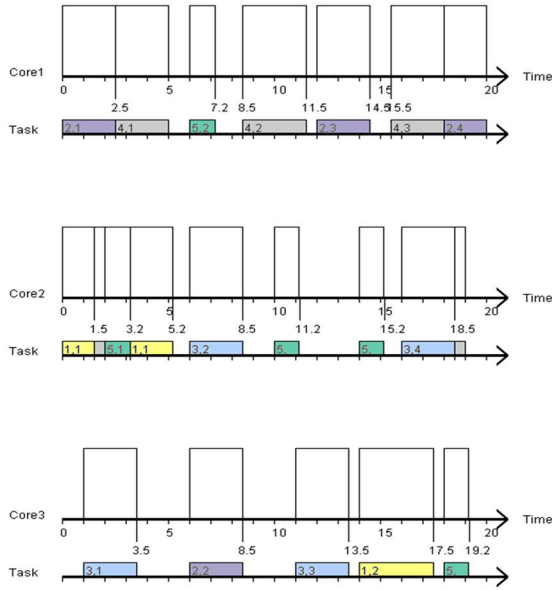
$$\alpha = \max \left\{ \frac{C_i}{P_i} \mid i = 1, \dots, n \right\}$$

Bu kısımda GRM algoritmalarının nasıl çalıştığı MCRTsim uygulaması kullanılarak gösterilmiştir. Burada görevler $T_n(r, C, P)$ şeklinde gösterilmiştir. Bu gösterimde r görevin işlemciye varış zamanını, C görev süresini ve P görevin periyodunu göstermektedir. Örnek olarak $T_1(0, 3.5, 14)$, $T_2(0, 2.5, 6)$, $T_3(1, 2.5, 5)$, $T_4(1.5, 3, 7)$, $T_5(2, 1.2, 4)$ görevleri çalıştırılmıştır. Denklem 2'deki formülden yararlanılarak işlemci sayısı 3 olarak belirlenmiştir.

Aşağıda Şekil-2.(a)'da ki sonuca (Total Utilization of Tasks) baktığımızda toplam kullanım 1.895 olarak görülmüştür. Böylece 2,25'den küçük olduğu için denklem 2'yi sağlamış ve simülasyon başarıyla çalışmıştır.



Şekil 2. (a). GRM Algoritması Simülasyon Sonucu



Şekil 2. (b). GRM Algoritması İşlemci Görev Grafiği

Ayrıca Şekil-2.(b)'deki işlemci görev grafiğine baktığımızda bütün işlemcilere görev dağılımının yapıldığı görülmüştür. Bu noktada dikkat edilmesi gereken diğer bir husus ise; bir görev başladığı bir işlemci tarafından sonlandırılmadan diğer işlemciye aktarılabilir. Örneğin Şekil-2.(b)'deki işlemci zaman grafiğine baktığımızda; 4 numaralı görevin ilk periyodu 2 numaralı işlemci ile başlayıp 1 numaralı işlemci ile devam etmiştir.

2.2. Evrensel Erken Biten Önce (Global Earliest Deadline First-GEDF)

Bir dizi periyodik görevin zamanlanması için kullanılan en yaygın dinamik öncelikli zamanlama yöntemi EDF algoritmasıdır (Palamut ve ark., 2019). Bu algoritmada öncelikle periyot süresi kısa olan işlem yapılmaktadır. Periyot sürelerine göre sıraya koyulmaktadır. Bu algoritmada en önemli noktalardan biri ise öncelik sıralaması yapıp yapılmayacağına karar verilebilmesidir. Yani önceliği yüksek olan bir görev diğer görevin geçici olarak durdurulmasına ve kendisinin icra edilmesine (preemptive) sebep olabilir ya da tam tersi olarak çalışabilir. Yani en kısa görev süresi olan görev bitene kadar yeni görev hiçbir şekilde icra edilmez(non preemptive).

1973 yılında Liu ve Layland tarafından yapılan çalışmaya göre eğer görev zamanlamaları denklem 3'de ki eşitliği (U işlemci kullanım yüzdesi, n görev sayısı, C hesaplanma süresi ve P periyot olmak üzere) sağlıyorsa EDF tek işlemcili durumlarda sorunsuz çalıştırılabilir (Liu ve Layland, 1973).

$$U = \sum_{i=1}^n \left(\frac{C_i}{P_i} \right) \leq 1 \quad (3)$$

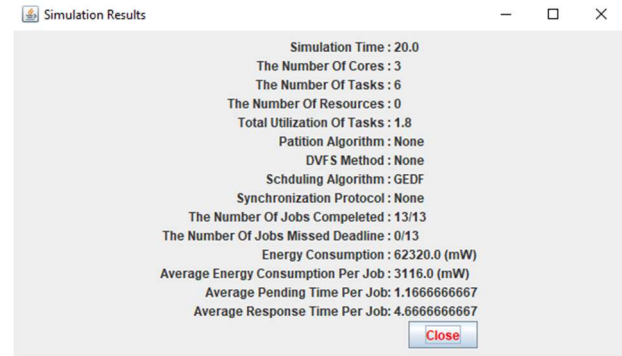
Yukarıda belirtilen denklem 3'de ki kurala göre, bir dizi periyodik görev, T_1, T_2, \dots, T_n tüm periyotların eşit olması kaydıyla, m işlemci üzerinde

preemptive GEDF algoritması kullanılırsa görevlerin başarıyla tamamlanması garantidir (Goossens ve ark., 2003).

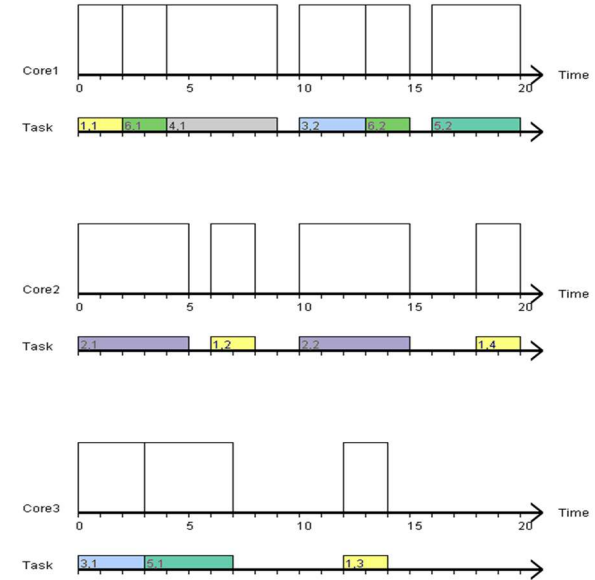
$$\sum_{i=1}^n \frac{C_i}{P_i} \leq m(1 - \alpha) + \alpha \quad (4)$$

$$\alpha = \max \left\{ \frac{C_i}{P_i} \mid i = 1, \dots, n \right\}.$$

Bu çalışmada PEDF algoritması ile çeşitli parçalama algoritmalarının kombinasyonları ve GEDF algoritması MCRTsim programında çalıştırılmıştır. Örnek olarak; $T_1(0, 2, 6)$, $T_2(0, 5, 10)$, $T_3(0, 3, 10)$, $T_4(0, 5, 20)$, $T_5(0, 4, 16)$, $T_6(0, 2, 12)$ görevleri çalıştırılmıştır. Simulasyon sonuçları Şekil-3 ve MCRTsim uygulaması simulasyon sonuçları başlığı altında yorumlanmıştır.



Şekil 3. (a). GEDF Algoritması Simülasyon Sonucu



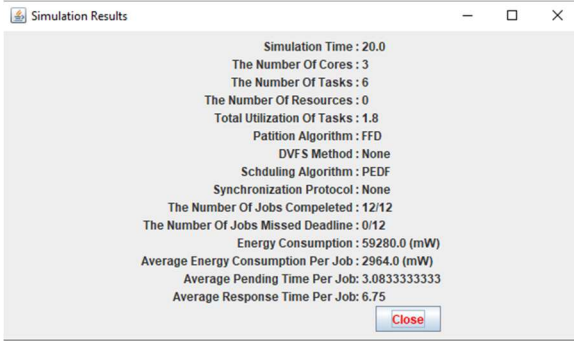
Şekil 3. (b). GEDF Algoritması İşlemci Görev Grafiği

2.3. Bölmeli Erken Biten Önce (Partitioned Earliest Deadline First-PEDF)

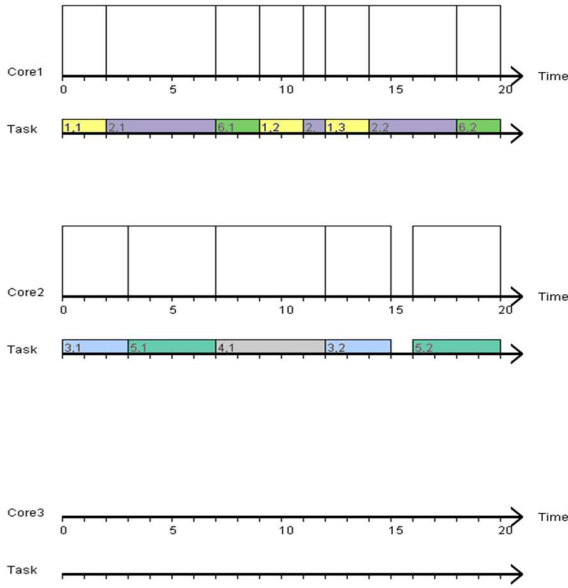
PEDF-WFD algoritmaları bütün işlemcileri kullanma gereği duyarken PEDF-FFD ve PEDF-BFD algoritmaları sadece 2 işlemciyi kullanarak görevleri başarıyla icra edebilmiştir.

Aşağıda Şekil-4.a'da PEDF-FFD algoritmasının başarılı şekilde çalıştığı görülmüştür. Diğer

algoritmalar ile karşılaştırma sonuçları Tablo 1.'de verilmiştir.

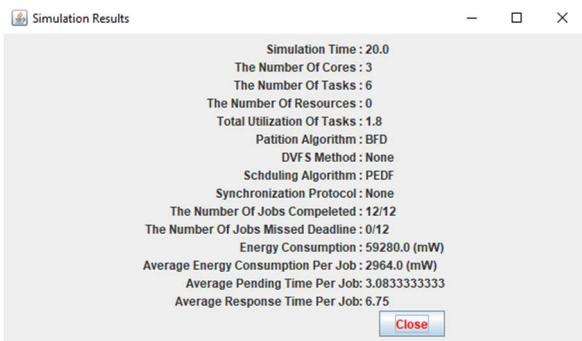


Şekil 4. (a). PEDF-FFD Algoritması Simülasyon Sonucu

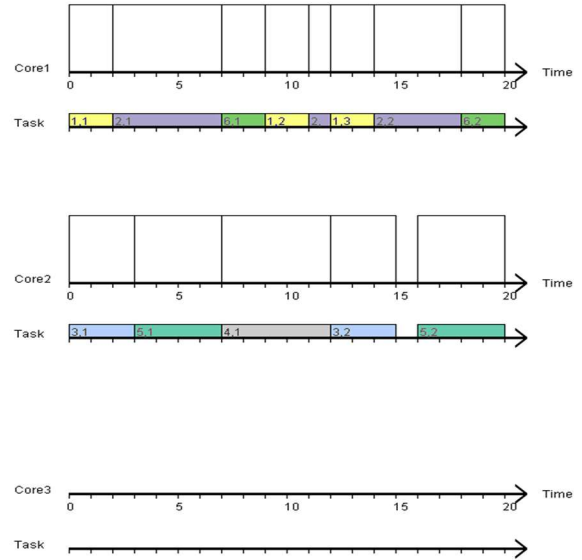


Şekil 4. (b). PEDF-FFD Algoritması İşlemci Görev Grafiği

Aşağıda Şekil-5.a'da PEDF-BFD algoritmasının başarılı şekilde çalıştığı görülmüştür. Diğer algoritmalar ile karşılaştırma sonuçları Tablo 1.'de verilmiştir.

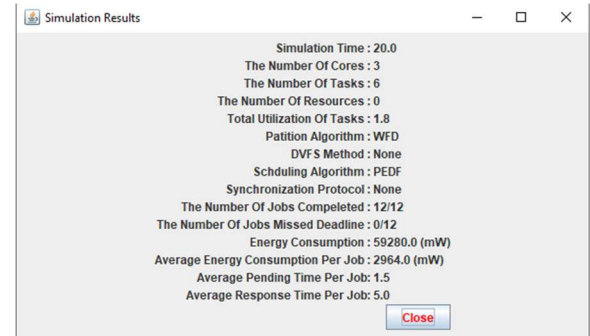


Şekil 5. (a). PEDF-BFD Algoritması Simülasyon Sonucu

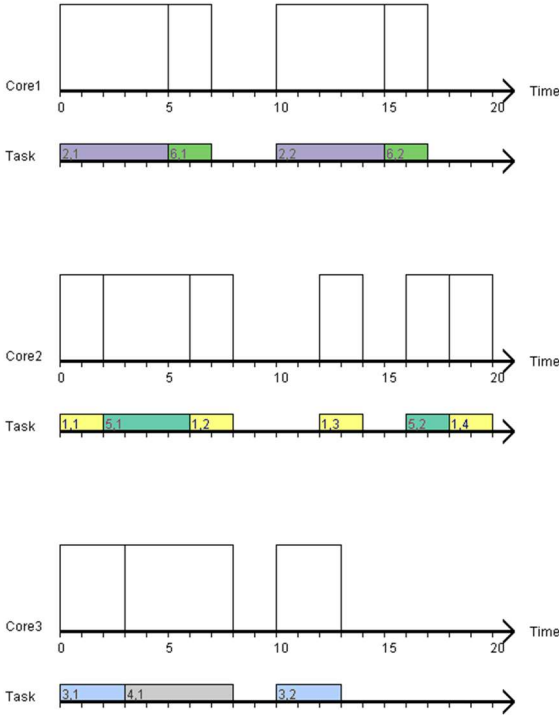


Şekil 5. (b). PEDF-BFD Algoritması İşlemci Görev Grafiği

Aşağıda Şekil-6.a'da PEDF-WFD algoritmasının başarılı şekilde çalıştığı görülmüştür. Diğer algoritmalar ile karşılaştırma sonuçları Tablo 1.'de verilmiştir.



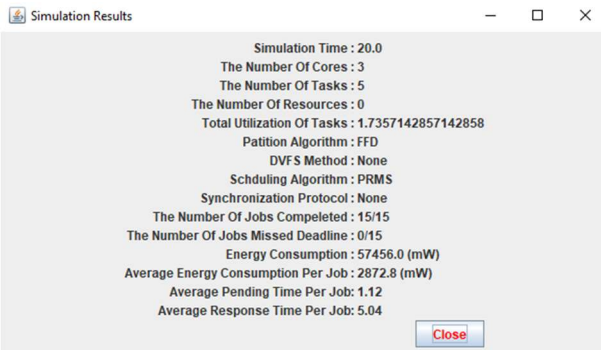
Şekil 6. (a). PEDF-WFD Algoritması Simülasyon Sonucu



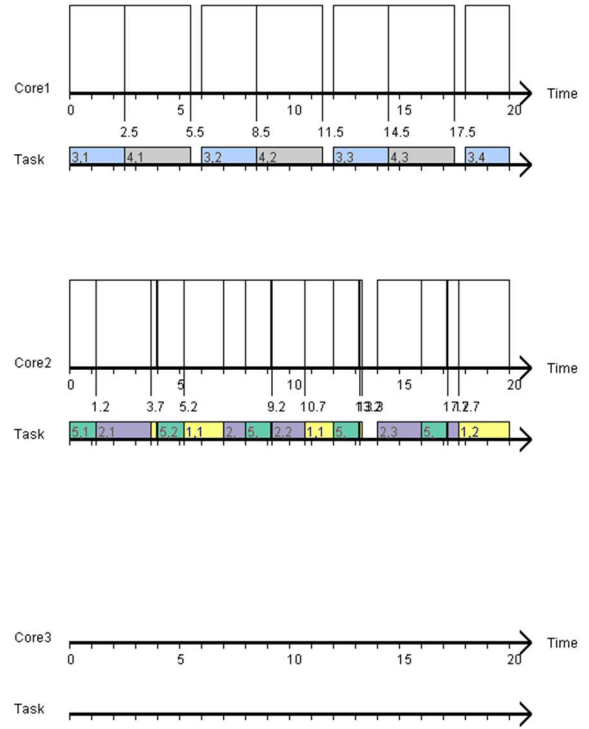
Şekil 6. (b). PEDF-WFD Algoritması İşlemci Görev Grafiği

2.4. Bölmeli Monoton Oran (Partitioned Rate Monotonic -PRM)

PRM algoritması FFD, BFD, WFD algoritmaları ile çalıştırılmıştır, ayrıca Şekil-7, Şekil-8 ve Şekil-9'da algoritmasının çalıştığı görülmüştür. Örnek görev olarak da $T_1(0, 3.5, 15)$, $T_2(0, 2.5, 7)$, $T_3(0, 2.5, 6)$, $T_4(0, 3, 7)$, $T_5(0, 1.2, 4)$ görevleri kullanılmıştır. Simülasyon sonuçlarına baktığımızda WFD algoritmasının daha kısa sürede cevap verebildiği görülmüştür. İşlemci görev grafiğine baktığımızda WFD algoritmasının 3 işlemciyi de kullandığı görülmüştür. BFD ve FFD algoritmasının ise sadece 2 işlemciyi kullandığı görülmüştür. Bitene kadar yeni görev hiçbir şekilde icra edilmez (non preemptive).



Şekil 7. (a). PRM-FFD Algoritması Simülasyon Sonucu
PRM-FFD algoritması başarılı çalışmıştır.

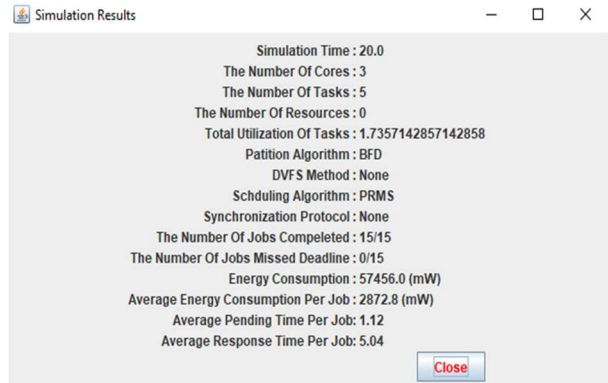


Şekil 7. (b). PRM-BFD Algoritması İşlemci Görev Grafiği

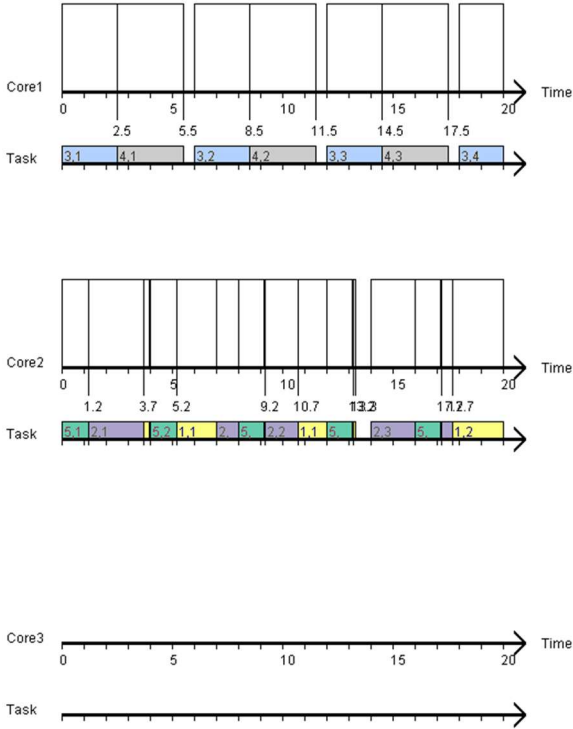
Yukarıda Şekil-7.a'da PRM-FFD algoritmasının başarılı şekilde çalıştığı görülmüştür. Ayrıca Şekil-7.b'de bu görevleri başarabilmesi için sadece iki işlemcinin yeterli olduğu görülmüştür. Diğer algoritmalar ile karşılaştırma sonuçları Tablo 2.'de verilmiştir.

PRM-BFD algoritması PRM-FFD algoritması ile aynı sonucu vermiştir. Aşağıda Şekil-8.a'da PRM-BFD algoritmasının başarılı şekilde çalıştığı görülmüştür. Diğer algoritmalar ile karşılaştırma sonuçları Tablo 2.'de verilmiştir.

Aşağıda Şekil-9.a'da PRM-WFD algoritmasının başarılı şekilde çalıştığı görülmüştür. Şekil-9.b'ye baktığımızda ise bu algoritmanın aynı işi yapabilmesi için üç işlemci kullandığı görülmüştür. Diğer algoritmalar ile karşılaştırma sonuçları Tablo 2.'de verilmiştir.

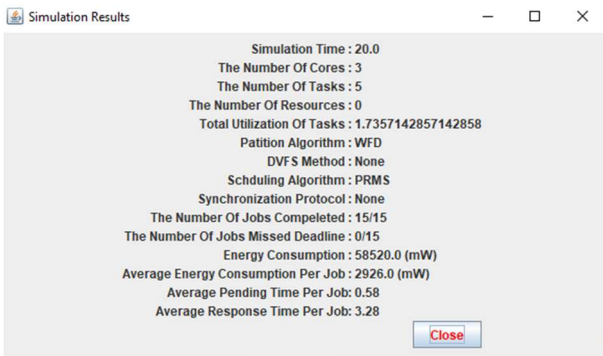


Şekil 8.(a). PRM-BFD Algoritması Simülasyon Sonucu

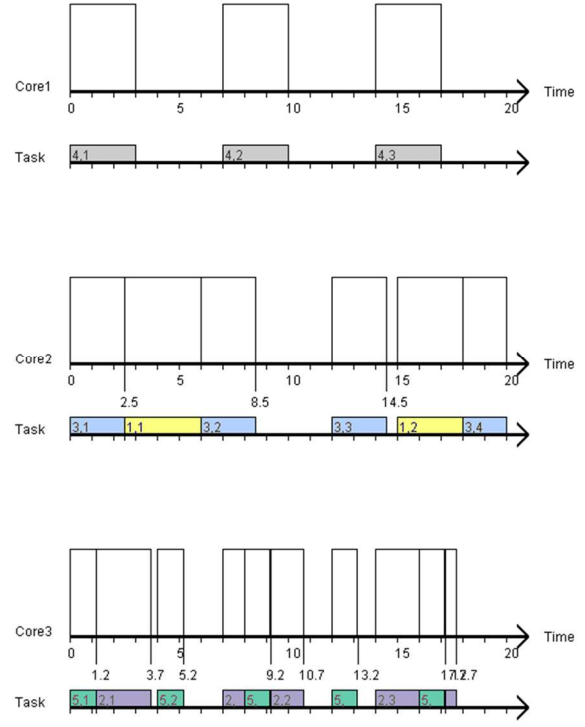


Şekil 8. (b). PRM-BFD Algoritması İşlemci Görev Grafiği

MCRTsim uygulamasında draw butonuna tıklayarak simülasyon sonucu grafiksel olarak gösterilebilmektedir. Çizim yaptırıldığında AllCores butonu tıklanarak aşağıda belirtilen işlemci görev grafikleri elde edilmektedir. Bu grafiklerde hangi işlemcinin hangi işi yüklediği ve hangi sürelerde çalıştığı gözlemlenebilmektedir. Grafikte belirtilen task satırında birinci rakam kaç numaralı görev olduğunu, ikinci rakam ise o görevi kaçınıcı defa çalıştırdığını göstermektedir. Örneğin 5,1 gösteriminde 5 görev numarası, 1 ise kaçınıcı defa çalıştığını göstermektedir.



Şekil 9. (a). PRM-WFD Algoritması Simülasyon Sonucu



Şekil 9. (b). PRM-WFD Algoritması İşlemci Görev Grafiği

3. MCRTsim UYGULAMASI SİMÜLYASYON SONUÇLARI

GRM algoritması, GEDF ve PEDF algoritmaları için yukarıdaki şekillerde verilen örnekler yapılmıştır. Bu örneklerle ilgi çıkarımlar bu kısımda anlatılmıştır.

3.1. Evrensel Monoton Oran (GRM) Algoritması Simülasyon Sonuçları

GRM algoritması için verilen denklem 2'deki kural gözetilerek 3 işlemcili simülasyon yapılmıştır. Simülasyon sonuçlarına göre denklem 2'de sağlanması gereken $m \geq 2$ koşulu başarıyla yerine getirilmiştir.

GRM algoritması için Şekil-2.(a)'daki Simülasyon sonucuna bakıldığında GRM algoritmasının bütün görevleri başarılı bir şekilde yerine getirdiği görülmüştür. Şekil-2.(b)'deki GRM Algoritması İşlemci Görev Grafiğine baktığımızda 3 işlemciyi de aktif olarak kullandığı görülmektedir. Ayrıca dikkat edilirse GRM algoritmasının Şekil-2.(b)'deki 4 numaralı görevin 1. icrası esnasında bu görevin bir kısmını 1. işlemciye yaptırırken diğer bir kısmını 2. işlemciye yaptırmaktadır.

3.2. Evrensel Erken Biten Önce (GEDF) Algoritması Simülasyon Sonuçları

GEDF algoritmasının Şekil-3.(a)'da ki Simülasyon sonucuna baktığımızda görevi başarılı bir şekilde tamamladığı görülmüştür. Tablo 1. deki ortalama cevap süresi karşılaştırmalarına göre

GEDF algoritmasının diğer PEDF algoritması kombinasyonlarına göre daha kısa sürede cevap verdiği görülmüştür. Şekil-3.(b)'deki GEDF Algoritması İşlemci Görev Grafiğine baktığımızda üç işlemcinin de aktif kullanıldığı ve bu algoritmanın nonpreemptive olarak çalıştığı görülmüştür. Bu yüzden bir görevi alan işlemci o görevi bitirene kadar yeni görev almamış ve elindeki görevi başka işlemciye devretmemiştir. Biten görev yeni görev olarak başka işlemcilere aktarılabilmiştir.

3.3 Bölmeli Erken Biten Önce (PEDF) Algoritması Simülasyon Sonuçları

PEDF algoritmasının hangi parçalama algoritmasında daha verimli çalıştığını anlamak için yukarıda belirtilen FFD, BFD ve WFD parçalama algoritmaları kullanılmıştır. Şekil-4.(a)'da ki PEDF-FFD Algoritması Simülasyon Sonucuna baktığımızda PEDF-FFD algoritmasının tüm görevleri başarılı bir şekilde icra ettiği görülmüştür. Ayrıca Şekil-4.(b)'deki PEDF-FFD Algoritması İşlemci Görev Grafiğine baktığımızda iki işlemcinin bu görevi yapması için yeterli olduğu görülmüştür. Şekil 5.(a)'da ki PEDF-BFD Algoritması Simülasyon Sonucuna baktığımızda bu parçalama algoritmasının da başarılı bir şekilde tüm görevleri icra ettiği görülmüştür. Şekil-5.(b)'deki PEDF-BFD Algoritması İşlemci Görev Grafiğine baktığımızda ise bu algoritmanın da aynı şekilde iki işlemci ile bu görevi başardığı görülmüştür. Şekil-6.(a)'da PEDF-WFD Algoritması Simülasyon Sonucuna baktığımızda bu algoritmanın da tüm görevleri başarılı şekilde icra ettiği görülmüştür. Fakat Tablo 1.'de de gösterildiği gibi PEDF-WFD algoritması PEDF-BFD ve PEDF-FFD algoritmalarına göre daha kısa sürede işlemi tamamlamıştır. Şekil-6.(b)'de PEDF-WFD Algoritması İşlemci Görev Grafiğine baktığımızda bu algoritmanın diğer iki algoritmadan farklı olarak üç işlemci kullandığı görülmüştür.

Tablo 1. GEDF ve PEDF Algoritmaları Görev Tamamlama Başarısı Sonuçları

	GEDF	PEDF-FFD	PEDF-BFD	PEDF-WFD
Görev Sayısı	6	6	6	6
Toplam Kullanım	1.8	1.8	1.8	1.8
Yapılan İş Sayısı	13	12	12	12
Tamamlanan İş Sayısı	13	12	12	12
Hatalı İş Sayısı	0	0	0	0
Ortalama Bekleme Süresi/İş Başına	1.17	3.08	3.08	1.5
Ortalama Tepki Süresi/İş Başına	4.67	6.75	6.75	5.0

3.4. Bölmeli Monoton Oran (PRM) Algoritması Simülasyon Sonuçları

Simülasyon sonuçlarına baktığımızda WFD algoritmasının daha kısa sürede cevap verebildiği görülmüştür. Şekil-9.(b)'deki İşlemci Görev Grafiğine baktığımızda WFD algoritmasının 3 işlemciyi de kullandığı görülmüştür. Şekil 8.(b)'de PRM-BFD algoritmasının ve Şekil-7.(b)'de PRM-FFD algoritmasının ise sadece 2 işlemciyi kullandığı görülmüştür. Ayrıca Şekil-9.(b)'de bir görevin hangi işlemci ile başladıysa sürekli o işlemci tarafından çalıştırıldığı görülmüştür.

Tablo 2'de GRM ve PRM algoritmalarının aynı örnek görevler için MCRTsim'de çalıştırılmasıyla elde edilen değerleri karşılaştırmalı olarak verilmiştir.

Tablo 2. GRM ve PRM Algoritmaları Görev Tamamlama Başarısı Sonuçları

	GRM	PRM-FFD	PRM-BFD	PRM-WFD
Görev Sayısı	5	5	5	5
Toplam Kullanım	1.895	1.736	1.736	1.736
Yapılan İş Sayısı	17	15	15	15
Tamamlanan İş Sayısı	17	15	15	15
Hatalı İş Sayısı	0	0	0	0
Ortalama Bekleme Süresi/İş Başına	0	1.12	1.12	0.58
Ortalama Tepki Süresi/İş Başına	2.78	5.04	5.04	3.28

4. SONUÇ

Yapılan bu çalışma sonucunda RM algoritması için evrensel algoritmanın cevap süresinin parçalı algoritmaya göre daha kısa olduğu görülmüştür. Ayrıca GRM algoritmasının işlemcide sürekli görev dağılımını değiştirdiği ve 3 işlemciyi de kullandığı gözlemlenmiştir. Fakat PRM-FFD ve PRM-BFD algoritmaları sadece 2 işlemciyi kullanmıştır. Bu da GRM algoritmasının daha kısa sürede cevap vermesi ve daha verimli çalışmasını sağlamıştır.

GEDF algoritması aynı süre içerisinde PEDF algoritmasına göre daha fazla görevi icra etmiştir. Ayrıca GEDF algoritmasının PEDF algoritmasına göre cevap verme süresi daha kısa olmuştur. Bu durumda GEDF algoritmasının daha verimli olduğu gözlemlenmiştir.

Parçalama algoritmalarından PEDF-WFD algoritması 3 işlemciyi kullanmıştır. Fakat PEDF-BFD ve PEDF-FFD algoritmaları 2 işlemciyi kullanmıştır. Bu da PEDF-WFD algoritmasının daha hızlı cevap vermesini sağlamıştır. Böylece PEDF-WFD algoritması daha verimli çalışmıştır.

Bir sonraki çalışmalarda ise bu algoritmaların enerji verimlilik değerlendirilmesi yapılabilir. Hangi algoritmanın aynı iş için daha fazla enerji tükettiği tespit edilebilir. Bu sonuçlara göre de hangi alanlarda hangi algoritmanın kullanılmasının daha mantıklı olabileceği tespit edilebilir.

KAYNAKÇA

- Andersson, B. (2003). Static Priority Scheduling in Multiprocessors. In *PhD Thesis, Department of Comp.Eng., Chalmers University, 2003*.
- Andersson, B. ve Jonsson, J. (2000). Fixed-Priority Preemptive Multiprocessor Scheduling: To Partition or not to Partition. In *IEEE Int'l Conference on Real Time Computing Systems and Applications, Dec. 2000*.
- Baker, T. P. (2003). Multiprocessor EDF and Deadline Monotonic Schedulability Analysis. In *IEEE Real-Time Systems Symposium, Dec, 2003*.
- Beitollahi, H. ve Deconinck, G. (2006). Fault-Tolerant Partitioning Scheduling Algorithms in Real-Time Multiprocessor Systems. In *12th Pacific Rim International Symposium on Dependable Computing (PRDC'06)*.
- Dhall, S. K. Ve Liu, C. L. (1978). On a Real-Time Scheduling Problem. In *Operations Research, vol. 26, number 1, 127-140, 1978*.
- Garey, M. R. ve Johnson, D. S. (1979). Computers and Intractability: A Guide to the Theory of NP-Completeness. In *W. H. Freeman, New York, 1979*.
- Goossens, J., Funk, S., & Baruah, S. (2003). Priority-Driven Scheduling of Periodic Task Systems on Multiprocessor. In *Real Time Systems, vol 25, 187-205, 2003*.
- Guan, N. (2016). Techniques for Building Timing-Predictable Embedded Systems. In *Springer International Publishing Switzerland, pp. 15, 2016*.
- Guochuan, Z. ve Enyu, Y. (1998). The FFD Algorithm for the Bin Packing Problem with Kernel Items. In *Applied Mathematics-A Journal of Chinese Universities, pp. 2, 1998*.
- Ha, R. ve Liu, J. (1994). Validating Timing Constraints in Multiprocessor and Distributed Real-Time Systems. In *Int'l Conf. on Distributed Computing Systems, pp. 162-171*.
- Davis, R. I. ve Burns, A. (2011). A survey of hard real-time scheduling for multiprocessor systems. In *ACM Comput. Surv. 43, 4, Article 35, 44 pages*.
- Lauzac, S., Melhem, R. & Mosse, D. (1998). Comparison of Global and Partitioning Schemes for Scheduling RM Tasks on a Multiprocessor. In *Euromicro Workshop on Real-Time Systems, June 1998*.
- Liu, L. ve Layland, J. W. (1973). Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment. *J. ACM, 1973*.
- Palamut, S., Gönültaş, T., Elewi A., & Avaroğlu, E. (2019). Task Scheduling Algorithms and Resource Access Protocols. In *2019 International Artificial Intelligence and Data Processing Symposium (IDAP), Malatya, Turkey, pp. 1-6*.
- Wu, J., ve Huang, Y. (2017). MCRTsim: A simulation tool for multi-core real-time systems. In *2017 International Conference on Applied System Innovation (ICASI), 461-464*.
- Zapata, O. ve Alvarez, P. (2005). EDF and RM Multiprocessor Scheduling Algorithms: Survey and Performance Evaluation. In *Dept. Seccion de Computacion, CINVESTAV-IPN, Mexico City, Tech. Rep. CINVESTAV-CS-RTG-02*.
- Elewi A., Shalan M., Awadalla M., & Saad E. M. (2014). Energy-efficient task allocation techniques for asymmetric multiprocessor embedded systems. In *ACM Transactions on Embedded Computing Systems (TECS) 13 (2s), 1-27*.
- URL-1: <https://github.com/SalihPalamut/MCRTsim> [Erişim Tarihi: 15.06.2020]



Araştırma Makalesi

Sosyal Medya Kullanıcılarının Cumhuriyet Halk Partisi Hakkındaki Görüşlerinin Makine Öğrenmesi Teknikleri ile Sınıflandırılması

Yaşar Safalı*¹

¹ Mersin Üniversitesi, Bilgisayar Mühendisliği Bölümü, Mersin, Türkiye

Anahtar Kelimeler:

Cumhuriyet Halk Partisi, Sosyal medya analizi, Metin Madenciliği, Kappa testi, Makine öğrenmesi

ÖZ

Sosyal ağların kullanım oranlarının artması ile sosyal medya kullanıcıları tarafından oluşturulan verilerin sayısı günden güne artmaktadır. Bu veriler ile duygu analizi, kişilik tespiti vb. akademik çalışmalar yapılabilmektedir. Bu çalışmada popüler sosyal ağlardan olan Facebook, Twitter, Instagram ve LinkedIn kullanıcılarının "Cumhuriyet Halk Partisi" hakkındaki görüşleri sınıflandırılmıştır. Bu amaçla, Facebook, Twitter, Instagram ve LinkedIn kullanıcılarının Cumhuriyet Halk Partisi hakkında paylaştıkları olumlu ve olumsuz görüşlere göre veri kümesi oluşturulmuştur. Bu veri kümesi üzerinde terim frekansı yöntemi uygulanarak öz nitelik çıkarımı yapılmıştır. Öz nitelik çıkarımı yapıldıktan sonra verilere sınıf etiketleri eklenerek eğitim kümesi oluşturulmuştur. Sınıf etiketine sahip eğitim kümesi denetimli makine öğrenmesi yöntemine uygun hale getirilmiştir. Oluşturulan eğitim kümesi üzerine makine öğrenmesi algoritmaları kullanılarak dört farklı model inşa edilmiştir. K En Yakın Komşu, Karar Ağacı, Sıralı Minimum Optimizasyon ve Bayes Sınıflandırma algoritmaları ile oluşturulan modeller kullanıcıları Cumhuriyet Halk Partisi'ni destekleyen ve desteklemeyen kullanıcılar olarak sınıflandırmıştır. Oluşturulan dört farklı modelin doğruluk oranları hesaplanırken Kappa istatistik testi kullanılmıştır. Yapılan test sonucunda K En Yakın Komşu sınıflandırma algoritmasının kullanıldığı model %97 oranında başarı vermiştir. Sosyal medya kullanıcılarının Cumhuriyet Halk Partisi hakkındaki görüşleri başarılı bir şekilde sınıflandırılmıştır.

Classification of Social Media Users' Opinions on the Cumhuriyet Alliance with Machine Learning Techniques

Keywords:

Cumhuriyet Alliance, Social media analysis, Text Mining, Kappa test, Machine Learning

ABSTRACT

With the increase in the usage of social networks, the number of data generated by social media users is increasing day by day. With these data, emotion analysis, personality detection, etc. academic studies can be done. In this study, the opinions of Facebook, Twitter, Instagram and LinkedIn users, which are popular social networks, about the "People's Alliance" were classified. For this purpose, a data set was created according to the positive and negative opinions shared by Facebook, Twitter, Instagram and LinkedIn users about the People's alliance. The term frequency method was applied on this data set and the attribute inference was made. After the self-attribute extraction, class labels were added to the data and the training set was created. The training set with the class label has been adapted to the supervised machine learning method. Four different models were built on the created training set using machine learning algorithms. Models created with K Nearest Neighbor, Decision Tree, Ordered Minimum Optimization and Bayes Classification algorithms classified users as those who support and do not support the Alliance of the People. Kappa statistical test was used to calculate the accuracy rates of four different models. As a result of the test, the model using the K Nearest Neighbor classification algorithm has achieved 97% success. Social media users' views on the People's Alliance have been successfully classified.

*Sorumlu Yazar

(yasarsafali.01@gmail.com) ORCID ID 0000-0001-9717-9892

e-ISSN: 2717-8579

Geliş Tarihi: 07/04/2020; Kabul Tarihi: 07/10/2020

Bilgisayar Bilimleri ve Teknolojileri Dergisi

1. GİRİŞ

Ülkemizde internet kullanım oranı her geçen yıl giderek artmaktadır. 2019 yılında Türkiye İstatistik Kurumu (TÜİK) tarafından yayınlanan Hanehalkı Bilişim Teknolojileri Kullanım Araştırması'na göre internetin Türkiye'deki kullanım oranı %75'dir. ("URL-1")

İnternet kullanım oranının artması ile sosyal ağların kullanım oranı da giderek artmıştır. Kullanıcılar günün önemli bir kısmını sosyal ağlarda geçirmektedir. Türkiye'de 58.000.000 kişi sosyal ağları aktif olarak kullanmaktadır. ("URL-2") Bu çoğunlukta kullanıcının paylaştığı görüşler sosyal ağlarda büyük miktarda veri oluşturmaktadır. Bu veriler aracılığıyla kullanıcıların duygu analizi, kullanıcılara ait siyasi görüşlerin çıkarılması gibi çeşitli analizler yapılabilmektedir. Ülkemizde sosyal ağlar etkin olarak siyasi görüşlerin, siyasi veya güncel olaylar hakkındaki fikirlerin paylaşıldığı alanlardır. Bu çalışmada, kullanıcıların Cumhuriyet İttifakı hakkındaki görüşleri sınıflandırılmıştır.

Sosyal medya kullanıcılarının Cumhuriyet İttifakı hakkındaki görüşlerinin sınıflandırılması işlemi temel olarak veri toplama, veri ön işleme, öz nitelik çıkarımı, modellerin oluşturulması ve deneysel sonuçlar olmak üzere 5 adımdan oluşmaktadır.

Veri kümesi, sosyal medya kullanıcılarının Cumhuriyet İttifakı hakkındaki görüşlerinden oluşmaktadır. Sosyal ağlarda herkese açık profillerden yararlanılmıştır. Veri kümesi Cumhuriyet İttifakını destekleyen ve desteklemeyen her platformda 500'er adet toplamda 4.000 kullanıcıdan oluşmaktadır. Her kullanıcı için 200 adet olmak üzere toplamda 800.000 paylaşım verisi kullanılmıştır.

Sosyal ağlardaki kullanıcı mesajları içerisinde sembol ve özel karakterler barındığından üzerinde çalışma yapmayı zorlaştırmakta, veriler üzerinde gürültü ve bozulmalara sebep olmaktadır. Verilerin çalışmaya uygun hale getirilebilmesi için üzerinde yer alan gürültülerin ve bozulmaların giderilmesi gerekmektedir. Bozulmaları ve gürültülü verileri temizlemek için veri kümesi üzerinde doğal dil işleme yöntemleri uygulanır (Akgül ve ark., 2016). Kullanılan doğal dil işleme yöntemleri sayesinde gereksiz veriler temizlenir, bozulmuş veriler onarılır. Veriler üzerinde işlem yapmaya hazır hale getirilir.

Veri kümesi üzerinde iyileştirme işlemi yapıldıktan sonra "terim frekansı" (TF) yöntemiyle öz nitelik çıkarımı yapılır. Terim frekansı yöntemi uygulanırken Cumhuriyet İttifakı'nı destekleyen ve desteklemeyen kullanıcıların paylaşımlarından elde edilen terimler, "terimler listesi" olarak tanımlanır. Bu listeler Cumhuriyet ittifakını destekleyen ve desteklemeyen paylaşımlara ait anahtar kelimelerden oluşmaktadır. Bu anahtar kelimeler seçilirken kullanıcılara ait olumlu ve olumsuz verilerden tekrar değerleri en yüksek 200 adet kelime referans alınmıştır. Anahtar kelimeler iki

sınıf içinde eşit sayıda 100'er adet seçilmiştir. Terimler listesi oluşturulduktan sonra veri seti bilgisayarın üzerinde işlem yapabileceği sayısal değerlere dönüştürülmüştür (Seker, 2014).

Öz nitelik çıkarımı yapıldıktan sonra verilere sınıf etiketleri eklenerek eğitim kümesi oluşturulmuştur. Eğitim kümesine sınıf etiketi eklenerek bilgisayara verilerin nasıl sınıflandırılacağı öğretilmek istenmiştir. Veri madenciliği yöntemlerinde sıkça başvurulan bu duruma supervised learning (denetimli öğrenme) denir (Çetin ve Amasyalı, 2015). Oluşturulan eğitim kümesi üzerine makine öğrenmesi algoritmaları kullanılarak dört farklı model inşa edilmiştir. K En Yakın Komşu, Karar Ağacı, Sıralı Minimum Optimizasyon ve Bayes Sınıflandırma algoritmaları ile oluşturulan modeller vasıtasıyla kullanıcılar, Cumhuriyet İttifakı'nı destekleyenler ve desteklemeyenler olarak sınıflandırılmıştır. Oluşturulan dört farklı modelin doğruluk oranları hesaplanırken Kappa istatistik testi kullanılmıştır. Kappa istatistik testi sonuçlarına göre K En Yakın Komşu (KNN) algoritmasının kullanıldığı model %97 oranında başarı vererek en başarılı model olmuştur.

Sosyal ağlarda yer alan kullanıcıların Cumhuriyet İttifakı hakkındaki görüşleri olumlu ve olumsuz sınıflandırılarak çalışma tamamlanmıştır. Sosyal ağ kullanıcılarının paylaşımları referans alınarak yapılan bu çalışmada kullanıcılar, Cumhuriyet İttifakı hakkında herhangi bir paylaşım yapmamış veya belli bir eşik değerinin altında paylaşım yapmışlarsa bu kullanıcılar nötr yani tarafsız olarak değerlendirilmiştir.

Çalışmanın ilerleyen bölümlerinde işleyiş şu şekilde düzenlenmiştir. İkinci bölümde Sosyal ağlar üzerinde metin madenciliği, veri madenciliği teknikleri kullanılarak yapılan çalışmalara, üçüncü bölümde veri toplanmasına, dördüncü bölümde toplanan veriler üzerinde temizleme çalışmalarının yapıldığı ön işleme adımlarına yer verilmiştir. Beşinci bölümde ön işleme adımından geçirilen veri seti üzerinde öz nitelik çıkarılması adımı, altıncı bölümde ise modellerin inşa edilip makine öğrenmesi algoritmaları ile sınıflandırıldığı sınıflandırma ve deneysel sonuçlar adımı yer verilmiştir. Çalışmanın deneysel sonuçlarının tartışıldığı sonuç adımı ise yedinci bölümde değinilmiştir.

2. LİTERATÜR TARAMASI

Twitter erişilmesi kolay ve büyük verileri bünyesinde barındırdığı için sosyal medya madenciliğinde, veri madenciliği uygulamalarında, metin madenciliği çalışmalarında ve doğal dil işleme gibi alanlarda yapılan çalışmalarda tercih sebebi olmuştur. Akgül ve arkadaşları tarafından yapılan çalışmada Twitter verileri üzerinde duygu analizi yapılmıştır. Bu makalede veriler olumlu, olumsuz ve

nötr sınıf etiketlerine göre el yordamıyla elde edilmiştir. Veri seti üzerinde sözlük tabanlı model ve n-gram modeli uygulanarak sınıflandırma işlemi gerçekleştirilmiştir. Modellerin başarı oranları incelendiğinde n-gram modeli daha başarılı bir sonuç vermiştir (Akgül ve ark., 2016).

Nizam ve Akın tarafından yapılan çalışmada ise makine öğrenmesi yöntemlerinden denetimli öğrenme yöntemi kullanılarak Twitter üzerinde dengeli ve dengesiz veri kümelerinin performanslarının karşılaştırılması yapılmıştır. Twitter üzerinde bir gıda firmasına ait ürünlere yapılan yorumlardan veri kümesi oluşturulmuştur. Bu veri kümesi üzerinde yorumlar pozitif, negatif ve nötr olarak sınıflara ayrılmıştır. Bu sınıflara ait veri dağılımının Weka kütüphanesinde yer alan sınıflandırma algoritmalarına etkisi incelenmiştir (Nizam ve Akın, 2014).

Türkmen ve Cemgil tarafından yapılan çalışmada 2013 yılında başlayan Gezi Parkı olaylarına Twitter kullanıcılarının destek verip vermediği incelenmiştir. Bu çalışmada destek verenler gösteri yanlısı, destek vermeyenler gösteri karşıtı ve tarafsız olan mesajlar nötral olarak değerlendirilmiştir. 1351 tweet üzerinden el ile oluşturulan veri kümesi üzerinde özellik çıkarımı yapılırken Ki-Kare İstatistik metodu kullanılmıştır. Özelik çıkarımı yapıldıktan sonra SVM ve RO sınıflayıcısı ile sınıflandırma işlemi gerçekleştirilmiştir (Türkmen ve Cemgil., 2014). 2011 yılında Pennacchiotti ve Popescu tarafından gerçekleştirilen çalışmada Twitter kullanıcılarının mesajlarındaki dilsel içerik bilgileri ve ağ yapıları kullanılarak makine öğrenmesi yaklaşımı ile etnik kökenleri ve siyasi yaklaşımları sınıflandırılmıştır (Pennacchiotti ve Popescu 2011). Nikfarjam ve arkadaşları ise Twitter üzerinde ilaçların gösterdiği yan etkilere yapılan hasta yorumlarını incelemişlerdir. Veri seti oluşturulurken DailyStrength ve Twitter üzerinde yer alan yorumlar kullanılmıştır. SVM yönteminin MetaMap yönteminden daha başarılı olduğu sonucu çıkarılmıştır (Nikfarjam ve ark., 2015).

Kılınç ve arkadaşları tarafından yapılan çalışmada metin madenciliği teknikleri kullanılarak akademik çalışmaların makine öğrenmesi algoritmaları ile sınıflandırılması yapılmıştır. Veri kümesi Research Gate üzerinde yer alan akademik bildirilerden oluşturulmuştur. Materials Science, Engineering ve Social Sciences & Humanities kategorilerine ait iki farklı sınıf etiketine sahip akademik bildirilerden oluşturulmuştur.

Oluşturulan veri kümesi üzerinde sınıflandırma işlemi yapılırken K En Yakın Komşu algoritması kullanılarak %96,67 başarı oranı elde edilmiştir (Kılınç ve ark., 2016).

Çalış ve arkadaşları tarafından yürütülen çalışmada reklam içerikli epostaların tespiti yapılmıştır. Bu tespit işleminde metin madenciliği tekniklerinden terim frekansı ve TF-IDF yöntemleri kullanılmıştır. Veri kümesi 400 reklam içerikli 400 tanede reklam içerikli olmayan Türkçe e-postalardan oluşturulmuştur. Veri kümesi içerisinde yer alan sınıf verileri eşit sayıda seçilmiştir. Daha sonra makine öğrenmesi algoritmalarından Naïve Bayes ve K En Yakın Komşu sınıflandırıcıları kullanılarak epostaların sınıflandırılması gerçekleştirilmiştir. En doğru sonucu %96,5 başarı oranı ile K En yakın Komşu algoritması vermiştir (Çalış ve ark., 2013).

Kaynar ve arkadaşları tarafından yapılan çalışmada IMDB üzerinde yer alan yorumlardan veri kümesi oluşturulmuştur. Veri kümesi 1000 adet olumlu yorumdan ve 1000 adet olumsuz yorumdan oluşturulmuştur. Oluşturulan veri kümesinin %75'i eğitim, %25'i ise test verisi olarak kullanılmıştır. Veri kümesi üzerinde TF-IDF yöntemi kullanılarak veriler makine öğrenmesi algoritmalarının çalışabileceği sayısal değerlere dönüştürülmüştür. Daha sonra makine öğrenmesi algoritmalarından Naïve Bayes, Merkez Tabanlı Sınıflayıcı, Destek Vektör Makineleri, Çok katmanlı Yapay Sinir Ağları kullanılarak veri kümesi üzerinde sınıflandırma işlemi gerçekleştirilmiştir. Kullanılan algoritmaların başarı oranları birbiri ile karşılaştırıldığında yapay sinir ağları ve destek vektör makineleri algoritmaları diğer algoritmalara göre daha başarılı olmuştur (Kaynar ve ark., 2016).

3. VERİ TOPLAMA

Veri kümesi, sosyal medya kullanıcılarının Cumhuriyet İttifakı hakkındaki görüşlerinden oluşmaktadır. Sosyal ağlarda herkese açık profillerden yararlanılmıştır. Veri kümesi Cumhuriyet İttifakını destekleyen ve desteklemeyen her platformda 500'er adet toplamda 4000 kullanıcıdan oluşmaktadır. Her kullanıcı için 200 adet toplamda 800000 paylaşım verisi kullanılmıştır. Veri kümesine ait bilgiler Tablo 1' de verilmiştir. Verilerin her sınıf için eşit sayıda seçilmesine özen gösterilmiştir. Bu sayede modellere ait başarı oranlarının artması hedeflenmiştir.

Tablo 1. Veri kümesinde kullanılan sosyal ağlar ve kullanıcı sayısı

Sosyal Ağ	Toplam Kullanıcı Sayısı	Cumhur İttifakını Destekleyen Kullanıcı Sayısı	Cumhur İttifakını Desteklemeyen Kullanıcı Sayısı	Paylaşım Sayısı
Facebook	1000	500	500	200000
Twitter	1000	500	500	200000
Instagram	1000	500	500	200000
Linkedin	1000	500	500	200000

4. VERİ ÖN İŞLEME

Sosyal ağlardaki kullanıcı mesajları içerisinde sembol ve özel karakterler barındırdığından üzerinde çalışma yapmayı zorlaştırmakta, veriler üzerinde gürültü ve bozulmalara sebep olmaktadır. Verilerin çalışmaya uygun hale getirilebilmesi için üzerinde yer alan gürültülerin ve bozulmaların giderilmesi gerekmektedir.

Bozulmaları ve gürültülü verileri temizlemek için veri kümesi üzerinde doğal dil işleme yöntemleri uygulanır (Akgül ve ark., 2016). Kullanılan doğal dil işleme yöntemleri sayesinde gereksiz veriler temizlenir, bozulmuş veriler onarılır. Veriler üzerinde işlem yapmaya hazır hale getirilir. Sık kullanılan doğal dil işleme yöntemlerinden birisi de cümlelerin anlamına etki etmeyen kelimelerin cümleden çıkarılmasıdır. Türkçe dili için kullanılan en önemli doğal dil işleme kütüphanelerinin başında Zemberek Kütüphanesi gelmektedir. (Müngen ve Kaya, 2018) Bu çalışmada da Türkçe doğal dil işleme kütüphanelerinden birisi olan Zemberek Kütüphanesi kullanılmıştır. Zemberek Kütüphanesi kelimeler üzerinde yapılan yazım yanlışlarını düzeltebilmesinin yanı sıra kelimeleri köklerine ayırabilmektedir. Bu kütüphane sayesinde elde edilen veri seti, çalışmada kullanılmak üzere köklerine ayrılmış ve üzerindeki bozulmalar giderilmiştir. Veri ön işleme adımı tamamlandıktan sonra veriler özellik çıkarımı için hazır hale getirilmiştir.

Veri ön işleme adımı tamamlandıktan sonra veriler öz nitelik çıkarımı için hazır hale getirilmiştir.

5. ÖZ NİTELİK ÇIKARIMI

Veri kümesi üzerinde iyileştirme işlemi yapıldıktan sonra terimler listesi oluşturulmuştur. Cumhur İttifakı'nı destekleyen ve desteklemeyen kullanıcıların paylaşımlarından elde edilen terimler terimler listesi olarak tanımlanır. Bu listeler Cumhur İttifakı'nı destekleyen ve desteklemeyen paylaşımlara ait anahtar kelimelerden oluşur. Bu anahtar kelimeler seçilirken kullanıcılara ait olumlu ve olumsuz verilerden tekrar değerleri en yüksek 200 adet kelime referans alınmıştır. Anahtar kelimeler iki sınıf içinde eşit sayıda 100'er adet seçilmiştir. Cumhur İttifakı'nı destekleyen terim

listesi Tablo 2'de, Cumhur İttifakı'nı desteklemeyen terim listesi Tablo 3'de gösterilmiştir.

Tablo 2. Cumhur İttifakı destekleyen terim listesi

Sıra	Kelime	Frekans
1	Cumhur	1087
2	Erdoğan	965
3	Vatan	942
4	Başkan	911
...
200	Millet	245

Tablo 3. Cumhur İttifakı desteklemeyen terim listesi

Sıra	Kelime	Frekans
1	Muharrem	1112
2	Laik	1075
3	Atatürk	980
4	Halk	850
..
200	İnce	350

Terim frekansı yöntemi kullanılarak terimler listesi oluşturulduktan sonra veriler bilgisayarın üzerinde işlem yapabileceği sayısal verilere dönüştürülmüştür.

Öz nitelik çıkarımı yapılırken veri madenciliği yöntemlerinden olan terim frekansı yöntemi kullanılmıştır. Terim frekansı yöntemi kullanılarak oluşturulan listeler içerisinde yer alan anahtar kelimelerin sosyal ağ kullanıcılarının mesajları içerisindeki tekrar sıklığı hesaplanmıştır. Hesaplama sonucu elde edilen değer sosyal ağ kullanıcılarının mesajlarında yer alan toplam frekans (TF) sayısına oranı hesaplanmıştır. Cumhur İttifakını destekleyenler için Cdestek, desteklemeyenler için Cdesteklemeyen değerleri hesaplanmıştır. Terimler listesinde her bir sınıf için 100 terim yer aldığından her terimin değeri ayrı ayrı hesaplanarak toplanan sonuç Cdestek ve Cdesteklemeyen değerleri ayrı ayrı hesaplanmıştır.

$$C_{destek} = \sum_1^{100} \left(\frac{frekans(k)}{TF} \right) \quad (1)$$

$$C_{desteklemeyen} = \sum_1^{100} \left(\frac{frekans(k)}{TF} \right) \quad (2)$$

Öz nitelik çıkarımı yapıldıktan sonra verilere sınıf etiketleri eklenerek eğitim kümesi oluşturulmuştur.

Eğitim kümesine Cdestekleyen ve Cdesteklemeyen sınıf etiketi eklenerek bilgisayara verilerin nasıl sınıflandırılacağı öğretilmek istenmiştir. Veri madenciliği yöntemlerinde sıkça başvurulan bu duruma Supervised Learning (Denetimli Öğrenme) denir (Çetin ve Amasyalı, 2015).

Cumhur İttifakı'nı destekleyen kullanıcılar için 2000 adet, desteklemeyen kullanıcılar için 2000 adet toplamda 4000 adet kullanıcıya ait

paylaşımlardan öz nitelik çıkarımı yapılmıştır. Çıkarılan öz nitelikler sayesinde eğitim kümesi oluşturulmuştur. Çıkarılan öz niteliklerden %80'i eğitim kümesi için kullanılmıştır. %20'si ise test verisi olarak seçilmiştir. Eğitim kümesindeki verilere sınıf etiketi eklenmiştir. Eğitim kümesine ait veriler Tablo 4 'de gösterilmiştir.

Tablo 4. Eğitim kümesi verileri

Paylaşım	Cdestek	Cdesteklemeyen	Sınıf
1	0.368717	0.047328	Cdestekleyen
2	0.021874	0.589650	Cdesteklemeyen
3	0.928461	0.410733	Cdestekleyen
4	0.319645	0.522839	Cdesteklemeyen
...	..		
640000	0.874429	0.946351	Cdesteklemeyen

6. MODELLERİN OLUŞTURULMASI VE DENEYSEL SONUÇLAR

Oluşturulan eğitim kümesi üzerine makine öğrenmesi algoritmaları kullanılarak dört farklı model inşa edilmiştir. K En Yakın Komşu, Karar Ağacı, Sıralı Minimum Optimizasyon ve Bayes Sınıflandırma algoritmaları ile oluşturulan modeller kullanıcıları Cumhuriyet İttifakı'nı destekleyenler ve desteklemeyenler olarak sınıflandırmıştır. Oluşturulan dört farklı modelin doğruluk oranları hesaplanırken Kappa istatistik testi kullanılmıştır. Bu test istatistik alanında gözlemciler arası uyumda kullanılabildiği gibi farklı testlerin uyumunda da kullanılmaktadır. Bu nedenle bu test sınıflandırma problemlerinde, asıl (gerçek) sonuç ile sınıflandırıcının verdiği kararın arasındaki uyumun göstergesidir (Nizam ve Akın,2014).

Kappa İstatistik Testi sınıflandırma işlemi sonucunda modelin doğruluk oranı analiz edilir. Yapılan analiz sayesinde doğru ve yanlış sınıflandırılmış veriler incelenir. Kappa değerinin sınır aralığı -1 ile 1 aralığıdır. Veriler %100 başarılı sınıflandırıldığında Kappa değeri 1 olur. Kappa değerinin bir diğer yorumlanan durum; 0 ile 1 arasında olduğu durumdur. Bu durumda veri setinin güvenilirliği söz konusudur. Kappa değerinin sıfırdan küçük olduğu durumlarda ise veri setinin güvenilirliği yoktur yorumu yapılabilir (Nizam ve Akın,2014). Kappa değeri hesaplanırken kullanılan matematiksel formül aşağıda verilmiştir. Kappa değeri K hesaplanırken kullanılan Po değeri asıl sonucun ve sınıflandırıcının verdiği kararın aynı sonuç olma oranı, Pc oranı ise aynı sonucu elde etmenin beklenen olasılığını verir.

$$K = \left(\frac{Po - Pc}{1 - Pc} \right) \quad (3)$$

Çalışmada Kappa değerleri hesaplanırken Weka Kütüphanesi'nden yararlanılmıştır. Weka Kütüphanesi, veri madenciliği uygulamalarında

kullanılan, içerisinde birçok sınıflandırma, denetleme, bağıntı fonksiyonları, yapay sinir ağı algoritmaları ve veri ön işleme yöntemleri barındıran bir programdır. Makine öğrenme algoritmaları yardımıyla veri analizi, grafiksel değerlendirme, görsel sonuç üretme yapabilmektedir.

K En Yakın Komşu algoritması, yaygın kullanılan bir sınıflandırma algoritmasıdır. Uzaklık hesabına göre sınıflandırma yapan bir algoritmadır. Öklid uzaklığına göre hesaplama yapan bu algoritma en yakın k komşuyu incelemektedir. Tespit ettiği bütün gözlemlere küme olarak davranmaktadır. Gözlemler sonucu oluşan kümelerin birleşmesiyle kümeler oluşturulur. Oluşturulan bu kümeler arasında mesafe tespiti yapılır. Algoritma kullanılırken dikkat edilmesi gereken nokta komşuluk değerinin (k değeri) çift sayıda değil de tek sayıda seçilmesidir. K değerinin çift sayıda seçildiği durumlarda eşitlik durumu ortaya çıkabilir. Bu durumda sınıflandırma sonucunda herhangi bir sonuç alınmayabilir. Öklid mesafe tespitine göre hesaplama işlemi şu şekilde gerçekleşir.

$$mesafe(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (4)$$

Karar ağaçları yaygın olarak kullanılan bir sınıflandırma algoritmasıdır. Enformasyona dayalı olarak oluşturulan verilerin özelliklerini otomatik olarak işleme özelliğine sahip sınıflandırma algoritmasıdır (Daş ve Türkoğlu, 2014). Kurallar kök ve yapraklar arasında oluşturulur. Veri kümesindeki en ayırt edici özellik kök olarak belirlenir. Veriler yinelemeli (recursive) olarak sınıflandırılarak ideal sonuç aranır. Zayıf dallar bu sınıflandırma tekniği ile budanarak bir sınıflandırma oluşturulmaya çalışır.

Bayes Sınıflandırma (Naive Bayes) makine öğrenmesi çalışmalarında sınıflandırma amacı ile

kullanılan bir algoritmadır. Yüksek performanslı, kolay uygulaması olan yöntemde sınıflandırılacak kümelerin ve örnek verilerin sınıfı belirlenir. Naive Bayes yaklaşımında, n boyutlu bir uzayda olan y vektörü (y_1, \dots, y_2), m tane, sınıf bulunan KC (K_1, \dots, K_n) data sınıfının son olasılığını maksimum eden bir sınıf etiketi K arar (Nizam ve Akın, 2014).

Sıralı Minimum Optimizasyon algoritması, fazladan bir matrise ihtiyaç duymadan, sayısal Quadratic Programming (QP) optimizasyon tekniklerini kullanmadan SVM QP problemlerini normalden yüksek bir performansta çözüme ulaştırır. Global olarak tüm kayıp değerleri yeni değerlerle değiştirir ve nominal nitelikleri ikili

olanlara dönüştürür. Ayrıca tüm nitelikleri önceden tanımlanmış değerlerle normalize eder (Daş ve Türkoğlu, 2014).

Tablo 5’de kullanılan modeller ve bu modellere ait başarı oranları ifade edilmiştir. Sonuçlar incelendiğinde Kappa İstatistik Testi’ne göre en başarılı model K En Yakın Komşu algoritmasının kullanıldığı modeldir. En doğru sınıflandırmayı yapan model seçilirken doğru sınıflandırılmış verilerin oranı ve Kappa İstatistik Testi değeri referans alınmıştır. Bu sebeple çalışmada K En Yakın Komşu algoritması kullanılarak %97 oranında başarı elde edilmiştir.

Tablo 5. Modellere ait başarı oranları

Modeller	Doğru Sınıflandırılan Verilerin oranı	Yanlış Sınıflandırmış Verilerin oranı	Kappa İstatistik Testi değeri
Bayes Sınıflandırma Modeli	%95	%5	0.9523
Sıralı Minimum Optimizasyon Modeli	%89	%11	0.8993
Karar Ağacı Modeli	%92	%8	0.9129
K En Yakın Komşu Modeli	%97	%3	0.9638

Analizde kullanılan sınıflandırma modellerinin başarı oranlarının yüksek olduğu görülmüştür. Veri madenciliği çalışmalarında başarı oranını etkileyen en önemli faktörlerden birisi veri kümesinin dengeli bir şekilde dağılımıdır. Bu nedenle Veri toplamada Cumhuriyet İttifakı’nı destekleyen ve desteklemeyen kullanıcı sayısı eşit alınmıştır.

Sosyal ağlarda yer alan kullanıcıların Cumhuriyet İttifakı hakkındaki görüşleri olumlu ve olumsuz sınıflandırılarak çalışma tamamlanmıştır. Sosyal ağ kullanıcılarının paylaşımları referans alınarak yapılan bu çalışmada kullanıcılar Cumhuriyet İttifakı hakkında herhangi bir paylaşım yapmamış veya belli bir eşik değerinin altında paylaşım yapmışlarsa bu kullanıcılar nötr yani tarafsız olarak değerlendirilmiştir.

7. SONUÇ

Sosyal ağlar bünyesinde büyük miktarda veri barındıran platformlardır. Kullanıcıların paylaşımları toplanarak veri kümesi oluşturulmuştur. Veri kümesi oluşturulurken sosyal ağ kullanıcılarının Cumhuriyet İttifakı hakkındaki görüşlerinden yararlanılmıştır. Oluşturulan veri kümesi ön işleme adımından geçirilerek öz nitelik çıkarımı yapılmıştır. Öz nitelik çıkarımı sonucu elde edilen sayısal değerlere sınıf etiketi eklenerek eğitim kümesi oluşturulmuştur. Eğitim kümesi oluşturulduktan sonra dört farklı sınıflandırma modeli üzerinden başarı oranları elde edilmiş ve sonuçlar karşılaştırılmıştır. Kappa İstatistik Testi ile Yapılan karşılaştırma sonucunda en başarılı sonucu veren K En Yakın Komşu sınıflandırma algoritması sosyal ağ kullanıcılarının Cumhuriyet İttifakı

hakkındaki görüşlerini %97 başarı oranı ile sınıflandırmıştır.

KAYNAKÇA

- Akgül, E. S., Ertano, C., & Diri, B. (2016). Twitter verileri ile duygu analizi. Pamukkale University Journal of Engineering Sciences, 22(2).
- ÇALIŞ, K., GAZDAĞI, O., & YILDIZ, O. (2013). Reklam İçerikli Epostaların Metin Madenciliği Yöntemleri ile Otomatik Tespiti. International Journal Of Informatics Technologies, 6(1), 1-7.
- Çetin, M., & Amasyalı, M. F. (2013, April). Supervised and traditional term weighting methods for sentiment analysis. In 2013 21st Signal Processing and Communications Applications Conference (SIU) (pp. 1-4). IEEE...
- Daş, B., & Türkoğlu, İ. (2014). DNA dizilimlerinin sınıflandırılmasında karar ağacı algoritmalarının karşılaştırılması. Elektrik-Elektronik-Bilgisayar ve Biyomedikal Mühendisliği Sempozyumu (ELECO 2014), 381-383.
- Kaynar, O., Yıldız, M., Görmez, Y., & Albayrak, A. (2016). Makine öğrenmesi yöntemleri ile Duygu Analizi. In International Artificial Intelligence and Data Processing Symposium (IDAP'16) (pp. 17-18).
- KILINÇ, D., BORANDAĞ, E., YÜCALAR, F., TUNALI, V., ŞİMŞEK, M., & ÖZÇİFT, A. (2016). KNN

algoritması ve r dili ile metin madenciliği kullanılarak bilimsel makale tasnifi.

Müngen, A. A., & Kaya, M. (2018). Extracting abstract and keywords from context for academic articles. *Social Network Analysis and Mining*, 8(1), 45.

Nikfarjam, A., Sarker, A., O'connor, K., Ginn, R., & Gonzalez, G. (2015). Pharmacovigilance from social media: mining adverse drug reaction mentions using sequence labeling with word embedding cluster features. *Journal of the American Medical Informatics Association*, 22(3), 671-681.

Nizam, H., & Akın, S. S. (2014). Sosyal medyada makine öğrenmesi ile duygu analizinde dengeli ve dengesiz veri setlerinin performanslarının karşılaştırılması. XIX. Türkiye'de İnternet Konferansı.

Pennacchiotti, M., & Popescu, A. M. (2011, July). A machine learning approach to twitter user classification. In *Fifth international AAAI conference on weblogs and social media*.

Seker, S. E. (2014). Sosyal Ağlarda Akan Veri Madenciliği. *YBS Ansiklopedi*, 1(3), 21-25.

Türkmen, A. C., & Cemgil, A. T. (2014, April). Political interest and tendency prediction from microblog data. In *2014 22nd Signal Processing and Communications Applications Conference (SIU)* (pp. 1327-1330). IEEE.

URL-1:

<http://www.tuik.gov.tr/PreHaberBultenleri.do?id=33679>

[Erişim Tarihi: 25.08.2019]

URL-2:

<https://wearesocial.com/uk/blog/2019/01/digital-in-2019-global-internet-use-accelerates>

[Erişim Tarihi: 31.01.2019]



Derleme Makalesi

Yeni Nesil Web Paradigması: Web 4.0

Betül Ersöz*¹

¹Gazi Üniversitesi Rektörlüğü, Ankara, Türkiye

Anahtar Kelimeler:

Web teknolojileri
Web 4.0
Simbiyotik Web

ÖZ

Teknolojinin gelişmesi ile birlikte insanlık tarihinde çığır açan internet, 21. Yüzyılda insanların vazgeçemediği bir unsur olmuştur. İnternetin sunmuş olduğu özellikler ile birlikte ortaya çıkan yeni teknolojiler de gün geçtikçe artmaktadır. İnsanların en büyük küresel bilgi ortamı olarak kullandığı World Wide Web (WWW) bilgisayarlar üzerinden bir Web tarayıcısı aracılığı ile internet kullanıcılarının verilerini paylaşabildiği, okuyabileceği ve yazabildiği önemli bir sanal ortamdır. Bu internet kanalıyla, Web 1.0, Web 2.0 ve Web 3.0 teknolojileri dönemi yaşanmış ve hala devam etmektedir. Ancak, 4. nesil web teknolojisi olan Web 4.0' ın literatürde belirli bir tanımı bulunmamaktadır. Daha ilk örneklerinin görülmeye başladığı Web 4.0, birkaç boyuttan oluşmaktadır. Önümüzdeki dönemlerde büyük veri (Big data), artırılmış gerçeklik, makineler arası iletişim (M2M), bulut bilişim ve yapay zeka teknolojileri hatta akıllı ajanlar ile bile entegrasyonu olabilecektir. Web 4.0, bir çok modelle etkileşimli yeni bir nesnelerin interneti olarak web teknolojisi devrimi olarak düşünülmektedir. Çalışmanın amacı, devasa ara yüz ve bağlantılar ile zeki ve simbiyotik (insan-makine etkileşimi) bir ağ olarak görülen Web 4.0 kavramını açıklığa kavuşturmak ve çeşitli boyutlarını keşfederek yeni nesil teknolojiler ile bağlantılarını inceleyerek literatüre katkı sağlamaktır.

New Generation Web Paradigm: Web 4.0

Keywords:

Web Technologies
Web 4.0
Symbiotic Web

ABSTRACT

With the development of technology, the Internet, which has revolutionized human history, has become an indispensable element for people in the 21st century. New technologies emerging with the features offered by the internet are increasing day by day. World Wide Web (WWW), which people use as the largest global information medium, is an important virtual environment where internet users can share, read and write their data through a Web browser. Through this internet channel, Web 1.0, Web 2.0 and Web 3.0 technologies period experienced and still continue. However, Web 4.0, which is a 4th generation web technology, does not have a specific definition in the literature. Web 4.0 which is first examples started to appear is consists of several dimensions. In the coming periods, big data, augmented reality, machine-to-machine communication (M2M), cloud computing and artificial intelligence technologies may even be integrated with smart agents. Web 4.0 is considered the Web technology revolution as the Internet of new objects interacting with many models. The aim of the study is to clarify the concept of Web 4.0, which is seen as a huge interface and with connections, a smart and symbiotic (human-machine interaction) network, and to contribute to the literature by exploring its various aspects and examining its connections with new generation technologies.

*Betül Ersöz

*(betul.ersoz@gazi.edu.tr) ORCID ID 0000-0001-6221-1530

1. GİRİŞ

İnternet sürekli güncellenen teknolojiler ile çağa uygun yenilikler geliştirmede, bağlantıların sağlanmasında insanlara büyük kolaylıklar sunmaktadır. Son 20 yılda, internet bağlantılarının sunduğu özelliklerin kullanılmasıyla web teknolojilerinde oldukça ilerleme yaşanmıştır. Web teknolojisinin ortaya çıktığı zamandan itibaren günümüzdeki gelişim sürecine bakıldığında, her dönem yeni özellikler ve fırsatlarla ortaya çıkmıştır. Bu ilerlemelerin oluşum süreci, internet kavramı ile kullanıcı arasındaki ilişkinin değişmesi ve şekillenmesiyle gün geçtikçe değişmektedir.

Avrupa Nükleer Araştırmalar Örgütü'nde (CERN)' de bilgisayar bilimcisi olarak çalışan Tim Berners-Lee ilk bilimciydi. HTML (HyperText Markup Language) etiket dilinin 1989 yılında ortaya çıkması ile The World Wide Web (WWW) kavramı ve web tarayıcılarının geliştirilmesi de bu dönemde ilerletilmiş oldu. Bu teknoloji, internet kullanıcıların Web üzerinden büyük veritabanlarında yer alan içeriklere erişimde, detaylı ve kolay bir şekilde ulaşmasına dayalı olarak geliştirilmiştir. Bu süreçte, "Web" de sosyal bilişimin meydana gelmesi ve sosyal makineler adlı yeni bir çağın ortaya çıkması gerekmektedir. Ayrıca artan teknolojik imkânlarla insanların yapmaya çalıştığı büyük miktardaki verilerin analizlerini gerçekleştiren etkileşimli yeni Web uygulamalarının geliştirilmesi" nin gerektiği savunulmuştur(Tim Berners, 1998). Yeni bir Web paradigmasının ortaya çıkışını belirleyen HTML ile web sayfalarının geliştirilmesinde ilk öğrenme ağı olarak Web 1.0 ortaya çıkmıştır. Daha sonra etkileşim ve iletişim ağı olarak Web 2.0 ve bir entegrasyon ağı olarak Web 3.0 etkin olarak kullanılmaktadır. Çeşitli teknolojiler ile entegre olan ve ultra akıllı Web olarak düşünülen Web 4.0, ise dördüncü nesil Web teknolojisi olarak ortaya çıkmıştır. Web 4.0 teknolojisi hakkında kesin bir yargı veya nasıl olacağına dair literatürde belirli bir tanım bulunmamasına rağmen bu teknolojinin nasıl işleyeceği ile ilgili düşünceler oldukça fazla görülmektedir. Literatürde bu kavram ile ilgili çeşitli tanımlamalar yapılmıştır. "Sosyal makinelerin" sosyal süreçlerimizi görünmez bir şekilde düzenleme misyonuna sahip olacağını belirtmektedir. Bu anlamda Web ve yapay zekâ alanındaki araştırmacılar arasında daha büyük bir işbirliğinin gerektiği düşünülmektedir (Shadbolt ve ark.,2016). 2020-2030'da yılları arasında önemli bir konumda olacak olan Web 4.0 yapay zeka, nanoteknoloji, telekomünikasyon ve kontrollü ara yüzler olarak dört teknolojinin yanı sıra Nesnelerin İnterneti kavramıyla da ilişkilendirilmektedir. Nesnelerin İnterneti (IoT)' nin destekli yaşam, e-sağlık, ileri öğrenme, perakende, otomasyon, finansal ve endüstriyel üretim, lojistik, iş-süreç yönetimi ve akıllı ulaşım imkânları potansiyel faydalarından bazıları olarak sıralanabilmektedir

(Parvathi ve Mariselvi, 2017). Bilişim çağında gelişen teknoloji ve yapılan çalışmalar doğrultusunda insanlar ve makineler arasındaki etkileşimi daha fazla sağlayan bir teknoloji olacağı da mümkün görülmektedir. İnsanların doğrudan makinelerle iletişim kurabileceği, dünyayı şekillendirebilecek ve bilgisayarların cevap verebileceği nitelikli ve etkileşimli bir web teknolojisi olarak düşünülmektedir(Khoo, ,2010),(Perera,2014). Web 4.0' ın yeni bir Web nesli olarak ortaya çıkması ile Web 4.0'ın Web 2.0 ve Web 3.0'ın tüm özelliklerini bir araya getireceğini ve her yerde her ortamda bulunabilecek bir teknoloji olarak görülmektedir(Luis,2017). Bilgisayar kullanıcılarının, bu teknolojinin sahip olduğu güçlü ve zeki ara yüzleri sayesinde web ortamını akıllı bir şekilde yürüten ve etkileşim sağlayan güçlü bir teknolojiye sahip olacağı söylenmektedir. Her geçen gün gelişen iletişim ve bilgi kanallarının artması ile Web dünyasının da bu ihtiyaçları çözümlenebilecek nitelikler geliştirmesi önemli bir unsurdur. Dolayısıyla, Web 4.0'ın teknolojinin eşzamanlı bir ağ olma özelliği ile önemli kitlelere erişim sağlayarak, küresel şeffaflık, yönetim, dağıtım sağlayan çevrimiçi ağlara katılımı, endüstri, politik, sosyal ve diğer önemli topluluklarda işbirliği yapabilen bir ortam olması da beklenmektedir(Khazode ve Sarode, 2016). Bu bağlamda, Web 4.0 insanlar ve akıllı cihazlar ile nesnelerin interneti (IoT) arasındaki veri etkileşimlerinin meydana geldiği simbiyotik (insan-makine etkileşimi) web olarak sınıflandırılmaktadır (Atzori, ve ark.,2010).

Çalışma kapsamında yeni nesil web teknolojisi olan zeki ve yüksek etkileşimli özellikleriyle Web 4.0' a dair literatür incelemesi yapılmış olup, gelecekte Web dünyasına sunacağı yenilikler ve özelliklere yönelik yorumlara yer verilmiştir. Ayrıca, Web 4.0 kavramı hakkında ne olduğu veya ne olabileceği konusunda oldukça farklı yaklaşımları açıklığa kavuşturarak bu teknolojinin boyutlarını analiz etmek amaçlanmaktadır.

2.WEB TEKNOLOJİLERİ

20.yüzyılın sonlarına doğru büyük bir hızla gelişen ve teknolojik bir devrim olan makine algoritmaları ve bilgisayar yazılımları, içinde bulunduğumuz çağın her alanında farklı şekillerde yer almaktadır(Kordon,2009). Günümüzde oldukça önemi artan yapay zekâ teknolojileri, Alan Turing tarafından İlk defa makine zekâsı kavramı diğer bir deyişle 'Düşünebilen makineler' olarak ortaya konuldu. 1950'de Turing Testi adını verilen testle bir makinenin insana eşdeğer veya ona benzeyen akıllı davranış sergileme becerisinin ölçülmesini sağlayarak büyük bir adım atıldı(Turing,2009). Akıllı web platformunun oluşumunda makine dilinin elde edilen veriyi okuma, anlama ve geri bildirim verme özelliğine sahip olan bir web teknolojisine ihtiyaç duyulmaktadır. Çünkü her

geçen gün artan büyük orandaki veriler ve bu verilerin anlamlı hale dönüştürülme ihtiyacı artırmaktadır. Dolayısıyla, Web' in 4. neslinde yapay zeka algoritmaları ile entegre edilen bir Web çağının yaşanması beklenmektedir. Web 'den elde edilen verilerin muhakeme edilerek anlamlı ve işlenmiş verilere dönüştürülecektir. Daha akıllı Web ortamlarının kurulması ile yapay zeka teknolojisinin sunduğu özelliklerden de yararlanılarak Web madenciliği yapılmasında kolaylıklar sunulacaktır. Çeşitli görüşlerin ve fikirlerin öne sürüldüğü akıllı web devrimi 4.0' in temelinde, yapay zekadaki gelişmeler ve buna bağlı Web 3.0 ile uygulamaya konulan özelliklerin geliştirilmesi ve bu özelliklerin kullanımıyla sürdürülen bir akıllı Web özelliği yatmaktadır. Sanal dünya üzerinden nesnelerin birbiri ile etkileşim halinde olduğu bir Web ortamıyla Web 4.0, Büyük veri (Big data), Makineler arası iletişim (M2M), Yapay zeka (AL), Bulut bilişim (Cloud Computing), Artırılmış gerçeklik (VR), Akıllı ajanlar ve Nesnelerin interneti (IoT) teknolojilerinin entegrasyonundan oluşan bir teknoloji pozisyonundadır.

2.1. Web 1.0

İnternetin ilk dönemi olarak bilinen Web 1.0 dönemi 1995 yılında Web' de araştırmalar yapıldığında kolay erişim sağlanması amacıyla oluşturulmuş statik HTML sayfaları içeren bir yapıdır. Bu yapıda kullanıcılar etkileşim sağlamadan bir web ağında pasif bir konumda yer almaktaydı(Berners,2001). İnternet kullanıcıları herhangi bir içeriği ekleme ya da içeriklere yorum yapma yetkisine de sahip değildi. Kullanıcılar web ortamlarını sadece bilgi edinmek için kullanmaktaydı. Web 1.0 temel olarak salt okunur ve sadece bilgiyi öğrenmeye odaklı statik ve tek yönlü bir ağıdır. Ancak teknolojinin günden güne gelişmesi ile birlikte bu Web ağı yerini Web 2.0 teknolojilerine bırakmıştır. Şekil 1'de Web 1.0 teknolojisinin bağlantı yönü gösterilmiştir.

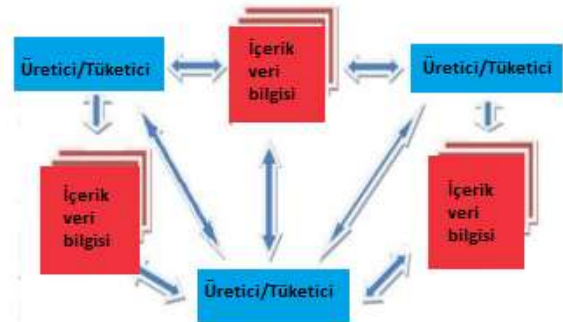


Şekil 1. Web 1.0 tek yönlü bağlantı platformu(thebusinessofsocia,2020)

2.2. Web 2.0

Web 2.0 terimi 2004 yılında O' Reilly başkan yardımcısı olan Dale Dougherty tarafından

MediaLive ile yapılan konferansta ilk defa kullanılmıştır. Web 2.0, internete taşınmanın neden olduğu bilgisayar endüstrisinde daha fazla insanı ağına sunmuş olduğu platformları kullanılarak uygulamalar oluşturabilmek amacıyla Web' in etkileşim haline gelmesi olarak tanımlanmaktadır(O'reilly,2005). Web' in popüler bir ortam haline geldiği Web 2.0 olarak bilinen ikinci nesli, 2000-2009 yılları arasında ortaya çıkmıştır. Web 2.0 ile kullanıcı merkezli ve işbirlikçi bir ortam oluşturulmuştur. Sosyal ağların devrimi, yüksek etkileşim, kullanıcıların içerik üretimine katılımı ve paylaşımların olduğu dinamik ve çift yönlü bağlantı platformu meydana gelmiştir. Bu dönemde, ortaya çıkan Wikiler, Youtube, Facebook, Twitter, Bloglar ve Instagram gibi çok popüler Web sitelerinin ortaya çıkması ile sosyal ağlarda ciddi bir bilgi akışı yaşanmaya başlanmıştır. Web siteleri etkileşimli hale gelerek, kullanıcılar arasında geri bildirimler sağlanmıştır. Web dünyasında meydana gelen bu gelişmelerle birlikte mobil erişim bağlantısı da önemli ölçüde büyümüştür. Akıllı telefonların piyasaya sürülmesiyle kullanıcılar birbiriyle etkileşim kurabilecekleri daha kolay ortamlara taşınmışlardır. Böylece web 2.0 ortamlarının kullanımı büyük oranda artmıştır. İnternet kullanıcıları, çeşitli web araçları kullanılarak fotoğraflar, dosyalar, müzikler, videolar gibi çeşitli paylaşımların yapıldığı bir platformun parçası olmuştur. Web 2.0' in alt yapısında bulunan bağımsız bir platform olan internetin omurgası XML (EXTensible Markup Language) etiket dilinin kullanılması ile Web' deki veritabanlarından alınan izinlerle bilgi akışı sağlanmaktadır. RSS (Rich Site Summary) Web sayfası bildiricisiyle, XML (EXTensible Markup Language) gibi çeşitli web kaynakları ile aktif yapılan bilgi akışları ve farklı sistemlerde yer alan verilerin sorgulanması mümkün kılınmıştır. Web 2.0 kullanıcı merkezli ve yüksek etkileşimli okuma-yazma web ağı olarak da bilinir. Web 2.0 kullanıcıları, Web 1.0'da alışkın oldukları kontrollerin çoğunu bırakıp, diğer kullanıcılarla daha fazla etkileşim yaşadıkları platforma taşınmışlardır. Şekil 2'de Web 2.0 teknolojisinin bağlantı yönleri gösterilmiştir.



Şekil2. Web 2.0 iki yönlü bağlantı platformu(Nath,2015)

2.3. Web 3.0

2001 yılı itibariyle W3C (World Wide Web Consortium) tarafından Web 3.0 ile ilgili devrim niteliğinde çalışmalar başlatıldı. Web 3.0 ya da Semantik Web olarak da bilinen Web' in üçüncü nesli, 2010 yılından günümüze kadar uzanmaktadır. Web 3.0, içeriğin kullanıcı tarafından nasıl aranacağını ve görüntüleneceğini düzenlemeyi amaçlamaktadır. RDF ve OWL dillerinin bu teknoloji ile gelmesiyle Web sayfalarından alınan bilgiler ile kullanıcıların sahip olduğu içeriklere dayalı bir Web ortamı hazırlanmış oldu. Arama motorlarından elde edilen kullanıcı verilerinin değerlendirilmesi ve yapılan çıkarımlar doğrultusunda veriyi yöneten bir platforma dönüştürüldü. XML (Extensible Markup Language)'den esinlenerek geliştirilen semantik ağın temeli OWL (Web Ontology Language) dili veriyi anlamlaştırımda Web 3.0' ın getirdiği bir yenilik olmuştur. Amaç kullanıcıların geçmişine, ilgi alanlarına ve isteklerine göre çevrimiçi aramayı özelleştirmek ve optimize etmektir. Web 3.0 Semantik Web ağı dünya çapındaki verileri tek bir platformda toplayan, ilgili süreçlerin bilgisayarlar tarafından Web üzerinden otomatik olarak yönetilmesini sağlayan bir uygulama olarak tanımlanabilir. Web 3.0, web üzerindeki tüm bilgi ve verilerin açıklamalar ile ilişkilendirilmesini sağlamaktadır. Geliştirilen yeni web teknolojisinde içerik (data) ile anahtar sözcüklerin (meta-data) iç içe geçmesi hedeflenmektedir (Yağcı,2015). Web 3.0 aynı zamanda birçok yazar tarafından akıllı Web olarak da adlandırılmaktadır, çünkü işlevleri geleneksel arama hizmetlerinin ötesine uzanmaktadır. İnternet kullanıcılarının Web üzerinden arama yaptıkları davranışlar, onların tercihlerine göre sıralanıp, özelleştirilebilir. Web 3.0' ın temel düşüncesi, yapı verilerini tanımlamak ve daha etkili olabilmek için bu verilerin birbiri ile bağlantı kurmasını sağlayarak, kullanıcı profillerinin oluşturulmasını desteklemektedir. Yeni bilgi akışlarını sağlamak amacıyla çeşitli veri kümelerinden elde edilen verileri içeriğine entegre ederek analiz etmek, veri yönetimini geliştirerek kullanıcıların memnuniyetini artırmak ve sosyal ağda iş birliğini sağlamaktadır. Web 3.0 ile günümüzde birçok kurumun da sahip olmak istediği büyük veri (Big data) oluşumunda, müşteri memnuniyetini artırmada ve beklentilerini karşılamada verilerin toplanmasında oldukça önemli aşamalar kaydedilmiştir. Kullanıcıların Web teknolojileri ile yaptıkları aramalar doğrultusunda oluşturulan veri ambarlarındaki veriler ile birçok kurum karar verme aşamalarında bu teknolojinin sunduğu imkânlardan faydalanmaktadır. Yapay zeka ve bulanık mantığın bir arada olduğu büyük verilerin analizinin yapılmasında Web 3.0 ile geçilmiştir.

3. WEB 4.0 VE BOYUTLARI

Web 4.0 hakkında henüz net bir tanım olmadığından, kalitesi ve performansının da ne derece etkin olacağı oldukça merak konusudur. Ancak Web 3.0 ile büyük ilerlemelerin kaydedilmeye başlaması, Web 4.0' ın Web içeriğinin okurken akıllıca davranması ve zeki ara yüzler oluşturarak karar vermeyi kolaylaştıracağı yönünde çok fazla görüş desteklenmektedir. WebOS (Linux tabanlı bir mobil işletim sistemi) olarak karakterize edileceği görüşleri bulunmaktadır. Özellikle 2011 yılında ilk kez Almanya'da ortaya çıkan Endüstri 4.0 kavramının Web 4.0 ile ilişkilendirileceği söylenmektedir. Endüstri 4.0 teknolojisinin amaçlarından biri olan akıllı fabrikaların etkileşimde Nesnelere İnterneti(IoT) ile ilişkili olacağı da düşünülmektedir. Bu süreçte daha verimli üretimlerin yapılması ve bu üretimlerin özelleştirilebilir hale gelebilme özelliğine sahip olması için Siber sistemler, Web hizmetleri ve Nesnelere İnterneti oldukça önem arz etmektedir[8]. Bu bağlamda. Web 4.0 ile bağlantıların geliştirilmesi söz konusudur. Örneğin; akıllı bir fabrikada birbiri ile iletişim halinde olan makine işlemlerinin bir veri tabanında toplanarak web ortamında analizini düşünün. Elde edilen işlenmiş veriler doğrultusunda ultra akıllı Web özelliğinin aktif bir rol oynaması ve karar verme aşamalarında bu nesnelere yön verilebileceği bir Web teknolojisine oluşması hiç uzakta olan fikir değildir.

Son yıllarda Müşteri ilişkileri Yönetiminde (CRM), kurumların müşteri ağı profillerinin oluşturulmasında büyük veri ile elde edilen bilgilerin analizi oldukça ivme kazanmıştır. Bu müşteri portföylerinin meydana gelmesinin altında yatan düşünce, kurumların pazarlama stratejilerinin oluşturulabilmesi ve müşterilerin satın alma özelliklerinin belirlenerek anlamlı verilere dönüştürülmesinde Nesnelere İnternetinin (IoT) aktif bir şekilde rol oynayacağı olması fikridir. Kurumlar bu profillerden edindikleri verileri genişletmede kendi kendine öğrenen, içerik oluşturucu akıllı ajanlar ve sensörler desteği ile edindikleri büyük verilerini kolaylıkla analiz edilebileceği bir platform oluşturma çabasıdadır. Örneğin, bir müşterinin mağazaya girdiğinde akıllı ajanların edindiği bilgiler doğrultusunda telefonuna bir anda mesaj geldiğini düşünün. Mesajda, müşterinin daha önce telefonu ile yapmış olduğu ürün araştırmaları ve satın aldığı ürünler ile ilişki hangi reyona yönlendirilmesi gerektiği bilgilerinin yer alması mümkün hale gelmesi söz konusu olabilecektir. Geliştirilen akıllı arayüzler ile makine zekasının geleceği pozisyon bu şekilde düşünülebilir. Geliştirilen algoritmalarla yaratıcı

özelliklere sahip olacağı ön görülen Web 4.0 teknolojisinin de bu süreçte insan ve makine zekasının bir arada olduğu ortak bir alan haline gelmesi beklenmektedir. Birbirine bağlı cihazların oluşturulduğu, büyük veri analitiğinin gerçekleştirildiği Akıllı Web özelliği çeşitli yazılım araçları ile tasarlanan insan-makine etkileşimin sağlandığı bir merkez olarak düşünülmektedir (Wu ve Unhelkar,2010). Örneğin; Web üzerinden gitmeyi planladığımız kongre için kayıt gerçekleştirdiniz. Gideceğiniz şehir için otel rezervasyonu gerçekleştireceğiniz zaman, kayıt yaparken eski verilerinizden yola çıkılarak karşınıza otel rezervasyonunuzu gerçekleştirebileceğiniz bir sayfa çıkabilir. Şekil 3'de Web 4.0 teknolojisinin bilgi alışverişine yönelik şekil yer almaktadır.



Web makineler tarafından öğrenilen veri ve belgeleri içerir.

Şekil 3. Web 4.0 ile makineler tarafından öğrenilen veriler (thebusinessofsocia,2020)

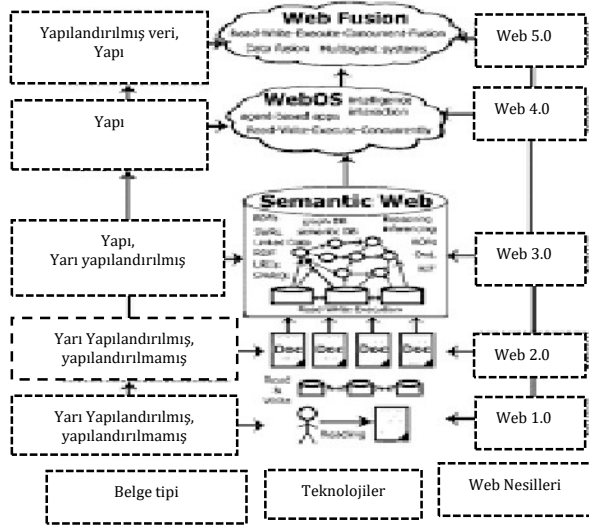
Web 4.0 ile yapay zekaya sahip kişiselleştirilmiş ajanlar için, makine-makine ve makine-insan etkileşim döngüsünde bulut teknolojileri ile sistemlerin yer-mekan gözetmeksizin her yerde yönetilebileceği bir platform olacağı görülmektedir (Nedeva ve Dineva,2012). Web 4.0 kavramı akıllı ajanlar, mobil teknolojiler ve günümüzde yayın bir şekilde kullanılan Bulut bilişim (Cloud Computing) teknolojiyle ilişkilendirilmektedir. Son yıllarda Bulut bilişim (Cloud Computing) sanayi şirketlerinin ve akademi kuruluşlarının da dikkatini çekmektedir (Adviser,2009). Bulut'un temelinde internete bağlı herhangi bir cihazdan her zaman, her yerde veri ve uygulamalara erişilebilirlik olması gerçeği yatmaktadır. Bulutun bu özelliği, bir bilgi altyapısının entegrasyonu ve bakımı ile ilgili hem maliyet hem de oluşan karmaşıklık ortadan kaldırılmaktadır. Bu özellikleri ile Web 4.0'ın Bulut bilişim ile entegrasyonu sonucu başarılı bir iş modellerinin artmasıyla iş dünyasında oldukça verimlilik sağlayabilecektir. Yer ve mekân gözetmeksizin internet kullanıcılarının erişim sağladıkları veri depolama alanı olan Bulut bilişim teknoloji sayesinde sabit disk gerektirmeyen, Yapay zekâya sahip Web üzerinden çalışabilen ortamların oluşum örnekleri görülmektedir. Web 4.0 örnekleri olarak günümüze bakıldığında Cortana (Android), Siri'nin (Iphone), Google Docs, YouOS, DesktopTWO uygulamalarının bu teknolojinin birer örnekleri olduğunu söyleyebiliriz (Kurgun ve Avşar,2018). Günümüzdeki oldukça popüler olan makine

öğrenimi uygulama topluluğu Kaggle, sunduğu ücretsiz GPU desteği ve her hangi bir yazılım programı kurulumunu gerçekleştirilmeden sunduğu editör desteğiyle topluluk tarafından yayınlanan Büyük veri ve kod havuzuna erişilerek, veri analizlerinin yapıldığı özelleştirilebilir bir ortam sunulmaktadır. İşlemlerin gerçekleştirilmesinde kullanılan Google Cloud ile yapılan bu işlemler Web 4.0 teknolojisinin bir ürünü olarak görülmektedir. Bu bağlamda, Bulut bilişim teknolojisinin sunulduğu özellikler kullanılarak fiziki disk kullanımı yapılmadan, kullanıcıların işlemlerini gerçekleştirmelerinde oldukça kolaylık sunulan bir örnek olarak gösterilebilir.

3D tarayıcılarının gelişmesi ile birlikte de Web üzerinden sanal olarak istenilen mağazaların gezilebilmesi söz konusu olabilir. Eğer bir ürün almaya karar verilirse web asistanı ile daha ucuz ürün seçilerek işlemlerin gerçekleştirileceği bir platform deneyimlenebilir. Web 4.0 ile cihazlar Web ortamında bilgileri işlemek ve bir eylem gerçekleştirmek için birbirleriyle iletişim kurabilir. Daha az insan müdahalesi ile Web'den alınan veriler ile istenilen işlemler cihazlara işlendikten sonra gerçekleştirilebilir. Web 4.0, insanlar ve makineler arasındaki sınırı ortadan kaldırmak, makinelere insan gibi düşünebilme özelliği kazandırılması ve insanların oldukça zorlandıkları işleri üstlenmede yardımcı olabilecektir. Bu doğrultuda zaman kayıpları önlenerek daha fazla kaynak tasarrufu olabilmektedir. Akıllı cihazların internete bağlanmasıyla geliştiren uygulamalar ile insanlar yaşam tarzlarına uygun kişisel makinelere de sahip olabilirler. Ayrıca daha iyi karar verme de Web 4.0, bilgisayarların yalnızca bilgi üretmesinden ziyade ortaya çıkan sorunlara da çözüm önerilebilecektir. Böylelikle kurumların Büyük verilerle üstesinden gelemedikleri zamanlarda etkili bir şekilde karar verebilmesi mümkün hale gelecektir (tech.netonboard,2020). Yapay zeka uygulamaları doğrultusunda yöneticilerin vermekte zorlandıkları kararların kolaylıkla verilebilmesi söz konusudur. Bu süreçte, Yapay zeka ile Web 4.0'ın entegrasyonundan oluşan uygulamaların yakın zamanda görülmesi kaçınılmazdır. Günümüzde kurumlar daha az maliyet ve daha çok iş gücü beklentisi içerisindedir. Dolayısıyla, kurumlarda yer alan alt yönetim kademeleri için geliştirilmiş uygulamalar sayesinde daha az insan gücü, daha fazla makine düşüncesi benimsenebilir.

Web 4.0'ın çeşitli uygulamalar ile getirebileceği bazı zorluklarına bakıldığında, kullanıcıların aktif olarak kullanacağı bu teknoloji ile internet yoğunluğu ve bağlantı fazlalığından kaynaklanan sistemsel problemlerin olması söz konusudur. Web'in en kritik gelişmelerinden biri muhtemelen çevrimiçi işlemlerinden kaynaklı yoğunluğun günlük yaşamı etkilemesidir. Çünkü fiziksel ve sanal olarak biri birine bağlantılı nesnelerin bağlantı

sorunu yaşandığında oluşturabileceği zorluklar mümkündür. Örneğin, kaybettiğiniz cep telefonunuzu bulmak için Web üzerinden arama yaparak işlem gerçekleştirdiğinizi düşünün. Sistemsel bir problem sonucu bağlantı sorunu ile karşılaşınca kontrolün sizde olmamasının olumsuzluğunu yaşayabilirsiniz. Gelecekte Web 4.0 ile gelen olumsuzlardan biri ise, kablosuz bağlantıların ve fiili bağlantı yapan telekomünikasyon hatlarının sadece belirli bir şirkete ait olunacağı düşünüldüğünde tüm cihazlar tarafından aynı standartlarda olması gereken bir dile gereksinim duyulacaktır(Choudhury,2014),(hubpages.com,2020). Bu durumun küresel dünyada bir rekabet halinde olan şirketleri göz önüne alırsak belirli bir standartta sahip bağlantıların kurulması oldukça zor görünmektedir. Ayrıca bilgi güvenliğinin tehdit altında olunduğu bu çağda, internet kullanıcıları tarafından oluşturulan veri tabanlarının gizliliği ve güvenilirliği güvence altına alınmaz ise kişisel verilerin üçüncü kişiler tarafından erişiminin ve bilgilerin güvenli saklanması da zor olabileceği olasılıklar arasındadır. Günümüzde Google çeviri gibi örnekleri olan doğal dil anlama (NLU) tekniğinin insan-makine arasında yeni iletişim ara yüz modeli olarak Web 4.0 ile gelen yeniliklerle ilerlemesi ve örneklerinin artması beklenmektedir. Ancak daha çok sosyal ağlar, Nesnelerin İnterneti, Büyük Veri, Yapay zeka ve M2M gibi teknolojilerin Web 4.0' ın benimsenmesinde ve uygulanmasında kilit bir rol oynadığı öne çıkmaktadır (Nath,2015).



Şekil 4. Web nesilleri ile ilgili olarak teknolojiler ve belge türleri(Algoşaibi ve ark., 2015)

Şekil 4'de Web teknolojileri ve belge tipleri gösterilmiştir. Web 1.0 ile başlayan uygulamaların ve örneklerinin geliştirilmesiyle Web 4.0' a kadar olan yolculuk, her adımda güncellemeleri ve web teknolojisinin zekileştirilmesi ile internet kullanıcılarının işlemlerini kolaylaştıran ultra akıllı

web olma yolundadır. Bu gelişmeler ve teknolojinin sunduğu bağlantılar ile endüstri alanında, siyasi, sosyal ve farklı ağlarda dünya çapında anlaşılabilirliği, idareyi, yaygınlaştırmayı, ilgiyi ve kilit ağlar arasında iş birliğini bir arada tutan bir teknoloji olacağı gözükmemektedir (Önday,2019).

4. SONUÇ

Web ortamı sınır tanımayan bir biçimde sürekli gelişmektedir. Son on yılda, literatüre bakıldığında Web 4.0 hakkında çeşitli tanımlar birçok yazar tarafından yapılmıştır. Ancak, Web 1.0, Web 2.0 ve Web 3.0' ın daha önce kabul edilmiş önceki tanımları ile ilgili olarak, Web 4.0' ı ile ilgili kesin bir tanım bulunmamaktadır. Son yıllarda, kablosuz ağların artması, akıllı telefonların ve tabletlerin gün içerisinde aktif olarak kullanıldığı bir çağda internet daha da yaygın hale gelmiştir. Teknolojilerin Web 4.0 ile olan bağlantısına bakıldığında, Büyük veri (Big data) analizlerinde Web teknolojilerinin bir sonraki aşaması olan Web 4.0 ile önemli aşamaları kaydedileceği ve kilit bir rol oynayacakları görülmektedir. Yapay zekâ teknolojisinin sunduğu uygulamalar ile kullanıcıların daha etkileşimli ve sürükleyici deneyimler yaşayabilmeleri için yeni teknolojilere ihtiyaç duyduğu görülmektedir. Web 4.0 ile akıllı ajanlar ve sensörlerin etkileşim halinde olduğu hizmetler, sanal gerçeklik hizmetleri ve doğal dil hizmetleri gibi gerçek ve sanal dünyadaki tüm cihazları gerçek zamanlı olarak bağlamak mümkün görünmektedir. Web 4.0 ile birbirine bağlı web sayfaları, web uygulamaları, videolar ve fotoğrafların etkileşimli olabileceği de başka bir boyuttur. Bu süreçte, bilginin iyi yapılandırılması büyük önem arz etmektedir. Bu bağlamda, internete erişim sadece insanlar tarafından değil, aynı zamanda Nesnelerin İnterneti kavramından kaynaklanan diğer fiziksel nesnelere, cihazlardan ve araçlardan da yapılabileceği düşünülmektedir. Arzulanan yeni teknoloji unsurları ile Web 4.0 kavramının yeni bir paradigma olarak ortaya çıkması görülmektedir. Web 4.0 ile yapay zeka işletim sistemlerinin ve bulut bilişim teknolojisinin devreye gireceği sanal bilgisayar ortamlarının sunulduğu bir etkileşim ortamının çok yakında yayınlanacağı düşünülmektedir. 2020 yılı itibari ile aktif bir rol alacağı görünen Web 4.0, günümüz teknolojilerine entegrasyonun yapılmaya başlanması da bu teknolojinin daha da ilerleyeceğinin kanıtıdır.

İnternet kullanıcılarının sürekli arttığı bu çağda web siteleri, bloglar, çevrimiçi yayınlar, sosyal ağlar, veritabanları ve daha çok etkileşim ağı kullanılarak yeni nesil teknolojiler ile bağlantılı olması mümkün olan Web 4.0, ultra akıllı ve zeki bir web teknolojisi olma yönünde ilerlemektedir. Bu sebeple, Web 4.0 kavramını çeşitli teknolojilerle ilişkilendirmek mümkün görülmektedir. Web 4.0 ile ilgili olumlu-olumsuz şüphesiz bir takım gelişmeler yaşanması

muhtemeldir. Ancak teknoloji odaklı küresel ağda, web teknolojilerinin toplumun merkezinde yer alan ağlar olduğu ve teknolojide oldukça ilerleme kaydettiği görülüyorken web nesillerinin ve aşamalarının da bu doğrultuda gelişmesi oldukça normaldir. Gelecek için bir bakış açısı geliştirildiğinde ise Web 4.0' ın Web 3.0 'ın getirdiği yeniliklere yeni güncellemeler ile oluşturulacağı bir gerçektir. Dolayısıyla, sonra gelmesi beklenen Web 5.0 teknolojisinin de Web 4.0' ın getirdiği yeni nesil teknolojilerinin güncellemesi ile geliştirilen ultra akıllı robotlar, makineler ve diğer teknolojilere ek olarak insanlara has özellik olan duygusallık ile duygusal düşünmeyi başaran makinelerin gelmesi hiç de uzak görünmemektedir.

KAYNAKÇA

- A. A. Algosai, S. Albahli, and A. Melton, "World Wide Web : A Survey of its Development and Possible Future Trends," The 16th International Conference on Internet Computing and Big Data/COMP'15, 2015.
- Adviser-Creeger, Mache. Cloud computing: An overview. Queue, 7.5: 2.2009.
- Almeida, Fernando Luis. "Concept and dimensions of web 4.0." *International Journal of Computers & Technology*.2017
- Atzori, Luigi, Antonio Iera, and Giacomo Morabito. "The internet of things: A survey." *Computer networks*.2787-2805. 2010
- Berners-Lee, Tim, *The World Wide Web: A very short personal history*, Tim Berners-Lee, 1998.
- Berners-Lee, Tim; Fischetti, Mark. *Weaving the Web: The original design and ultimate destiny of the World Wide Web by its inventor*. DIANE Publishing Company, 2001.
- Choudhury, N." World Wide Web and Its Journey from Web 1.0 to Web 4.0", *International Journal of Computer Science and Information Technologies*, 5(6), 8096-8100,2014.
- İnternet:thebusinessofsocia,<https://thebusinessofsocial.wordpress.com/tag/web-5-/>,20.05.2020.
- İnternet:By Net Onboard - October 22, 2019, <https://tech.netonboard.com/web-4-0-18.05.2020>.
- İnternet: Janik D., hubpages- Technology, <http://hubpages.com/hub/The-Internet-of-Things-Web-4-0#>, 01.05.2020.
- Khazode, K., and Sarode, R. "Evolution of the World Wide Web: From Web 1.0 to 6.0.", *International Journal of Digital Library Services*, 6(2), 1-11. 2016.
- Khoo, B.," Rfid- from tracking to the internet of things: A review of developments. "In *Proceedings of the 2010 IEEE/ACM International Conference on Green Computing and Communications*, Washington, USA, 533-538. 2010.
- Kordon, Arthur. *Applying computational intelligence: how to create value*. Springer Science & Business Media, 2009.
- Nath,Keshab; ISWARY, Raja." What comes after Web 3.0? Web 4.0 and the Future. In", *Proceedings of the International Conference and Communication System (I3CS'15)*, Shillong, India. p. 337-341. 2015.
- Nath, K., and Iswary, R. ,"What comes after Web 3.0? Web 4.0 and the Future", In *Proceedings of the International Conference and Communication System (I3CS'15)*, Shillong, India, 337-341,2015.
- Nedeva, V., and Dineva, S., "New learning innovations with Web 4.0.", In *Proceedings of the 7th International Conference on Virtual Learning (ICVL)*, Bucharest, Romania, 316-321. 2012.
- Kurgun, Hülya; Kurgun, Osman Avşar; AKTAŞ, Erdem." What does Web 4.0 Promise for Tourism Ecosystem? A Qualitative Research on Tourism Ecosystem Stakeholders' Awareness. "*Journal of Tourism and Hospitality Management*, 2018, 6.1: 55-65.
- O'reilly, Tim. *What is web 2.0*. 2005.
- Önday, Özgür."Web 6.0: Journey From Web 1.0 To Web 6.0", *Journal of Media & Management*,SRC/JMM-102,2019.
- Parvathi, M., and Mariselvi, R. "A bird's eye on the evolution - Web 1.0 to Web 5.0: Lib 1.0 to Lib 5.0." *International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)*, 4(4), 167-176.2017.
- Perera, Charith, et al. "Sensing as a service model for smart cities supported by internet of things.",*Transactions on emerging telecommunications Technologies*.25.181-93. 2014
- Shadbolt, N., Kleek, M., and Binns, R."The Rise of Social Machines",*IEEE Consumer Electronics Magazine*,106-111, 2016.

Turing, Alan M. "Computing machinery and intelligence." ,Parsing the Turing Test. Springer, Dordrecht, 23-65.2009.

Wu, M.C. and Unhelkar, B. "Mobile service oriented architecture (MSOA) for business in the Web 2.0 era", In Electronic services: Concepts, methodologies, tools and applications, New York: Information Science Reference.2010

Yağcı, Yalçın. Web teknolojisinde yeni bilgi fırtınası: Web 3.0. Ünak 2009 (2011).



Araştırma Makalesi

Son İşlem Algoritmalarının İncelenmesi

Didem Yosunlu^{*1}, Erdinç Avaroğlu¹

¹Mersin Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği, Mersin, Türkiye

Anahtar Kelimeler:
Rasgele Sayı Üreteçleri
Son İşlem
Son İşlem Algoritmaları

ÖZ

Rasgele sayılar, belirli bir aralık için tanımlanmış, oluşma olasılıkları birbirine eşit ve bu sayılar arasında belirli bir ilişki olmayan sayılar olarak tanımlanabilir. Rasgele sayı üreteçleri de rasgele sayıları üretmek için kullanılan araçlardır. Bu rasgele sayı üreteçlerinden elde edilen bit dizilerinin bazıları zayıf istatistiksel özellikler göstermektedir. Bu nedenle üretilen dizilere bu zayıflığın giderilmesi amacıyla çeşitli son işlem algoritmaları uygulanmaktadır. Bu makalede şimdiye dek önerilen belli başlı son işlem algoritmaları incelenerek, karşılaştırılmaları yapılmıştır.

Examination of Post Processing Algorithms

Keywords:
Random Number
Generator
Post Processing
Post Processing
Algorithms

ABSTRACT

Random numbers can be defined as numbers that are defined for a certain range, their probability of occurrence is equal to each other, and there is no specific relationship between these numbers. Random number generators are tools used to generate random numbers. Some of the bit sequences obtained from these random number generators show poor statistical properties. For this reason, various post-processing algorithms are applied to the produced sequences in order to eliminate this weakness. In this article, major post-processing algorithms proposed so far have been examined and compared.

*Sorumlu Yazar

*(didemoz80@gmail.com) ORCID ID 0000 - 0001 - 6917 - 4912
(eavaroglu@mersin.edu.tr) ORCID ID 0000-0003-1976-2526

e-ISSN: 2717-8579

Geliş Tarihi: 04/11/2020; Kabul Tarihi: 06/12/2020

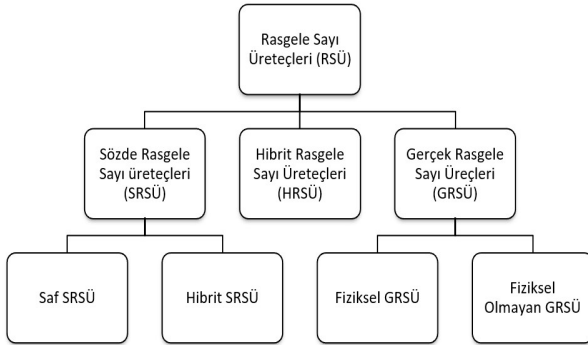
Bilgisayar Bilimleri ve Teknolojileri Dergisi

1. GİRİŞ

Rasgelelik en basit şekliyle deterministik olmayan yani belirli olmayandır. Rasgele sayılar ise, verilen iki koşulun yerine getirildiği bir dizide bulunan sayılardır; değerler belirli bir aralıkta eşit dağılmış veya ayarlanmış olmalı ve geçmiş değerleri veya şimdiki değerleri temel alarak gelecekteki değerleri tahmin etmek mümkün olmamalı.

Rasgele sayı üretimi, elektronik iletişimin gizliliğini korumak için temel bir süreçtir (Yadav, 2013). Günümüzde, rasgele sayı üreteçleri telekomünikasyon, bilgi güvenliği, oyun endüstrisi ve simülasyon başta olmak üzere birçok alan için çok önemli hale gelmektedir. Özellikle kriptografi, anahtar üretimi, kimlik doğrulama gibi telekomünikasyon uygulamaları için oldukça kritik bir öneme sahiptir (Anikin ve Alnajjar, 2016). Bu tarz uygulamalarda sistemin güvenliği üretilen bu sayıların gerçek rasgeleliğine dayanmaktadır. Bundan dolayı kullanılan rasgele sayılar tahmin edilememe, tekrar üretilmeme ve iyi istatistiksel özellikler gibi sıkı gereksinimleri sağlamaları gerekmektedir (Avaroğlu ve Türk, 2013). Bu sebeple de rasgele sayı üreteçlerine ihtiyaç doğmuştur.

Rasgele sayı üreteçleri (RSÜ), sözde rasgele sayı üreteçleri (SRSÜ), gerçek rasgele sayı üreteçleri (GRSÜ) ve hibrit rasgele sayı üreteçleri olmak üzere üç ana sınıfa ayrılır. SRSÜ'leri kendi içinde Saf ve Hibrit olarak ikiye ayrılırken GRSÜ'leri Fiziksel ve Fiziksel olmayan sayı üreteçleri olarak iki alt sınıfa ayrılmıştır. Şekil 1'de sınıflar gösterilmiştir.



Şekil 1. Rasgele Sayı Üreteçlerinin Sınıflandırılması

Sözde rasgele sayı üreteçleri tohum adı verilen bir başlangıç değerine bağlıdır. Belirlenen tohum değeri deterministik algoritmalarla tabi tutularak uzun rasgele bit dizileri üretilir. Ucuz ve hızlı olmalarına karşın SRSÜ'ler periyodik olabilmektedir ve periyodun açığa çıkması ile üretilen rasgele sayıların tamamen tahmin edilebilir olmasını mümkün kılacaktır. Ayrıca tohum değerinin korunamaması da bu duruma neden olacaktır. Bu sebepten dolayı kriptografik uygulamalarda kullanımı uygun değildir.

Gerçek rasgele sayı üreteçleri entropi kaynağı olarak deterministik olmayan fiziksel olaylar kullanır. Faz gürültüsü, termal gürültü, titreşim,

rastgele telgraf gürültüsü, fotoelektrik etki ve doğada istatistiksel olarak rasgele olan diğer fenomenler gibi fiziksel süreçlere dayanan yüksek hızlı ve gerçek rastgele diziler oluştururlar (Tehranipoor ve ark., 2016). Entropi kaynağı tarafından elde edilen sinyaller sayısal olarak dönüştürülerek örnekleme yapılır. Örnekleme sürecinden sonra üretilen sayılar bazen zayıf istatistiksel özellikler gösterebilmektedir. Bunun giderilmesi amacı ile üretilen bu bit dizileri son işleme tabi tutulmaktadır.

Son işlem algoritmaları zayıflıkları gidermeyi amaçlamakla beraber aynı zamanda sistemi saldırganlara ve çevresel değişikliklere karşı dirençli hale getirmeyi de amaçlamaktadır. Son işlem algoritmasına bağlı olarak üreticinin güvenliği artmaktadır. Ancak, son işlem uygulamaları zayıflıkları giderirken uygulamaya bağlı olarak bit oranında azalmaya neden olabilmektedir. Buna rağmen kriptografik uygulamalar için kritik olan tahmin edilememe, tekrar üretilmeme ve iyi istatistiksel özellikler sağlama sebebiyle kriptolojide birçok uygulamada kullanılmaktadır.

Hibrit rasgele sayı üreteçleri GRSÜ'den elde edilen rasgele sayının SRSÜ'de tohum değeri olarak kullanılmasıyla her iki sistemin birlikte çalıştığı rasgele sayı üreteçidir (Avaroğlu ve Türk, 2013).

Bu makalede amacımız şimdiye kadar önerilen ve sık kullanılan son işlem algoritmalarından bazılarının incelenerek karşılaştırılmalarının yapılmasıdır.

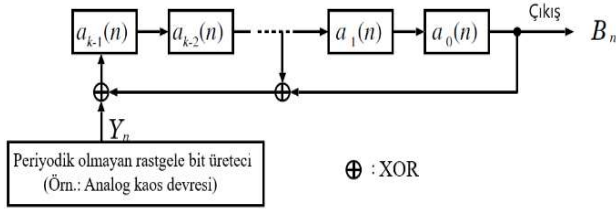
Bölüm 2'de literatür taraması yapılarak son yıllarda önerilen son işlem algoritmalarından bahsedilmektedir. Bölüm 3'te seçilen son işlem algoritmaları açıklanırken Bölüm 4'te ise bu algoritmaların karşılaştırmaları yer almaktadır.

2. LİTERATÜR TARAMASI

Günümüze kadar gerçek rasgele sayı üreteçlerindeki zayıflıkları gidermek ve istatistiksel özellikleri güçlendirmek amacı ile birçok son işlem algoritması önerilmiştir. Bu algoritmalarından bazıları; Doğrusal Geri Beslemeli Kayan Yazmaç (DBKY), Lojistik Harita, SHA-256, Mixing Bits in Steps and XORing, High-Throughput Von Neumann, Keccak Algoritması ve SBOX.

Doğrusal Geri Beslemeli Kayan Yazmaç birçok uygulamada son işlem algoritması olarak kullanılmıştır (Tsuneda ve ark., 2008; Tsuneda ve Morikawa, 2013). Bunun sebepleri ise uygulamasının kolay ama kullanışlı olması, iyi istatistiksel özelliklere sahip diziler üretmesi ve aynı zamanda geniş tekrarlama periyodudur.

Örneğin (Tsuneda ve ark., 2008)'te üretilen rasgele sayıların periyodikliğini gidermek ve iyi istatistiksel özellikler göstermesi sağlamak için aşağıdaki şekilde bir yapı kullanılmıştır;



Şekil 2. DBKY kullanılan RSÜ'nün genel yapısı

Şekilde görüleceği üzere sistem bir rasgele sayı üretici, k hafıza hücresi ve XOR dan oluşmaktadır. Her bir zaman biriminde her hafıza hücresi bir sağa kaydırılır, ayrıca $a_{k-1}(n)$ Denklem (1)'deki gibi güncellenir.

Denklem (1): DBKY hücre kaydırma

$$a_{k-1}(n+1) = a_0(n) \oplus a_1(n) \oplus Y_n$$

Yapılan çalışma göstermektedir ki önerilen son işlem algoritması k büyük olduğunda çok etkili olmaktadır.

Lojistik harita; donanım üzerinde uygulaması kolay olan, bir başlatma koşulu ve bir kontrol parametresi gerektiren birinci dereceden bir denklemdir (Avaroğlu ve ark., 2015). Lojistik haritanın yapısı en basit hali ile Denklem (2)'deki gibidir.

Denklem (2): Lojistik Harita

$$x(n+1) = r x(n)(1-x(n))$$

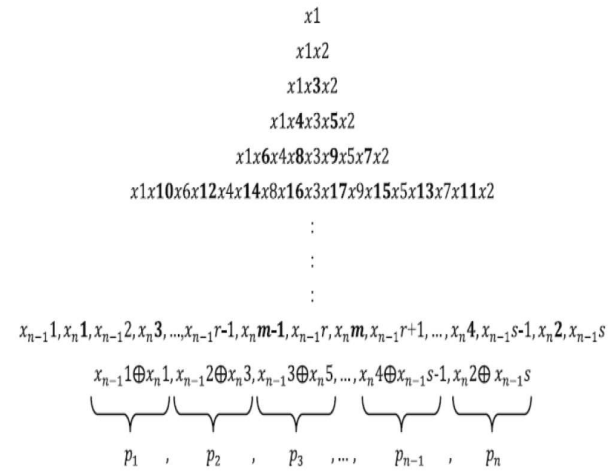
Burada r sistem parametresi n ise tekrarlamaya sayıdır. Aynı zamanda $0 \leq x(n) \leq 1$ koşulunun sağlanması gerekmektedir. $x(n)$ Değeri $x(0)$ başlangıç değerinin seçilmesi ile hesaplanır. Sistem r parametresinin değerine göre farklı davranışlar sergileyebilmektedir. Aynı zamanda lojistik haritada başlangıç koşulunun değiştirilmesi ile farklı sayı dizileri üretilebilmektedir. Bu sebeple lojistik harita kullanan sistemler saldırılara karşı daha güvenlidir. Yüksek güç tüketimi sebebi ile dezavantajlı olmasına rağmen birçok son işlem algoritması veri çıkış oranını azaltırken lojistik harita veri çıkış oranını değiştirmemektedir.

Kaynak (Loza ve Matuszewski 2014)'te ise halka osilatörler ile tasarlanmış bir GRSÜ ile birlikte son işlem algoritması olarak SHA-2 özet fonksiyonları ailesinden olan SHA-256 kullanımı önerilmiştir. Kurulan tasarıma göre 512 rasgele biti giriş olarak alan SHA-256 algoritmasına yollanmak üzere GRSÜ'den gelen rasgele bitler 8 bitlik bloklara ayrılarak, her blok 64 byte genişliğinde olan FIFO buffer içerisinde saklanır. Bu algoritmanın sonucunda ise 8 tane 32 bitlik kelime (256 bit toplam) üretilir.

Önerilen bir diğer son işlem algoritmasında rasgele sayı üretmek için mikrofon aracılığı ile standart bir bilgisayarın, yeni nesil bir dizüstü bilgisayarın, tabletlerin ya da akıllı telefonların ses

kartı kullanılarak bir rasgele gürültü sinyali sağlanmıştır (Nikolic ve Veinovic, 2016). Son işlem algoritması olarak da otokorelasyonu azaltmanın ve çıktı bit dizisinin toplam entropisini artırmanın yenilikçi bir yolu olduğu söylenen Mixing Bits in Steps and XORing of Adjacent Bits yöntemi kullanılmıştır. Kurulan sistemin 3 aşaması bulunmaktadır. İlk aşama, gürültü kaynağından elde edilen analog sinyalin sayısallaştırılmasıdır. İkinci aşamada bu sinyaller zayıflığın azaltılması amacı ile son işlem algoritmasından geçirilerek dahili rasgele sayılar üretilir. Üçüncü aşamada ise harici rasgele sayılar elde edilir.

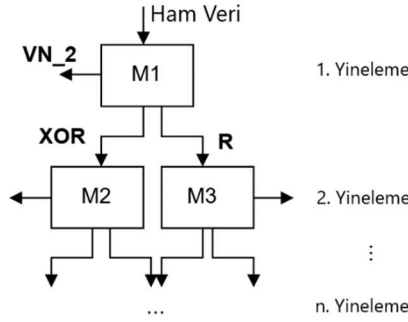
Karıştırma algoritması gelen ilk iki biti alır. İkinci aşamada üçüncü bit alınır ve ilk iki bitin arasında yerleştirilir. Üçüncü aşamada ise dördüncü bit 1. ve 3. bitin arasına yerleştirilir, beşinci bit ise 3. ve 2. bit arasına yerleştirilir. Bu aşamada görünüm $x_1; x_4; x_3; x_5; x_2$ şeklindedir. Dördüncü adımda, altıncı bit 1. ve 4., yedinci 2. ve 5., sekizinci 4. ve 3. ve dokuzuncusu 5. ve 3. arasına yerleştirilir. Bu, dördüncü adımı tamamlar ve ortaya çıkan düzen $x_1; x_6; x_4; x_8; x_3; x_9; x_5; x_7; x_2$ şeklinde olur. Tüm yerleştirmeler yapıldıktan sonra komşu olan bitler XOR işlemine tabi tutularak rasgele bit dizisi son halini almış olur. Bu uygulama yavaş olmasına rağmen istatistiki açıdan oldukça iyi sonuçlar vermiştir. Makalede aynı zamanda yavaşlığı azaltmak için de farklı bir yöntem önerilmektedir.



Şekil 3. Mixing Bits in Steps and XORing of Adjacent Bits yöntemi gösterimi

İlk olarak J.von Neumann tarafından 1951'de önerilen Von Neumann algoritması (Von Neumann, 1951) ise hala birçok rasgele sayı üreticinde son işlem algoritması olarak kullanılmaktadır. Veri çıkış oranını düşürdüğünden dolayı geliştirilmeye devam edilmektedir. Örneğin kaynak (Peres, 1992)'de Iterating Von Neumann (IVN) yöntemi sunulmuştur. Bu yöntemde IVN atılan bitleri toplayıp onları bir sonraki aşamaya yollayarak tekrardan kullanılmaktadır. Aşağıdaki şekilde de görüleceği gibi her IVN modülü üç blok içerir: VN_2 (giriş çifti "01" veya "10" ise, ardından "0" veya "1" çıkışı verir, diğerleri çıkış olmaz), XOR ("00" veya "11" çifti

oluşursa çıkış '0', '01' veya '10' çiftleri daha sonra '1' çıktısı oluşur), R ('00' ise '0', '11' ise '1' çıktı, diğerleri çıktı olmaz). XOR ve R işleminden sonra çıkış verisi tekrardan işlenmemiş veri olarak bir sonraki yinelemeye verilir.



Şekil 4. Iterating Von Neumann (IVN)

1972'de ise Elias N bits Von Neumann (Elias, 1972) algoritmasını önermiştir. Orijinal Von Neumann (VN_2) algoritmasında 2 bit ile işlem yapılırken VN_N algoritmasında N bit ile işlem yapılmaktadır. Burada N'nin değerini artırarak, VN_N, yalnızca 1 kez işleme ile Shannon entropisine yakın yüksek çıktı oranı elde edebilir. Bununla birlikte, teorik değer ile gerçekçi değer arasında bir çıktı oranı farkı vardır. Teorik çıktı oranı N = 4, 8, 16 olduğunda sırasıyla %49,2, %68,2, %81,0'dır. Gerçekçi çıktı oranı sırasıyla %40,6, %55,2 ve %73,1'dir. Bu arada, N = 17'nin ötesinde verimlilik doymuş hale gelir. Bu yüzden uygun N değerlerinin seçilmesi gerekiyor.

2018'de hazırlanan makalede (Zhang ve ark., 2018) ise VN_N algoritması geliştirilerek teorik değer ile gerçekçi değer birbirine yaklaştırılması amaçlanmıştır. Burada herhangi bir N (N ≥ 2) değeri için bir eşleşme algoritması hazırlanmıştır. Böylece örneğin VN_2 algoritması 25.0% veri çıkış oranına sahipken VN_4 40.6% oranında çıkış vermiştir. Makalenin devamında VN_N+waiting adı verilen bir bekleme stratejisi de önerilerek VN_4+waiting için çıkış oranı 46.9% 'a kadar çıkarılmıştır. Böylece teorik çıktı oranı olan 49.2% a oldukça yaklaşmıştır.

Kriptolojik uygulamalar için güvenliği arttırmak amaçlı kullanılan bir diğer son işlem yöntemi ise özet fonksiyonlar için son standart olan Keccak Algoritması. Selman Yakut, Taner Tuncer ve Ahmet Bedri Özer'in yayınladığı makalede (Yakut ve ark., 2019) güvenli rasgele sayılar üreten Keccak algoritması ve tahmin edilebilirliği ve yeniden üretimi engelleyen ek girdiler üreten halka osilatörler beraber kullanılmıştır. Çalışmada üretici daha verimli hale getirmek için Keccak algoritması yeniden düzenlenip kullanılmıştır. Önerilen üretilen, ek girdi olarak 512 bit ham gerçek rasgele sayı alınıp 1024 bitlik gerçek rasgele sayı dizisi üretilmektedir.

Aynı yazarların yayınladığı bir diğer makalede ise yine Keccak algoritması kullanılarak 1600 bit

giriş için sıfır veri kaybı ile 1600 bitlik rasgele sayı dizisi üretilmiştir (Yakut ve ark., 2020).

Önerilen bir diğer son işlem yöntemi ise S-box (Avaroğlu ve Tuncer, 2020). SBOX DES, AES gibi şifreleme sistemlerindeki doğrusal olmayan tek bileşendir. Bu yüzden güçlü şifreleme sistemlerinin tasarlanması için kriptolojik özellikleri iyi olan S-Box'ların tasarlanması gerekmektedir. (Avaroğlu ve Tuncer, 2020)'de önerilen algoritmada RO tabanlı bir GRSÜ tanımlanmış ve son işlem algoritması olarak da DES için tasarlanmış olan S-box kullanılmıştır. Kullanılan S-box 6 bit giriş için 4 bit çıkış üretmektedir. 6 bit giriş dizisinin en önemli biti (MSB) ve en az anlamlı biti (LSB) satırı belirlerken, diğerleri sütunu belirlemektedir. Bu satır ve sütundaki sayının ikili sistemdeki karşılığı çıkış bit dizisi olarak kabul edilir. Veri çıkış oranını 2/3 oranında azaltmaktadır.

3. SON İŞLEM ALGORİTMALARI

Rasgele sayı üreticilerinden elde edilen bit dizilerinden bazıları zayıf zayıf istatistik özellikler göstermektedir. Bu nedenle üretilen dizilere bu zayıflığın giderilmesi amacıyla çeşitli son işlem algoritmaları uygulanmaktadır. Son işlem algoritmasına bağlı olarak üreticiler, saldırganlara ve çevresel değişikliklere karşı dirençli hale gelirken üreticinin güvenliği de artmaktadır.

a. Von Neumann Son İşlem

Von Neumann (Suresh ve Burleson 2010) algoritması bit dizisindeki düzensizlikleri gidermeyi amaçlayan en basit ve en eski son işlem yöntemlerindedir. Algoritma giriş olarak üreticiden gelen eşzamanlı bitleri alır. Tabloda gösterildiği gibi bit dizisi (1,0) ise çıkış biti 1, eğer bit dizisi (0,1) ise çıkış biti 0 olarak üretilir. (0,0) ve (1,1) bit dizileri dikkate alınmaz.

Bu dikkate alınmayan bit dizilerinden dolayı sistemin çıkış bit hızı sabit değildir.

Tablo 1. Von Neumann son işlem çıkışı

Bit Dizisi	Von Neumann Çıkışı
00	Çıkış yok
01	0
10	1
11	Çıkış yok

b. XOR Son İşlem

XOR (Davies, 2002) doğrulama, elde edilen bit dizisinin n bit(n=2) bloklara ayrılıp bu blokların kendi içerisinde XOR işlemine tabii tutulması ile 1 çıkış biti üretilmesi işlemidir. Çıkış bit verimini 1/n kez azaltmaktadır.

Tablo 2. XOR son işlem çıkışı

Bit Dizisi	XOR Çıkışı
00	0
01	1
10	1
11	0

c. H Fonksiyonu Son İşlem

Bir diğer son işlem yöntemi ise Dichtl tarafından önerilen, Quasigroup dizi dönüşümüne dayalı olan H son işlem algoritmasıdır (Dichtl, 2007). Şekil 1.1'de de görüleceği gibi 16 bit giriş bitinden 8 bit çıkış elde edilir. Son işlem için giriş bitleri $a_0, a_1, a_2, \dots, a_{15}$ şeklinde tanımlanmıştır. $b_i = a_i \oplus a_{(i+1) \bmod 8}$ ile b_0, b_1, \dots, b_7 8 bit tanımlanır. c_0, c_1, \dots, c_7 çıkışı $c_i = b_i \oplus a_{(i+8)}$ ile tanımlanır.

16 bit giriş dizisini a_1 ve a_2 olarak ikiye ayırdığımızda Denklem (2) deki gibi olmaktadır. sözde kod olarak gösterimi aşağıdaki şekilde olmaktadır;

Denklem (3): H fonksiyonu sözde kod gösterimi

$$H(a_1, a_2) = \text{XOR}(\text{XOR}(a_1, \text{rotateleft}(a_1, 1)), a_2)$$

Daha sonra H fonksiyonu geliştirilerek Denklem (4)'te gösterilen H2 ve Denklem (5)'te gösterilen H4 fonksiyonları tanımlanmıştır.

Denklem (4): H2 fonksiyonu sözde kod gösterimi

$$H_2(a_1, a_2) = \text{XOR}(\text{XOR}(\text{XOR}(a_1, \text{rotateleft}(a_1, 1)), \text{rotateleft}(a_1, 2)), a_2)$$

Denklem (5): H4 fonksiyonu sözde kod gösterimi

$$H_4(a_1, a_2) = \text{XOR}(\text{XOR}(\text{XOR}(\text{XOR}(a_1, \text{rotateleft}(a_1, 1)), \text{rotateleft}(a_1, 2)), \text{rotateleft}(a_1, 4)), a_2)$$

d. SBOX Son İşlem

SBOX ile amaç bir tablo vasıtası ile kimin kiminle yer değiştireceğinin belirlenmesidir. Oluşturulan S-kutusunun tipine veya kategorisine bakılmaksızın, herhangi bir S-kutusunun etkili olabilmesi için belirli özellikler göstermesi gerekir. İstedığınız herhangi bir yer değiştirme şemasını bir araya getirip iyi bir S-kutusu oluşturamazsınız. Bir S-kutusu karışıklık (confusion) ve difüzyon (diffusion) sağlamalıdır. S-kutularında bu girişleri üretmenin bir yolu, n giriş bitlerini m çıkış bitlerine

eşleyen doğrusal olmayan bir Boole fonksiyonu (Boolean function) oluşturmaktır.

(Avaroğlu ve Tuncer, 2020)'de önerilen son işlem yönteminde DES için hazırlanan 8 SBOX kullanılmıştır.

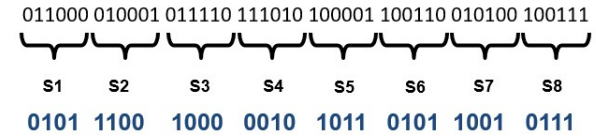
DES S-Kutuları 6 bit giriş alır ve 4 bit çıkış verir. Görüldüğü üzere 6 bitin dışta olanları (1. ve 6. bit) satırları, iç tarafta kalan bitler (2.,3.,4.,5.) sütunları oluşturmaktadır. Buna göre seçilen satır ve sütundaki sayı da çıkışı verir. Şekil 5'te gösterilen örnekle açıklayacak olursak;



Şekil 5. Örnek bit dizisi

S1 için ilk 6 bitlik diziyeye baktığımızda MSB ve LSB yani 00 bize satırı verecektir. Bu durumda bakmamız gereken satır 0. Arada kalan bitler ise sütunu verecektir; 1100 yani; 12. Bu durumda bakmamız gereken yer S-box 1'de 0. Satır 12. Sütun. Karşılığında denk gelen sayı 5. Bunun 2li sistemdeki karşılığı ise bize çıkış bit dizimizi verecek yani; 0101.

Bu şekilde tüm dizi için SBOX üzerindeki karşılıklarını bularak Şekil 6'da gösterilen yeni çıktı dizimizi üretiyoruz.

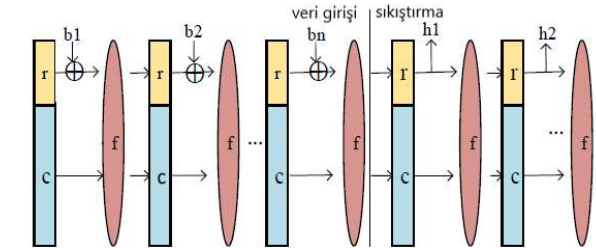


Şekil 6. SBOX son işlem algoritması uygulandıktan sonra oluşan bit dizisi

e. Keccak Algoritması

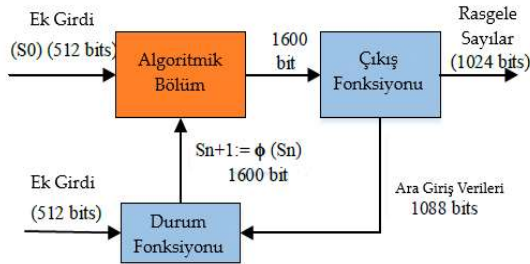
Özet algoritmalarının en son standardı olan Keccak, sünger yapısına dayandığı için herhangi bir uzunlukta bir girdi dizisi alabilir ve çeşitli uzunluklarda özet değeri üretebilir (Yakut ve ark., 2019).

Şekil 7'de gösterildiği gibi bir yapıya sahip olan Keccak algoritması 2 temel parçadan oluşmaktadır. İlk parçada giriş mesaj blokları işlenirken ikinci bölümde özet değerleri üretilmektedir.



Şekil 7. Keccak algoritmasının genel yapısı

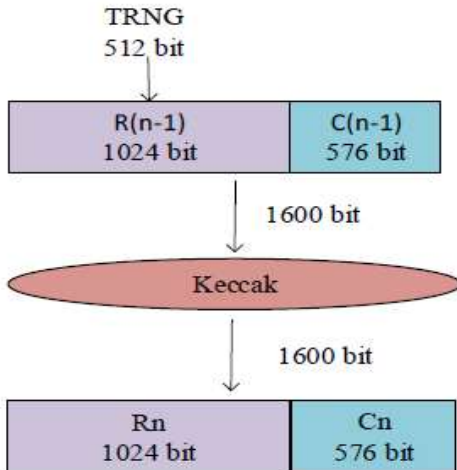
(Yakut ve ark., 2019)'da Keccak algoritması yeniden düzenlenerek kullanılmıştır ve 512 bit giriş dizisi ek girdi olarak alınıp 2014 bit gerçek rasgele ayı üretimi sağlanmıştır. Kullanılan yapı Şekil 8'de gösterilmiştir. 512 bit ham gerçek rasgele sayı ek girdi olarak kullanılmıştır. Üretimin ilk çalışmasında ek girdi(S0) Keccak fonksiyonuna verilirken, sonraki Durum Fonksiyonuna verilir. Keccak fonksiyonunda algoritmik işleme tabi tutulan saf veriden 1600 bitlik özet değeri üretilir. Bu algoritmik işlemlerde 1600 bitlik giriş alınır ve 24 tur işlendikten sonra 1600 bitlik çıktı oluşur. Çıkış fonksiyonu kullanılarak alınan 1600 bit özet değerinden 1024 bit gerçek rasgele sayı ve 1088 bit ara giriş verisi üretilir. Ara giriş verisi ve ek girdi kullanılarak bir sonraki çalışmada Keccak algoritması için 1600 bitlik giriş verisi oluşturulur.



Şekil 8. (Yakut ve ark., 2019)'da önerilen rasgele sayı üreticinin genel yapısı

Yine aynı yazarların bir diğer çalışmasında ise yine Keccak algoritması kullanılarak 1024 bit ham gerçek rasgele sayı girişinden, 1024 bit gerçek rasgele sayı çıkışı üretilmiştir (Yakut ve ark., 2020).

Şekil 9'da gösterildiği üzere oluşturulan yapı R giriş vektörü ve C ara vektöründen oluşur. Her çalışmada 1024 bit ham gerçek rasgele sayı R, bir önceki çalıştırmadan gelen 576 bit vektör kısmı C olarak alınır. Bu 1600 bitlik giriş vektörü, algoritma tarafından işlenir ve 1600 bitlik bir çıktı vektörü oluşturulur. Bu çıkış vektörü, R ve C bloklarından oluşur.



Şekil 9. (Yakut ve ark., 2020)'de önerilen rasgele sayı üreticinin genel yapısı

f. Resilient Fonksiyon

(n, m, t)-resilient fonksiyonu $F(x_1, x_2, \dots, x_n) = (y_1, y_2, \dots, y_m)$ şeklinde, $Z_2^n \rightarrow Z_2^m$ olan ve i_1, \dots, i_t koordine eden herhangi bir t için Z_2^m 'den herhangi bir a_1, \dots, a_t sabiti için ve ortak etki alanının herhangi bir y ögesi için $\text{Prob}[F(x) = y | x_{i_1} = a_1, \dots, x_{i_t} = a_t] = \frac{1}{2^m}$ özelliğini sağlayan bir fonksiyondur.

Daha basit bir ifade ile açıklayacak olursak; fonksiyon girdi olarak verilen herhangi bir t değerinin bilinmesi kişinin çıktıda rasgele bir tahminden fazlasını yapmasına izin vermez. Bu yönüyle oldukça güvenlidir ve birçok kriptografik uygulamada kullanılmaktadır.

Sunar, Martin ve Stinson tarafından yapılan bir çalışmada da resilient fonksiyonlar deterministik bitlerin filtrelenmesi için kullanılmıştır (Sunar ve ark., 2007). Burada hata düzeltme kodlarından resilient fonksiyon oluşturulmuştur. Bunun için de basit bir teknik verilmiştir: G, [n, m, d] doğrusal kod için bir üretici matrisi olsun. $f(x) = xG^T$ kuralına göre bir $f: \{0, 1\}^n \rightarrow \{0, 1\}^m$ fonksiyonu tanımlandığında, bu fonksiyon f(n, m, d-1)-resilient fonksiyondur. Uygulama için kullanılan kod ise H_m Hamming kodunun ikilisi olan, tek yönlü $[2^m - 1, m, 2^{m-1}]$ doğrusal kod H_m^\perp 'dir.

Makalede 114 halka osilatörden alınan çıktı resilient fonksiyonuna verilir. Resilient fonksiyon için bilinen bir genişletilmiş BCH kodu olan [256, 16, 113] kodu kullanılmaktadır. Yani 256 bitlik bloklara ayrılan osilatör çıktıları resilient fonksiyona verilir ve fonksiyondan yalnızca 16 bitlik çıkış alınır.

4. SONUÇ

Tablo 3. Son işlem algoritmaları ve bit dizisi azalma oranları

Son İşlem Algoritmaları	Bit Dizisi Azalma Oranı
Von Neumann	3/4
XOR	1/2
H-Fonksiyonu	1/2
SBOX	2/3
Keccak Algoritması	0/1
Resilient Fonksiyon	1/16

XOR son işlem algoritması basit ve sabit bit çıkış hızı sağlamasından dolayı tercih edilse de sistemin güvenliğini sağlama konusunda zayıf kalmaktadır. Von Neumann algoritması ise entropiyi ideal değer olan 1'e yaklaştırırken bazı giriş bitleri atıldığından dolayı sistemin bit çıkış hızı sabit değildir ve 3/4 oranında azalmaya sebep olur.

H fonksiyonu 15/16 oranında bir azalmaya sebep olurken entropi değerinin 1 olarak elde edilmesi en büyük avantajıdır.

SBOX son işlem algoritması ise NIST testlerinde başarılı sonuçlar elde etmesinin yanı sıra otokorelasyon ve karmaşıklık ölçümlerinde de

başarılı sonuçlar elde etmiştir. Ayrıca H fonksiyonu ve Von Neumann algoritmalarında olduğu gibi entropi değeri 1 olarak elde edilmektedir. Bu başarılı sonuçlardan dolayı kriptografi gibi birçok alan için kullanıma uygun olduğu görülmektedir.

Resilient fonksiyonu ise saldırılara karşı güçlü olmasına karşın H fonksiyonu gibi 15/16 oranında bir veri kaybına sebep olmaktadır.

Keccak algoritmasında ise ek girdiler kullanılarak rasgele sayıların yeniden üretilmesi ve tahmin edilebilirlik önlenmiştir. Yapılan testler göstermektedir ki kriptografik uygulamalar için gerekli olan güvenlik gereksinimleri karşılanmaktadır. Algoritmanın en önemli avantajı ise veri kaybına sebep olmamasıdır.

KAYNAKÇA

- Yadav, A. (2013). Design and Analysis of Digital True Random Number Generator, M.Sc. Thesis Virginia Commonwealth University, Virginia.
- Anikin, I. V. & Alnajjar, K. (2016). Pseudo-Random Number Generator Based on Fuzzy Logic. In 2016 International Siberian Conference on Control and Communications (SIBCON), 2016 International Siberian Conference, 1-4.
- Avaroğlu, E. & Türk, M. (2013) Son İşlemin Gerçek Rasgele Sayı Üreteçleri Üzerindeki Etkisinin İncelenmesi. In 6th International Information Security and Cryptology Conference, ISCTURKEY 2013, 290-294.
- Tehranipoor, F., Yan, W. & Chandy, J. A. (2016). Robust Hardware True Random Number Generators using DRAM Remanence Effects. In 2016 IEEE International Symposium on Hardware Oriented Security and Trust (HOST), 79-84.
- Tsuneda, A., Mitsuishi, S. & Inoue, T. (2008). A Study on Generation of Random Bit Sequences with Post-Processing by Linear Feedback Shift Registers. *International Journal of Innovative Computing, Information & Control*, 4(10), 2631-2638.
- Tsuneda, A. & Morikawa, K. (2013). A Study on Random Bit Sequences with Prescribed Auto-Correlations by Post-Processing Using Linear Feedback Shift Registers. In 2013 European Conference on Circuit Theory and Design (ECCTD).
- Avaroğlu, E., Tuncer, T., Özer, A.B., Ergen, B., Türk, M. (2015). A novel chaos-based post-processing for TRNG. *Nonlinear Dynamics*, 81, 189-199.
- Loza, S. & Matuszewski, L. (2014). A True Random Number Generator Using Ring Oscillators and SHA-256 as Post-Processings. In *International Conference on Signals and Electronic Systems (ICSES) 2014*, 1-4.
- Nikolic, S. & Veinovic, M. D. (2016). Advancement of True Random Number Generators Based on Sound Cards Through Utilization of a New Post-processing Method. *Wireless Personal Communications*, 91(2), 603-622.
- Von Neumann, J. (1951). Various Techniques Used in Connection With Random Digits. *National Bureau of Standards Applied Math Series 12*, 36-38.
- Peres, Y. (1992). Iterating Von Neumann's Procedure for Extracting Random Bits. *Annals Statistics*, 20(1), 590-597.
- Elias, P. (1972). The efficient construction of an unbiased random sequence. *Ann. Math. Statist*, 43(3), 864-870.
- Zhang, R., Chen, S., Wan, C. & Shinohara, H. (2018). High-Throughput Von Neumann Post-Processing for Random Number Generator. In 2018 International Symposium on VLSI Design, Automation and Test (VLSI-DAT), 1-4.
- Yakut, S., Tuncer, T., & Özer, A. B. (2019). Secure and Efficient Hybrid Random Number Generator Based on Sponge Constructions for Cryptographic Applications. *Elektronika Ir Elektrotehnika*, 25(4), 40-46.
- Yakut, S., Tuncer, T., & Özer, A. B. (2020). A New Secure and Efficient Approach for TRNG and Its Post-Processing Algorithms. *Journal of Circuits, Systems and Computers*
- Avaroğlu, E. & Tuncer, T. (2020). A novel S-box-based postprocessing method for true random number generation. *Turk. J. Elec. Eng. & Comp. Sci.* (2020) 28, 288-301.
- Suresh, V. B., & Burleson, W. P. (2010). Entropy extraction in metastability-based TRNG. In *Proceedings of the IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, 135-140.
- Davies, R. B. (2002). Exclusive OR (XOR) and hardware random number generators. 1-11. <http://www.robertnz.net/pdf/xor2.pdf>
- Dichtl, M. (2007). Bad and Good Ways of Post-processing Biased Physical Random Numbers. In *Proceedings of International Workshop on Fast Software Encryption (Luxembourg, Luxembourg, Mar. 26-28, 2007). FSE '07. Lecture Notes in Computer Science*, 4593, Springer, Berlin, Germany, 137-152.

Sunar, B., Martin, W. J. & Stinson, D. R. (2007). A Provably Secure True Random Number Generator with Built-in Tolerance to Active Attacks. *IEEE Transactions on Computers* 2007, 56 (1), 109-119.



Araştırma Makalesi

A Distributed Computer System Architecture Acting As A High Performance Computing System For Big Data Analytics

Hamza EROL*¹, Timuçin KORKMAZ²

¹Mersin University, Engineering Faculty, Department of Computer Engineering, Mersin - Turkey

²Mersin University, Graduate School of Natural and Applied Sciences, Department of Computer Engineering, Mersin - Turkey

Keywords:

Big data analytics,
Distributed computer
system architecture,
Emotion analysis,
Image classification,
Parallel programming,
High performance
computing system,
Face expression
recognition

ABSTRACT

A distributed computer system architecture, which acts as a high-performance computing system for big data analytics, was created. In the distributed computer system architecture, the volume of big data and the load of the threads to be processed were taken into account. The distributed computer system architecture was designed based on the principle of using personal computers together and on the same computer network in normal standards and features. For the distributed computer system architecture, a computer featured on the common computer network is planned as the main computer and the others as slave computers. Threads created for big data analytics were processed simultaneously in the processors of both the main computer and slave computers. Two different databases were used in the processing of threads created for big data analytics in the distributed computer system architecture. A database was used for the process of reading data files for threads and another database was used for writing process results to output files. Visual C # parallel programming language was used to process the threads in the processors of computers in the distributed computer system architecture. In this study, the emotional expression classification of facial images was made. For this purpose, first the image processing was done in the MATLAB software, and the creation of threads was carried out. The threads created later were processed simultaneously on the processors of computers in the distributed computer system network. By processing the threads in the distributed computer system architecture, image files are classified for facial emotion prediction. Process outputs and file classification labels for threads are printed as output to the second database. Big data analytics can be done using high performance computing systems. Such systems are costly, and the number of installed systems is low. With this study, the distributed computer system architecture was created for big data analytics. High performance computing system architecture was realized by using databases and parallel programming. The image files in the database are classified and processed as threads in the high-performance computing system architecture created. The cost of the created system is low, and the speed of processing is high. With the distributed computer system architecture, big data analytics has been made convenient and applicable.

*Sorumlu Yazar

*(herol@mersin.edu.tr) ORCID ID 0000 - 0001 - 8983 - 4797

(timucinkorkmaz33@gmail.com) ORCID ID 0000 - 0003 - 0888 - 1327

e-ISSN: 2717-8579

Geliş Tarihi: 06/10/2020; Kabul Tarihi:13/12/2020

Bilgisayar Bilimleri ve Teknolojileri Dergisi

Büyük Veri Analitiği İçin Yüksek Performans Hesaplama Sistemi Gibi Davranan Bir Dağıtık Bilgisayar Sistemi Mimarisi

Anahtar Kelimeler:

Büyük veri analitiği,
Dağıtık bilgisayar sistemi
mimarisi,
Duygu analizi,
Görüntü sınıflandırma,
Paralel programlama,
Yüksek performans
hesaplama sistemi,
Yüz ifadesi tanıma

ÖZET

Büyük veri analitiği için yüksek performans hesaplama sistemi gibi davranan bir dağıtık bilgisayar sistemi mimarisi oluşturulmuştur. Dağıtık bilgisayar sistemi mimarisinde, büyük verinin hacmi ve işlenecek iş parçacığının yükü dikkate alınmaktadır. Dağıtık bilgisayar sistemi mimarisi, kişisel bilgisayarların birlikte ve aynı bilgisayar ağı üzerinde normal standartlarda ve özelliklerde kullanılması esasına göre tasarlanmıştır. Dağıtık bilgisayar sistemi mimarisi için ortak bilgisayar ağında yer alan bir bilgisayar ana bilgisayar, diğerleri de bağımlı bilgisayar olarak planlanmıştır. Büyük veri analizi için oluşturulan iş parçacıkları hem ana bilgisayar hem de bağımlı bilgisayarların işlemcilerinde aynı anda işlenmektedir. Dağıtık bilgisayar sistemi mimarisinde büyük veri analitiği için oluşturulan iş parçacıklarının işlenmesinde iki farklı veri tabanı kullanılmaktadır. İş parçacıkları için veri dosyalarını okuma işlemi için bir veri tabanı ve işlem sonuçlarını çıktı dosyalarına yazmak için başka bir veri tabanı kullanılmaktadır. Visual C# paralel programlama dili, dağıtılmış bilgisayar sistemi mimarisindeki bilgisayar işlemcilerindeki iş parçacıkları işlemek için kullanılmıştır. Bu çalışmada yüz görüntülerinin duygusal ifade sınıflandırması yapılmıştır. Bu amaçla önce MATLAB yazılımında görüntü işleme yapılmış ve internet protokol numaralarının oluşturulması gerçekleştirilmiştir. Daha sonra oluşturulan iş parçacıkları, dağıtılmış bilgisayar sistemi ağındaki bilgisayarların işlemcilerinde eşzamanlı olarak işlenmiştir. Dağıtılmış bilgisayar sistemi mimarisinde iş parçacıkları işlenerek, görüntü dosyaları yüz ifadesi tahmini için sınıflandırılmıştır. İşlem çıktıları ve iş parçacıkları için dosya sınıflandırma etiketleri, ikinci veri tabanına çıktı olarak yazdırılmıştır. Büyük veri analizi, yüksek performanslı bilgi işlem sistemleri kullanılarak yapılabilmektedir. Bu tür sistemler maliyetlidir ve kurulu sistem sayısı düşüktür. Bu çalışma ile büyük veri analitiği için dağıtık bilgisayar sistemi mimarisi oluşturulmuştur. Veri tabanları ve paralel programlama kullanılarak yüksek performanslı bilgi işlem sistemi mimarisi gerçekleştirilmiştir. Veri tabanındaki görüntü dosyaları, oluşturulan yüksek performanslı bilgi işlem sistemi mimarisinde iş parçacıkları olarak sınıflandırılır ve işlenir. Oluşturulan sistemin maliyeti düşük ve işlem hızı yüksektir. Dağıtılmış bilgisayar sistemi mimarisi ile büyük veri analitiği rahat ve uygulanabilir hale getirilmiştir.

*Sorumlu Yazar

*(herol@mersin.edu.tr) ORCID ID 0000 - 0001 - 8983 - 4797
(timucinkorkmaz33@gmail.com) ORCID ID 0000 - 0003 - 0888 - 1327

1. INRODUCTION

In this study, the distributed computer system architecture was created by installing an intranet among the required number of personal computers. It was made by pinging on the manual IP route, which is connected to each other with personal computers. While IP is given to personal computers, network and internet settings are regulated. Network and internet settings are started. Adapter options are changed. Ethernet card is set. The properties of the Ethernet card are regulated. Internet protocol version 4 (TCP / IPv4) features are regulated. IP values are entered.

By entering the IP address when entering IP values, the connection is provided by entering the desired IP and subnet mask values. Connection control is carried out with the ping command via cmd. While making transactions, IP addresses based on 192.168.0.1 have been preferred to computers. In this study, the IP address of the main computer (PC1_M00) was determined as 192.168.0.14 and the IP addresses of the slave computers (PC2_Sxx) was determined as the number required, starting from 192.168.0.15. Slave operations are required to access the main computer, as the operations are done on the main computer. Access test was done by typing ping 192.168.0.14 code on the cmd screen of the slave computers.

Pinging from the slave computers (PC2_Sxx) ensures that access to the main computer (PC1_M00) has been successful. The connection between the number of personal computers required in the basic distributed computer system architecture is an intranet connection. The distributed computer system architecture also has switches with the required number of ports for the common computer network. If desired, parallel switches can be made with the processor of more computers by increasing the number of slave computers by placing switches in the distributed computer system architecture.

In this study, a distributed computer system architecture (Yadav et al., 2013), which acts as a high performance calculation system (Cliff et al., 2019), was created for big data analytics (Ezhilraman and Srinivasan, 2018; Kouanou et al., 2018). In this section, big data analytics, high performance calculation system, distributed computer system architecture (Czarnul et al., 2020), parallel programming (Hager and Wellein, 2011), database programming, computer architecture (Murdocca and Heuring, 1999), computer network (Dordal, 2020) concepts will be examined in relation.

Distributed computer system architecture, created to classify image files in the database and acting as a high-performance calculation system, will be explained with the concepts of database and parallel programming.

2. DISTRIBUTED COMPUTER SYSTEM ARCHITECTURE

2.1. System and Application Softwares

A computer network system is called a distributed computer network in the system where computers communicate with each other using the network infrastructure of hardware or software tools (Coulouris et al., 2012). A distributed computer system can have

two or more computers and/or devices. These devices can make data exchange with each other over the computer network. In the distributed computer system architecture, MSSQL database has been installed on the main computer. In the distributed computer system architecture designed in this study, all personal computers were connected to the MSSQL database installed on the main computer. Information about the operations performed in the classification of face images and the transaction outputs are printed on the database. In addition, database programming has been checked to see if the image is processed in the database. MSSQL Management v17.7 version was installed and used on the main computer.

In the distributed computer system architecture, MSSQL database server permissions were established by opening 1433 port over the network so that all personal computers can access the MSSQL database server. This port must be opened in order to access the MSSQL database server over the network.

The process of opening ports on the MSSQL database server are: (1) SQL Server Configuration Manager opens. SQL Server Configuration Manager can be accessed via the path C:\Windows\SysWOW64\SQLServerManager14.msc. In this screen, SQL Server Network Configuration is selected. (2) The TCP/IP setting is set to Enabled on the incoming screen. (3) Right click TCP/IP option and select features. The IP Addresses tab is displayed on the screen. (4) By finding the IP address to be used, the Enabled field is set to Yes and the port information to be forwarded is entered. (5) After all these operations are done, the screen is closed by saying Apply changes. Restart is done by selecting SQL Server Services for the changes to be applied. (6) After opening the port, the permission process for this port is required from the Firewall. Permission is made so that another device can access it. For this, Control Panel->System and Security->Firewall screen opens. Advanced settings are selected on the selected screen. (7) Select Rules from Screen, Click New Rule button. (8) Select the port on the screen and press the next button. (9). On the incoming screen, TCP/IP is selected, as a rule, by selecting specific local ports, enter the port information to be allowed here and click on the next option. (10). Select the Allow Connection button on the Action screen and click Next. (11) Marked in 3 options on the Profile screen and called Next. (12) Finally, the Last Option is selected by naming the rule. (13) The Rule allowed after this Transaction will appear on the rules screen. Access to the SQL Server is permitted via the IP address allowed. (14) You can enter the system by calling Connect. When it is entered into the system by connecting to it, it writes the IP address determined as the name of SQL Server in the section marked in red. This SQL Server can be accessed through the same network infrastructure.

Another software used for distributed computer system architecture is the Entity framework, a Microsoft-based framework. Entity framework is a process that enables data to be sent to the database in object oriented programming with a relational database (Lerman, 2009). The necessary transactions are provided by transmitting the data to the database through the framework. Relational database operations

with the Entity framework are performed using the model structure.

Entity Framework is based on Microsoft. The ADO.NET (ActiveX Data Objects.NET) infrastructure is used when making data connections. ADO.NET is a Microsoft-based data access and management technology. Entity Framework is a more advanced structure than ADO.NET technology. It is necessary to use query clauses for all database operations in ADO.NET. Performing a query requires more processing than current technologies. All database operations in Entity framework; is done on the created model. Entity Framework uses 3 different methods for 4 different situations in the workflow (Hung et al., 2016). There are advantages and disadvantages to using the Entity Framework. The advantages of using Entity Framework are: (1) It provides object-oriented project development. (2) Allows to perform transactions on the database without database knowledge. (3) Used with all database applications. (4) No need to back up the database with Code first technique. Operation can be done by creating a database at the desired location. (5) It positively contributes to software development time and costs.

The disadvantages of using Entity Framework are: (1) Processing speed is slower than ADO.NET. (2) All control operations are carried out by the entity framework. This sometimes creates complex queries in an easy check and negatively affects performance. (3) Linq is used as the language of use.

2.2. Databases in Basic Distributed Computer System Architecture

First, Information on two different databases used in the distributed computer system architecture created in this subsection will be given. The first database is the MSSQL database used by the distributed computer system architecture. The second database is the MS Access database used in the classification process. The first database used in the distributed computer system architecture is SQL based. In the basic distributed computer system architecture, this database accessed by computers is the database containing information about the data analyzed in the system.

In the database, there is an automatically increasing Id related to the image, the name of the image being processed, the emotional state detected in the image, the computer information processing and the date of the transaction. A database named GoruntuAktar was created in the database by entering the appropriate formats for the state of the data. The created database table and the format information of the data are kept in the DataAktar table.

In this study, while the application developed with visual C# parallel programming is running and before the application analysis result is saved, the registration process is performed by recording the data status in this table about the related image. If there is no information about the image to be analyzed in the related table, the analysis process begins. If there is information about the image to be analyzed in the related table, the application continues to operate with the other image. The developed application process makes another final check before saving the result to the database at the end

of the analysis. This is because the image may have been processed and recorded in the same time period on another computer's processor. Particular attention was paid to the absence of duplicate records for the analysis results to be correct. As a result of the recheck, if the result related to the image information is not in the database, the result of the analysis is recorded in the database. Application software developed with visual C# parallel programming saves the analysis results to the table in the database after data processing, data labeling and classification.

MS Access database is the second database used in data analysis in the distributed computer system architecture. MS Access database has been preferred because the analysis process can be done in a short time and because there is not a lot of processing load. By recording an image about each emotion in the MS Access database, the analysis process is performed with similarity to these images. The database created contains right eye, left eye position information, mouth opening rate, and emotional state corresponding to this image. In the data analysis application, the analysis process of the image is completed and the emotion value closest to the values found is calculated and the result of the operation is determined according to the emotion in that image. Right eye, left eye position information, mouth opening rate, and mood values corresponding to this image are added to the created MS Access database. In the database used, a test data table was created containing different emotions from different data sets. According to the results obtained, emotion results are obtained by making a comparison with the data table. While obtaining the results, transactions are made according to the values in the data table.

With some images that are recorded in the system and provide the formation of data sets, training was made according to the different emotional states of the same person. There are differences in facial expressions according to the images on the faces. Using different faces with the same emotion enables the data to be analyzed correctly. The fact that the facial data used is different and diverse increases the accuracy and reliability of the results obtained. Different emotions of the same person in the image files recorded in the database are shown in Figure 1.



Figure 1. Different emotions of the same person in the image files stored in the database.

The application software developed with visual C# parallel programming determines the emotional values corresponding to the image according to five different facial emotions using the right eye, left eye position information, mouth opening ratio values in the created MS Access database. In the MS Access database created by the application software, using the values of right eye, left eye position, mouth opening ratio, the

mood values corresponding to the image were obtained according to five different facial emotions and added to the table. Processing is performed according to the id data of the line containing the results of the closest values. The obtained Id gives the mood information of the person.

3. THE PROPOSED METHOD

The distributed computer system (Czarnul et al., 2020) proposed in this study consists of the required number of normal standard personal computers. The computer, which has better features than other computers, defined as the main computer (PC1_M00) and others are designated as slave computers (PC2_Sxx). All computers defined and introduced to each other in the computer network. General features of personal computers used in the distributed computer system architecture: Main computer (PC1_M00): Processor: Intel Core I7-4720HQ CPU 2.60 GHz; RAM Memory: 12 GB; System Type: 64 bits. Slave computers (PC2_Sxx): Processor: Intel Core I5-3210M CPU 2.50 GHz; RAM Memory: 6 GB; System Type: 64 bits. The personal computers are connected to each other via an Ethernet cable via CAT5 / CAT6 local area network port.

In the architecture of the distributed computer system created in this section, the operations performed in parallel programming phase for big data analytics will be explained.

3.1. Threads and Parallel Programming For Image Files In The Database

In the distributed computer system architecture, in order to perform big data analytics with database and parallel programming, threads are taken in parallel programming stage as much as the number of image files in the database. The data of the received image files are sent for processing in the processors on the computer network and the analysis process is started. System allows to keep information about which thread is processed with which file with which line of code:

```
FileVM dd = new FileVM {ThreadID = Thread.CurrentThread.ManagedThreadId, ThreadName = folder};
```

The process, which completes the thread, comes to the selection stage and takes an image file again and the process takes place. The image files are taken as threads and processing steps are performed in parallel programming. The content and information of the image received as a thread is sent to a computer on the common network in the distributed computer system architecture for processing in the processor. The properties of the image taken as a thread can also be displayed. The features of the images in the database include information such as name, width, length. While the image file is processed in the processor, the software classification developed with visual C# parallel programming is performed according to the image's width and height information. After the face shape determination process, the edge and the edge of the eyes are detected by performing pixel processing in the width and length of the image.

3.2. Processing of Image Files in The Database with Parallel Programming

In the application developed using visual C# parallel programming, MATLAB software was first used for processing image files. In practice, it first separates an image sent to the processors in the distributed computer system architecture as pixels in RGB. The face in the image is emphasized by performing a 3% cut/trim on each edge of the image. With the imcrop function of the MATLAB program, cutting/trimming is performed from the edges of the image. After determining the face shape from the image, the edge and eye edge detection process is performed with pixel operations from the width and length of the image. With the developed application, the edges of the images to be processed in the database are cropped. Facial elements in the cropped image are more easily determined. This facilitates the detection event of edge detection filters.

3.3. Processing of Image Files in The Database with Parallel Programming

Within the scope of big data analytics, using distributed computer system architecture, detailed face edge detection operations were performed in image files in the database (Acharjya et al., 2012; Shrivakshan and Chandrasekar, 2012). Edge detection filters have been applied to image files in the database. The 4 different edge detection methods used in the study are: (1) Canny edge detection. (2) Laplace Edge Detection. (3) Sobel Edge Detection and (4) Robert Edge Detection.

Four different edge detection algorithms have been applied to face image files in the database for the face recognition system (Tsankashvili, 2018). Accordingly, all face image files in the database are classified into five different emotion categories. The results are given in Table 1.

TABLE 1. CLASSIFICATION OF ALL FACE IMAGE FILES IN THE DATABASE INTO FIVE DIFFERENT EMOTION CATEGORIES USING FOUR DIFFERENT EDGE DETECTION ALGORITHMS FOR THE FACE RECOGNITION SYSTEM.

		Facial Expression					
		Discrimination (%)	Normal	Sad	Smile	Surprise	Angry
Edge detection method	Canny	100.0	91.3	99.4	95.7	94.8	
	Laplace	93.2	87.1	89.5	85.6	87.4	
	Sobel	63.4	49.7	66.8	64.9	62.6	
	Robert	49.8	30.1	42.5	48.2	48.6	

4. CLASSIFICATION OF IMAGES IN THE DATABASE WITH PARALLEL PROGRAMMING AND EXPERIMENTAL RESULTS

Use facial emotion classification was made using the software developed with visual C# parallel programming language on the distributed computer system architecture created within the framework of big data analytics of image files in the database. Sobel filter was used in the selection of facial expressions (Acharjya et al., 2012; Shrivakshan and Chandrasekar, 2012; Tsankashvili, 2018). Face selection was made

thanks to the convolution matrices of the Sobel filter. The face image is divided into 4 equal parts and the edge detection processes are performed for eye and mouth detection. The results obtained at the end of the operations are assigned to the variables. Appropriate mood status was determined by comparing eye and mouth opening values with the values in the database. Before registering to the database, the analyzes were checked to ensure that the same results were not recorded in the database.

The database connection in the distributed computer system architecture was made using the Entity framework. This connection method is faster than adonet, so this is preferred for interrogation. A code blog has been developed to process the selected images in the processors and start the process. The process starts by selecting the image lines. Then, with the parallel for method, the images were taken to the threads and the processing lines were continued in this way. Face detection was done with the Sobel filter. Code section was created to take the eye areas.

A code section was prepared in which processes related to the mouth area were made. The code section was prepared, in which the database was checked and the records were made to the database. In the developed application, the working principle of facial emotion analysis was realized on these code blocks. Recording processes for emotion classification and data labeling were also carried out using these code blocks. The existing, previously known label values of the facial images in the database were compared with the labeling results obtained as a result of the classification calculations made in this study. The operation was carried out for 58 face image files registered in the database. The manual classification results of these were compared with the results obtained from the calculation in practice. The emotional categories obtained as a result of the classification of the facial images and the correct classification percentages piercing these categories are given in Table 2.

TABLE 2. EMOTION CATEGORIES OBTAINED AS A RESULT OF CLASSIFICATION OF FACE IMAGES AND CORRECT CLASSIFICATION PERCENTAGES CORRESPONDING TO THESE CATEGORIES.

	Facial Expression				
	Normal	Sad	Smile	Surprise	Angry
Correct classification percentages	95%	92%	90%	92%	86%

4.1. Application Developed for The Classification Of Image Files In The Database With Parallel Programming

The application is a Windows form application developed with Windows based visual C# parallel programming language. Visual Studio was used as IDE during the development phase. .NET 4.6 was used as the framework while developing in Visual Studio. There is no parallel programming library in frameworks before .Net framework 4. It is preferred because it contains libraries related to parallel programming in .Net framework 4.6 and it is one of the most current

versions. Application file of the application's Windows desktop .exe extension has been created.

In the developed application, the select file button is used to start the processes. When the Select file button is pressed, it will open the My Documents-> Pictures file on the first boot as it is set as picture files in the OpenFileDialog toolbox. In later use of the program, the most recently used file extension is opened. In the application, analysis of the desired number of images is performed by clicking the select file button. The window that opens and opens when the select file button is clicked is shown in Figure 2.

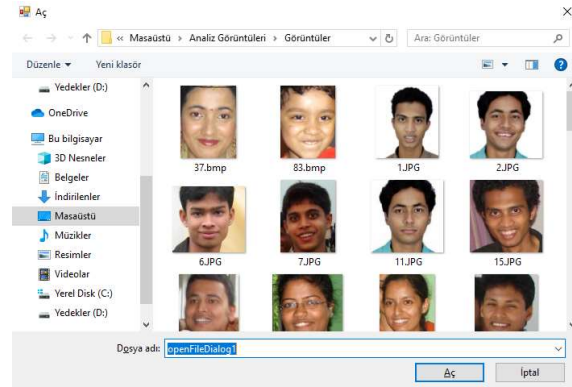


Figure 2. The window that appears when the Select File button is clicked.

The images selected from the opened window are processed in the processors of the computers in the distributed computer system architecture created randomly due to parallel programming; The name of the image is recorded with information such as mood, processing date. If one of the selected images was previously analyzed and saved in the database, another image is analyzed without analyzing the image again. The image processed during the analysis can be displayed on the form screen (PictureBox toolbar) for a short time.

The database is checked for the image currently processed in a computer processor. It is checked whether or not any action has been taken regarding the image. The first image related to the image is recorded as a result of the analysis. If the image is analyzed in another processor or on another computer in the distributed system while it is being processed, it is not saved again. The system prevents repetitive processes in the analysis process.

4.2. Classification Results of Images In The Database

By using parallel programming on the distributed computer system architecture, each image in the database is assigned to one of five different categories. The classification results of the images in the database are taken as in the figures shown in Figure 3.

When the operations are carried out in the architecture of the distributed computer system, data on their results are recorded in the database. Saving the result data about the transactions in the database provides the opportunity to be used for analysis. In the

distributed computer architecture, information on which computer in the database image files are processed as threads is also recorded.



Figure 3. Classification results for each of the images in the image files in the database.

5. CONCLUSIONS AND DISCUSSIONS

In this research, distributed computer architecture, which acts as a high performance system, was created. Databases and parallel programming are used in distributed computer system architecture. Face image files in the database are classified for emotion analysis using the distributed computer system. It was concluded from the facial expressions that the formation of emotions was influenced by many factors. These factors can be divided into two environmentally and psychologically. Information was obtained about the psychological state of the person or the environment in which the person is directly affecting the emotions of the person. It is revealed that the person creates reactions with these gestures and gestures against these factors.

The importance of mimic movements was understood in the perception of emotions from facial expression. Based on these movements, we have been informed about what emotion the person has at that moment. The collected information about the reactions of the person according to his emotional state. A hundred datasets were created with this collected information. The data set contains images of different people with different emotional states. The emotions in the image were taken according to the test values of the application by taking the expressions in the mimics of the people. The results obtained are shown in the application success table.

Different programming languages and methods were used in this study. The programming languages used are Matlab and visual C# parallel programming language. In the software prepared for the study subject: Matlab program was used to process image files and simplify the images. Visual C# predicts facial emotional

states from parallel programming image files and has been applied in the classification process.

Visual C# parallel programming coding and classification methods and performances of algorithms are used. Threads created using the basic distributed computer system architecture were processed on other computer processors on the common computer network. Thanks to the filters used in basic distributed computer architecture, the distortion and noise in the data are minimized and classification results are obtained with higher accuracy percentages. In this study, sobel filter was used. With the sobel filter used, realistic emotion values were estimated in the facial emotion classification process. High accuracy rates were obtained in the classification results. In the next study, performance comparisons of Sobel filter and other filters in image processing will be made. In addition, the performance of the deep learning method will be investigated in facial emotion analysis.

Acknowledgments

We would like to thank the editor and three anonymous referees for their valuable comments and suggestions on the manuscript.

REFERENCES

- Acharjya, Pinaki Pratim; Ritaban Das and Dibyendu Ghoshal (2012). Study and Comparison of Different Edge Detectors for Image Segmentation. *Global Journal of Computer Science and Technology Graphics & Vision* Volume 12 Issue 13 Version 1.0 Online ISSN: 0975-4172 & Print ISSN: 0975-4350.
- Cliff, Ashley; Jonathon Romero, David Kainer, Angelica Walker, Anna Furches and Daniel Jacobson (2019). Article A High-Performance Computing Implementation of Iterative Random Forest for the Creation of Predictive Expression Networks. *Genes*, 10, 996; doi:10.3390/genes10120996.
- Coulouris, George; Jean Dollimore, Tim Kindberg and Gordon Blair (2012). *Distributed Systems: Concepts and Design Fifth Edition*. Addison-Wesley. 1067 pages.
- Czarnal, Paweł; Jerzy Profcz and Krzysztof Drypczewski (2020). Survey of Methodologies, Approaches, and Challenges in Parallel Programming Using High-Performance Computing Systems. *Scientific Programming*. Volume 2020, Article ID 4176794, 19 pages <https://doi.org/10.1155/2020/4176794>.
- Dordal, Peter L. (2020). *An Introduction to Computer Networks Release 1.9.21*. <http://intronetworks.cs.luc.edu/current/ComputerNetworks.pdf>, 896 pages.
- Ezhilraman, S. Vahini and Sujatha Srinivasan (2018). State of the art in image processing & big data analytics: issues and challenges. *International Journal of Engineering & Technology*, Volume 7, Pages 195-199.
- Hager, George and Gerhard Wellein (2011). *Introduction to High Performance Computing for Scientists and Engineers*. Chapman & Hall/CRC Computational Science Series. 349 pages.
- Hung, Jason C., Neil Y. Yen and Kuan-Ching Li (2016). *Frontier Computing: Theory, Technologies and Applications*. Springer (Lecture Notes in Electrical Engineering), 1252 pages.
- Kouanou, Aurelle Tchagna; Daniel Tchiotsop, Romanic Kengne, Djoufack Tansaa Zephirin, Ngo Mouelas Adele Armele and René Tchinda (2018). An optimal big data workflow for biomedical image analysis. *Informatics in Medicine Unlocked*. Volume 11, Pages 68-74.
- Lerman, Julia (2009). *Programming Entity Framework*. O'Reilly Media, Inc. 731 pages.
- Murdocca, Miles J. and Vincent P. Heuring (1999). *Principles Of Computer Architecture*. Prentice Hall. 654 pages.
- Shrivakshan, G. T. and C. Chandrasekar (2012). A Comparison of various Edge Detection Techniques used in Image Processing. *IJCSI International Journal of Computer Science Issues*, Vol. 9,

Issue 5, No 1, September 2012 ISSN (Online): 1694-0814
www.IJCSI.org.

- Tsankashvili, Nika (2018). Comparing Edge Detection Methods. <https://medium.com/@nikatsanka/comparing-edge-detection-methods-638a2919476e>.
- Yadav, Shikha; Preeti Dhanda and Nisha Yadav (2013). High Performance Computing through Parallel and Distributed Processing. Journal Of Harmonized Research in Engineering 1(2), 54-64.