



**T.C.
İSTANBUL ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**



Doktora Tezi

**BİLGİYE ERİŞİMDE KULLANILABİLİRLİĞİN YENİ BİR YAPAY
ZEKÂ YÖNTEMİYLE GELİŞTİRİLMESİ**

Veli Özcan BUDAK

Enformatik Anabilim Dalı

Enformatik Programı

**DANIŞMAN
Doç. Dr. Çiğdem SELÇUKCAN EROL**

Aralık, 2020

İSTANBUL

Bu alıřma, 25.12.2020 tarihinde ařağıdaki jüri tarafından Enformatik Anabilim Dalı, Enformatik Programında Doktora tezi olarak kabul edilmiştir.

Tez Jürisi

Do. Dr. iğdem SELUKCAN EROL(Danışman)
İstanbul Üniversitesi
Enformatik Bölümü

Prof. Dr. Sevin GÜLSEEN
İstanbul Üniversitesi
Enformatik Bölümü

Prof. Dr.Oya KALIPSIZ
Yıldız Teknik Üniversitesi
Elektrik-Elektronik Fakültesi

Do. Dr. İrfan ŞİMŞEK
İstanbul Üniversitesi-Cerrahpařa
Hasan Ali Yücel Eğitim Fakültesi

Do. Dr. Seda TOLUN TAYALI
İstanbul Üniversitesi
İřletme Fakültesi

İntihal Programı Beyanı

20.04.2016 tarihli Resmi Gazete’de yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince; Bu Lisansüstü teze, İstanbul Üniversitesi’nin aboneli olduğu intihal yazılım programı kullanılarak Fen Bilimleri Enstitüsü’nün belirlemiş olduğu ölçütlere uygun rapor alınmıştır.

Tezden Üretilmiş Yayınların Künye Bilgileri

1. Budak, V. Ö., Selçukcan Erol, Ç., 2020, Web Kullanıcılarının Bilgi Erişim ve Ziyaret Desenlerinin Web Madenciliği ile Keşfi: Kırklareli Üniversitesi Örneği, *AJIT-e: Bilişim Teknolojileri Online Dergisi*, 11(43), 37-55.



ÖNSÖZ

Beş yılı aşkın süredir birlikte çalıştığım, destek ve önerileriyle hem kişisel hem de akademik gelişimimde en büyük pay sahibi olan, benim için bir eğitimsiden çok daha fazlası olan, sorgulayıcı ve eleştirel yaklaşım tarzıyla farklı ve kapsamlı düşünebilmemde bana büyük katkılar sunan canım hocam Sayın Doç. Dr. Çiğdem Selçukcan Erol'a sadece bu tez çalışması için değil, tüm katkılarından dolayı en içten duygularıyla teşekkür ederim.

Enformatik bölümüne adım attığım ilk günden itibaren hem yüksek lisans hem de doktora eğitimim süresince bana destek olan, samimiyeti ve yol göstericiliğiyle akademik gelişimime katkılar sunan değerli hocam Sayın Prof. Dr. Sevinç Gülseçen'e,

Tez izleme sürecinde önerileriyle bu tez çalışmasına önemli katkılar sunan değerli hocam Sayın Prof. Dr. Oya Kalıpsız'a,

Lisansüstü eğitimim süresince beni hep destekleyen değerli hocam Sayın Öğr. Gör. Gökhan Doğan'a,

Enformatik bölümünü bana sevdiren, hayatım boyunca unutamayacağım bir eğitim ve gelişim ortamı sağlayan, birbirinden değerli tüm öğretim üyeleri ve çalışanlarıyla Enformatik ailesine,

TÜBİTAK 2211-C Yurt İçi Öncelikli Alanlar Doktora Burs Programı ile bu tez çalışmasını destekleyen TÜBİTAK'a,

Beni bugünlere getiren ve hep yanımda olan canım aileme, tanıdığım ilk günden itibaren neşesi ve desteğiyle beni hep motive eden canım eşim Nurhüda Budak'a ve bana destek olan tüm dostlarıma sonsuz teşekkürlerimi sunarım.

Aralık 2020

Veli Özcan BUDAK

İÇİNDEKİLER

Sayfa No

ÖNSÖZ	iv
İÇİNDEKİLER.....	v
ŞEKİL LİSTESİ	vii
TABLO LİSTESİ.....	viii
SİMGE VE KISALTMA LİSTESİ	ix
ÖZET	xi
SUMMARY	xiii
1. GİRİŞ	1
2. GENEL KISIMLAR.....	5
2.1. BİLGİ ERİŞİM.....	5
2.2. BİLGİ ERİŞİM SİSTEMİ	6
2.2.1. Bilgi Erişim Sistemi Bileşenleri	9
2.2.2. Bilgi Erişim Sistemi Çeşitleri	11
2.3. BİLGİ ERİŞİM MODELLERİ	15
2.3.1. Boolean Tam Eşleşme Modeli.....	15
2.3.2. Vektör Uzayı Modeli	17
2.4. BİLGİ ERİŞİMDE DEĞERLENDİRME	22
2.4.1. Etkililik Metrikleri.....	23
2.4.2. Verimlilik Metrikleri	28
2.4.3. Kullanıcı Merkezli Değerlendirme ve Kullanılabilirlik	29
2.4.4. Değerlendirmelerin İstatistiksel Geçerliliğinin Sağlanması	34
2.5. BİLGİ ERİŞİMDE İLGİLİLİK GERİBİLDİRİMİ VE SORGU GENİŞLETME MEKANİZMASI.....	35
2.5.1. İlgililik Geribildirimi (<i>Relevance Feedback</i>)	35
2.5.2. Sorgu Genişletme (<i>Query Expansion</i>).....	38
2.6. YAPAY ZEKÂ	39
2.7. LİTERATÜR.....	45
3. MALZEME VE YÖNTEM.....	50
3.1. ÇALIŞMA KAPSAMINDA GELİŞTİRİLEN ALGORİTMA	50
3.2. ALGORİTMANIN ENTEGRASYONU	59

3.3. ALGORİTMANIN MAKİNE ÖĞRENMESİ İÇİN VERİ TOPLAMA SÜRECİ	61
3.4. DEĞERLENDİRME.....	67
3.5. ÇALIŞMADA KULLANILAN YAZILIM VE SİSTEM ALTYAPISI	69
4. BULGULAR.....	70
4.1. GENEL BULGULAR.....	70
4.2. ETKİLİLİK BULGULARI	77
4.3. VERİMLİLİK BULGULARI	81
4.4. MEMNUNİYET BULGULARI	87
5. TARTIŞMA VE SONUÇ	88
KAYNAKLAR.....	92
EKLER	98
EK 1. Veri Kullanım İzin Dilekçesi.	98
EK 2. Veri Kullanım İzin Dilekçesi Cevabı.	99
ÖZGEÇMİŞ	100

ŞEKİL LİSTESİ

Sayfa No

Şekil 2.1: Bilgi erişimde gerçekleşen etkileşim (Göker ve Davies, 2009).....	6
Şekil 2.2: Bir BES üzerinde gerçekleştirilen işlemler (Ceri ve diğ., 2013).	8
Şekil 2.3: Temel anlık bilgi erişim işlemi (Grossman ve Frieder, 2004).	15
Şekil 2.4: “information retrieval evaluation” aramasına karşılık Yahoo ve Google arama motorları tarafından sunulan öneriler (Baeza-Yates ve Ribeiro-Neto, 2011).	31
Şekil 3.1: RPR hesaplamasına yönelik K_1 ve K_2 kümeleri kesişimi örneği.....	55
Şekil 3.2: Algoritmanın öneri sunma arayüzü.....	60
Şekil 3.3: 2 farklı web sayfalarına yapılmış olan ziyaretler ve kalma süreleri grafiği.....	63
Şekil 3.4: 2 farklı web sayfasına yapılmış olan ziyaretlerdeki kalma sürelerinin tekrarlanma miktarı grafiği.....	64
Şekil 3.5: Memnuniyet değerlendirme barı.....	68
Şekil 4.1: DED’lerin 12 haftalık süreçteki değişim grafiği.....	70
Şekil 4.2: Arama araçlarının kullanım istatistikleri.	71
Şekil 4.3: Arama araçlarının ortalama kaynak öneri miktarları.	72
Şekil 4.4: Kullanıcıların periyodik kaynak kaydetme istatistikleri.	73
Şekil 4.5: Periyodik GEM ve KOKAM istatistikleri.	77
Şekil 4.6: Algoritmanın periyodik sorgu durumlarına ait istatistik.....	86

TABLO LİSTESİ

	Sayfa No
Tablo 2.1: Örnek bir vektör uzay modeli havuzu.	17
Tablo 2.2: bit vector yaklaşımına göre oluşan vektör uzayı matrisi.	18
Tablo 2.3: inner product yaklaşımına göre oluşan vektör uzayı matrisi.	19
Tablo 2.4: Verimlilik göstergeleri (Croft ve diğ., 2009).	29
Tablo 3.1: Arama terimleri ve incelenen kaynaklar.	52
Tablo 3.2: Terim temsiliyet olasılıkları.	54
Tablo 3.3: Terimlerin ve bilgi kaynaklarının sıralanmasına yönelik örnek senaryolar.	57
Tablo 4.1: KOOR ve kural miktarlarının normallik test sonuçları.	75
Tablo 4.2: KOOR ve KOKAM normallik test sonuçları.	76
Tablo 4.3: RR değerleri üzerinden işaret testi sonuçları.	78
Tablo 4.4: DED gruplarına ait normallik test sonuçları.	79
Tablo 4.5: DED gruplarına ait etkililik karşılaştırmaları.	80
Tablo 4.6: TÖS değerleri üzerinden işaret testi sonuçları.	82
Tablo 4.7: Sorgu süresi ve önerilen kaynak miktarlarına ait normallik test sonuçları.	82
Tablo 4.8: İTS değerleri üzerinden işaret testi sonuçları.	84
Tablo 4.9: İTS ve önerilen kaynak miktarlarına ait normallik test sonuçları.	85

SİMGE VE KISALTMA LİSTESİ

Simgeler	Açıklama
d	: Yoğunluk
F	: <i>F measure</i> (F Ölçüsü)
H	: Hipotez
idf	: <i>Inverse Document Frequency</i> (Ters Doküman Frekansı)
m	: Kütle
p	: Olasılık
P	: <i>Precision</i> (Hassasiyet)
P@n	: <i>Average Precision at n</i> (n. Sıradaki Ortalama Hassasiyet)
r	: Korelasyon Katsayısı
R	: <i>Recall</i> (Geri Çağırma)
td	: Terim Yoğunluğu
tf	: <i>Term Frequency</i> (Terim Frekansı)
v	: Hacim

Kisaltmalar	Açıklama
BES	: Bilgi Erişim Sistemi
DED	: Disiplin Eşik Değeri
FIRE	: <i>Forum for Information Retrieval Evaluation</i>
GEM	: Toplam Arama Etkileşim Miktarı
GMAP	: <i>Geometric Mean Average Precision</i> (Ortalama Hassasiyetin Geometrik Ortalaması)
IQR	: Kartiller Arası Aralık
IS	: <i>Information Source</i> (Bilgi Kaynağı)
ISO	: <i>International Organization for Standardization</i> (Uluslararası Standardizasyon Örgütü)
İK	: Bilgi Kaynaklarının Sıralanmasında İzlenen Kural
İTS	: İlk Tıklama Süresi

KLU	: Kırklareli Üniversitesi
KOKAM	: Kural Oluşturmada Kullanılan Arama Miktarı
KOOR	: Kural Oluşturma Oranı
LISA	: <i>Library and Information Science Abstracts</i>
LOKT	: Laboratuvar Ortamında Kullanılabilirlik Testleri
MAP	: <i>Mean Average Precision</i> (Ortalama Hassasiyetin Aritmetik Ortalaması)
MRR	: <i>Mean Reciprocal Rank</i> (Ortalama Karşılıklı Sıra)
NDCG	: <i>Normalized Discounted Cumulative Gain</i> (Normalleştirilmiş İndirgemeli Kümülatif Kazanç)
NTCIR	: <i>NII Testbeds and Community for Information Access Research</i>
ÖK	: Önem Katsayısı
RPR	: <i>Representation Rate</i>
RR	: <i>Reciprocal Rank</i> (Karşılıklı Sıra)
RSS	: <i>Rich Site Summary</i> (Zengin Site Özeti)
SD	: Sayfa Değeri
SN	: Senaryo Numarası
SOÇ	: Sorgu Çıktısı
SOG	: Sorgu Gecikmesi
STE	: Sorgu Terk Etme
SVM	: <i>Support Vector Machines</i> (Karar Destek Makineleri)
WWW	: <i>World Wide Web</i>
TÖS	: Tekil Kaynak Öneri Süresi
TREC	: <i>Text REtrieval Conference</i>
TZ	: Tüm Web Sayfalarına Yapılan Ziyaretler
UK	: Birleşik Krallık
US	: Birleşik Devletler
YD	: Yeterlilik Değeri
YZ	: Yapay Zekâ

ÖZET

DOKTORA TEZİ

BİLGİYE ERİŞİMDE KULLANILABİLİRLİĞİN YENİ BİR YAPAY ZEKÂ YÖNTEMİYLE GELİŞTİRİLMESİ

Veli Özcan BUDAK

İstanbul Üniversitesi

Fen Bilimleri Enstitüsü

Enformatik Anabilim Dalı

Danışman : Doç. Dr. Çiğdem SELÇUKCAN EROL

Bilgi, her bireyin temel ihtiyaçlarından biridir. Günümüzde, neredeyse her birey günlük zamanın çoğunu internet üzerinde çeşitli bilgi ihtiyacını gidermek ya da bilgi alışverişi yapmak amacıyla geçirmektedir. Bilgiye erişimin çok daha kolay olduğu bu dönemde artık daha fazla odaklanılması gereken noktanın “kullanışlı bilgiye erişim” konusu olduğu düşünülmektedir.

“Kullanışlı bilgiye erişim” konusunda günümüzde en çok faydalandığımız uygulamalar Google, Yandex, vb. arama motorlarıdır. Bu uygulamalar, literatürde “Bilgi Erişim Sistemleri (BES)” adı altında değerlendirilmekte olup, bilgiye ihtiyaç duyan kişi ile bilgiyi barındıran kaynağı buluşturan bir köprü vazifesi üstlenmektedir. Çeşitli türleri olmasına rağmen, BES’lerin ortak hedefi, ihtiyacı karşılayacak bilginin ortaya çıkarılmasıdır. Bu tez çalışmasında, BES altyapısına entegre edilebilecek bir algoritma geliştirilerek, hem kullanıcıların değişken bilgi ihtiyaçlarına göre kendini güncelleyebilen hem de hâlihazırda kullanılmakta olan bir BES altyapısını daha fazla kullanıcıyla bütünleştirebilen bir ortamın sağlanmasına odaklanılmıştır. Bu doğrultuda, bilgiye erişimin etkili bir biçimde gerçekleştirilebilmesi amacıyla bir yapay zekâ algoritması geliştirilmiş ve Kırklareli Üniversitesi (KLU) bünyesinde yer alan 170 tane web sitesinin içinde çalışan site-içi arama araçlarına entegre edilmiştir. Çalışmanın amacı, kullanıcı gerçek davranış verisi üzerinden, kullanıcı dilinden bağımsız olarak bilgi erişim sürecinin iyileştirilmesi ve geliştirilen algoritmanın (Analogy) bu süreçteki başarımının değerlendirilmesidir.

Tez kapsamında geliştirilen Analogy algoritması üç aylık kullanıcı davranış verisi ile eğitilmiş, süreç sonrasında, “Bilgi Erişim” ve “Kullanılabilirlik” literatürleriyle ilişkili olacak şekilde üç farklı açıdan (etkililik, verimlilik ve memnuniyet) bilgi erişim sürecindeki katkısı analiz edilmiştir. Çalışma sonuçları, Analogy algoritmasının etkililik ve verimlilik bakımından bilgi erişim sürecine belirgin bir katkı sağladığını göstermiştir. Reciprocal Rank metriği temel alınarak gerçekleştirilmiş olan işaret testi sonuçları, etkililik bakımından algoritmanın başarısına işaret eden bulguları ortaya çıkarmıştır ($z=-2,421$; $p=0,015<0,05$). “Sorgu Gecikmesi”, “İlk Tıklama Süresi” ve “Sorgu Terk Etme” olmak üzere üç farklı göstergeye dayalı gerçekleştirilmiş olan verimlilik analizlerinde de benzer bulgular ortaya çıkartılmıştır. İşaret testinden faydalanan *Sorgu Gecikmesi* incelemelerinde, 0,05 anlamlılık düzeyinde algoritmanın kullanıcılara hızlı bir şekilde yanıt verdiğine işaret eden bulgulara rastlanmıştır ($z=-38,453$; $p=0,000<0,05$). *İlk Tıklama Süresi* analizlerinde, yine işaret testinden faydalanılmış olup, algoritmanın başarısına işaret eden bulgular ortaya çıkartılmıştır ($z=-4,936$; $p=0,000<0,05$). Tek Örneklem Ki-Kare Testi’nden faydalanılarak analizi gerçekleştirilen *Sorgu Terk Etme* göstergesindeyse, algoritmanın az sayıda kullanıcıyı kaybettiğine işaret eden bulgular ortaya çıkartılmıştır ($\chi^2= 38,407$; $p=0,000<0,05$). Diğer taraftan, memnuniyet açısına yönelik olarak algoritmayı değerlendirmek için yeterli veri elde edilememiştir. Kullanıcı dilinden bağımsız olarak, kullanıcı davranışından öğrenerek aksiyon alan Analogy algoritmasının, kullanışlı bilgiye etkili ve verimli erişim amacıyla sadece arama motorlarında değil birçok alanda kullanılma potansiyeli olduğu düşünülmektedir. Algoritmanın memnuniyet açısından da kullanılabilirliğinin değerlendirilmesi için gelecekte araştırma yapılması önerilmektedir.

Aralık 2020, 116 sayfa.

Anahtar kelimeler: Bilgi Erişim, Yapay Zekâ, Kullanılabilirlik, Algoritma, Günlük Dosyası

SUMMARY

Ph.D. THESIS

IMPROVING USABILITY IN ACCESS TO INFORMATION BY A NEW ARTIFICIAL INTELLIGENCE METHOD

Veli Özcan BUDAK

İstanbul University

Institute of Graduate Studies in Sciences

Department of Informatics

Supervisor : Assoc. Prof. Dr. Çiğdem SELÇUKCAN EROL

Information is one of the basic needs of each individual. Today, almost every individual spends most of their daily time on the internet to meet their various information needs or to exchange information. In this period that accessing information is much easier, it is thought that the point that needs to be focused more is “accessing useful information”.

The applications which we use the most about "accessing useful information" are search engines such as Google, Yandex, etc. These applications are considered under the name of "Information Retrieval Systems (IRS)" in the literature and they act as a bridge that brings the person who needs information and the source that contains the information together. Despite having various types, the common goal of IRS is to find out the information that will meet the needs. In this thesis, by developing an algorithm that can be integrated into any IRS infrastructure, we focused on providing an environment that can both update itself according to the various information needs of users and unify a currently used IRS infrastructure with more users. In this direction, for the purpose of achieving effective access to information, an artificial intelligence algorithm has been developed and has been integrated into site-in search tools operating within 170 websites belong to Kırklareli University (KLU). The aim of the study is to improve the information retrieval process independently of user language through real user behavior data and to evaluate the performance of the developed algorithm (Analogy) in this process.

The Analogy algorithm developed within the scope of the thesis was trained with three months real user behavior data, and after that process, the contribution of the algorithm to the information retrieval process was analyzed in three different aspects (effectiveness, efficiency and satisfaction) in relation to the "Information Retrieval" and "Usability" literature. The results of the study showed that the Analogy algorithm made a significant contribution to the information retrieval process in terms of effectiveness and efficiency. The sign test results, which were carried out on the basis of the Reciprocal Rank metric, revealed the findings that indicate the success of the algorithm in terms of effectiveness ($z=-2,421$; $p=0,015<0,05$). Similar findings were revealed in the efficiency analysis based on three different indicators: "Response Time", "Time to First Click" and "Query Abandonment". In the analysis of *Response Time* based on the sign test, it was found that the algorithm responded more quickly to the users ($z=-38,453$; $p=0,000<0,05$). In the *Time to First Click* analysis, the sign test was also used, the findings indicating the success of the algorithm were revealed ($z=-4,936$; $p=0,000<0,05$). In the *Query Abandonment* indicator, which was analyzed using the Single Sample Chi-Square Test, the findings indicating the algorithm only lost a small number of users were revealed ($\chi^2=38,407$; $p=0,000<0,05$). On the other hand, not enough data could be obtained to evaluate the algorithm in terms of satisfaction. It is thought that the Analogy algorithm, which takes action by learning from real user behavior regardless of user language, has the potential to be used not only in search engines but in many areas for effective and efficient access to useful information. Future researches are recommended to evaluate the usability of the algorithm in terms of satisfaction.

December, 2020, 116 pages.

Keywords: Information Retrieval, Artificial Intelligence, Usability, Algorithm, Web Access Log

1. GİRİŞ

Türk Dil Kurumu (2020a) tarafından “insan zekâsının çalışması sonucu ortaya çıkan düşünce ürünü” şeklinde tanımlanmış olan bilgi, çeşitli şekillerde geçmişten günümüze kadar artarak gelen ve neredeyse tüm yaşamımızı yönlendiren önemli bir kavramdır. İngilizce’de “information (enformasyon)” ve “knowledge (bilgi)” olmak üzere iki farklı şekilde literatürde yer alan bu kavram, dilimizde çoğunlukla sadece “bilgi” olarak (enformasyona karşılık gelen) tek bir kelimeyle kullanılmaktadır. Enformasyon, yapılandırılmış/organize edilmiş ham veri olarak tanımlanırken, bilgi ise, enformasyon, anlayış, yetenek, deneyim ve değer kavramlarını karışık şekilde içeren bir kavram olarak ifade edilmektedir (Rowley, 2007). Dilimizde “bilgi erişim” olarak yer almış olan “information retrieval” kavramıyla ilgili literatür incelendiğinde, bilgi kelimesinin kavramsal tanımından ziyade bilgi erişim kavramının bütün olarak ne anlama geldiği ve bu kavramın içinde yer alan bileşenlerin aktarılmaya çalışıldığı tespit edilmiştir. Türkçe literatürde “information retrieval” kavramının çevirisi “bilgi erişim” olarak yaygın kullanılmasından dolayı bu çalışmada da enformasyon değil bilgi olarak yer verilmiştir.

İlk kayıtları milattan önce 2. yüzyılda Yunan parşömenlerinde keşfedilmiş olan bilgi kavramı (Ceri ve diğ., 2013), 21. yüzyılın başında olduğumuz bu dönemde teknolojiyle birlikte sürekli olarak genişleyen bir evrene dönüşmüştür. Bu evren, yeni bir bilginin üretilmesi veya var olan bir bilginin güncellenerek yenilenmesi sonrasında çeşitli yollarla paylaşarak büyümesini sürdürmektedir. Bilginin paylaşılması ve hızlı erişilebilir olması noktasında görev yapan günümüzün en önemli araçlarından biri (belki de ilki) internet’tir. Kowalski (2011), internet’in bir kişinin ihtiyaç duyabileceği herhangi bir bilginin deposu haline geldiğini ve daha rahat bir araştırma aracı olarak kütüphanelerin yerine geçtiğini ifade etmiştir. Teknolojinin en güçlü araçlarından biri olan bu dijital platform, bilginin paylaşılma ve yayılma hızını takip edilemez bir seviyeye ulaştırmıştır. 1990 yılında bir tarayıcı aracılığıyla web dokümanlarına ulaşmayı sağlayan World Wide Web (WWW)’in Tim Berners-Lee tarafından icat edilmesinin, bilgi evreninin hızla büyümesinde görev yapan en önemli unsurlardan biri olduğu belirtilebilir. Ryan (2010)’ın WWW’yi “insanlık iletişim tarihindeki en önemli gelişmelerden biri” olarak nitelendirmesi de bu icada yönelik dikkat çekici bir değerlendirmedir.

Bilgi miktarının sürekli olarak artış göstermesi, üzerinde durulması gereken bir dezavantajı da ortaya çıkarmaktadır. Bu dezavantaj, aşırı bilgi çokluğudur. Metzler (2011) bu durumu, “bilgisayar maliyetlerinin düşmesi ve çevrimiçi popülerliğin artması doğrultusunda, internet’te mevcut olan bilgi miktarının patlaması” şeklinde tanımlamaktadır. Bilgi boyut ve çeşitliliğinin artmasıyla birlikte asıl ihtiyaç duyulan, konuya ya da duruma özel bilgiye erişme eyleminin giderek zorlaştığı belirtilebilir. Bu durum, gerekli bilgiye erişme işleminin “samanlıkta iğne arama” biçimine dönüşmesi şeklinde de ifade edilebilir. Benzer şekilde, Van Rijsbergen (1979), doğru ve hızlı erişimin daha da zorlaştığı çok fazla bilgiye sahip olduğumuzu söylemiştir. Bilgi gereksiniminin karşılanabileceği çok fazla yol olmasına rağmen, bu yollardan hangisinin seçileceği ve seçilen yolda ihtiyacı karşılama potansiyeli olan bilgiden hangisinin değerlendirileceğinin kararını vermek oldukça zor bir sürece dönüşmüştür. Dolayısıyla, teknoloji çağında yaşadığımız bu dönemde “kullanışlı bilgiye erişim” temalı yapılacak olan çalışmaların oldukça önemli olduğu düşünülmektedir.

Günümüzde, farklı amaçlar doğrultusunda ihtiyaç duyulan bilgiye erişmek amacıyla arama motorları kullanılmaktadır. Farklı altyapılarla hizmet veren çeşitli arama motorları (Google, Yandex vb.) biz fark etmesek de günlük yaşamımızın önemli bir bölümünde yer almaktadır. Temel olarak, faydalı bilgiye erişmek amacıyla kullandığımız bu tür uygulamalar, literatürde “Bilgi Erişim Sistemleri (BES)” adıyla ifade edilmekte olup, bu çalışmanın da odağında yer almaktadır. BES’ler farklı işlevsellikte tasarım ve hizmet şekillerine sahip olmalarına rağmen, en önemli ortak yönlerinin “kullanışlı bilgiye erişim” olduğu rahatlıkla ifade edilebilir.

Bu tez çalışmasında, herhangi bir BES altyapısına entegre edilebilen, kullanıcı davranış verisinden öğrenen ve kullanıcı dilinden bağımsız olarak aksiyon alan, Analogy algoritması geliştirilerek (Analogy algoritması için 09.07.2019 tarihinde 2019/10260 başvuru numarası ve “Analogy: Yoğunluk Tabanlı Bir Bilgi Erişim Optimizasyon Yöntemi” başlığıyla patent başvurusu da yapılmıştır), hem kullanıcıların değişken bilgi ihtiyaçlarına göre kendini güncelleyebilen hem de hâlihazırda kullanılmakta olan bir BES altyapısını daha fazla kullanıcıyla bütünleştirebilen bir ortamın sağlanmasına odaklanılmıştır. Bu doğrultuda, Kırklareli Üniversitesi (KLU) bünyesinde yer alan 170 tane web sitesinin içinde entegre çalışan site-içi arama araçlarından faydalanılmıştır. Bu araçlar, çalışmanın 2. bölümü olan “GENEL KISIMLAR” bölümünde açıklanmış olan BES’lerle benzer bir altyapıya sahip olan araçlardır. Çalışmanın amacı, kullanıcı gerçek davranış verisi üzerinden, kullanıcı dilinden bağımsız

olarak bilgi erişim sürecinin iyileştirilmesi ve geliştirilen algoritmanın (Analogy) bu süreçteki başarımının değerlendirilmesidir. Geliştirilmiş olan algoritma, halihâzırda çalışmakta olan site-içi arama araçlarına entegre edilmiş ve var olan yapının performans olarak ne derece geliştirilebildiği incelenmiştir. Performans değerlendirmeleri genel olarak “Kullanılabilirlik” kavramı bağlamında gerçekleştirilmiş olup, “Bilgi Erişim”de kullanılan performans ölçütleriyle ilişkili olacak şekilde araştırma sonuçları paylaşılmıştır.

Tezin genel kısımlar bölümünde, “Bilgi Erişim” kavramına odaklanılarak BES’in ortaya çıkışından itibaren günümüzdeki kullanım alanlarına yönelik literatür paylaşılmıştır. Bununla birlikte, BES’in çalışma altyapısı detaylı bir şekilde aktarılmış ve performans açısından nasıl değerlendirildiğine yönelik çeşitli yöntemler sunulmuştur. Ek olarak, performans değerlendirmeleri noktasında “Kullanılabilirlik” kavramı bakımından yararlanılan değerlendirme yöntemlerine de yer verilmiştir. Yine aynı bölümde, tezin bir diğer odak noktası olan “Yapay Zekâ” kavramının tanımı yapılmaya çalışılmış ve bu tez çalışmasındaki “Yapay Zekâ” vurgusunun ne anlama geldiği ifade edilmiştir. Bu bölümde, son olarak, literatürde bu tez çalışmasıyla ilgili olduğu düşünülen çeşitli araştırmalara yer verilmiştir.

Malzeme ve yöntem bölümünde, tez kapsamında geliştirilmiş olan algoritmanın çalışma mekanizması açıklanmış, algoritmanın, çalışan bir BES üzerindeki entegrasyon süreci paylaşılmış ve yapay zekâ olarak hareket edebilmesi için nasıl bir yöntem izlendiği aktarılmıştır. Ek olarak, algoritmanın veri toplama süreci sonrasındaki başarımının nasıl değerlendirildiğine yönelik faydalanılan performans ölçütleri açıklanmış ve bu algoritmanın geliştirilmesinde ve veri toplama sürecinde görev yapan tüm sistem altyapısı hakkında bilgi verilmiştir.

Tezin bulgular bölümünde, “Kullanılabilirlik” ve “Bilgi Erişim” kavramlarına bağlı kalınarak “etkililik, verimlilik ve memnuniyet” bağlamında araştırma sonuçları paylaşılmıştır. Bununla birlikte, veri toplama sürecinde ortaya çıkmış olan genel kullanım istatistikleri aktarılmıştır. Ayrıca, ortaya çıkarılan tüm sonuçların anlamlılığının belirlenebilmesi amacıyla, çeşitli hipotez testleri doğrultusunda ortaya çıkarılmış olan bulgulara yer verilmiştir.

Tartışma ve sonuç bölümünde, algoritmanın ortaya çıkardığı sonuçlar değerlendirilerek, BES’lerin geliştirilmesinde ne tür faydalar sağlayabileceğine değinilmiştir. Bununla birlikte,

algoritmanın performans ölçümlerinde ortaya çıkarılmış olan ancak yine de daha fazla çalışmaya ihtiyaç duyulan noktalar paylaşılmıştır.

Tezin son bölümlerinde ise, çalışma boyunca yararlanılan kaynaklar ve ekler aktarılmıştır.

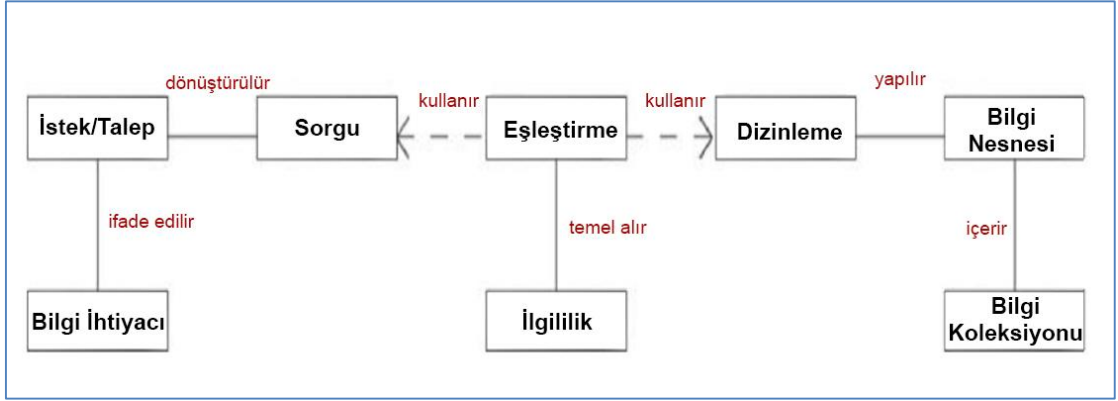


2. GENEL KISIMLAR

Çalışmanın bu bölümünde, “Bilgi Erişim” kavramının ne anlama geldiği ifade edilmeye çalışılmış ve bu kavram çerçevesinde ortaya çıkarılmış olan “Bilgi Erişim Sistemleri (BES)”ne değinilmiştir. Bununla birlikte, BES’in altyapısı ile ilgili bilgiler aktarılmış olup, bu sistemlerin performans değerlendirmelerinde faydalanan çeşitli yöntemler paylaşılmıştır. Ek olarak, bu çalışmanın, “Yapay Zekâ” kavramıyla olan ilişkisi açıklanmaya çalışılarak, bu çalışmaya ilgili literatürde yapılmış olan çeşitli araştırmalara yer verilmiştir.

2.1. BİLGİ ERİŞİM

Bilgi erişim (*Information Retrieval*), Britannica Concise Encyclopedia (2007a) tarafından “özellikle bir bilgisayarda yer alan veri tabanından bilginin elde edilmesi”, Türk Dil Kurumu (2020b) tarafından da “bellekte saklı verilerden belli bir konuda bilgi alma yöntem ve yordamları” şeklinde tanımlanmıştır. Bilgi ihtiyacını daha belirginleştirecek şekilde “kullanıcıların ilgi duydukları bilgiye hızlı erişimlerine odaklanmış bir alan” olarak tanımlanmaktadır (Baeza-Yates ve Ribeiro-Neto, 2011). Benzer olarak, “büyük koleksiyonlardan (genellikle bilgisayarlarda depolanan) bir bilgi ihtiyacını karşılayan yapılandırılmamış bir nitelikteki (genellikle metin) materyalin (genellikle belgeler) bulunması” şeklinde de tanımlanmaktadır (Manning ve diğ., 2008). Bilgi erişim, bilgi ihtiyacı, arama ve erişim başlıklarını kapsamaktadır (Göker ve Davies, 2009). Bilgi erişimde amaç, kullanıcının işine yarayacak/ilgili bilginin elde edilmesidir (Ceri ve diğ., 2013). Çok çeşitli bilgi türleri üzerine yapılan çalışmaları ve bilginin araştırılmasıyla ilgili olan çeşitli uygulamaları kapsayan (Croft ve diğ., 2009) ve bilginin yapısı, analizi, düzenlenmesi, depolanması, araştırılması ve elde edilmesiyle ilgilenen (Salton, 1968; aktaran Croft ve diğ., 2009) bir alandır. Benzer şekilde, Larson (2011) da, bilgi koleksiyonunun depolanması, organize edilmesi ve araştırılmasıyla ilgilendiğini belirtmiştir. Göker ve Davies (2009) tarafından “dizinlenmiş olan bilgi nesnesinin bir sorguya karşılık eşleştirilmesi süreci” olarak daha özelleştirilmiş çerçevede tanımlanmış olan bu kavram, Şekil 2.1’deki biçimde ifade edilmektedir.



Şekil 2.1: Bilgi erişimde gerçekleşen etkileşim (Göker ve Davies, 2009).

Şekil 2.1'e göre, bilgi erişimde bilgi ihtiyacı olan taraf ve ihtiyacın giderilebileceği bir bilgi koleksiyonu olmak üzere iki taraf bulunmaktadır. Bilgi ihtiyacı, bir istek olarak ifade edilir ve araştırılmak için bir sorgu ifadesine çevrilmektedir. Buna karşılık, ihtiyacı giderebilecek bilgiyi içeren bir bilgi koleksiyonu bulunmaktadır. Gerekli olan bilginin bulunabilmesi için bu bilginin bir şekilde temsil edilmesi ve dizinlenmesi gerekmektedir. Bu noktada, kişinin gereksinimiyle, dizinlenmiş bilgi nesnesinin ilgili olma durumlarına göre iyi bir şekilde eşleştirmelerinin sağlanması, bilgiye erişimde ortaya çıkan zorluk olarak ifade edilmiştir (Göker ve Davies, 2009).

2.2. BİLGİ ERİŞİM SİSTEMİ

Kowalski (2011), BES'i, "bilgiyi yutan, araştırılabilir bir biçime dönüştüren ve kullanıcının bilgi aramasına ve erişimine imkân verecek bir arayüz sağlayan sistem" olarak tanımlamıştır. Dokümanlardaki (çoğunlukla metin içeren fakat aynı zamanla çoklu ortam verisi de içerebilen) bilgiyi depolayan ve yöneten yazılım programıdır (Göker ve Davies, 2009). Bilgiye erişimi bir teknoloji olarak nitelendiren Metzler (2011), BES teknolojisinin ilk müşterilerinin çoğunun hükümet araştırmacıları, bilim insanları ve kütüphaneciler olduğunu belirtmiştir. Bu durumun değişmeye başlamasının ise World Wide Web (WWW)'in icadı sonrası olduğu ifade edilmiştir (Metzler, 2011). Kütüphaneler bilgiye erişimde BES'i kullanan ilk kurumlar olmakla birlikte (Baeza-Yates ve Ribeiro-Neto, 2011), bu BES'ler, kütüphanelerde el ile yapılan dizinleme işleminin otomatikleştirilmesine odaklanmıştır (Kowalski, 2011; Latha, 2018). Bu işlem ise genellikle yazar ismi ya da başlıklar üzerinden gerçekleştirilen kart katalogu aramaları şeklinde olup, BES'in ilk nesli olarak nitelendirilmiştir (Baeza-Yates ve Ribeiro-Neto, 2011). Bilgisayarların ortaya çıkmasıyla birlikte hızlı ve zeki BES'lerin tasarlanmasına yönelik

yoğunlaşma yaşanmıştır (Van Rijsbergen, 1979). BES'in evrimleşmesinde, bilgisayar işlemcilerinin gücünde yaşanan gelişmeler de etkili olmuştur (Kowalski, 2011; Latha, 2018). Bu gelişmeler, konu başlıkları, anahtar kelimeler ve sorgu operatörlerini içeren güçlendirilmiş arama işlevselliğine sahip ikinci nesil BES'leri, son olaraksa günümüzde kullandığımız, açık sistem mimarisine, gelişmiş arayüz ve hipertext özelliklere odaklanan üçüncü nesil BES'leri ortaya çıkarmıştır (Baeza-Yates ve Ribeiro-Neto, 2011).

BES'in amacı, bir kullanıcının ihtiyaç duyduğu değerli bilgiye ulaşmasına yönelik olarak bu işlem için harcanması gereken sürenin/emeğin azaltılmasıdır (Kowalski, 2011; Latha, 2018). Büyük bir öge koleksiyonu içerisinde bilgi ihtiyacı olan kullanıcı için ilgili olması beklenen öğelerin (metin, resim, video, vb.) seçilmesidir (Larson, 2011). Ayrıca, kullanıcının isteğini karşılamaya yönelik gerekli bilginin sağlanması olmakla birlikte, konuya özel tüm bilgiyi bulmak değildir (Kowalski, 2011). Hangi eylem bilgi ihtiyacını başlatmış olursa olsun, kullanıcının ilerlemesine yardımcı olacak yeterli bilgiyi sağlamaktır (Kowalski, 2011).

Bir BES'in desteklemek zorunda olduğu üç temel işlem bulunmaktadır. Bunlar aşağıdaki şekilde belirtilebilir (Göker ve Davies, 2009):

1. *Kaynakların içeriğinin betimlenmesi:* dizinleme işlemi olarak da ifade edilmektedir. Bu işlem, kullanıcının dâhil olmadığı bir biçimde gerçekleştirilir. Çoğunlukla belgeler parçalar halinde (sadece başlık, özet ve asıl belgenin konumu ile ilgili bir bilgi) depolanır, bazen de asıl belgenin depolanmasını içerebilir. Bu işlemin en basit hali, her bir metnin içinde yer alan kelimelerin çıkartılması ve bu kelimelerin hangi metne ait olduğunu belirten bir gösterge ile birlikte bir dizin değeriyle depolanması şeklinde gerçekleştirilmektedir (Larson, 2011).
2. *Kullanıcının bilgi ihtiyacının betimlenmesi:* çoğunlukla sorgu formülleştirme olarak ifade edilmektedir. Uygun bir sorgulamaya ve kullanıcı ihtiyaçlarının daha iyi anlaşılmasına önderlik eden sistem ve kullanıcı arasındaki etkileşimli iletişim olarak değerlendirilmektedir. Buradaki “betimleme” kavramı kullanıcının yaptığı sorgulama işlemidir. Yani; bir kullanıcının ihtiyacını tanımlayan bir terimle (kelime ya da kelimeler) BES üzerinde sorgulama yapmasıdır. Kullanıcı, bilgi ihtiyacını betimleyen bir terimle yaptığı sorgulama sonucunda karşısına getirilen kaynakları yeterli görmediği

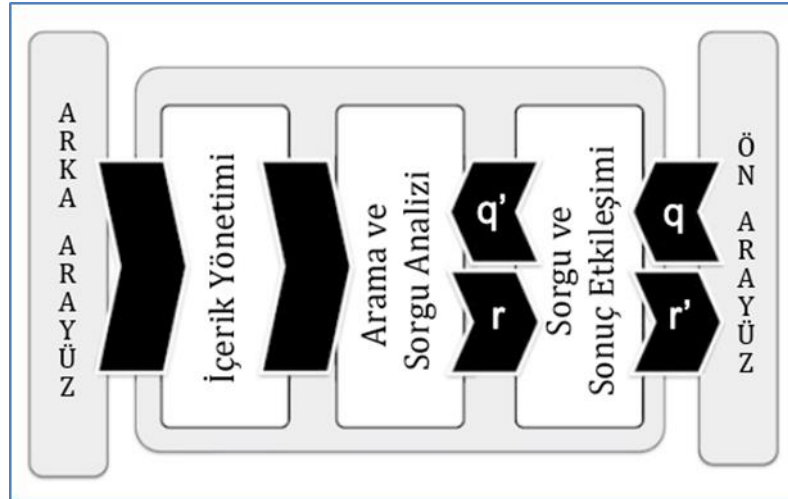
takdirde, betimleme şeklini değiştirebilir. BES ise, değişen bu betimleye yönelik olarak yeni bilgi kaynaklarını listeleyebilir.

3. *Her iki betimlemenin eşleştirilmesi*: kullanıcının yaptığı sorgulamayla, dizinlenmiş olan bilgi kaynaklarının eşleştirilerek, en uygun kaynakların kullanıcıya sunulması anlamına gelmektedir. Kullanıcı, karşısına getirilen listede yukarıdan aşağıya doğru tarama yaparak ihtiyacına uygun olan kaynağı bulmaya çalışacaktır.

Göker ve Davies (2009)’in 3 farklı başlıkta ifade ettiği BES işlemleri, Larson (2011) tarafından “dizinleme” ve “erişim” olmak üzere iki şekilde nitelendirilmekte olup, bu kavramların açıklamaları aşağıdaki gibidir:

1. *Dizinleme*: belge formundaki girdilerin alınması, bilgisinin çıkartılıp kullanıcı aramalarıyla eşleşebilmesine yönelik olarak hızlıca erişilebilecek bir formatta depolanması
2. *Erişim*: kullanıcı sorgulamalarının alınması ve depolanmış olan bilgiyle karşılaştırılabilecek bir formata dönüştürülmesi

Ceri ve diğ. (2013) ise, BES üzerinde gerçekleşen işlemleri Şekil 2.2’deki gibi görselleştirmiştir:



Şekil 2.2: Bir BES üzerinde gerçekleştirilen işlemler (Ceri ve diğ., 2013).

Şekil 2.2’ye göre, BES üzerindeki kullanıcı arayüzü (tarayıcı vb. bir arayüz) üzerinden bir sorgu (q) ile başlatılan işlemler, sorgu etkileşim modülü sayesinde kullanıcı sorgusunun bilgisayarın

anlayabileceği bir biçime (q') dönüştürülmesiyle devam etmektedir. q' sorgusunun, arama ve sorgu analizi modülünü beslemesi sonrasında BES sisteminin bağlı olduğu içerik yönetimine bağlantı sağlanmaktadır. İçerik yönetimi daha sonra arka planda yer alan veritabanı vb. bir bilgi koleksiyonuyla iletişim kurmaktadır. Sonuçlar, arama modülü sayesinde hazır olur olmaz kullanıcının karşısına sonuç etkileşim modülü ile listelenmektedir. İsteğe bağlı olarak, sonuç, kullanıcı tamamen tatmin olana kadar değiştirilmekte veya güncellenmektedir (r').

Genel itibariyle özetlemek gerekirse; BES, belirli bir konuya ya da duruma özel bilgi ihtiyacını karşılayabilecek kaynakları organize edip depolayan, bilgi ihtiyacını giderebilmek için kullanıcıyla bir arayüz üzerinden etkileşime giren ve kullanıcıyla ihtiyacı giderebilme potansiyeli olan bilgi kaynaklarını ortak bir çerçevede buluşturabilen sistem olarak tanımlanabilir.

2.2.1. Bilgi Erişim Sistemi Bileşenleri

Web üzerinde çeşitli amaçlarla dolaşan kullanıcıların tamamı birbirinden farklı bilgi ihtiyacına sahip olabilir. Benzer şekilde, web arama motorlarından faydalanan kullanıcıların da değişken karmaşıklıkta bilgi ihtiyaçları bulunmaktadır (Baeza-Yates ve Ribeiro-Neto, 2011). Bu bilgi ihtiyacı, bir kişinin arama motoruna gönderdiği sorgunun temel nedeni olarak ifade edilmiştir (Croft ve diğ., 2009). Çeşitli türlerde hizmet veren BES'lerin bu ihtiyacı iyi tespit edebilmesi ve buna karşılık olarak kullanıcının ihtiyacını mümkün olduğunca yüksek seviyede giderebilmesi önemlidir. Kowalski (2011), bu noktada yaşanan zorluğu kullanıcının dili ile bilgi kaynağı üreticisinin dili arasındaki uyumsuzluk olarak ifade etmiştir. Yani; çoğu durumda hem bilgi kaynağı üreticisinin hem de kullanıcının aynı kelimeleri bilmesine rağmen, aynı kavramı temsil eden terimlerin çoğunlukla iki taraf arasında farklılık taşıyacağı belirtilmiştir (Kowalski, 2011). Bu zorluğu BES penceresinde detaylı bir şekilde ifade etmek gerekirse, şu örnek verilebilir; bir kullanıcının, ihtiyaç duyduğu bilgiyi araştırırken, bu bilginin hangi arama terimiyle ilişkili olabileceğini düşünerek sorgulama yapacağı terimi belirlemesi gerekir. Bu eylem, kullanıcının bir düşünsel çabayı ortaya koyması anlamına gelmektedir. Kullanıcının erişmek istediği bilgiyi nasıl bir terimle sorgulayabileceğini bilmemesi, farklı kullanıcıların aynı terimi kullanmasına rağmen farklı bilgiye erişmek istemeleri ya da belki de aynı kullanıcının geçmişte kullandığı sorgu terimiyle gelecekte yapacağı sorgulamalarda farklı bir bilgiye erişme düşüncesinde olabileceği gibi çeşitli durumlar bu çabalara örnek gösterilebilir. Diğer taraftan, kullanıcının bir bilgiye erişmeye çalışırken kullandığı terimle ne tür bir bilgiye

erişmeye çalıştığının anlaşılması ve bu doğrultuda uygun bilgi kaynaklarının kullanıcıya sunulması da BES’ler tarafından ortaya konulması gereken çabaya işaret etmektedir.

BES’in ilk kullanıcılarından olduğu daha önce belirtilmiş olan kütüphanelerde, kullanıcılar bilgiye erişmede sorun yaşadıkları zaman kütüphane yetkililerinin yönlendirilmesi sayesinde bu sorun aşılabilmekteydi (Nordlie, 1999). Bu yetkililer, “arabulucu” olarak da ifade edilmiştir (Nordlie, 1999; Kowalski, 2011). Kütüphane ya da belirli sınırlar içindeki bu çözümün, WWW penceresinden bakıldığında uygulanabilirliğinin imkânsız olduğu söylenebilir. Her ne kadar bilgiye erişimde kullanıcıların ihtiyaçlarını iyi ifade edebilmeleri önemli olsa da, arabulucusuz bu dijital platformda kullanıcıyı anlama ve ihtiyacı giderme noktasında BES’lerin oldukça kritik görevler üstlendiği rahatlıkla belirtilebilir. Bu görevin başarılı gerçekleştirilebilmesi için BES altyapısında yer alan çeşitli bileşenlerin literatürde farklı başlıklarla ifade edildiği görülmüştür. Croft ve diğ. (2009), “Dizinleme Süreci (*Indexing Process*)” ve “Sorgu Süreci (*Query Process*)”, Baeza-Yates ve Ribeiro-Neto (2011) ise “Bilgi Kaynağı Tarama Süreci (*Crawling Process*)”, “Dizinleme Süreci (*Indexing Process*)” ve “Erişim ve Sıralama Süreci (*Retrieval & Ranking Process*)” olmak üzere farklı süreçlerin BES’in bileşeninde yer aldığını belirtmiştir. Diğer taraftan, Melucci (2015) tarafından “Dizinleme (*Indexing*)” ve “Erişim (*Retrieval*)”, Latha (2018) tarafından da “Belge/Bilgi Kaynağı İşleme (*Document Processing*)”, “Sorgu İşleme (*Query Processing*)” ve “Bilgi Erişim Geri Bildirimi (*Retrieval Feedback*)” olmak üzere farklı işlemlerin BES’in bileşenlerinde yer aldığı ifade edilmiştir.

“Dizinleme” ve “Erişim” olmak üzere iki kavram olarak düşünüldüğünde, *Dizinleme*, aramayı mümkün kılan yapıların oluşturulması, *Erişim* ise, bu oluşturulan yapıların ve kullanıcı sorgularının kullanılarak ilgili bilgi kaynaklarının ortaya çıkarılması şeklinde tanımlanmıştır (Croft ve diğ., 2009). Daha detaylı bir şekilde açıklamak amacıyla *Dizinleme* ve *Erişim* kavramları aşağıdaki şekilde kendi içinde farklı bileşenlere ayrılabilir (Croft ve diğ., 2009):

- **Dizinleme**

- *Metin edinimi*: aramalarda faydalanılacak bilgi kaynaklarının belirlenmesidir. Kimi zaman kullanılan sistemde var olan doküman koleksiyonu kullanılırken, kimi zaman da web’in, intranet ya da diğer dış bilgi kaynaklarının taranması ile oluşturulan koleksiyonlar kullanılmaktadır.

- *Metin dönüştürümü:* kullanılan koleksiyondaki bilgi kaynaklarından, aramalarda kullanılacak olan terimlerin oluşturulma sürecidir. En basit haliyle, bir dokümanda yer alan kelimelerin ayrıştırılması buna örnek gösterilebilir.
- *Dizin oluşturumu:* metin dönüşümü sonrasında oluşturulan çıktının, hızlı bir aramaya olanak sağlayacak şekilde dizinlenmesi/depolanmasıdır.
- **Erişim**
 - *Kullanıcı etkileşimi:* arama yapan kişi ile BES arasında etkileşim kurmaya yarayan arayüzü ifade etmektedir. Bu arayüzün, kullanıcı sorgusunu alıp, arama yapılabilir terime dönüştürerek aramayı gerçekleştirme ve arama sonrası sonuçları kullanıcıya sunma olmak üzere iki görevi bulunmaktadır.
 - *Sıralama:* BES'in çekirdeğini oluşturmaktadır. *Kullanıcı etkileşimi*'nde dönüşümü yapılan arama terimiyle, çalışmanın 2.3 bölümünde de değinilmiş olan bilgi erişim modeline göre (sadece iki tane model aktarılmış olup, farklı modeller de bulunmaktadır) ilgili bilgi kaynaklarının ortaya çıkarılmasıdır.
 - *Değerlendirme:* BES'in etkililik ve verimlilik bakımından değerlendirilmesi süreci olup, genellikle çevrimdışı bir şekilde yapılmaktadır. BES'ler için oldukça kritik bir noktada yer almaktadır.

2.2.2. Bilgi Erişim Sistemi Çeşitleri

Hitap ettiği kullanıcı kitlesinin çeşitli bilgi ihtiyaçlarını karşılamak amacıyla hizmet vermekte olan BES'lere, günümüzde sıklıkla faydalanmakta olduğumuz Google, Yahoo, Bing, vb. arama motorları en bilinen örnekler olarak verilebilir. Bununla birlikte, çoğu üniversitelerin ve kütüphanelerin kitaplara, dergilere ve belgelere erişim sağlamak amacıyla kullandıkları sistemler (Latha, 2018) de BES'lere örnek verilebilir. BES olarak değerlendirilmekte olan sistem çeşitlerinden bazıları aşağıda açıklanmıştır:

1. *Web Arama (Web Search) Motorları:* Kullanıcıların web üzerinden ihtiyaçlarını giderebilmelerine yardımcı olan arama motorları, herkes tarafından görülebilen en önemli bilgi erişim teknolojilerinden birisidir (Latha, 2018). Farklı nitelikteki bilgiye hızlı bir şekilde erişim imkânını arttıran önemli bir BES çeşididir. Bilgi ihtiyacı bulunan kişi, arama motoru arayüzlerindeki metin yazma kısmına ihtiyacını tanımlayan terimleri girerek sorgulamalar gerçekleştirebilmekte ve bu sorgulamalar doğrultusunda ihtiyacına cevap olabilecek en uygun sonuçlara ulaşabilmektedir. Bu sistemin

kullanıcıları, kısa sorgulamalarına (birkaç kelimedenden oluşan terimler) karşılık olarak ihtiyaçlarına doğru ve hızlı cevap beklemektedir (Büttcher ve diğ., 2010). Kullanıcı tarafında bu kadar basit bir işlem yapılmasına rağmen, arka planda bilgi ihtiyacını giderebilecek web sayfalarının sıralamasını yapan çok fazla bilgisayar koordineli bir şekilde çalışmaktadır (Büttcher ve diğ., 2010). Aynı zamanda bu sistemler, kullanıcıların bilgi ihtiyaçlarına cevap verebilmek için “Web Tarayıcısı (*Crawler*)” adı verilen bir yapı kullanırlar. Bu yapı, ihtiyaca karşılık olarak doğru sonuçların sunulması için web’de yer alan sayfaların belirli zaman aralıklarında kopyasını alıp, kayıt altında tutmaktadır (Büttcher ve diğ., 2010).

2. *Masaüstü ve Dosya Sistemi Arama (Desktop and File System Search) Araçları*: Yerel bir hard disk üzerinde ya da belki de yerel bir internet ağı üzerinden ulaşılabilen diskler üzerinde bulunan dosyalara erişmeye yarayan uygulamalardır (Büttcher ve diğ., 2010; Latha, 2018). Web arama motorlarının aksine, dosya format ve üretilme sürelerine yönelik olarak farkındalık bakımından daha hassas çalışan bir yapıya sahiplerdir (Büttcher ve diğ., 2010; Latha, 2018).
3. *Dijital Kütüphaneler (Digital Libraries)*: Kaliteli ve çoğu zaman özel mülkiyete sahip materyal koleksiyonuna erişim amacıyla kullanılan bir BES çeşididir (Büttcher ve diğ., 2010). Bu koleksiyon, telif hakkı sebebiyle web siteleri üzerinde barındırılmayan makaleler, kitaplar, vb. kaynakları içerir (Latha, 2018). Yayıncı kalitesi ve sınırlı kapsamı göz önüne alındığında, arama isteğini daraltıp, bilgi erişim verimliliğini arttırmak için yapısal özelliklerinin (yazar, başlık, tarih ve diğer yayın verisi) avantajından faydalanmak çoğunlukla mümkündür (Büttcher ve diğ., 2010).
4. *Seçme Bilgi Yayımları (Selective Dissemination of Information; SDI) Sistemleri*: Arama sistemleri “çekme (pull)”, dağıtım sistemleri ise “itme (push)” sistemleri olarak ifade edilmektedir (Meadow ve diğ., 2007; Kowalski, 2011). Arama sistemlerinde bir kullanıcı, ihtiyacını giderebilmek için sisteme sorgu yöneltirken; dağıtım sistemlerindeyse kullanıcı kendine bir profil (arama yapılan terimlerin bir arada kayıtlı tutulmasına benzetilmektedir) tanımlamaktadır (Kowalski, 2011). Büttcher ve diğ. (2010) bu davranışı bilgi erişim sürecinin tersten işletilmesi şeklinde ifade etmiştir. Latha (2018) tarafından dokümantasyon izleme araçları olarak tanımlanmış olan bu sistem, yeni bir bilgi eklendiğinde otomatik olarak kullanıcının profiliyle karşılaştırma

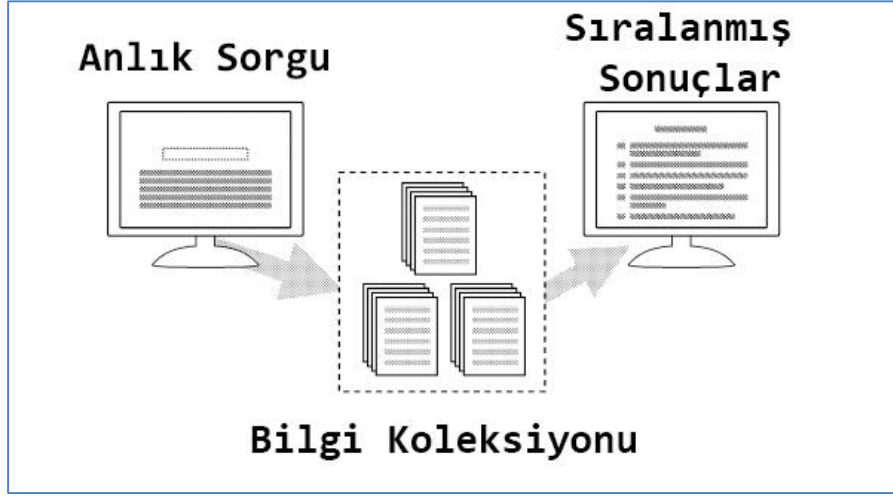
yapmaktadır (Kowalski, 2011). Eşleşme bulunduğundaysa, kullanıcıyı bilgilendirmektedir. Bu bilgilendirmeler, e-posta, sesli mesaj, anlık mesajlaşma, cep telefonu mesajı ve Zengin Site Özeti (Rich Site Summary - RSS) beslemeleri gibi çeşitli şekillerde yapılabilmektedir (Latha, 2018). Genel olarak bu sistemler, kullanıcı profiline bağlı kalarak yeni bir bilgi kaynağı ortaya çıktığında kullanıcıyı bilgilendiren otomatik arama sistemleri şeklinde ifade edilebilir. Web arama motorlarında sonuçların gösterilmesi esnasında bir ürünün reklamının sayfada yer alması da buna örnek gösterilebilir (Zhai ve Massung, 2016).

5. *Bilgi Filtreleme (Information Filtering) Sistemleri*: Bir kullanıcıya sunulmadan önce otomatik/yarı otomatik veya bilgisayarlı yaklaşımları kullanarak gereksiz bilgileri bilgi akışından kaldıran sistemlerdir (Latha, 2018). Büttcher ve diğ. (2010) tarafından “tersten işletilen bilgi erişim süreci” olarak ifade edilmiş diğer bir BES çeşididir. Bu sistemlere, spam filtreleme sistemleri örnek gösterilebilir.
6. *Öneri Sistemleri (Recommender Systems)*: Latha (2018), bu sistemleri, kullanıcıların ilgisini çekebilecek bilgi öğelerini kullanıcıya sunan bilgi filtreleme sistemleri olarak tanımlamıştır. Bu sistemler, kullanıcıya iletilen bilgi akışından bilgi öğelerini çıkarmak yerine, bu akışa yeni öğeler ekleyen sistemlerdir (Latha, 2018). Bu sistemlerde yapılan analizlerin genellikle kullanıcılar ve öğeler arasındaki geçmiş etkileşimlere dayanmakta olduğu, çünkü geçmiş eğilim ve ilgilerin gelecek tercihlerinin iyi bir göstergesi olduğu belirtilmiştir (Aggarwal, 2016).
7. *Uzman Arama (Expert Search) Sistemleri*: Kurumlarda belirli bir alanda uzmanlaşmış kişilerin aranabilmesi amacıyla kullanılan sistemlerdir (Büttcher ve diğ., 2010). Bu sistemde, kullanıcıların bir konu üzerinden gerçekleştirdikleri sorgulamalara karşılık olarak bilgi koleksiyonunda sorguyla güçlü ilişkisi olan kişilerin bulunması işlemi gerçekleştirilmektedir (Latha, 2018). Bulunan uzmanlar, sorguyla ilişkilerinin kuvvetlilik durumlarına göre sıralanarak listelenmektedir (Latha, 2018). Laudon ve Laudon (2012) tarafından “Bilgi Ağı Sistemleri (*Knowledge Network Systems*)” olarak da isimlendirilmiş olan bu BES, çalışanların bir şirkette uygun uzmanı bulmasını kolaylaştırmak için iletişim teknolojilerini kullanan bir sistem olarak tanımlanmıştır.

8. *Soru Cevaplama (Question Answering) Sistemleri*: İnsanların doğal dilleriyle sordukları soruları otomatik olarak cevaplamaya yönelik işlem yapan sistemlerdir (Latha, 2018). Bu işlemi yapabilmek için farklı kaynaklardan bilgi toplamaktadır (Büttcher ve diğ., 2010). Ek olarak, doğal dille oluşturulmuş, yapılandırılmamış belge koleksiyonlarındaki cevapları da bünyesine alabilmektedir (Latha, 2018). Bilgi kaynaklarını listelemek yerine sorulara karşılık özel cevaplar vermek bu sistemlerin amacıdır (Croft ve diğ., 2009).
9. *Multimedya Bilgi Erişim (Multimedia Information Retrieval) Sistemleri*: Resim, müzik, video ve konuşma gibi farklı multimedya kaynaklarından anlamsal bilgiye erişmeyi amaçlayan sistemlerdir (Büttcher ve diğ., 2010).
10. *Özetleme (Summarization) Sistemleri*: Belgeleri, içeriklerini açıklayan az miktarda ana paragraf, cümle veya kelime öbeklerine küçülten sistemlerdir (Latha, 2018). Web arama motorlarında yapılmış olan bir sorgulama sonucunda, ortaya çıkmış olan sonuç listesindeki metin parçacıkları buna örnek gösterilebilir (Büttcher ve diğ., 2010).

Meadow ve diğ., (2007) ve Kowalski (2011) tarafından “çekme” ve “itme” olmak üzere iki farklı biçimde kategorize edilmiş olan BES’ler, Zhai ve Massung (2016) tarafından da bilgiye erişimde zaman kavramı temel alınarak farklı bir şekilde açıklanmıştır. Zhai ve Massung (2016)’un “kısa vadeli” olarak ifade ettiği çekme biçimindeki BES’lerde, kullanıcının geçici bir bilgi ihtiyacının (*ad hoc information need*) olması söz konusudur. Bu bilgi ihtiyacı karşılanır karşılanmaz, bilgi erişim isteği ortadan kaybolmaktadır. Web arama motorlarında gerçekleştirilen anlık, ihtiyaca özel bilgi erişim davranışı bu duruma örnek olarak gösterilebilir. Diğer taraftan, “uzun vadeli” olarak ifade edilmiş olan BES’lerde ise, sistem, kullanıcıya birtakım ilgili bilgi öğelerini tavsiye etme işlemini gerçekleştirmektedir (Zhai ve Massung, 2016). Bu BES’ler, bir kullanıcının uzun süredir devam eden bilgi ihtiyacını karşılamak için genellikle daha kullanışlıdır (Zhai ve Massung, 2016). Bilgi filtreleme ve öneri sistemleri bu türdeki BES’lere örnek olarak gösterilebilir.

Zhai ve Massung (2016), anlık bilgi erişimin (*ad hoc retrieval*) daha önemli olduğunu, çünkü anlık bilgi ihtiyacının uzun vadeliden daha sık ortaya çıktığını belirtmiştir. Bu çalışma kapsamında gerçekleştirilmiş işlemler de *anlık bilgi erişim* çerçevesinde yer alan web arama motorlarına dayanmakta olup, bu BES’lerdeki temel işlemler Şekil 2.3’de aktarılmıştır:



Şekil 2.3: Temel anlık bilgi erişim işlemi (Grossman ve Frieder, 2004).

Şekil 2.3'e göre bir kullanıcı ihtiyaç duyduğu bilgi için arama motoruna anlık olarak sorgu göndermekte ve arama motoru, içinde barındırdığı bilgi koleksiyonunu kullanıcının sorgusuna göre analiz edip, ilgili olan sonuçları sıralayarak kullanıcıya listelemektedir.

2.3. BİLGİ ERİŞİM MODELLERİ

Kullanıcı ihtiyaçlarına yönelik olarak belirli bir sıralamayla kaynakların listelenmesi işlemi, BES'in kullanıcıdan bağımsız olarak kendi içinde gerçekleştirmiş olduğu çeşitli dizinlemeler doğrultusunda yapılmaktadır. Bu dizinlemeler yapılırken kullanılan çeşitli bilgi erişim modelleri, BES'in başarısına etki eden en önemli unsurlardan birisidir. Çünkü bilgi erişim modelleri, listelemede gösterilecek olan kaynakları farklı şekillerde ağırlıklandırılarak sıralamaya yardımcı olmaktadır. Öyle ki, Larson (2011), BES'ler arasındaki gerçek farklılığın sıralamayla ilişkili olduğunu vurgulamıştır. Literatür incelendiğinde, kaynak listelemede faydalanan çeşitli bilgi erişim modellerinin olduğu görülmektedir. Bunlardan bazıları: boolean tam eşleşme (*boolean exact match*) modeli, vektör uzayı (*vector space*) modeli, olasılıksal (*probabilistic*) model ve dil (*language*) modelidir. Bu modellerden, boolean tam eşleşme ve vektör uzayı modeli olmak üzere iki temel bilgi erişim modeli bu bölümde açıklanmıştır.

2.3.1. Boolean Tam Eşleşme Modeli

Boolean tam eşleşme modeli, klasik bir bilgi erişim modeli olmakla birlikte (Latha, 2018) ilk bilgi erişim modelidir (Göker ve Davies, 2009). Bu modelde *George Boole*'un matematiksel mantık operatörleri (AND, OR, NOT) kullanılarak, kullanıcı sorguları ve ilgili kaynaklar birleştirilmektedir (Latha, 2018). Tam eşleşme olarak ifade edilmesinin sebebi ise, bir kullanıcı

ihtiyacını tanımlayan bir terimle sorgulama yaptığında, kaynakların bu sorgu terimiyle tam olarak eşleşen kayıtlar arasından listelenmesidir. *George Boole*'un matematiksel operatörleri ile ilgili kaynakların listelenmesine yönelik aşağıdaki senaryolar örnek gösterilebilir:

- Bir kullanıcı kelime1 ve kelime2 terimlerinin her ikisini de barındıran bir sorgulama yaptığında, listelemede ortaya çıkarılacak kaynakların bulunmasında “kelime1 AND kelime2” şeklinde bir sorgulama ifadesi kullanılmaktadır. Bu sorgulama sonucunda ise, bilgi havuzunda yer alan kaynaklarda her iki terimi içeren ($\text{kelime1} \cap \text{kelime2}$) sonuçlar getirilmektedir.
- Bir kullanıcı kelime1 ve kelime2 terimlerinin herhangi birini ya da her ikisini birden barındıracak bir sorgulama yaptığında, listelemede ortaya çıkarılacak kaynakların bulunmasında “kelime1 OR kelime2” şeklinde bir sorgulama ifadesi kullanılmaktadır. Bu sorgulama sonucunda ise, bilgi havuzunda yer alan kaynaklarda iki terimden herhangi birini ya da her ikisini içeren ($\text{kelime1} \cup \text{kelime2}$) sonuçlar getirilmektedir.

Boolean tam eşleşme modelinde daha farklı senaryolar da oluşturulabilir ancak bu modelin kaynak ağırlıklandırması yapmaya yönelik olarak bir dezavantajı söz konusudur. Bu dezavantaj, modelin, eşleşen kaynaklar arasında hangisinin daha üstte olacağına yönelik bir ağırlık değeri atayamamasıdır (Latha, 2018). Bununla birlikte, bir kullanıcının üç kelimedenden (A, B ve C) oluşan bir terimle her üç kelimeyi de içeren bir sorgulama yaptığı ($A \text{ AND } B \text{ AND } C$) düşünölsün. Belki de A ve B terimlerini içeren bir kaynak bulunmasına ve bu kaynağın kullanıcının bilgi ihtiyacını karşılayacak nitelikte olmasına rağmen, C kelimesini içermemesinden dolayı bu kaynak listelemede gösterilmeyecektir. Bu durum da, kullanıcının harcayacağı emek ve zamanın artması anlamına gelmektedir. Ek olarak, kullanıcıların hatalı yazdıkları terimlerle yaptıkları sorgulamalarda da bu modelin dezavantaj oluşturabileceği ifade edilebilir. Böyle bir senaryoda, hatalı terimle herhangi bir eşleşme olmayacağı için yine ihtiyaç duyulan kaynakların listelenmeme durumu ortaya çıkacaktır.

Bu çalışmada geliştirilmiş olan algoritmanın entegre edildiği yapı, bu modeli kullanarak hizmet vermekte olan bir yapıdır. Algoritma sayesinde, bu modelle ilgili aktarılan dezavantajlar giderilmeye çalışılmıştır.

2.3.2. Vektör Uzayı Modeli

Bir diğer model olan vektör uzayı modeli, her bir boyutun aynı kelime havuzundaki bir terime denk geldiği bir uzayda, dokümanların/kaynakların ve sorguların birer vektör olarak temsil edildiği bir modeldir (Salton ve diğ., 1975; Ceri ve diğ., 2013). Daha açık bir şekilde ifade etmek gerekirse; kullanıcı tarafından sorgulanan terim/terimler ve kullanıcının ihtiyacını karşılama potansiyeli olan kaynaklar/dokümanlar, t-boyutlu (t tüm doküman ve terimlerde yer alan kelimelerin sayısı) bir vektör uzayının birer parçasıdır (Croft ve diğ., 2009). Burada her bir kelime uzayda yer alan farklı boyutları temsil etmektedir. Bu model, boolean tam eşleşme modeline nazaran kaynaklara ağırlık değeri atabilmekte ve bu sayede de hangi kaynağın daha üst sıralarda yer alacağı belirlenebilmektedir.

Vektör uzay modelinde yer alan vektör ve boyut kavramlarının daha iyi anlaşılması amacıyla, t-boyutlu bir vektör uzayında, kaynak havuzunda yer alan bir dokümanın $D_i = (d_{i1}, d_{i2}, d_{i3}, \dots, d_{ij})$ şeklinde ifade edildiği varsayalım. Burada D_i dizinlenmiş terimlerden oluşan bir vektörü ve d_{ij} ise, j'ninci terimin D_i dokümanındaki ağırlığını/bulunma durumunu ifade eder (Croft ve diğ., 2009). n tane dokümanın (vektör) ve t kadar terimin (boyut) bulunduğu bir havuz Tablo 2.1'deki şekilde bir matris olarak ifade edilebilir:

Tablo 2.1: Örnek bir vektör uzay modeli havuzu.

	<i>Terim₁</i>	<i>T₂</i>	<i>T₃</i>	...	<i>T_t</i>
<i>Doküman₁</i>	d_{11}	d_{12}	d_{13}	...	d_{1t}
<i>D₂</i>	d_{21}	d_{22}	d_{23}	...	d_{2t}
<i>D₃</i>	d_{31}	d_{32}	d_{33}	...	d_{3t}
...	...	...	...	...	...
<i>D_n</i>	d_{n1}	d_{n2}	d_{n3}	...	d_{nt}

Kullanıcılar tarafından sorgulanan terimler de tıpkı dokümanlar gibi ifade edilebilir. Bu durumda bir sorgu vektörü $Q = (q_1, q_2, q_3, \dots, q_t)$ şeklinde gösterilebilir. Bir sorgu vektöründe q_j , j'ninci terimin Q sorgusundaki ağırlığını/bulunma durumunu ifade eder (Croft ve diğ., 2009).

Vektör uzay modelinde, bir doküman, kullanıcı tarafından yapılan sorgulama terimine başka bir dokümandan daha fazla benziyorsa, bu dokümanın diğer dokümandan daha ilgili olduğu kabul edilmektedir. Bu karşılaştırma biçimi, vektör uzay modelinde dokümanların ağırlık değeri olarak sıralandırılmasının temelini oluşturmaktadır (Zhai ve Massung, 2016). Bu modelde, kullanıcı sorgusuyla bilgi kaynakları arasındaki benzerliklerin tespit edilip

sıralanması işleminde faydalanan çeşitli yaklaşımlar bulunmaktadır. Bu yaklaşımlardan bazıları aşağıda açıklanmıştır:

“bit vector” yaklaşımı

Bu yaklaşımda terimlerin dokümanlarda ve sorgularda ortak olarak var olup olmama durumuyla ilgilenilmektedir (Zhai ve Massung, 2016). Doküman vektörüyle sorgu vektörünün benzerliği Denklem 2.1’deki formül kullanılarak hesaplanmaktadır:

$$Sim(D_i, Q) = \sum_{j=1}^t d_{ij} \cdot q_j \quad (2.1)$$

Bu yaklaşımda yapılan hesaplama, aşağıdaki örnek sorgu (Q) ve dokümanlarla (D) açıklanmıştır:

Q = “yapay zeka”

D_1 = “... yapay sinir ağları ...”

D_2 = “... kurumsal zeka ...”

D_3 = “... yapay zeka çalışmaları ... yapay ...”

Tablo 2.2: bit vector yaklaşımına göre oluşan vektör uzayı matrisi.

	Boyutlar t						
	yapay	zeka	sinir	ağları	kurumsal	çalışmaları	...
Q	1	1	0	0	0	0	...
D_1	1	0	1	1	0	0	...
D_2	0	1	0	0	1	0	...
D_3	1	1	0	0	0	1	...

Tablo 2.2’de ortak olarak tespit edilen terimler doğrultusunda vektörler şu şekilde ifade edilebilir:

$Q = (1, 1, 0, 0, 0, 0, 0, \dots)$

$D_1 = (1, 0, 1, 1, 0, 0, 0, \dots)$

$D_2 = (0, 1, 0, 0, 1, 0, 0, \dots)$

$D_3 = (1, 1, 0, 0, 0, 1, 0, \dots)$

Benzerlikler, yukarıdaki vektörlerle birlikte bit vector formülü kullanılarak aşağıdaki şekilde hesaplanır:

$$Sim(D_1, Q) = 1*1 + 1*0 + 0*1 + 0*1 + 0*0 + 0*0 + \dots = 1$$

$$Sim(D_2, Q) = 1*0 + 1*1 + 0*0 + 0*0 + 0*1 + 0*0 + \dots = 1$$

$$Sim(D_3, Q) = 1*1 + 1*1 + 0*0 + 0*0 + 0*0 + 0*1 + \dots = 2$$

Verilen örnekte görüleceği üzere, boyutları oluşturan terimler vektörler içinde bulunuyorsa 1, bulunmuyorsa da 0 değeri almaktadır. Sorgu vektörüyle doküman vektörleri arasındaki benzerlik hesaplanırken, her iki vektörde ortak olarak bulunan terimlerin almış olduğu bu ikili (binary) değerler çarpılarak toplanmakta ve karşılaştırılan iki vektörün benzerlik ağırlığı tayin edilmektedir. Verilen örneğe göre, D3 dokümanının D1 ve D2 dokümanından daha üst sıralarda yer alacağı belirtilebilir. D1 ile D2 arasında ise eşitlik söz konusudur. Bu sıralama yaklaşımının dezavantajı, terimlerin vektörler içindeki frekans değerlerine bakılmadan sadece var olup olmama durumunun incelenmesidir. Bu durumda, belki de çok farklı başlıkları içeren bir doküman ile gerçekten kullanıcının ihtiyacını giderebilecek bir doküman aynı ağırlığa sahip olabilmektedir.

“inner product” yaklaşımı

Bu yaklaşımın *bit vector* yaklaşımından tek farkı, terimlerin vektörler içindeki bulunma miktarlarının (*term frequency - tf*) kullanılmasıdır. *Bit vector* yaklaşımında kullanılan formülün aynısı (Denklem 2.1) kullanılmakta olup, *bit vectorde* verilen örnek üzerinden gidildiğinde Tablo 2.3’deki sonuçlara ulaşılmaktadır.

Tablo 2.3: inner product yaklaşımına göre oluşan vektör uzayı matrisi.

Vektörler	Boyutlar t						
	yapay	zeka	sinir	ağları	kurumsal	çalışmaları	...
Q	1	1	0	0	0	0	...
D ₁	1	0	1	1	0	0	...
D ₂	0	1	0	0	1	0	...
D ₃	2	1	0	0	0	1	...

Tablo 2.3’de ortak olarak tespit edilen terimler doğrultusunda vektörler şu şekilde ifade edilebilir:

$$Q = (1, 1, 0, 0, 0, 0, \dots)$$

$$D_1 = (1, 0, 1, 1, 0, 0, \dots)$$

$$D_2 = (0, 1, 0, 0, 1, 0, \dots)$$

$$D_3 = (2, 1, 0, 0, 0, 1, \dots)$$

Benzerlikler, yukarıdaki vektörlerle birlikte *inner product* formülü kullanılarak aşağıdaki şekilde hesaplanır:

$$Sim(D_1, Q) = 1*1 + 1*0 + 0*1 + 0*1 + 0*0 + 0*0 + \dots = 1$$

$$Sim(D_2, Q) = 1*0 + 1*1 + 0*0 + 0*0 + 0*1 + 0*0 + \dots = 1$$

$$Sim(D_3, Q) = 1*2 + 1*1 + 0*0 + 0*0 + 0*0 + 0*1 + \dots = 3$$

Sonuçlarda görüleceği üzere, boyutları oluşturan terimler sadece sorgu vektörü için eğer bulunuyorsa 1, bulunmuyorsa 0 değeri almaktadır. Doküman vektörlerinde ise terimlerin frekans değerleri hesaplanmaktadır. Yani, kullanılan formülde yer alan d_{ij} , tf_{ij} olarak değer almıştır. Hesaplamalarda yapılan işlemler ise tıpkı *bit vector* yaklaşımında olduğu gibi yapılmaktadır. Verilen örneğe göre D_3 dokümanının D_1 ve D_2 dokümanından daha üst sıralarda yer alacağı belirtilebilir. D_1 ile D_2 arasında ise şu haliye eşit benzerlik görülmektedir. Bu sıralama yaklaşımının dezavantajı, terimlerin vektörler içindeki frekans değerlerine bakılmasıdır. Buna örnek olarak, oldukça uzun bir yapısı olan bir dokümanın kısa ama ihtiyaç giderme ihtimali daha yüksek olan başka bir dokümandan daha üstte yer alabilme olasılığı verilebilir. Ayrıca bu yaklaşımda frekanslar hesaplanırken “ve”, “veya” gibi tek başına bir benzersizlik unsuru olamayacak olan terimlerin benzerlik ağırlığını arttırabileceği unutulmamalıdır.

“cosine correlation” yaklaşımı

Bu yaklaşımda kullanıcının sorguladığı sorgulama vektörü ile kullanıcının ihtiyacını giderme potansiyeli olan kaynak/doküman vektörü arasındaki benzerlik ölçüsü, her iki vektör arasındaki açının kosinüsüyle ifade edilmektedir (Göker ve Davies, 2009). Bu yaklaşımın hesaplanmasında kullanılan formül Denklem 2.2’de aktarılmıştır:

$$Cosine(D_i, Q) = \frac{\sum_{j=1}^t d_{ij} \cdot q_j}{\sqrt{\sum_{j=1}^t d_{ij}^2 \cdot \sum_{j=1}^t q_j^2}} \quad (2.2)$$

Denklem 2.2’de farkedileceği üzere, bölünen kısım *bit vector* ve *inner product* yaklaşımlarında kullanılan formülün (Denklem 2.1) aynısıdır. Bölen kısım ise, öklit uzaklık formülünün kullanılmasıyla ortaya çıkarılan vektör uzunluğudur (Büttcher ve diğ., 2010). *Cosine correlation* yaklaşımıyla yapılacak olan ağırlıklandırma hesaplamaları aşağıdaki örnek üzerinden gösterilebilir:

Vektörler: $Q = (1,1,1)$, $D_1 = (2,1,2)$, $D_2 = (0,2,1)$

$$\begin{aligned} \text{Cosine}(D_1, Q) &= \frac{1 * 2 + 1 * 1 + 1 * 2}{\sqrt{(1^2 + 1^2 + 1^2)(2^2 + 1^2 + 2^2)}} \\ &= \frac{5}{\sqrt{27}} = 0.962 \end{aligned}$$

$$\begin{aligned} \text{Cosine}(D_2, Q) &= \frac{1 * 0 + 1 * 2 + 1 * 1}{\sqrt{(1^2 + 1^2 + 1^2)(0^2 + 2^2 + 1^2)}} \\ &= \frac{5}{\sqrt{15}} = 1.291 \end{aligned}$$

Sonuçlar incelendiğinde, D_2 ’nin D_1 ’den daha büyük olduğu, bu sebeple de sıralamada daha üst sıralarda yer alacağı belirtilebilir. Yukarıda belirtilen örnekte kullanılan formülün bölünen kısmında *inner product* yaklaşımı kullanılmasına rağmen, *bit vector* yaklaşımı da kullanılabilir. Buradaki karar, BES için yapılacak tercihe göre değişkenlik gösterebilir.

TF (Term Frequency) x IDF (Inverse Document Frequency) yaklaşımı

Inner product yaklaşımında belirtilmiş olan *tf* değeriyle ilgili dezavantajın giderilmesi amacıyla *idf* değerini hesaplamalara katan bir yaklaşımdır. Farklı hesaplama biçimleri olmasına rağmen, bu yaklaşımda da *inner product* formülünden yararlanılmaktadır. Bu yaklaşımda, *idf* sayesinde doküman havuzunda yer alan kaynaklarda fazla geçmeyen terimlerin ödüllendirilmesi, çok fazla bulunan terimlerin ise cezalandırılması sağlanmıştır (Zhai ve Massung, 2016). Böylelikle *tf* değeri sebebiyle benzersizlik özelliği barındırmayan “ve”, “veya” benzeri terimlerin, dokümanlara ağırlık atamaları esnasında etkisinin azaltılması sağlanmaktadır. *idf* hesaplaması, Denklem 2.3’deki formül kullanılarak yapılmaktadır:

$$idf_j = \log \frac{N}{n_j} \quad (2.3)$$

Denklem 2.3’de N , kaynak havuzunda bulunan toplam doküman sayısını, n_j ise j teriminin bulunduğu doküman sayısıdır. Bu formüle göre, bir terim ne kadar çok dokümanda yer alıyorsa idf değeri o kadar düşecektir. Bu sayede, dokümanlarda çok fazla geçen terimler cezalandırılmaktadır. Bu hesaplamanın yapılmasıyla birlikte *inner product* formülünde (Denklem 2.1) yer alan d_{ij} değeri Denklem 2.4’deki şekilde hesaplamalarda yer almaktadır:

$$d_{ij} = tf_{ij} \cdot idf_j = tf_{ij} \cdot \log \frac{N}{n_j} \quad (2.4)$$

d_{ij} değeriyle birlikte *tf-idf* yaklaşımının formülü Denklem 2.5’de görülmektedir:

$$Sim(Di, Q) = \sum_{j=1}^t tf_{ij} \cdot \log \frac{N}{n_j} \cdot q_j \quad (2.5)$$

tf-idf yaklaşımının çok fazla sayıda biçimi olmasına rağmen, vektör uzay modelinde en popüler olan yaklaşımdır (Bates, 2011).

2.4. BİLGİ ERİŞİMDE DEĞERLENDİRME

Bir BES’in farklı kullanıcı ihtiyaçlarından doğan bilgi gereksinimini karşılayıp karşılamadığının belirlenmesi, sistemin çeşitli yönlerden değerlendirilmesi doğrultusunda gerçekleştirilmektedir. Baeza-Yates ve Ribeiro-Neto (2011), uygun bir değerlendirme olmadan bir BES’in ne kadar iyi hizmet verdiğinin tespit edilmesinin bir yolu olmadığını ve hatta farklı BES’lerle bilgi erişim kalitesinin karşılaştırılmasının yapılamayacağını ifade etmişlerdir. Croft ve diğ. (2009) ise, değerlendirmenin BES’in etkili kullanılıp kullanılmadığının anlaşılması için bir gereklilik olduğunu belirtmişlerdir.

Bu bölümde, ilk olarak, BES’in etkililik ve verimlilik olmak üzere iki farklı kavram açısından değerlendirilmesinde faydalanılan çeşitli ölçüm metrikleri paylaşılmıştır. Sonrasında, “Kullanılabilirlik” kavramıyla ilişkili olacak şekilde aktif kullanıcı katılımı içeren değerlendirme yöntemleri aktarılmıştır. En son olaraksa, ölçümler sonucunda ortaya çıkarılan sonuçların anlamlılığının test edilmesinde faydalanılan çeşitli istatistik yöntemleri paylaşılmıştır.

2.4.1. Etkililik Metrikleri

Etkililik açısından yapılan değerlendirmelerde, test edilen BES'in altyapısında kullanılan bilgi erişim modelinin ne derece başarılı olduğu tespit edilmeye çalışılmaktadır. Kullanıcıların yaptıkları sorgular doğrultusunda, kullanılan BES'in, bilgi kaynaklarını ne derece etkili sunduğunun belirlenmesi işlemi gerçekleştirilmektedir. Bir başka ifadeyle, BES'in sunduğu bilgi kaynağı öğelerinin ne kadar doğru olduğunun tespit edilmesi ve sistemin, ilgili dokümanları ilgisizlerden daha üstte sıralayabilme yeteneğinin ölçülmesi (Zhai ve Massung, 2016) söz konusudur. Bu kısımda ilk olarak, etkililik metriklerinin kullanılmasında yararlanılan çeşitli referans koleksiyonlarından bahsedilmiştir. Bunun sebebi, aktarılmış olan tüm etkililik metriklerinin, halihazırda var olan referans koleksiyonları doğrultusunda ölçüme olanak tanınmasıdır. Referans koleksiyonlarının aktarılması sonrasında ise, literatürde yer alan ve BES'lerin değerlendirilmesinde sıklıkla faydalanılan çeşitli metrikler paylaşılmıştır.

Mortimer Taube tarafından geliştirilmiş olan “*Uniterm System*” adlı bir dizinleme sisteminin, 1952 yılında Birleşik Krallık'ın (UK) Cranfield köyündeki Havacılık Koleji'nde görev yapmakta olan Cyril Cleverdon isimli bir kütüphanecinin dikkatini çekmesiyle başlayan deney yapmakta olan Cyril Cleverdon isimli bir kütüphanecinin dikkatini çekmesiyle başlayan deney çalışmaları, günümüzdeki referans koleksiyonlarının temelini oluşturmaktadır (Baeza-Yates ve Ribeiro-Neto, 2011). Cleverdon'ın, Birleşik Devletler'deki (US) Ulusal Bilim Vakfı'ndan aldığı hibe ile gerçekleştirdiği ilk deney (Cranfield-1), dört farklı dizinleme sisteminin birbiriyle karşılaştırmasını içeren ve 18.000 havacılık dokümanının 1.200 arama sorgusuyla test edildiği bir araştırmadır. Bu araştırma sonuçlarında, bu kısımda da aktarılacak olan Precision (Hassasiyet) ve Recall (Geri Çağırım) metriklerinin hiçbir BES için aynı anda yükseltilemeyeceği ve aralarında ters orantılı bir ilişki olduğu ortaya çıkartılmıştır (Baeza-Yates ve Ribeiro-Neto, 2011). Bir sonraki deney, her bir sorguya karşılık ilgili dokümanların belirlendiği daha küçük bir doküman koleksiyonu kullanılarak gerçekleştirilmiş olup, toplam 1.400 dokümana karşılık 279 tane sorgudan oluşan bir test ortamından oluşmaktadır. Sorgulara karşılık ilgili dokümanların tespit edilmesi amacıyla üç ay boyunca toplam altı öğrenci görev yapmıştır. Bu işlemler sonucunda, her bir doküman-sorgu çifti için ilgililik değerlendirmeleri, sorgular ve dokümanlardan oluşan bir referans koleksiyonu oluşturulmuştur ki bu da Cranfield-2 koleksiyonu olarak bilinmektedir (Baeza-Yates ve Ribeiro-Neto, 2011). Cleverdon'ın önderlik ettiği 2. deney sonrasında, yüksek geri çağırımından ziyade, kullanıcıların, sadece sorgularına karşılık birkaç ilgili cevabı bulabilmeyi tercih ettikleri

gözlemlenmiştir. Cleverdon tarafından ilk olarak gerçekleştirilen bu deneyler sonrasında, referans koleksiyonlar üzerinden BES'in değerlendirme işlemi literatürde "Cranfield Paradigması" olarak yerini almıştır (Büttcher ve diğ., 2010).

Cranfield referans koleksiyonu, BES'lerin değerlendirilmesinde faydalanan birincil veri seti olmakla birlikte (Kowalski, 2011), çoğunlukla başlangıç pilot deneylerde tercih edilmektedir (Manning ve diğ., 2008). Cranfield dışında, farklı koleksiyonlar da zaman geçtikçe oluşturulmuş ve BES'lerin etkililik bakımından değerlendirilmesinde kullanılmıştır. Günümüzde bilgi erişim alanında en çok bilinen konferanslardan biri olan Text REtrieval Conference (TREC¹), çeşitli referans koleksiyonlar sunan bir oluşumdur. Benzer şekilde, NTCIR² (*NII Testbeds and Community for Information Access Research*) tarafından kullanıma sunulan referans koleksiyonları da yine etkililik değerlendirmelerinde kullanılmaktadır. Bir diğer referans koleksiyonu FIRE³ (*Forum for Information Retrieval Evaluation*) tarafından kullanıma sunulmaktadır. LISA⁴ (*Library and Information Science Abstracts*) referans koleksiyonları da yine bir başka veri seti olarak örnek gösterilebilir.

Cranfield paradigmasının en temel dezavantajı, kullanıcı ihtiyaçlarının statik olarak görülmesi ve bir grup tarafından yapılan ilgililik değerlendirmelerinin tüm kullanıcı popülasyonunu temsil ettiğinin varsayılmasıdır (Baeza-Yates ve Ribeiro-Neto, 2011). Bu durumun, özellikle web üzerinden hizmet veren uygulamalar göz önüne alındığında, yetersiz etkililik değerlendirmelerine sebep olacağı ifade edilebilir. Çünkü web kullanıcılarının ihtiyaçlarının sürekli değişkenlik gösteren bir yapıda olduğu rahatlıkla belirtilebilir. Buna rağmen, referans koleksiyonları, farklı BES'lerin karşılaştırmasında objektiflik taşıdığından (Baeza-Yates ve Ribeiro-Neto, 2011) ve hızlı bir şekilde değerlendirme imkânı tanıdığından günümüzde oldukça fazla tercih edilmektedirler.

Literatür incelendiğinde, CG (*Cumulative Gain*), DCG (*Discounted Cumulative Gain*), NCG (*Normalized Cumulative Gain*) ve NDCG (*Normalized Discounted Cumulative Gain*) isimli kazanç tabanlı ve dereceli ilgililik değerlendirmesi içeren etkililik metrikleri (Järvelin ve Kekäläinen, 2000; Järvelin ve Kekäläinen, 2002), ERR (*Expected Reciprocal Rank*) isimli bir başka dereceli ilgililik değerlendirmesi içeren etkililik metriği (Chapelle ve diğ., 2009) ve

¹ <https://trec.nist.gov/>

² <http://research.nii.ac.jp/ntcir/index-en.html>

³ <http://fire.irs.res.in/>

⁴ http://ir.dcs.gla.ac.uk/resources/test_collections/lisa/

Moffat ve Zobel (2008) tarafından önerilen RBP (*Rank-Biased Precision*) isimli farklı bir metriğin etkililik ölçümlerinde kullanıldığı görülmektedir. Bu metrikler, kullanıcılar üzerinden elde edilen bir değerlendirmeye (kullanıcıdan direk geribildirim alma zorunluluğu söz konusudur) bağlı kullanılmalarından ya da referans koleksiyonları üzerinden değerlendirmeye imkân tanıdıklarından dolayı bu çalışma kapsamında yer almamaktadır. Bunun birincil sebebi, çalışmada geliştirilen ve detayları 3.1 bölümünde açıklanan algoritmanın, aktif bir sistem üzerinde hizmet verebilen bir algoritma olması ve dolayısıyla referans koleksiyonu gibi statik olan veri setleriyle analizin algoritmanın çalışma felsefesine uymaması olarak gösterilebilir. İkincil sebep olaraksa, kullanıcıların direk geribildirim (oylama, yorum yapma, vb.) konusunda isteksiz davranabilme ihtimalleridir. Bu durum, çalışmanın 4.4 bölümünde bariz bir şekilde ortaya çıkartılmış olup, çalışmada, kullanıcıya direk bağlı bir ölçüm metriğinin kullanılmamasının doğru bir karar olduğu görülmüştür. Diğer taraftan, etkililik ölçümlerinde, referans koleksiyonlarından faydalanılarak gerçekleştirilen testlerde **çoğunlukla** tercih edilen bazı temel metrikler aşağıda sıralanmış ve açıklanmıştır. Bu metriklerden, sadece Mean Reciprocal Rank (*MRR*) metriğine temel teşkil eden Reciprocal Rank (*RR*) değeri bu çalışmadaki altyapıyla kullanılabilmiş olup, detayları hem *MRR* metriğinin açıklandığı kısımda hem de çalışmanın 4.2 bölümünde aktarılmıştır:

- Precision (P - Hassasiyet)
- Recall (R - Geri Çağırma)
- Average Precision at n ($P@n$ - n. Sıradaki Ortalama Hassasiyet)
- Mean Average Precision (MAP - Ortalama Hassasiyetin Aritmetik Ortalaması)
- Geometric Mean Average Precision (GMAP - Ortalama Hassasiyetin Geometrik Ortalaması)
- F measure (F - F Ölçüsü)
- Mean Reciprocal Rank (MRR - Ortalama Karşılıklı Sıra)

Precision (P)

Herhangi bir BES altyapısında kullanılan modelin, ilgili olmayan dokümanları bulma eylemini ne kadar iyi gerçekleştirdiğini ölçen bir metriktir (Croft ve diğ., 2009). Bir başka ifadeyle, bir sorguya karşılık bulunan tüm dokümanlar içinde ilgili olanların oranıdır. *P* metriğine ait formül Denklem 2.6'da aktarılmıştır.

$$P = \frac{\text{Bulunan Dökümanlar} \cap \text{Bulunan İlgili Dökümanlar}}{\text{Bulunan Dökümanlar}} \quad (2.6)$$

Recall (R)

Bir sorguya karşılık tüm ilgili dokümanların bilindiği bir sorgulamada, kullanılan BES tarafından ilgili olarak tespit edilen dokümanların, tüm ilgili dokümanlara oranını ifade etmektedir (Denklem 2.7).

$$R = \frac{\text{Tüm İlgili Dökümanlar} \cap \text{Bulunan İlgili Dökümanlar}}{\text{Tüm İlgili Dökümanlar}} \quad (2.7)$$

Average Precision at n (P@n)

R metriği, tahmin edileceği üzere bir sorguya karşılık tüm ilgili dokümanların bilinmesi durumunda kullanılabilir. Web ortamını düşündüğümüzde, farklı bilgi kaynaklarının sürekli internet ortamına eklendiğini ve herhangi bir sorguya karşılık gelen tüm ilgili dokümanların tespit edilmesinin mümkün olmadığı rahatlıkla söylenebilir. Öyle ki, R metriğinin web ortamında yapılan aramalar için anlamsız olduğu ifade edilmiştir (Büttcher ve diğ., 2010). Cranfield-1 deneylerinde de ortaya çıkan “kullanıcıların sadece birkaç cevap bulmayı istemesi” durumu $P@n$ metriğinde öne çıkmaktadır. Çünkü bu metrik ilgili dokümanların kullanıcılara sunulurken en üstte yer alıp almadığını değerlendirmektedir (Croft ve diğ., 2009; Baeza-Yates ve Ribeiro-Neto, 2011). $P@n$ metriğinde yer alan $@n$, BES tarafından belirli bir miktarda doküman gösterildiğinde ortaya çıkan P değerine işaret etmekte olup, kesme noktası şeklinde ifade edilebilir. $@n$ değeri çoğunlukla $@5$, $@10$ ve $@20$ şeklinde kullanılabilir (Baeza-Yates ve Ribeiro-Neto, 2011). $@5$ göz önüne alındığında, bir sorguya karşılık BES tarafından bulunan dokümanlardan ilk 5 içinde yer alan P değeri göz önüne alınmaktadır. Bu metrik temel alınarak farklı iki sistemin karşılaştırması yapılırken, belirlenen bir kesme noktası üzerinden çok fazla sorgu gerçekleştirilip, ortaya çıkan tüm P değerlerinin her bir sistem için ortalaması alınmaktadır. Son durumdaysa, farklı sistemler, ortaya çıkarılan bu ortalama değerler göz önüne alınarak kıyaslanabilmektedir.

Mean Average Precision (MAP)

P değerini kullanan bir diğer metriktir. Bir sorguya karşılık BES tarafından bulunan dokümanlarda, her bir ilgili dokümanın gözlemlendiği sıradaki P değerinin ortalamalarının alınmasıyla yapılan bir hesaplama yöntemidir. Kullanıcının, her sorgu için ilgili birçok belgeyi bulmakla ilgilendiğini varsaymaktadır (Croft ve diğ., 2009). Bu metriğe ait formül Denklem 2.8 ve 2.9’de aktarılmıştır.

$$MAP_i = \frac{1}{|R_i|} \sum_{k=1}^{|R_i|} P(R_i[k]) \quad (2.8)$$

$$MAP = \frac{1}{N_q} \sum_{i=1}^{N_q} MAP_i \quad (2.9)$$

Denklem 2.8 için, R_i , q_i sorgusuna yönelik ilgili dokümanlar olmak üzere $|R_i|$ ilgili dokümanların boyutunu ifade etmektedir. $R_i[k]$, R_i ’deki k ’nıncı dokümana karşılık gelmekte, $P(R_i[k])$ ise q_i ile yapılan sorgulama sonrasında ortaya çıkan doküman sıralamasındaki $R_i[k]$ ’nın gözlemlendiği P değerine karşılık gelmektedir (Denklem 2.8). Bu dokümanın gözlemlenmemesi durumunda, $P(R_i[k])$ sıfır “0” değeri almaktadır. Denklem 2.9 içinse, N_q toplam yapılan sorgulama miktarını ifade etmektedir.

Geometric Mean Average Precision (GMAP)

MAP , aritmetik ortalama kullanan bir ölçüttür. $GMAP$ (Robertson, 2006) ise ortalama P değerlerini çarparak düşük performanslı sorguların etkisini vurgulayan bir ölçüttür (Croft ve diğ., 2009). TREC 2004’de önerilen (Voorhees, 2004) bu metriğe ait formül Denklem 2.10’da aktarılmıştır.

$$GMAP = \sqrt[n]{\prod_{i=1}^n (AP_i + \varepsilon)} - \varepsilon \quad (2.10)$$

Denklem 2.10 için, n toplam sorgu miktarını, AP_i , i sorgusu için ortalama P değerini, ε ise herhangi bir AP_i değerinin sıfır “0” olduğu durumda ortaya çıkacak hataları engellemek için kullanılan bir sabittir.

F measure (F)

P ve R değerlerinin birlikte kullanıldığı ve aritmetik ortalama yerine harmonik ortalama hesaplamasının yapıldığı bir etkililik metriğidir. Etkililiği tek bir değerde özetleme avantajına sahiptir (Croft ve diğ., 2009; Baeza-Yates ve Ribeiro-Neto, 2011; Ceri ve diğ., 2013; Latha, 2018). F metriğinin formülü Denklem 2.11’de görülmektedir.

$$F = \frac{2RP}{R + P} \quad (2.11)$$

Aritmetik ortalama yerine harmonik kullanılmasının sebebi, aritmetik ortalamanın uç değerlerden (*outliers*) fazla etkilenebilmesidir. Oysaki, harmonik ortalama küçük değerlerin önemini de ön plana çıkaran bir hesaplama yöntemidir (Croft ve diğ., 2009).

Mean Reciprocal Rank (MRR)

Reciprocal Rank (RR - Karşılıklı Sıra), kullanıcıların yaptıkları sorguya karşılık tek bir ilgili dokümanı görmek istedikleri ve bu dokümanın da mümkün olduğunca en üstte yer alması gerektiği varsayımına dayanmaktadır (Büttcher ve diğ., 2010). RR , yapılan bir sorguya karşılık bulunan ilk ilgili dokümanın sıralamasına karşılık gelen bir metriktir (Croft ve diğ., 2009; Harman, 2011). MRR metriği ise, bir örnekle açıklanmak istenirse; bir sorguya karşılık kullanılan BES tarafından 20 tane ilgili dokümanın bulunduğu varsayılın. Bulunan ilk ilgili dokümanın 4. sıradaki doküman olduğu düşünölsün. Bu durumda, RR değeri $4/20$ işleminden 0.2 olarak bulunur. Farklı bir sorguda ise RR değerinin 0.4 olarak hesaplandığı düşünölsün. Sonuç itibariyle, her iki RR değerinin aritmetik ortalaması olan MRR değeri 0.3 olarak bulunur. Buna benzer şekilde, birçok aramadan elde edilecek MRR değeri ile farklı BES’ler karşılaştırılabilmektedir. Bu çalışmada faydalanılan etkililik metriği RR olmakla birlikte, analiz işlemlerinde nasıl faydalanıldığına yönelik detaylar çalışmanın 4.2. bölümü olan “ETKİLİLİK BULGULARI” bölümünde açıklanmıştır.

2.4.2. Verimlilik Metrikleri

Verimlilik, kullanıcının ihtiyaç duyduğu bilginin ne kadar sürede sistem tarafından karşısına getirildiği ve sistemin bu bilgiyi kullanıcıya sunarken ne kadar kaynak tükettiği (Zhai ve Massung, 2016) ile ilgili bir performans göstergesidir. Croft ve diğ. (2009), verimlilik ölçümüne dair Tablo 2.4’deki şekilde göstergelerin bulunduğunu belirtmişlerdir.

Tablo 2.4: Verimlilik göstergeleri (Croft ve diğ., 2009).

GÖSTERGE	TANIM
Dizinleme Süresi	Bir sistemin dizinleme için harcadığı süredir.
Dizinleme İşlemci Süresi	Dizinleme için işlemcinin kullandığı süreye tekabül etmekte olup, I (Input)/O (Output) için harcanan süreden bağımsızdır.
Sorgu Çıktısı	Saniyede işlenen sorgu miktarıdır.
Sorgu Gecikmesi	Kullanıcının sorgulama yaptığı süre ile cevap aldığı süre arasında geçen zamandır.
Dizinleme Geçici Alanı	Dizinleme için kullanılan geçici disk kullanım miktarıdır.
Dizinleme Boyutu	Dizinleme dosyalarının depolanması için ihtiyaç duyulan disk alanıdır.

Sorgu Çıktısı (SOÇ), çoğunlukla kullanılan ölçüm yöntemi olmakla birlikte, aynı referans koleksiyonunun aynı sistem donanımı üzerinde test edilmesi doğrultusunda karşılaştırmaya olanak sağlamaktadır (Croft ve diğ., 2009). SOÇ'un, Response Time (Cevap Süresi) olarak da tanımlanmış olan (Büttcher ve diğ., 2010) Sorgu Gecikmesi (SOG) ile ters orantılı bir davranış gösterdiği belirtilmiş olup, kullanıcıların 150 milisaniyenin üstünde bekletilmemesi gerektiği ifade edilmiştir (Croft ve diğ., 2009). Her iki göstergenin tek bir sorgu yerine milyonlarca sorguyu içeren bir veri setinde ortalama kullanılarak hesaplanmasının daha anlamlı bir analiz sağlayacağı söylenmiştir (Büttcher ve diğ., 2010). Bu açıklamalar doğrultusunda, sistem donanımının el verdiği ölçüde her iki gösterge arasında kullanıcıyı memnun edecek bir dengenin oturtulmasının önemli olduğu rahatlıkla söylenebilir. Her iki gösterge dışında diğer göstergeler de verimlilik ölçümünde tercih edilmekte olup, bu çalışma için odak alınan verimlilik göstergesi SOG olmuştur.

2.4.3. Kullanıcı Merkezli Değerlendirme ve Kullanılabilirlik

Kullanılabilirlik, Budak (2016) tarafından “*kullanım ortamlarının farklılığına rağmen günlük yaşamımızda çeşitli işlemleri gerçekleştirmek için yararlandığımız her türlü ürünün işlevsellik, pratiklik, kolay öğrenilme ve kullanım memnuniyetinin seviyesi*” şeklinde; ISO (1998) tarafından “*hedef kullanıcının yapması amaçlanan işlemleri özelleştirilmiş içerik bağlamında etkili, verimli ve memnuniyet verici bir şekilde gerçekleştirme derecesi*” şeklinde tanımlanmıştır. Kullanılabilirlik, bir sistemin, gerçek kullanıcı eylemlerinde ne kadar kullanışlı olduğu (Zhai ve Massung, 2016) ile ilgilenen bir araştırma alanıdır. Bir amaca ulaşmak için kullanılan sistemin ne kadar etkili olduğuyla ilgilidir (Nielsen, 1993). ISO (1998)'nin kullanılabilirlik tanımı dikkate alındığında, temel olarak üç farklı unsurun bir sistemin

kullanılabilirliğine işaret ettiği belirtilebilir: etkililik, verimlilik ve memnuniyet. Bu kısma kadar çalışmada aktarılan değerlendirme yöntemlerinin de etkililik ve verimlilik olmak üzere kullanılabilirlikle ortak iki kavramının olduğu görülmektedir. Kelly (2009), bilgi erişim literatüründe kullanılabilirliğin ayrı bir varlık olarak ele alınmasının sebebini, “performans ölçümünün bilgi erişim sistemlerinin değerlendirmesinde her zaman var olması” şeklinde belirtmiştir. Yani, kullanılabilirlik kavramının BES’lerin ilk değerlendirme çalışmalarının çok sonrasında ortaya çıkmasının, sanki ayrı bir kavrammış gibi konu edilmesine sebep olduğu belirtilebilir.

Kullanılabilirlik perspektifinden bakıldığında, etkililik, bir sistemin, kullanıcının beklediği şekilde davranması ve kullanıcının amacını gerçekleştirmek için kolay bir şekilde kullanılabilmesine işaret etmektedir (Rubin ve Chisnell, 2008). Bilgi erişim açısdansa, sistemin, kullanıcının ihtiyaç duyduğu bilgi ihtiyacını ne kadar karşılayabildiğinin ölçümü söz konusudur. Tanımlar farklı gibi görünse de, her ikisinin de ortak paydasının, “kullanıcının amacını en doğru ve en kolay şekilde gerçekleştirmesi” olduğu rahatlıkla ifade edilebilir. Verimlilik, kullanılabilirlik açısından, zamansal olarak kullanıcının hedefini en hızlı biçimde gerçekleştirebilmesi şeklinde belirtilmiştir (Rubin ve Chisnell, 2008). Bilgi erişim açısdansa, aktarılan ölçüm yöntemlerinin hemen hepsinde zaman ve hız faktörünün ölçümünün odak alındığı görülmektedir. Dolayısıyla, farklı literatürler altında geçmelerine rağmen, verimlilik kavramıyla, zaman açısından bir kullanıcının amacına daha hızlı erişebilmesi noktasında her iki literatürün buluştuğu ifade edilebilir.

Bilgi erişimde kullanıcı merkezli değerlendirmeler, Cranfield Paradigmasının ötesine geçilen çeşitli metodlarla gerçekleştirilmektedir (Baeza-Yates ve Ribeiro-Neto, 2011). Manning ve diğ. (2008), kullanıcılar tarafından belirtilen bilgi ihtiyacının, sistem tarafından karşılandığının belirlenmesi için insan deneklerle yapılacak çalışmalara vurgu yapmıştır. Bu çalışmalar, “bir görevi tamamlama süresi” gibi nesnel ölçütleri, “arama motorundan memnuniyet puanı” gibi öznel ölçütleri veya “arama arayüzündeki kullanıcı yorumları” gibi nitel ölçütleri içerebilir (Manning ve diğ., 2008). Genel itibarıyla, BES’in kullanıcı merkezli değerlendirmesinde faydalanan yöntemlerden bazıları aşağıda aktarılmış olup, kullanılabilirlik literatürüyle ilişkili olacak şekilde açıklanmıştır:

Labaratuvar Ortamında Kullanılabilirlik Testleri (LOKT)

LOKT, “labaratuvar ortamında genellikle çeşitli görevler içeren belirli bir senaryoya bağlı kalınarak, insan deneklerle gerçekleştirilen bir arayüz test yöntemi” şeklinde tanımlanabilir. Temel amacı, bir ürün hakkında mümkün olduğunca fazla kullanılabilirlik problemlerinin tespit edilmesi veya iki farklı ürünün karşılaştırılmasıdır (Dumas ve Loring, 2008). Statik referans koleksiyonları kullanılarak ölçülemeyecek dinamik kullanıcı-sistem etkileşimlerinin anlaşılmasına olanak sağlamaktadır (Baeza-Yates ve Ribeiro-Neto, 2011). Her ne kadar test ortamının kurulmasının masraflı olduğu, az sayıda insan denekle yapılan ve az sayıda bilgi ihtiyacı içeren bir yapıda olduğu ifade edilse de, referans koleksiyonları üzerinden elde edilen bulguları tamamlayıcı bir niteliğe sahip olduğu belirtilmiştir (Baeza-Yates ve Ribeiro-Neto, 2011).

Yan Yana Panel Testi

Genellikle iki farklı BES tarafından sabit bir sorguya karşılık gösterilen belirli bir miktardaki (ör; ilk 10) sonucun yan yana olacak şekilde insan deneklere gösterilerek sorgulandığı test yöntemidir. Bu test yöntemi, LOKT için bir örnek teşkil etmektedir (Baeza-Yates ve Ribeiro-Neto, 2011). Çünkü karşılaştırmalar laboratuvar ortamında insan deneklerle gerçekleştirilmektedir. Şekil 2.4, bu test yönteminde kullanılan bir yan yana panel testi görselidir.

[pdf] Pharmaceutical Information Flyer PDF/Adobe Acrobat PHARMACEUTICAL INFORMATION RETRIEVAL AND EVALUATION SERVICE. Future Solutions Now ... Information need, • retrieval of the appropriate documents, • evaluation ... www.slowa.edu/~dis/Pharm_info_Flyer.pdf ROMIP: Russian Information Retrieval Evaluation Seminar Russian information retrieval evaluation initiative was launched in 2002 with ... a basis for independent evaluation of information retrieval methods, aimed to be ... romip.ru/en [pdf] Reflections on Information Retrieval Evaluation Mei-Mei Wu & Diane ... PDF/Adobe Acrobat Reflections on Information Retrieval Evaluation. Mei-Mei Wu ... Research and evaluation in information retrieval. Journal of Documentation, 53 (1), 51-57 ... pnclink.org/annual/annual1999/1999pdf/wu-mm.pdf Information retrieval - Wikipedia, the free encyclopedia Information retrieval (R) is the science of searching for ... that was needed for evaluation of text retrieval methodologies on a very large text collection. ... en.wikipedia.org/wiki/Information_retrieval The Music Information Retrieval Evaluation eXchange (MIREX) The 2005 Music Information Retrieval Evaluation eXchange (MIREX 2005). Preliminary Overview. ... Music Information Retrieval Systems Evaluation Laboratory: ... www.dlib.org/dlib/december05/downie/12downie.html	[pdf] Reflections on Information Retrieval Evaluation Mei-Mei Wu & Diane ... PDF/Adobe Acrobat digital library initiatives. Information retrieval (R) evaluation has Evaluation of evaluation in information retrieval. Proceedings of the ... pnclink.org/annual/annual1999/1999pdf/wu-mm.pdf [pdf] Retrieval Evaluation with Incomplete Information PDF/Adobe Acrobat The philosophy of information. retrieval evaluation. In Evaluation of Cross-Language. Information Retrieval Systems. Proceedings of CLEF ... www.nist.gov/fdi/aiad/papers/2004/p102-buckley.pdf Evaluation criteria for information retrieval systems. - [Traduzir esta página] The contrast between the value placed on discriminatory power in discussions of indexing and classification and on the transformation of a query into a set ... informationr.net/ai/4-4/paper62.html - 36k Information retrieval - Wikipedia, the free encyclopedia - [Traduzir esta página] The aim of this was to look into the information retrieval community by supplying the infrastructure that was needed for evaluation of text retrieval ... en.wikipedia.org/wiki/Information_retrieval - 59k [pdf] Information Retrieval System Evaluation: Effort, Sensitivity, and ... PDF/Adobe Acrobat Information Retrieval System Evaluation: Effort, Sensitivity, and Reliability. Mark Sanderson. Department of Information Studies, University of ... dis.shef.ac.uk/mark/publications/my_papers/SIGIR2005.pdf
--	--

Şekil 2.4: “information retrieval evaluation” aramasına karşılık Yahoo ve Google arama motorları tarafından sunulan öneriler (Baeza-Yates ve Ribeiro-Neto, 2011).

Bu test yönteminde değerlendirmeler genellikle “hangi öneri listesinin daha iyi sonuç sunduğunun yargılanması” şeklinde olup, internet ortamında arama sonuçları sürekli

değişkenlik içereceğinden tekrar ettirilemezler (Baeza-Yates ve Ribeiro-Neto, 2011). Herhangi bir sistemin, başka bir sistemden daha iyi olduğunun belirlenemediği bu yöntem, diğer değerlendirme yöntemlerini destekleyici bir özelliكتedir (Baeza-Yates ve Ribeiro-Neto, 2011).

A/B Testi

Bir kısım kullanıcının bir tasarıma “A”, geri kalan kullanıcıların ise farklı bir tasarıma “B” yönlendirilerek, iki farklı tasarımın karşılaştırıldığı test yöntemidir (Baxter ve diğ., 2015). Modifiye edilmiş bir tasarımın olumlu ya da olumsuz olduğunun anlaşılmasını mümkün kılan bir yöntemdir (Baeza-Yates ve Ribeiro-Neto, 2011). Nitel ve nicel olmak üzere her iki türdeki çalışmalarda kullanılabilen (Rosenzweig, 2015) bu yöntemde, tıklamalar gibi çeşitli kullanıcı eylemlerinin incelenmesi yapılarak, tasarımın, kullanıcı davranışlarına olan etkisi görülmeye çalışılmaktadır (Ritter ve diğ., 2014).

Kitle Kaynak Testi (Crowdsourcing)

Cranfield Paradigması’nda, her bir sorguya karşılık ilgili dokümanların belirlendiği ve zaman ve emek gerektiren bir işlem söz konusudur. Bu tarz bir eşleştirmenin yapılabilmesinde, büyük miktarda çalışana ihtiyaç duyulmasından dolayı ön plana çıkan bir yöntemdir. Bu yöntemde, büyük miktarda çevrimiçi katılımcının yer aldığı bir grubun değerlendirmelere katıldığı bir etkileşim süreci söz konusudur (Baeza-Yates ve Ribeiro-Neto, 2011). Bilgi erişim uzmanı olmayan bu grubun, kullanılabilirlik kurallarına uygun olarak tasarlanmış bir sistemi test etmesi doğrultusunda değerlendirmeler gerçekleştirilmektedir (Baeza-Yates ve Ribeiro-Neto, 2011). Bu sebeple, grubun doğru bir şekilde yönlendirilmesi noktasında, tasarımcının açık bir şekilde yapılacak eylemleri sunması oldukça önemlidir (Baeza-Yates ve Ribeiro-Neto, 2011). Howe (2006) tarafından, Wired dergisi makalesinde icat edilen "kitle kaynak kullanımı (Crowdsourcing)" kavramı, bir sorunu çözmek veya bir görevi yerine getirmek için yapılan açık bir çağrı olarak tanımlanmakta ve genellikle bu tür bir hizmet, karşılığında parasal bir değer içermektedir (Alonso ve diğ., 2008).

Tıklama Verisi Testi (Clickthrough Data)

BES’lerin bir referans koleksiyonu kullanılarak değerlendirilmesi çeşitli sınırlamalar içermektedir. Bunlara, kısıtlı miktardaki sorguların test edilebilmesi, veri setinin statik oluşundan dolayı güncelliğini yitirebilmesi veya koleksiyon oluşturma maliyetli

olması ve tüm kullanıcı kitlesinin ihtiyaçlarını temsil ediyormuş varsayımıyla hareket edilmesi örnek olarak verilebilir. Diğer taraftan, arama motorlarının tipik olarak milyonlarca sayıda sorgudan oluşan günlük (*log*) dosyaları barındırması (Baeza-Yates ve Ribeiro-Neto, 2011), yararlanılabilecek ve sürekli kendini güncelleyen bir veri setine işaret etmektedir.

Croft ve diğ. (2009), tıklama verisi barındıran günlük dosyalarının, BES'lerin eğitilmesinde ve değerlendirilmesinde kullanıldığını ve insanlarla birebir yapılan ilgililik değerlendirmeleriyle yüksek derecede benzerlik içerdiğini ifade etmiştir. Kullanıcıların, rahatsız edilmeden normal işlemlerini gerçekleştirdikleri süreçte depolanabilen tıklama verisi, düşük bir maliyet içermesinden dolayı oldukça cazip bir değerlendirme veri setidir (Baeza-Yates ve Ribeiro-Neto, 2011). Veri toplama sürecinin dikkatli bir şekilde organize edilmesinin gerektiği belirtilen bu yöntem için, uzman değerlendiriciler tarafından sunulan ilgililik geribildirimlerine kıyasla oldukça gereksiz verinin yer aldığı da belirtilmiştir (Baeza-Yates ve Ribeiro-Neto, 2011). Manning ve diğ. (2008) tarafından tıklama madenciliği (*clickstream mining*) olarak da ifade edilmiş olan bu yöntemde, kullanıcıların BES'ler tarafından önerilen dokümanlarda üst sıralarda yer alan tıklama davranışlarının, frekans vb. ölçümlerle değerlendirilebileceği söylenmiştir.

Kullanıcılar bir BES üzerinde doğal eylemlerini gerçekleştirirken, bu davranışların kaydedildiği günlük (*log*) dosyaları, “ilk tıklama süresi (*time to first click*)” ve “sorgu terk etme (*query abandonment*)” olmak üzere çalışmanın odağında olan iki farklı performans göstergesinin elde edilebileceği bir etkileşim kaydını da içinde barındırabilmektedir. Tıklama davranışı, bir kullanıcının yaptığı sorguya karşılık karşısına getirilen doküman önerilerinden herhangi bir tanesine tıklaması şeklinde ifade edilebilir. Baxter ve diğ. (2015), tıklama etkileşiminin bir bakıma verimlilik göstergesi olduğunu belirtmiştir. İlk tıklama süresi ise, bir kullanıcının arama yaptığı an ile karşısına getirilen önerilerden herhangi birine ilk tıkladığı ana kadar geçen süre şeklinde tanımlanabilir. Bu iki an arasında harcanan zaman, arama sistemi kullanıcıları için temel maliyet olduğundan, bu sürenin kısaltılmasının önemli olduğu vurgulanmıştır (Hofmann ve diğ., 2016). Bu noktada, kullanılan BES'in herhangi bir sorguya karşılık en doğru sonuçları üst sıralarda kullanıcıya sunmasının önemli olduğu düşünülmektedir. Çünkü kullanıcının bilgi ihtiyacını gidereceği dokümana hızlı ulaştırılmasının, ilgili dokümanların, aranan cevaba en kısa sürede ulaşabilecek bir konumda

sunulması (üst sıralar) doğrultusunda gerçekleştirilebileceği rahatlıkla ifade edilebilir ki bu yaklaşımın verimlilik açısından da kazanç sağlayacağı söylenebilir.

Kullanıcının yaptığı sorgu sonrasında karşısına getirilen önerilerden herhangi bir tanesine tıklamaması durumu “sorgu terk etme” olarak tanımlanmış olup (Radlinski ve diğ., 2008; Büttcher ve diğ., 2010), kullanıcı memnuniyetsizliğinin bir işaretidir (Joachims ve Radlinski, 2007). Kullanıcıların, bilgi ihtiyaçlarını gidermek amacıyla gerçekleştirdikleri sorgulamalara karşılık, bir cevap bulamamaları şeklinde de ifade edilebilir. “Cevap bulamama” açısından düşünüldüğünde, terk etme eyleminin bir bakıma kullanıcının “zaman ve emek açısından karşılık bulamadığı bir süreç” olarak nitelendirilebileceği ve bu durumun da verimlilikle doğrudan ilişkili olduğu rahatlıkla ifade edilebilir.

Kullanıcıların, tıklama davranış takibinin ön planda olduğu bu yöntemde, belirtilen göstergeler haricinde çeşitli başarımlar ölçümleri gerçekleştirilebilmektedir. Bu çalışmada ise, açıklaması yapılmış olan göstergelerin kullanılması yoluna gidilmiştir.

2.4.4. Değerlendirmelerin İstatistiksel Geçerliliğinin Sağlanması

BES’lerin başarımlarının belirlenmesinde, daha önce de ifade edildiği üzere iki ya da daha fazla sistemin etkililik, verimlilik ve memnuniyet (kullanılabilirlik literatürü) açısından karşılaştırmaları söz konusudur. Çalışmada aktarılan metriklerle ya da farklı ölçüm yöntemleri üzerinden elde edilen sayısal veriyle, karşılaştırılan sistemler arasında anlamlı bir farklılığın olup olmadığına karar vermek için anlamlılık testlerine (*significance tests*) ihtiyaç duyulacağı belirtilmiştir (Croft ve diğ., 2009). Bu karşılaştırmalar, sıfır hipotezi’ne (*null hypothesis*) dayandırılarak yapılmaktadır (Croft ve diğ., 2009). “Karşılaştırılan sistemler arasında bir farklılık yoktur” şeklinde kurulan sıfır hipotezine alternatif olarak “sistemler arasında fark vardır” şeklinde bir hipotez kurulmaktadır (Croft ve diğ., 2009). Alpar (2016), hipotez testlerini, “elde edilen sonuçların rastlantıya/şansa bağlı olup olmadığının incelenmesinde kullanılan istatistiksel yöntemler bütünü” olarak tanımlamıştır. Dolayısıyla, farklı BES’lerin kıyaslanmasında istatistiksel geçerliliğin belirlenmesi için hipotez testlerinin bir gereklilik olduğu rahatlıkla belirtilebilir.

BES’lerin karşılaştırılmasında genel itibariyle t-testi (parametrik) ve parametrik olmayan Wilcoxon işaretli sıralar testi ile işaret testi’nden yararlanılmaktadır (Hull, 1993; Croft ve diğ., 2009). Belirli bir miktar sorgudan elde edilen veriyle iki farklı sistemin karşılaştırılmasında

gerçekleştirilen anlamlılık testlerinde, genel itibariyle aşağıdaki adımlar izlenmektedir (Croft ve diğ., 2009):

1. Her bir sorgunun her iki sistem üzerinde ürettiği sonuçların belirli bir etkililik (yazarlar, etkililik açısından ifade etmiş olmalarında rağmen, farklı metrikler için de hipotez testleri yapılabilir) metriğiyle bulunması
2. Kullanılan metriktan elde edilen sonuçların bir anlamlılık testine tabi tutulması
3. p değerinin ortaya çıkarılması
4. Tip 1 hatasının önüne geçmek amacıyla, genellikle 0.05 ya da 0.1 olarak seçilen anlamlılık seviyesine (*significance level*) göre karşılaştırmanın yapılması ve eğer p değeri seçilen seviyeden küçük ise, sıfır hipotezinin reddedilmesi

Peters ve diğ. (2012), daha kapsamlı bir anlayış kazanmak için yaygın olarak bir veya daha fazla istatistiksel anlamlılık testinin kullanılabileceğini belirtmişlerdir. Bu noktada, elde edilen veri niteliğinin de belirleyici bir unsur olduğu, çünkü kullanılacak olan test yönteminin veri bütününe göre değişkenlik göstereceği rahatlıkla ifade edilebilir. Öyle ki, Alpar (2016), kullanılacak hipotez testinin araştırma planına, veri tipine, gözlem miktarına, verilerin normal dağılım durumlarına vb. bağlı olarak değişkenlik gösterebileceğini ifade etmiştir.

2.5. BİLGİ ERİŞİMDE İLGİLİLİK GERİBİLDİRİMİ VE SORGU GENİŞLETME MEKANİZMASI

Kullanıcıların bilgi gereksinimlerini gidermek amacıyla BES'ler üzerinde gerçekleştirdikleri sorgulamalar, çeşitli sebeplerden dolayı tekrar eden sorgulamalarla devam ettirilebilmektedir. Bu sebeplere, kullanılan BES'in ilgili dokümanları doğru sırada listeleyememesi veya kullanıcının ihtiyaç duyduğu bilgi için yeterli ipucu taşımayan terimlerle sorgulama yapması örnek olarak gösterilebilir. Bu gibi durumlarda, sebep ne olursa olsun hem sistemin daha iyi yanıt verebilmesi hem de kullanıcının ihtiyacını betimleyen terimi daha özelleştirilmiş biçimde keşfedebilmesi için bilgi erişimde “İlgililik Geribildirimi” ve “Sorgu Genişletme” olmak üzere iki farklı kavram bulunmaktadır. Her iki kavram bu bölüm altında sırasıyla açıklanmıştır.

2.5.1. İlgililik Geribildirimi (*Relevance Feedback*)

İlgililik geribildirimi, nihai sonuç kümesini iyileştirmek için kullanıcıyı bilgi erişim sürecine dahil etme işlemidir (Manning ve diğ., 2008). İlk olarak 1960 yılında önerilmiş olan ve belirli

bir sorguya karşılık ortaya çıkarılan sıralamadaki ilgili dokümanların belirlenmesi için kullanıcı etkileşimlerine dayanan bir sorgu geliştirme tekniğidir (Croft ve diğ., 2009). Kullanıcı tarafından yapılan bir sorguya karşılık, ilgili olarak tespit edilen dokümanlar kullanılarak, yapılan sorgulamanın modifiye edilmesinin gerçekleştirildiği bir geribildirim döngüsüdür (Baeza-Yates ve Ribeiro-Neto, 2011). Literatür incelendiğinde üç farklı türde ilgililik geribildiriminin bulunduğu görülmektedir. Bunlar, “Açık İlgililik Geribildirimi (*Explicit Relevance Feedback*)”, “Sözde İlgililik Geribildirimi (*Pseudo Relevance Feedback*)” ve “Örtük İlgililik Geribildirimi (*Implicit Relevance Feedback*)”dir (Manning ve diğ., 2008; Croft ve diğ., 2009; Baeza-Yates ve Ribeiro-Neto, 2011; Zhai ve Massung, 2016).

Açık ilgililik geribildirim, yapılan bir sorgulamaya karşılık, kullanılan BES tarafından getirilen doküman listesinde, sorgulamayı yapan kullanıcıdan yapılan önerilerin ilgili olup olmadığının değerlendirilmesinin istenmesi ve kullanıcıdan alınan değerlendirmeler doğrultusunda gelecekte önerilecek olan doküman sıralamasının geliştirilmesi (Zhai ve Massung, 2016) mantığına dayanmaktadır. *Açık ilgililik geribildirim*in temel çalışma prosedürü aşağıdaki şekildedir (Manning ve diğ., 2008; Büttcher ve diğ., 2010):

- Kullanıcı bir sorgu gönderir
- Sistem başlangıç doküman öneri listesini getirir
- Kullanıcı, dokümanlardan bazılarını “ilgili” ve “ilgisiz” olacak şekilde işaretler
- Sistem, kullanıcı geri bildirimlerine dayalı olarak bilgi ihtiyacının daha iyi bir temsilini hesaplar
- Sistem, revize edilmiş sonuçları görüntüler

Croft ve diğ. (2009), *açık ilgililik geribildirim* sürecinde gerçekleşen işlemleri makine öğrenmesinin bilgi erişimde kullanımının basit bir örneği olarak belirtmiş ve eğitim verisinin kullanıcılar tarafından yapılan değerlendirmeler olarak nitelendirilebileceğini ifade etmiştir. Bilgi erişimin geliştirilmesinde başarılı sonuçlar ortaya koyan bir geribildirim yöntemi olarak ifade edilmesine rağmen, çoğu kullanıcının bu geribildirimi verme noktasında isteksiz olduğu (Baeza-Yates ve Ribeiro-Neto, 2011; Zhai ve Massung, 2016) ve bu sebeple eğitim verisinin sınırlı olduğu (Croft ve diğ., 2009) belirtilmiştir. Bunun sebebi, *açık ilgililik geribildirim* yönteminde, sistemin kullanıcıyı arama yaptığı esnada kesintiye uğratma durumu (Yang ve diğ., 2016) olarak gösterilebilir.

Açık ilgililik geribildirim yöntemindeki kullanıcıya olan bağımlılığın ortadan kaldırılması için önerilen bir diğer geribildirim yöntemi ise *sözde ilgililik geribildirimidir*. Kullanıcıyla yapılan değerlendirmelerin otomatik olarak yapıldığı (Manning ve diğ., 2008) *sözde ilgililik geribildirimi*, kullanıcılardan geribildirim almak yerine, sistemin sıralamada en üstte yer alan dokümanları ilgili olarak varsaymasına (Croft ve diğ., 2009) dayanan bir yöntemdir. Bu varsayım temel alınarak, BES tarafından önerilen ilk 10 ya da ilk 20 doküman sistem arkaplanında incelenip, kullanıcılar tarafından yapılan sorgunun geliştirilmesi söz konusudur. Çünkü ilgili olmasalar bile BES altyapısında kullanılan bilgi erişim modelinin belirlediği bu sıralamada, üstte yer alan dokümanların ilgili olduğu kabul edilmektedir (Zhai ve Massung, 2016). *Sözde ilgililik geribildirim* yöntemi için örnek olarak, bir kullanıcının “yapay zeka” terimiyle bir arama yaptığı varsayalım. BES, bu terimle alakalı olarak tespit ettiği ilk 10 doküman incelemesinde, “makine öğrenmesi” ve “yapay sinir ağları” gibi iki yeni terimle, kullanıcının aradığı terimin ilişkili olduğunu tespit etsin. Bu durumda, BES, kullanıcının gönderdiği sorgu terimine ek olarak arkaplanında kendi bulduğu diğer terimleri de kullanıcı terimine ekleyip yeni bir sorgulama gerçekleştirmektedir. Son durumdaysa üç terimle ilgili dokümanlar kullanıcıya belirli bir sıralamayla sunulmaktadır. *Sözde ilgililik geribildiriminin* tamamen güvenilir bir yöntem olmadığı belirtilmiş olup (Zhai ve Massung, 2016), buna sebep olarak, keyfi bir kesinti noktasının belirlenmesi ve arkaplanda işlem yapan bilgi erişim modelinin en azından bazı yararlı dokümanları elde etmede yeterince iyi olduğu varsayımına dayanması olarak ifade edilmiştir (Croft ve diğ., 2009; Zhai ve Massung, 2016).

Açık ilgililik geribildiriminden daha az güvenilir fakat *sözde ilgililik geribildiriminden* daha kullanışlı (Manning ve diğ., 2008) bir diğer yöntem ise *örtük ilgililik geribildirimidir*. Bu yöntemde, kullanıcılar kesintiye uğratılmadan geribildirim döngüsüne dahil edilmekte ve kullanıcıların arama sonuçlarıyla nasıl etkileşime girdiği, tıklama verisi (*clickthrough data*) üzerinden elde edilerek incelenmektedir (Zhai ve Massung, 2016). Bu yöntemde, sistem tarafından oluşturulacak olan gelecekteki sıralamalarda kullanıcıların yaptıkları sorgular doğrultusunda inceledikleri/tıkladıkları dokümanların temel alınması söz konusudur. Kullanıcıları uğraştırmamasından ve düşük maliyetlerde elde edilebilir olmasından dolayı etkili bir yöntemdir (Baeza-Yates ve Ribeiro-Neto, 2011). *Örtük ilgililik geribildirimi*, çalışmanın 2.4.3 bölümünde aktarılan ve kullanıcı merkezli değerlendirme yöntemlerinden biri olan “Tıklama Verisi Testi” ile birebir ilişkili bir yöntem olup, bu çalışmada geliştirilmiş olan algoritmanın da temelini oluşturmaktadır.

2.5.2. Sorgu Geniřletme (*Query Expansion*)

Açık ilgililik geribildirim yönteminde kullanıcılar BES tarafından sunulan dokümanları ilgili olup olmamalarına göre işaretlerken, *sorgu genişletmede* bu durum kullanıcının yapmayı düşündüğü sorguda kullandığı arama terimine ek terimler eklemesi/ekletmesi şeklinde gerçekleştirilmektedir (Manning ve diğ., 2008). *Sorgu genişletmeye* örnek olarak, Google arama motorunun herhangi bir arama gerçekleştirilirken yazılmaya başlayan terimi tamamlayan farklı terimleri önermesi ya da arama yapıldıktan sonra sorulan “Bunu mu demek istediniz?” şeklinde düzenlenmiş arama terimi önerisi verilebilir. Croft ve diğ. (2009), *sorgu genişletmeyi* bir tür *ilgililik geribildirim* tekniğı olarak tanımlamıştır. Google örneğı düşünöldüğünde, tıpkı *ilgililik geribildirim* yöntemlerinde olduğı gibi bir sorgu geliştirme durumunun söz konusu olduğı görölmektedir. *Sorgu genişletme*, sistemde barındırılan dokümanlarda, yapılan sorgulamalarda, eşanamlılar sözlüğünde (*thesaurus*) ya da BES tarafından önerilen sonuç listesinde en üstte yer alan dokümanlarda geçen terimlerin ya da kelimelerin, birlikte var olma durumlarının analizine dayanmaktadır (Croft ve diğ., 2009). Uygun terimlerin seçilmesinin önemli olduğı ifade edilmiş olan *sorgu genişletmede*, terimler arasındaki ilişkinin derecesini belirlemede faydalanan çeşitli yaklaşımlar bulunmaktadır (Croft ve diğ., 2009). Bu yaklaşımlara, Dice's Coefficient, Mutual Information, Expected Mutual Information ve Pearson Chi-squared yaklaşımları örnek olarak gösterilebilir.

Sorgu genişletme işleminde genellikle ilişkilerinin tespiti yapılan bir terimler havuzu görev yapmaktadır. İlişkili terimler havuzunun oluşturulmasında kullanılan yöntemler Manning ve diğ. (2008) tarafından aşağıdaki şekilde ifade edilmiştir:

- İnsan editörler tarafından sağlanan kontrollü bir sözlük kullanımı
- İnsan editörlerin, kanonik bir terim belirtmeden kavramlar için eşanamlı isimler oluşturduğı bir eş anamlılar sözlüğünün kullanımı
- Bir doküman koleksiyonu üzerindeki terimlerin birliktelik istatistiklerinin kullanılmasıyla otomatik olarak oluşturulan bir eşanamlılar sözlüğünün kullanımı
- Bir kullanıcıya önerilerde bulunmak için diğ er kullanıcıların sorgularının da tutulduğı günlük (*log*) dosyalarının madenciliğıne dayalı yöntem

Croft ve diğ. (2009), *sorgu genişletme* için terimler arasındaki ilişki tespitinde ve benzer sorgu terimlerinin bulunmasında en iyi kaynağın günlük (*log*) dosyaları olduğunu ifade etmiştir. Bu

çalışmada geliştirilmiş olan algoritma, doğal kullanıcı davranışlarını barındıran günlük dosyalarına benzer bir yapıdan faydalanarak, *sorgu genişletme* işlemini gerçekleştirmektedir. Detayları çalışmanın 3. bölümünde açıklanmış olan bu yapıda, günlük dosyalarındaki etkileşim kaydı tutma biçimine benzer bir işlem farklı bir yolla gerçekleştirilmiş ve Croft ve diğ. (2009) tarafından yukarıda ifade edilmiş olan dördüncü yöntem çerçevesinde *sorgu genişletme* işlemi gerçekleştirilmiştir.

2.6. YAPAY ZEKÂ

Bu bölümde, ilk olarak Yapay Zekâ (YZ) teriminde yer alan “Yapay” ve “Zekâ” kelimelerinin kavramsal anlamına değinilmiş, sonrasında YZ’nin ne anlama geldiği, tarihçesi, hangi alanlarda kullanıldığı ve makine öğrenmesinin YZ literatüründe ne anlam ifade ettiği aktarılmıştır. Bunlara ek olarak, çalışmada geliştirilmiş olan algoritmanın ne açıdan bir YZ algoritması olduğu açıklanmaya çalışılmıştır.

“Yapay” terimi Lucci ve Kopec (2015) tarafından *sentetik (insan yapımı)* olarak tanımlanmış ve bu türdeki nesnelerin genellikle gerçek veya doğal nesnelerden daha üstün olduğu ifade edilmiştir. Bu nesnelere, ateşe ihtiyaç duyduğumuz zaman kullandığımız çakmak veya günlük hayatımızda sıklıkla faydalandığımız parfümler örnek olarak gösterilebilir. Yapay Zekâ kavramı da bir bakıma doğallık içermeyen, tam aksine, insan yapımı şeklindeki bir oluşuma (Lucci ve Kopec, 2015) işaret etmektedir. Diğer taraftan, “zekâ” kelimesinin kavramsal olarak aşağıdaki mental eylemleri içerdiği ifade edilmiştir (Mueller ve Massaron, 2018):

- *Öğrenme*: Yeni bilgileri alma ve işleme becerisine sahip olmak.
- *Sebeplendirme*: Bilgiyi çeşitli şekillerde manipüle edebilmek.
- *Anlama*: Bilgi manipülasyonunun sonucunu göz önünde bulundurabilmek.
- *Gerçekleri ayırt edebilme*: Değiştirilen bilginin geçerliliğini belirleyebilmek.
- *İlişkileri tespit edebilme*: Doğrulanmış verilerin diğer verilerle nasıl etkileşime girdiğini belirlemek.
- *Anlamları göz önünde bulundurma*: Gerçekleri, ilişkileriyle tutarlı bir şekilde belirli durumlara uygulamak.
- *Gerçek ile inanışı birbirinde ayırma*: Verilerin tutarlı bir şekilde geçerli olduğu kanıtlanabilen kanıtlanabilir kaynaklar tarafından yeterince desteklenip desteklenmediğini belirleyebilmek.

Yukarıda ifade edilen her bir eylem, günlük yaşamda yaptığımız çoğu etkileşimde ayrı ayrı ya da birlikte olacak şekilde farkında olmasak da gerçekleştirdiğimiz eylemlere karşılık gelmektedir. Herhangi bir amacı yerine getirmek için ilk “başlama” kararının verilmesinden itibaren başarılı ya da başarısız olarak amaca yönelik sonuca ulaşma anına kadar geçen süreçte, zihinsel olarak geçilen aşamalar olarak da belirtilebilir. Bir bilgisayar sistemi açısından zekânın nasıl taklit edilebileceği ise şu şekilde ifade edilmiştir (Mueller ve Massaron, 2018):

1. İstek ya da ihtiyaca göre bir hedef belirleme
2. Hedefi desteklemek için halihâzırda bilinen herhangi bir bilginin önemini değerlendirme
3. Hedefi destekleyebilecek ek bilgiler toplama
4. Verileri, mevcut bilgilerle tutarlı bir form elde edecek şekilde değiştirme
5. Mevcut ve yeni bilgiler arasındaki ilişkileri ve doğruluk değerlerini tanımlama
6. Hedefe ulaşıp ulaşılmadığına karar verme
7. Yeni veriler ışığında hedefi ve başarı olasılığına etkisini değiştirme
8. Hedefe ulaşılan (doğru bulunana) veya hedefe ulaşma olasılıkları tükenene (yanlış bulunan) kadar 2'den 7'ye kadar olan adımları gerektiği kadar tekrarla

Genel itibariyle, hem “yapay” hem de “zekâ” terimlerinin bir araya getirilmesiyle oluşturulan YZ kavramı, “zihinsel hesaplama gerektiren soyut işlemleri, fiziksel etkileşim içeren somut işlemleri veya hem soyut hem somut eylem gerektiren çeşitli işlemleri, insan doğasında var olan eylemleri taklit ederek gerçekleştirebilen ve insan üretimi doğrultusunda hayat bulan sistemlerin, yazılımların veya algoritmaların oluşturulmasıyla ilgilenen bir alan” şeklinde tanımlanabilir. Bu tanıma, “YZ özelliğindeki herhangi bir insan yapımı oluşumun/varlığın, farklı bir YZ oluşumunu/varlığını insan yardımı olmadan ortaya çıkarabileceği bir alan” şeklinde biraz daha fütüristik bir ekleme yapmak da mümkündür.

YZ, düşünceyi mekanize etmenin mümkün olduğu hipotezi üzerine inşa edilmiştir (Mueller ve Massaron, 2016). YZ kavramının öncülerinden biri olan John McCarthy, 1955 yılında “zeki gibi davranan makineler” tanımı yaparak YZ’nin temel hedefini ifade etmiştir (Ertel, 2017). Mueller ve Massaron (2018), YZ’nin insan zekâsıyla hiçbir ilgisinin olmadığı ve sadece bir zekâ simülasyonu olduğunu belirtmiştir. Poole ve Mackworth (2017) tarafından, “akıllıca hareket eden hesaplama ajanlarının sentezi ve analiziyle ilgilenen bir araştırma alanı” olarak ifade edilmiş olan bu kavram, Britannica Concise Encyclopedia (2007b) tarafından da “bir

makinenin insan zekâsı gerektirdiği düşünölen görevleri yerine getirme yeteneđi” şeklinde tanımlanmaktadır. YZ’nin insan hedefleri veya bu hedeflere ulaşma yöntemleri ile ilgisi olabilecek veya olmayabilecek bir sonuca ulaşmak için algoritmalara güvendiđi belirtilmiş olup (Mueller ve Massaron, 2018), bir bakıma zekiymiş gibi işlem yapan bir yapının tarifi yapılmıştır. Dolayısıyla, YZ, sanki bir zekâya sahipmiş gibi davranabilen yapılar olarak nitelendirilebilir. Bu yapılara, robotlar, yazılımlar veya algoritmalar örnek olarak gösterilebilir.

Bir yapının YZ kavramına karşılık gelip gelmediğinin belirlenmesi noktasında, Turing (1950) tarafından “taklit oyunu (*imitation game*)” olarak adlandırılan bir test (Turing testi olarak da bilinmektedir) önerilmiştir. Bu test, bir insan sorgulayıcı ve bu sorgulayıcının göremediđi bir yerde bulunan bir bilgisayar ve bir insandan oluşan bir ortamda gerçekleştirilmektedir. Bu ortamda, sorgulayıcı ve diğör taraf arasında haberleşmenin sağlanabilmesi için bir cihaz kullanılmaktadır. Bunun sebebi, sorgulayıcının sorduđu sorular doğrultusunda alacađı cevapların, herhangi bir sesli dönüş yerine tek bir cihaz üzerinden elde edilmesinin sağlanması ve böylelikle sorgulayıcının, cevaplayanın insan mı ya da bilgisayar mı olduğunu tespit edememesinin sağlanmasıdır. Sorgulama süreci sonrasında, sorgulayıcının aldıđı cevaplar doğrultusunda hangi cevabın kime ait olduğunu tespit etmesi istenmektedir. Eğer sorgulayıcı böyle bir ayrımı gerçekleştiremezse, bilgisayar (yapay) zekâsının, insanın zekâsı ile aynı olduđu sonucuna varılmaktadır. Herhangi bir fiziksel etkileşimin olmadığı bu deneyin bir diğör türü ise “toplam Turing testi (*total Turing test*)” şeklinde adlandırılmaktadır. Bu testte, sorgulayıcının yapay varlığın dış görünüşünden haberdar olduđu bir durum söz konusu olup, sorgulayıcı, varlığın algılama ve gezinme yeteneğini test edebilmektedir (Neapolitan ve Jiang, 2018).

Turing tarafından önerilen teste alternatif olarak Searle (1980) tarafından “Çin odası deneyi (*Chinese room*)” adıyla farklı bir test önerilmiştir. Bu test, bir odada yer alan ve Çince karakterlerle yazılmış bir soruyu alıp işledikten sonra yine Çince çıktı veren bir bilgisayar programını (Çinceyi işleyebilen) içermektedir. Programın sorgulayıcıyı ikna edebildiđi durumda testi (*Turing testi*) geçtiđi varsayılmaktadır. Bu testin diğör aşamasındaysa, Çince bilmeyen bir insan ve Çince yazım kitabının yer aldıđı bir durum söz konusudur. Bu noktada, sorgulayıcının sorularına karşılık insan denek doğru cevap verdiđi takdirde, şu soru Searle (1980) tarafından sorulmaktadır: “Odadaki denek Çince bilmekte midir?”. Odadaki deneğın Çince bilmemesine rağmen, doğru cevaplar üretebilmesi, “bilgisayarın da Çince’yi anlamadıđı” sonucunu ortaya çıkarmaktadır. Bu doğrultuda, Searle (1980), **güçlü** ve **zayıf** olmak üzere iki

farklı YZ kavramını ortaya atmıştır. Güçlü YZ (strong AI), uygun şekilde programlanmış ve anlayabilen ve bilişsel durumu olan bilgisayarlar olarak tanımlanmışken, Zayıf YZ (weak AI), hipotezleri daha titiz ve kesin bir şekilde formüle etmemizi ve test etmemizi sağlayan güçlü bir araç olarak tanımlanmıştır (Searle, 1980). Bu açıklamalara ek olarak, Searle (1980), güçlü YZ'nin mümkün olmadığını da ifade etmiştir.

Güçlü YZ'nin var/mümkün olup olmadığı tartışması geçmişten beridir yapılıyor olmasına rağmen, göz ardı edilemeyecek olan şey, YZ'nin, günümüzde birçok alanda yararlanılan bir konumda bulunmasıdır. YZ'nin kullanıldığı alanlardan bazıları aşağıdaki şekildedir (Baeza-Yates ve Ribeiro-Neto, 2011; Mueller ve Massaron, 2018):

- Doğal dil işleme
- Dolandırıcılık tespiti
- Kaynak planlama
- Karışık analiz/tıbbi teşhis
- Otomasyon
- Bilgisayarlı görme
- Market analizi
- Bilgi erişim
- Müşteri servisi
- Güvenlik sistemleri
- Makine kaynak verimliliği
- Genetik
- Sağlık

Akıllı telefonlar, arabalar, evde kullandığımız dijital ısı termometreleri, vb. günlük hayatımızda faydalanmakta olduğumuz çoğu aracın aslında, YZ'yi içinde barındıran bir tasarıma sahip olduğu rahatlıkla belirtilebilir. Her ne kadar günümüzün vazgeçilmez bir parçası olsa bile, YZ için “YZ kışı (*artificial intelligence winter*)” olarak nitelendirilen dönemler bulunmaktadır (Mueller ve Massaron, 2018; Schuchmann, 2019). İlkinin 1973, ikincisinin ise 1988 yılında başladığı belirtilen bu dönemlerde (Schuchmann, 2019), YZ'nin, savunucuları tarafından mümkün olanın abartıldığı bir yolu takip etmesinden kaynaklanan ve bu doğrultuda, teknoloji bilgisi olmayan, ancak çok parası olan insanların yatırım yapmaya teşvik edildiği bir dönemden

geçilmiştir (Mueller ve Massaron, 2018). Bu süreç sonunda, beklenenin karşılanamadığı görüldüğündeyse, YZ'ye olan yatırımların kısıldığı bir sürece girilmiştir (Mueller ve Massaron, 2018). Bu geçilen süreçlere rağmen, YZ'nin yeni bir döneme girdiği ve bu dönemin başlamasındaki temel faktörün “Makine Öğrenmesi (*Machine Learning*)” olduğu belirtilmiştir (Mueller ve Massaron, 2018). Makine Öğrenmesi, “programın yazarı tarafından açıkça programlanmayan bir davranış üretmeyi öğrenebilen bir bilgisayar programı” şeklinde tanımlanmaktadır (Joshi, 2020). Alpaydın (2016) tarafından YZ için bir gereklilik olarak ifade edilmiş olan bu kavram, bir insanın yapabileceğinden daha hızlı analiz ve tahmin performansına sahip bir yapıya (Mueller ve Massaron, 2016) işaret etmektedir.

Makine öğrenmesi, temel olarak “Denetimli Öğrenme (*Supervised Learning*)”, “Denetimsiz Öğrenme (*Unsupervised Learning*)” ve “Pekiştirmeli Öğrenme (*Reinforcement Learning*)” olmak üzere üç farklı öğrenme türünü içinde barındırmaktadır. *Denetimli öğrenme*, bir algoritmanın, yeni örnekler verildiğinde doğru yanıtı tahmin edebilmek için, var olan örnek verilerdeki sayısal değerler veya sınıflar/etiketler gibi dize etiketlerinden oluşabilen ilişkili hedef yanıtlarından, öğrenmeyi gerçekleştirdiği bir öğrenme biçimidir (Mueller ve Massaron, 2016). *Denetimli öğrenmenin* amacı, daha önce hiç görülmemiş örneklerle de çalışması gereken bir sistemi eğitmektir (Bonaccorso, 2017). *Denetimli öğrenmede* doğru değerleri bir denetçi tarafından sağlanan girdiden çıktıya bir eşleştirmeyi öğrenmek amaç iken, bir diğer öğrenme biçimi olan *denetimsiz öğrenme* için sadece denetçisi olmayan bir girdi söz konusudur (Alpaydın, 2014). *Denetimsiz öğrenme*, bir dizi ögenin benzerliklerine göre nasıl gruplanabileceğinin öğrenilmesi gerektiğinde kullanışlıdır (Bonaccorso, 2017). Girdideki düzenlerin bulunmaya çalışılması *denetimsiz öğrenmenin* amacıdır (Alpaydın, 2014). Elde var olan verinin içyüzünü anlama noktasında oldukça kullanışlı olan *denetimsiz öğrenme*, aynı zamanda, *denetimli öğrenmeye* yönelik girdi olarak nitelendirilebilecek veriyi sağlamada da faydalıdır (Mueller ve Massaron, 2016). Yani, *denetimsiz öğrenme* süzgecinden geçirilmiş olan veri, bir bakıma, *denetimli öğrenme* için girdi olarak kullanılabilir.

Detayları Bölüm 3’de aktarılmış olan ve bu çalışma kapsamında geliştirilmiş olan algoritmanın işleyiş biçimi, *denetimsiz öğrenmenin* çalışma mantığıyla birebir uyumaktadır. Dolayısıyla bu tez çalışması, *denetimsiz öğrenme* özelliğindeki bir öğrenme algoritmasına ait performansın incelendiği bir çalışma olarak nitelendirilebilir.

Diğer bir öğrenme türü olan *pekiştirmeli öğrenme*, herhangi bir girdi denetçisi içermemesinden dolayı *denetimsiz öğrenme* ile benzerlik taşımaktadır. Buna ek olarak, çevresel olarak elde ettiği geribildirimden beslenen bir yapıya sahiptir (Bonaccorso, 2017). Hangi durumda hangi eylemin gerçekleştirileceğine karar vermek için gereken ilkeleri öğrenmenin bir yoludur (Flach, 2012). Tek bir eylem yerine, hedefe ulaşmak için yapılması gereken eylemlerden oluşan bir dizi ilkeler bütününe önemli olduğu (Alpaydın, 2014) bir öğrenme biçimidir.

Bilgi erişim literatüründe, kullanıcılardan elde edilen geribildirime bağlı kalınarak, kaynak önerisindeki sıralamayı geliştirme tekniği olan *ilgililik geribildirimi* ilk olarak 1960’larda tanıtılmış olup, bu yöntemin, eğitim verisi üzerinden ilgili dokümanları ilgisizlerden ayırt edebilmeyi sağlayan basit bir makine öğrenmesi algoritmasına karşılık geldiği ifade edilmiştir (Croft ve diğ., 2009). Bilgi erişim açısından makine öğrenmesinin kullanılmasında çoğunlukla odaklanılan alanın “metin sınıflandırma (*text classification/categorization*)” olduğu söylenmiş olup (Croft ve diğ., 2009; Baeza-Yates ve Ribeiro-Neto, 2011), baskın olarak kullanılan algoritmanın, aynı zamanda bir *denetimli öğrenme* algoritması olan “Karar Destek Makineleri (*Support Vector Machines - SVM*)” olduğu belirtilmiştir (Manning ve diğ., 2008; Croft ve diğ., 2009). Makine öğrenmesi içeren çoğu BES’in, düşük miktardaki sorgu sayısından ve *ilgililik geribildiriminden* dolayı öğrenme noktasında kısıtlı veriye sahip olduğu söylenmiş, bu dezavantaja rağmen, *örtük ilgililik geribildirim* verisi taşıyan günlük (*log*) dosyalarının bu konuda büyük bir eğitim verisi potansiyeli taşıdığı ifade edilmiştir (Croft ve diğ., 2009).

Bilgi erişimde herhangi bir bilgi ihtiyacına karşılık ilgili dokümanların önerilebilmesi temel problem olarak görülmektedir (Liu, 2009). Bu noktada, eğitim verisinden faydalanarak, ilgili/önemli dokümanların sıralanmasını sağlayabilecek bir modelin otomatik olarak oluşturulması (Liu, 2009) şeklinde tanımlanan ve bilgi erişim literatüründe “sıralamayı öğrenme (*learning to rank*)” şeklinde adlandırılan bir çalışma alanı öne çıkmaktadır. Bu alan, temel olarak aşağıdaki şekilde üç farklı yaklaşımı içinde barındırmaktadır (Liu, 2009; Baeza-Yates ve Ribeiro-Neto, 2011; Yang ve diğ., 2016):

- *Noktasal (pointwise)*: her bir tekil belgenin ikili (*binary*) ilgililik durumu bir sınıflandırıcı kullanılarak öğrenilir. Bir başka ifadeyle, bir sorguya karşılık ilgili olan dokümanların belirlenmesidir.
- *Eşsel (pairwise)*: doğrudan bir ilgililik durumu öğrenmek yerine, eşli olarak ikili belgelerin doğru sıralaması öğrenilir. Bir başka ifadeyle, iki bilgi kaynağı arasındaki

sıralama ilişkisinin barındırıldığı bir öğrenme yaklaşımıdır (örn; $S_1 > S_2$ ise, S_1 kaynağının S_2 'den daha ilgili olduğu anlaşılır).

- *Listesel (listwise)*: ilgili dokümanların tüm sıralamasının barındırıldığı bir öğrenme yaklaşımıdır.

Üstte ifade edilmiş olan *sıralamayı öğrenme* yaklaşımlarının tamamı, makine öğrenmesindeki *denetimli öğrenme* üzerinden gerçekleştirilmektedir. Yani, tamamında bir eğitim ve bu eğitim doğrultusunda öğrenme eylemi söz konusudur. Her bir yaklaşıma yönelik olarak, literatürde hem farklı algoritmalar önerilmiş hem de var olan *denetimli öğrenme* algoritmaları kullanılmıştır. Böylelikle, bilgi erişimin, makine öğrenmesinin gücüyle birlikte gelişimine katkı sağlanmıştır. Bu çalışmada geliştirilen algoritma, *sıralamayı öğrenme* yaklaşımı altında değerlendirilmemesine rağmen, *denetimsiz öğrenme* mantığıyla çalıştığından dolayı bilgi erişimde kullanılabilecek bir makine öğrenmesi algoritması olarak nitelendirilmiştir.

2.7. LİTERATÜR

Bu bölümde, literatürde yapılmış olan ve bu tez çalışmasının altyapısıyla benzerlik taşıyan çeşitli araştırmalar aktarılmıştır:

Bilgi erişim kalitesinin otomatik olarak geliştirilebilmesinde “tıklama verisi”nin kullanılmasına yönelik bir yöntem önerilen çalışma Joachims (2002) tarafından gerçekleştirilmiştir. *denetimli öğrenme* algoritmalarından karar destek makineleri'nin kullanıldığı bu çalışmada, önerilen yöntemin, pratikte gayet iyi çalıştığı ortaya çıkarılmıştır. Buna ek olarak, birkaç yüz veriyle modelin eğitilmesi sonrasında, erişim kalitesinde Google'ın geçildiği ifade edilmiştir.

Hiyerarşik doküman kümeleme işleminin, sorguya özgü kümeleme işleminden geçirilmiş olan arama sonuçlarına uygulanmasının, bilgiye erişimde etkililiği arttırıp arttırmadığına yönelik olası potansiyelinin incelendiği bir çalışma Tombros ve diğ. (2002) tarafından gerçekleştirilmiştir. Dört farklı hiyerarşik kümeleme algoritmasının, beş farklı doküman koleksiyonu üzerinden testinin gerçekleştirildiği bu çalışma sonucunda, sorguya özgü kümeleme işleminin etkililik açısından oldukça iyi performans ortaya çıkardığı ifade edilmiştir.

Etkileşimli (*interactive*) bilgi erişimde, kullanıcı ihtiyaçlarının zamanla değişkenlik göstereceğinin ifade edildiği bir çalışmada, bu değişkenliklerin takibiyle birlikte bilgi ihtiyacını daha iyi temsil edebilecek terimlerin önerilebileceğini temel alan bir *örtük ilgililik geribildirim*

yaklaşımı sunulmuştur (White ve diğ., 2006). 24 kişiden oluşan bir denek grubuyla gerçekleştirilmiş olan bu çalışma sonucunda, yazarlar, önerdikleri *örtük ilgililik geribildirim* temelli terim öneri yapısının kullanışlı olduğunu belirtmişlerdir. Ek olarak, bilgi ihtiyacındaki değişikliklerin izlenmesiyle üretilmiş olan kanıtların, denekler tarafından beğenilen erişim stratejilerinin oluşturulduğuna işaret ettiği ifade edilmiştir.

Odağında *örtük ilgililik geribildirim* olarak “tıklama verisi” kullanımını barındıran bir başka çalışma Jung ve diğ. (2007) tarafından gerçekleştirilmiştir. *Örtük ilgililik geribildirim*inin tüm kullanıcılara yönelik olarak arama sonuçlarındaki kaliteyi geliştirip geliştiremeyeceğinin incelendiği bu çalışmada, aynı zamanda, *açık ilgililik geribildirim* yönteminden de faydalanılmıştır. Çalışma sonuçlarında, *örtük ilgililik geribildirim*inin hem P hem de R değerlerinde artışa sebep olduğu ortaya çıkartılmış olup, “en son ziyaret edilmiş doküman” üzerinden hareket edildiği takdirde, en yüksek P değerine ulaşıldığı ifade edilmiştir.

Biyomedikal BES alanında yapılmış olan bir çalışmada, bilgi erişimde etkililik faktörünün geliştirilmesi amacıyla PubMed veritabanı sorgulamasına yönelik bir sistem önerisi yapılmıştır (Huang ve Hsu, 2008). P , R ve F metrikleriyle ölçüm yapılan çalışmanın temelinde, kullanıcı sorgusuyla ilişkili örtük kelimelerin bulunması amacıyla tasarlanmış bir “kelime-ilişki ağacı” yapısının test edilmesi yatmaktadır. Çalışma sonuçlarında, her ne kadar P değerinin arttırılamadığı belirlenmiş olsa da, R değerinde oldukça yüksek bir başarı yakalandığı yazarlar tarafından ifade edilmiştir.

Arama motorlarında bir sorguya karşılık önerilen bilgi kaynaklarının sıralanmasına yönelik olarak Akbari Torkestani (2012) tarafından bir *sıralamayı öğrenme* algoritması önerilmiştir. Bilgi kaynaklarının daha hassas bir şekilde bulunabilmesinin amaç olduğu bu çalışmada, kullanıcı geribildirimi, “kullanıcının bir bilgi kaynağına olan ilgisi” şeklinde tanımlanmıştır. Bir başka ifadeyle, kullanıcılar tarafından ziyaret edilen dokümanların kullanıcı sorgusuyla daha fazla ilgili olduğu düşüncesi çalışmada yer almaktadır. Çalışmada, popüler veri koleksiyonları üzerinden MAP , $P@n$ ve bu çalışmada açıklanmamış olan bir başka metrik olan $NDCG$ (*Normalized Discounted Cumulative Gain*) kullanılarak etkililik ölçümleri gerçekleştirilmiştir. Çalışma sonuçları, arama motoruna gönderilen sorgu sayısı arttıkça, önerilen algoritmanın, en ilgili bilgi kaynaklarını en üst sıralarda sunabilmeyi öğrendiğini ortaya çıkarmıştır. Buna ek olarak, algoritmanın, SVMRank, LREG ve LRDRS gibi farklı algoritmalarından daha başarılı performans gösterdiği belirlenmiştir.

Çok araçlı tıbbi bir BES önerisinde bulunan bir başka çalışma Mourão ve diğ. (2015) tarafından gerçekleştirilmiştir. Önerilen sistem, görüntü ve metin işleme algoritmalarına dayalı olmakla birlikte, *sorgu genişletme* altyapısı da bulunan birçok araçlı arama arayüzüne sahiptir. *Sorgu genişletme* için farklı bir sözlükten faydalanılmış olan bu çalışmada, yazarlar, önerdikleri sistemin başarılı bir şekilde işlem gerçekleştirdiğini ifade etmişlerdir.

Bilgi erişimde kelime anlam ayrıştırması yapabilen bir *denetimsiz öğrenme* algoritmasının önerildiği bir başka çalışma Chifu ve diğ. (2015) tarafından gerçekleştirilmiştir. Geliştirilen yöntem, spektral kümelemeye dayanan ve hedef sorguya anlamsal olarak benzer olan bilgi kaynaklarını geliştirerek başlangıçta alınan kaynak öneri listesinin yeniden sıralama işlemini gerçekleştirmektedir. $P@n$ metriği üzerinden etkililik ölçümlerinin gerçekleştirildiği çalışma sonuçlarında, belirsiz terimlerden oluşan sorgulara yönelik olarak önerilen kaynak listesinin daha hassas bir şekilde geliştirilebildiği ortaya çıkarılmıştır.

Malizia ve diğ. (2017), karınca kolonisi optimizasyonu'ndan (ant colony optimization) esinlenerek kullanıcı etkileşimlerinden faydalanmak ve bu doğrultuda daha ilgili bilgi kaynaklarını ortaya çıkarmak için üç farklı türde algoritma önermişlerdir. Veriseti olarak AOL ve Yahoo tarafından sağlanan günlük (*log*) dosyaları kullanılmış olup, etkililik ölçümü için NDCG'den faydalanılmıştır. Yazarlar, gerçek zamanlı örtük işbirliğine dayalı bilgi arama algoritmaları önermeyi amaçlayan bu çalışma sonucunda, etkililiğini kanıtladıkları her bir algoritmanın farklı amaçlar doğrultusunda kullanılabileceğine yönelik açıklamalarda bulunmuşlardır.

K-ortalamlar algoritmasıyla benzer dokümanların kümelemesinin yapıldığı, DCI_Closed algoritmasıyla bu kümelerdeki sıklıkla tekrar eden terimlerin ortaya çıkarıldığı ve son olarak da arı sürüsü optimizasyonu (bees swarm optimization) doküman kümelerinin derinlemesine incelendiği bir diğer çalışma, Djenouri ve diğ. (2018) tarafından gerçekleştirilmiştir. Çalışmada, yazarlar tarafından önerilen yöntem, farklı yaklaşımlarla kıyaslanmış ve bu kıyaslamalarda birbirinden farklı doküman koleksiyonlarından faydalanılmıştır. Çalışma sonuçlarında, önerilen yaklaşımın, kıyaslama yapılan diğer yaklaşımlarla karşılaştırıldığında rekabetçi bir hesaplama süresine sahip olduğu ortaya çıkarılmıştır. Bu sonuca ek olarak, önerilen yaklaşım sayesinde, sorgu sonucunda önerilen bilgi kaynaklarındaki kalitenin önemli ölçüde arttırıldığı tespit edilmiştir.

Yahoo tarafından sağlanan bir veri setinden faydalanılan ve bilgi kaynakları arasındaki ilişkilerin tespiti doğrultusunda sorgu sonuçlarının yapılandırılmasına odaklanan hibrid bir modelin önerildiği başka bir çalışma Khalifi ve diğ. (2018) tarafından gerçekleştirilmiştir. *Denetimsiz öğrenmeye* yönelik k-ortalamlar, *denetimli öğrenmeye* yönelik olarak karar destek makineleri algoritmasından faydalanılmış olan bu çalışmada, P , R ve F metrikleri üzerinden etkililik değerlendirmesi yapılmıştır. Yazarlar, önerdikleri modelin önemli derecede tatmin edici sonuçlar verdiğini ifade etmişlerdir.

İşlem süresinin “verimlilik”, önerilen bilgi kaynaklarının kalitesinin ise “etkililik” olarak nitelendirildiği bir çalışma, Djenouri ve diğ. (2018) tarafından gerçekleştirilmiştir. Bu çalışma, bir *denetimsiz öğrenme* algoritması olan k-ortalamlar algoritmasından ve kapalı yaygın öge kümesi algoritmalarından DCI_Closed algoritmasının kullanıldığı bir çalışmadır. Yazarlar tarafından yeni bir bilgi erişim yaklaşımının önerildiği bu çalışma sonucunda, önerilen yöntem sayesinde, hem etkililik hem de verimlilik bakımından geleneksel yaklaşımlardan daha iyi performans gösterildiği ortaya çıkarılmıştır.

Farklı türde *sorgu genişletme* tekniklerinin incelendiği, bilgi tabanlı (*knowledge-based*) ve kitaplık tabanlı (*corpus-based*) teknikleri içeren yeni bir *sorgu genişletme* modelinin önerildiği bir başka çalışma, Nasir ve diğ. (2019) tarafından gerçekleştirilmiştir. MAP metriği ile birlikte bu çalışmada aktarılmayan 11 noktalı ara değerli ortalama hassasiyet (*11-point interpolated average precision*) metriğinden faydalanılan bu çalışmada, bilgi ve kitaplık tabanlı *sorgu genişletme* teknikleri sayesinde bilgi erişim performansının *ilgililik geribildirimi* açısından geliştirilebildiği ortaya çıkarılmıştır.

Kullanıcıların orijinal sorgusunun uygun ek terimlerle genişletilmesi ve daha ilgili bilgi kaynaklarının döndürülmesi amacıyla, k-medoit ve k-ortalamlar olmak üzere iki farklı *denetimsiz öğrenme* algoritmasının kullanıldığı bir çalışma yapılmıştır (Khennak ve diğ., 2019). Yazarların, MAP ve $P@5$ etkililik metrikleri temelinde değerlendirme yaptıkları bu çalışmada, MEDLINE üzerinden elde edilmiş veri setini kullanmış olup, bilgi erişim performansının geliştirildiği ifade edilmiştir.

Büyük bilgi koleksiyonlarından kullanıcı merkezli bilginin elde edilmesi amacıyla Bhopale ve Tiwari (2020) tarafından gerçekleştirilen bir çalışmada, sürü zekası destekli optimize edilmiş küme tabanlı bir model önerisi yapılmıştır. Bu model, aynı zamanda veri madenciliğinde

çoğunlukla market sepet analizlerinde kullanılan örüntü madenciliği (*pattern mining*) tekniğini de içermektedir. Bilgi kaynaklarının kümelenmesi işleminin, yazarlar tarafından önerilen ve *denetimsiz öğrenme* altyapılı bir “K-Flock” isimli bir algoritmayla gerçekleştirildiği bu çalışmada, önerilen model sayesinde, geleneksel bilgi erişim yaklaşımlarından daha etkili ve verimli performans gösterildiği belirtilmiştir.

Sosyal medyada yer alan kısa metinlerdeki arama performansını artırabilecek örtük ilişkileri yakalamayı amaçlayan bir başka çalışma, Albishre ve diğ. (2020) tarafından gerçekleştirilmiştir. Yazarlar tarafından sorgu temelli yeni bir *denetimsiz öğrenme* modelinin önerildiği bu çalışmada, birden fazla popüler modelle kıyaslama yapılmış olup, *MAP* ve *P@30* olmak üzere iki farklı etkililik metriği kullanılmıştır. Etkililik bakımından başarı sağlandığı ifade edilen çalışma sonucunda, yazarlar, önerdikleri model sayesinde, soru cevaplama ve zaman çizelgesi özetleme gibi potansiyel bilgi erişim görevlerinde de sosyal medya arama etkinliğinin arttırılabileceğini belirtmişlerdir.

Bir sorguya karşılık gösterilen bilgi kaynaklarının kalitesinin geliştirilmesinde kullanılmak amacıyla, *sözde ilgililik geribildirim* mantığıyla çalışan, ilgililik ve anlamsal eşleştirme içeren bir model Wang ve diğ. (2020) tarafından önerilmiştir. Anlamsal eşleştirmenin, “iki metin arasındaki anlamsal ilişki ve bilginin tespiti”, ilgililik eşleştirmesinin, “bir sorguya karşılık bir dokümanın ilgili olup olmadığının belirlenmesi” şeklinde tanımlandığı bu çalışmada, *MAP* ve *P@10* metrikleri kullanılmıştır. Farklı modellerle, önerilen modelin kıyaslamasının da yapıldığı bu çalışma sonucunda, önerilen modelin etkililik bakımından başarılı olduğu belirtilmiş ve tek başına ilgililik ya da anlamsal eşleştirmeden faydalanmak yerine, her ikisinin kullanılmasının daha etkili bir yaklaşım olduğu ifade edilmiştir.

3. MALZEME VE YÖNTEM

Çalışmanın bu bölümünde, bilgi erişimde “kullanışlı bilgiye erişim” hedefinden yola çıkılarak geliştirilmiş olan bir algoritmanın (Analogy) çalışma felsefesi, bu algoritmanın Kırklareli Üniversitesi (KLU)’nde hâlihazırda kullanılmakta olan arama araçlarına entegrasyon süreci, algoritmanın yapay zeka olarak çalıştırılma yöntemi ve analizlerde kullanılacak veri kümesinin elde edilme süreci, algoritma başarısının nasıl değerlendirildiği ve tüm çalışma sürecinde kullanılan yazılımsal ve sistemsal altyapı hakkında bilgiler aktarılmıştır.

3.1. ÇALIŞMA KAPSAMINDA GELİŞTİRİLEN ALGORİTMA

09.07.2019 tarihinde 2019/10260 başvuru numarası ve “Analogy: Yoğunluk Tabanlı Bir Bilgi Erişim Optimizasyon Yöntemi” başlığıyla patent başvurusu yapılmış olan algoritma, herhangi bir bilgi erişim modeliyle hizmet veren bir Bilgi Erişim Sistemine (BES) entegre edilebilen ve kullanıcıların, ihtiyacı olan bilgi kaynağına (doküman vb.) erişmesinde BES'lere destek sağlayan bir algoritmadır. Entegre çalıştığı BES üzerinde kullanıcıların bir bilgi kaynağına erişmeye çalışırken sorgulama yaptıkları farklı arama terimlerine odaklanmaktadır. Farklı/aynı kullanıcıların, benzer bilgi kaynaklarına erişmeye çalışırken kullandıkları farklı sorgulama terimlerinin tespitinde aktif rol almakta ve bu terimlerle birlikte, bu terimlerin ilgili olduğu bilgi kaynaklarını organize edebilmektedir. Başka bir ifadeyle, kullanıcıların farklı arama terimleriyle gerçekleştirdikleri sorgulamaları daha düzenli hale getirebilmekte ve ilgili bilgi kaynaklarını sunabilmektedir. Kullanıcılar, BES üzerinde algoritma tarafından önerilmemiş olan terimlerle sorgulama yaptıklarında ve BES'in bu terimlerle ilgili önerdiği bilgi kaynaklarını inceledikleri takdirde, algoritmanın analiz yapabileceği bir süreç oluşmaktadır. Yani algoritma, BES üzerinde kullanıcıların kullandıkları sorgulama terimlerini ve bilgi kaynaklarını incelemekte ve bu incelemeler doğrultusunda, terim önerilerini ve bilgi kaynaklarını daha verimli kullanılacak şekilde organize edebilmektedir. Bu organizasyonun yapılması ise, çalışmanın 2.5.1 bölümünde aktarılmış olan “Örtük İlgililik Geribildirim” mekanizması sayesinde gerçekleştirilmektedir. Algoritmanın işleyişinde yer alan temel aşamalar aşağıdaki şekilde olup, sırasıyla açıklanmıştır:

1. Disiplin Eşik Değerinin (DED) kararlaştırılması (*the determination of discipline threshold value*)

2. Terim yoğunluklarının hesaplanması (*the calculation of term densities with term comparisons*)
3. Terimlerin birbirlerini temsil edebilme durumlarının denetlenmesi (*the analysis of term representativeness*)
4. İlişkili terim kuralları sözlüğünün oluşturulması (*the creation of term rules thesaurus*)
5. Terim kurallarına bağlı ilgili dokümanların sıralı bir şekilde dizinlenmesi (*ranked document indexing*)

1. Disiplin Eşik Değerinin (DED) kararlaştırılması (*the determination of discipline threshold value*)

DED, kullanıcıların geçmiş arama ve ziyaret davranışları doğrultusunda, kullanıcılara gelecekte önerilecek olan arama terimlerinin ve bu terimlerin ilgili olduğu bilgi kaynaklarının belirlenmesinde kullanılan bir parametredir. 0 (sıfır) – 1 (bir) arasında olacak şekilde sayısal değerler alabilmektedir. DED özelliğinin 0' a yaklaşması, algoritmanın, sorgulama terimlerini ve bu terimlerle ilgili bilgi kaynaklarını kullanıcılara önerirken daha esnek davranmasına neden olmaktadır. Yani, önerilecek sorgulama terimleri ve bilgi kaynakları arasındaki ilişkilerin tespitinde, birbirleriyle daha az ilgili sorgulama terimleri ve bilgi kaynakları öneri sürecine dahil edilmektedir. Buna rağmen, kullanıcıların sorgulama yaptıkları terimler ve bu sorgulamalar doğrultusunda inceledikleri bilgi kaynakları içerisinden seçim yapılmakta olup, indirgenmiş bir öneri yapısı oluşturulmaktadır. DED parametresinin 1' e yaklaşması ise, algoritmanın, sorgulama terimlerini ve bu terimlerle ilgili bilgi kaynaklarını kullanıcılara sunarken daha disiplinli ve katı davranmasına neden olmaktadır. Bu haliyle de, önerilecek sorgulama terimleri ve bilgi kaynakları arasındaki ilişkilerin tespitinde, birbirleriyle daha fazla ilgili sorgulama terimleri ve bilgi kaynakları öneri sürecine dahil edilmektedir.

DED parametresi, 3. aşamada kullanılan bir parametre olup, algoritmanın, veriden öğrenerek bir yapay zekâ olarak çalıştırılmasında görev yapmaktadır. DED'in bu çalışmada periyodik olarak nasıl değiştirildiğine dair açıklamalar, çalışmanın 3.3 bölümünde detaylı bir şekilde yapılmıştır.

2. Terim yoğunluklarının hesaplanması ve karşılaştırmalar (the calculation of term densities with term comparisons)

Algoritmanın en önemli özelliklerinden biri olan terim yoğunluğu (td) hesaplama aşaması, kullanıcıların yaptıkları aramalarda kullandıkları terimler arasındaki benzerliklerin/üstünlüklerin tespitinde görev yapan bir aşamadır. td , algoritmanın, kullanıcıların arama yaparken kullandıkları sorgulama terimlerini tahmin edebilmesine ve bu terimleri düzeltilmiş bir şekilde önerebilmesine yardım etmektedir. Bu sayede, kullanıcıların bilgi kaynağına ulaşmaya çalışırken kullanacakları sorgulama terimini belirlemelerine olanak sağlamaktadır. Aynı zamanda, kullanıcıların düşünsel yükü hafifletilerek hem zaman hem de emek açısından kazanç sağlanmaktadır.

DED parametresiyle kontrollerin gerçekleştirildiği 3. aşamaya geçilmeden önce geçilen bu aşamada, belirli bir süre zarfında kullanıcıların BES üzerinde yaptıkları aramalarda kullandıkları sorgulama terimleri birbirleriyle kıyaslanmakta ve bir terimin bir diğer terimi temsil edebilme durumunun ön kontrolü yapılmaktadır. Bu kontrollere başlanmadan önce ilk olarak, her bir terim için td değeri, cisimlerin yoğunluklarının hesaplanmasında kullanılan denklem 3.1'deki formül sayesinde belirlenmektedir:

$$d = \frac{\text{kütle}}{\text{hacim}} = \frac{m}{V} \quad (3.1)$$

td hesaplamaları Tablo 3.1'de verilen bilgiler doğrultusunda aşağıda örneklenmiştir.

Tablo 3.1: Arama terimleri ve incelenen kaynaklar.

Sorgulanan Terimler (t)	İncelenen Kaynak ID'leri (v)	İncelenme Miktarı (m)	Terim Yoğunluğu (td)
t_1	100	20	$60/3 = 20$
	200	10	
	500	30	
t_2	300	25	$30/2 = 15$
	200	5	
t_3	100	5	$5/1 = 5$

Tablo 3.1 incelendiğinde, t_1 araması yapılarak üç farklı kaynağa (100, 200, 500) ulaşıldığı görülmektedir. Bu değer, Denklem 3.1'de hacime ($v = 3$) karşılık gelmektedir. Bu kaynakların

her birinin toplamda ne kadar miktarda (20, 10, 30) incelendiği ise kütle ($m = 60$) olarak ifade edilmektedir. Her iki değer kullanılarak tüm td 'ler hesaplanmaktadır. Yoğunluğun algoritmadaki görevi ise, benzer kaynaklarla ilişkili olan farklı arama terimlerinden hangisinin bir diğerini temsil edebileceğini belirlemektir. Bu sayede, aynı kaynaklara ulaşılırken kullanılmış olan farklı arama terimleri indirgenerek, kullanıcıların hedefine uygun terim önerilerinin ortaya çıkarılması sağlanmaktadır. Bu indirgeme esnasında, yoğunluğu büyük olan terimin, küçük olanı temsil edebilme durumunun bulunduğu düşünülerek hareket edilmektedir. Bu durum, su içerisinde bulunan farklı yoğunluktaki cisimlerin dibe batması, askıda kalması ya da yüzmesi şeklinde hayal edilebilir. Algoritmada, suda batan cisimler (yoğunluğu yüksek olan terimler), askıda kalanları ya da yüzenleri (yoğunluğu düşük olan terimler) temsil edebilir düşüncesi yer almaktadır.

Bir terimin hacmi ne kadar büyükse, o kadar yoğunluğu düşecektir. Yani, bir BES üzerinde arama yapan kullanıcının, ulaşmak istediği bilgiyle ilgili çok çeşitli kaynakları az miktarda taraması BES'in kullanıcıya tatmin edici sonuçları sunamadığı anlamına gelebilir. Kullanıcıyı bu davranışa iten sebebin ise, erişmeye çalıştığı bilgi kaynağını bir türlü bulamamasından kaynaklandığı söylenebilir. Diğer taraftan, daha az benzer kaynağı daha fazla inceletmiş olan farklı bir terimin yoğunluğu ise büyük olacaktır. Bu noktada, ilişkili kaynakları daha az çeşitte içeren yoğunluğu büyük terimin, küçük olanı temsil etmesiyle birlikte, kullanıcının göstermesi gereken düşünsel çabadan tasarruf ettirilmesi sağlanmaktadır.

Tablo 3.1'e tekrar bakıldığında, t_1 ve t_3 'ün ortak bir kaynağa (100) gittiği görülmektedir. t_1 'in yoğunluğunun t_3 'den büyük olması durumu t_3 'ün, t_1 olarak ifade edilebilme potansiyelinin olduğunu göstermektedir. Bu durumda, bir BES üzerinde " t_3 arama terimini yazan kullanıcıya t_1 önerilecektir ($t_3 \Rightarrow t_1$)" şeklinde bir yol izlenebilir. Bu işlemin, pratikte gerçekleştirilmesi ise " t_3 teriminin tamamını yazmadan kullanıcıya t_1 teriminin otomatik olarak önerilmesi" şeklinde yapılabilir ki bu işlem, çalışmanın 3.2 bölümünde detaylı bir şekilde açıklanmıştır. Elbette ki, farklı terimler arasında ortak olarak incelenmiş olan kaynaklarla birlikte ortak olmayan kaynakların da bulunabilme olasılığı her zaman vardır. Bu noktada, hangi terimin başka bir terimi temsil edeceği ve bu temsiliyet doğrultusunda hangi bilgi kaynaklarının kullanıcıya sunulacağı, *DED* parametresi kullanılarak geçilen 3. aşamada kararlaştırılmaktadır. Temelde, algoritma, 2. ve 3. aşamayı aynı anda geçmektedir. Yani, yoğunluk bakımından bir temsiliyet potansiyelinin olması durumunda bir sonraki aşamanın kontrolleri sağlanmakta, aksi

durumdaysa, terimler arasındaki yoğunluk bakımından temsiliyet potansiyellerinin kontrolleri devam ettirilmektedir. Miktar olarak, 2. ve 3. aşamanın algoritma tarafından geçilmesi, Denklem 3.2' deki formül üzerinden belirlenmektedir.

$$\text{kontrol miktarı} = \frac{n \cdot (n - 1)}{2} \quad (3.2)$$

Denklem 3.2'ye örnek olarak, belirli bir süre zarfında (periyot) toplam 100 farklı terimle farklı bilgi kaynaklarının araştırıldığı varsayalım. Bu durumda, 2. ve 3. aşama birlikte olacak şekilde $100 \cdot 99 / 2$ hesaplamasıyla 4950 defa tekrar ettirilmektedir. Bu tekrarlamalarda her bir terim bir başka terimle kıyaslanmaktadır. Yani, her bir tekrar edişte iki farklı terimin birbirini temsil edebilme durumu kontrol edilmektedir. Tekrarlı bir şekilde temsiliyet kontrollerinin yapılmasının sebebi, Tablo 3.2'deki olasılıkların tamamının kontrol edilmesidir.

Tablo 3.2: Terim temsiliyet olasılıkları.

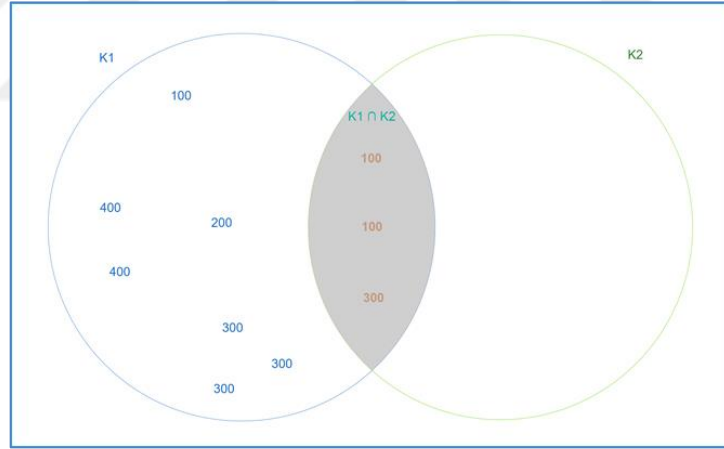
#	Olasılık	Sonuç
1	Bir terim başka bir terim tarafından temsil edilmez.	Bulunmamaktadır
2	Bir terim başka bir terim tarafından temsil edilir.	t_1 (temsilen) $\Rightarrow t_2$ (temsilen)
3	Karşılaştırılan iki terim birbirini eşit derecede temsil edebilir.	$t_1 \Rightarrow t_2$ veya $t_2 \Rightarrow t_1$
4	Temsilen bir terimin temsiliyeti başka bir terim tarafından devralınabilir.	<i>önceki durum:</i> $t_1 \Rightarrow t_2$ <i>yeni durum:</i> $t_1 \Rightarrow t_3$
5	Temsilen bir terim başka bir terim tarafından da eşit derecede temsil edilebilir.	$t_1 \Rightarrow (t_2, t_3)$
6	Temsilen bir terim başka bir terim tarafından temsil edilebilir.	$t_1 \Rightarrow (t_2 \Rightarrow t_3)$
7	Başka bir terim tarafından temsil edilen ve kendisi başka bir terimi temsil eden bir terim başka bir terim tarafından da eşit derecede temsil edilebilir.	$t_1 \Rightarrow (t_2 \Rightarrow t_3, t_4)$

Tablo 3.2 incelendiğinde, iç içe olacak şekilde sınırsız sayıda temsiliyetin olabileceği bir olasılıklar kümesinin olduğu görülmektedir. Bu gibi durumlarda, algoritma geriye dönük olarak var olan temsiliyetleri inceleyip, üzerinde bulunduğu potansiyel temsiliyet kontrolünü geçmiş temsiliyetlerle kıyaslayabilmektedir. Genel olarak, algoritma, bu aşamada tüm olasılıkları kontrol ederek, 3. aşamaya geçiş öncesinde potansiyel temsiliyetleri belirlemektedir.

3. Terimlerin birbirlerini temsil edebilme durumlarının denetlenmesi (the analysis of term representativeness)

Bu aşamada, 2. aşamada karşılaştırılan iki farklı terimden herhangi birinin, bir diğerini temsil edebilme potansiyeli bulunduğu, her iki terimle yapılan sorgulamalar doğrultusunda incelenmiş olan bilgi kaynaklarının benzer olup olmadıkları kontrol edilmektedir. Bir başka ifadeyle, var olan bir temsiliyet potansiyelinin değerli olup olmama durumu kontrol edilmektedir. Bu kontrol için, ilk olarak td değeri büyük olan terimin, küçük olan terimi temsil edebilme oranı (*Representation Rate - RPR*) hesaplanmaktadır. Bu işlemin yapılması için aşağıdaki şekilde bir senaryo örnek gösterilebilir:

t_1 terimiyle yapılan bir aramada erişilen bilgi kaynaklarının ID kümesinin (K_1) {100, 100, 100, 200, 300, 300, 300, 300, 400, 400} elemanlarından oluştuğu varsayalım. t_2 terimininkinin (K_2) ise, {100, 100, 300} elemanlarından oluştuğu varsayalım. Her iki kümenin kesişimi Şekil 3.1'deki gibi gösterilebilir.



Şekil 3.1: RPR hesaplamasına yönelik K_1 ve K_2 kümeleri kesişimi örneği.

Şekil 3.1'e göre, $td(t_1) = 2.5$, $td(t_2)$ ise 1 olarak hesaplanır. $td(t_1) > td(t_2)$ şartı sağlandığından dolayı, t_1 ' in t_2 ' yi temsil etme oranı $RPR(t_1, t_2)$ Denklem 3.3'deki formülle hesaplanır:

$$RPR(t_1, t_2) = \frac{K_1 \cap K_2}{K_2} \quad (3.3)$$

Denklem 3.3 formülü üzerinden t_1 'in t_2 'yi temsil etme oranı aşağıdaki şekilde hesaplanır:

$$RPR(t_1, t_2) = \frac{3}{3} = 1$$

Yukarıda verilen örnekte, $RPR(t_1, t_2)$ değerinin 1 (%100) olarak ortaya çıkması sonrasında bu değer, 1. aşamada kararlaştırılmış olan ve 0-1 aralığında değer alabilen DED parametresi ile kıyaslanmaktadır. RPR değerinin, DED değerine eşit ya da büyük olması durumunda t_1 'in t_2 'yi temsil edebildiğine karar verilmektedir. Aksi durumdaysa, td üzerinden ortaya çıkarılan potansiyelin değerli olmadığına kararı verilmektedir.

Tablo 3.2'deki 3. olasılık söz konusu olduğunda, hangi terimin bir diğerini temsil edebileceğinin kıyaslaması gerçekleştirilmektedir. Bu durumda, hem $RPR(t_1, t_2)$ hem de $RPR(t_2, t_1)$ DED değeriyle kıyaslanmakta ve RPR değeri büyük olana göre karar verilmektedir. Her iki RPR değerinin DED değerinden büyük olduğu ve eşit olduğu durumda da hangi terimin bir diğerini temsil edebileceğinin kararı verilemeyeceği için bu potansiyel algoritma tarafından göz ardı edilmektedir.

4. İlişkili terim kuralları sözlüğünün oluşturulması (the creation of term rules thesaurus)

Bu aşama, 5. aşamayla birlikte gerçekleştirilen ve 2. ve 3. aşamaların tekrarlı bir şekilde geçilerek bitirilmesi sonrasında aktif olan aşamadır. 2. aşamada ifade edilmiş olan Tablo 3.2'deki olasılıkların kontrol edilmesi sonrasında birbirleri arasında temsiliyetleri belirlenen terimlerin, aynı tabloda yer alan "Sonuç" sütunundaki ifade biçimleriyle bir kurallar sözlüğü olarak kayıt altına alınma işlemi yapılmaktadır. Bir diğer ifadeyle, her bir kuralın, diğer kurallarla çakışmayacak şekilde bir kimlik (*identification*) değeri ile kayıt altına alınma işlemi gerçekleştirilmektedir. Kayıt işlemleri sonucunda, ilişkili terimlerden oluşan bir kurallar sözlüğü oluşmakta ve bu sözlük, kullanıcılara arama yaparken terim önerileri yapılabilir şekilde kullanılabilir. Bu çalışma için, yapay zekâ olarak çalıştırılan algoritma sayesinde, günlük olarak yeni sözlükler oluşturulmuştur.

5. Terim kurallarına bağlı ilgili dokümanların sıralı bir şekilde dizinlenmesi (ranked document indexing)

Bu aşamada, 4. aşamada kaydedilen kurallara ait kimlik değerleri üzerinden, aralarında temsiliyet ilişkisi bulunan terimlerle yapılmış aramalar doğrultusunda incelenmiş olan bilgi kaynakları (doküman vb.), çeşitli kurallar çerçevesinde sıralanarak dizinlenmektedir. Bir başka ifadeyle, kurallarda geçen terimlerin ilişkili oldukları bilgi kaynaklarının (*Information Source*

- IS) düzenlenmesi gerçekleştirilmektedir. Bu aşamadaki işlemlerin nasıl gerçekleştirildiğine yönelik Tablo 3.3'deki senaryolar örnek gösterilebilir:

Tablo 3.3: Terimlerin ve bilgi kaynaklarının sıralanmasına yönelik örnek senaryolar.

SN	KURAL	İK
1	$t_1 \Rightarrow t_2$	$\text{Order}(\text{IS}(t_1) \cap \text{IS}(t_2), \text{DESC}) \cup \text{Order}(\text{IS}(t_1), \text{DESC}) \cup \text{Order}(\text{IS}(t_2), \text{DESC})$
2	$t_1 \Rightarrow (t_2, t_3)$	$\text{Order}(\text{IS}(t_1) \cap \text{IS}(t_2), \text{DESC}) \cup \text{Order}(\text{IS}(t_1) \cap \text{IS}(t_3), \text{DESC}) \cup$ $\text{Order}(\text{IS}(t_1), \text{DESC}) \cup \text{Order}(\text{IS}(t_2), \text{DESC}) \cup \text{Order}(\text{IS}(t_3), \text{DESC})$
3	$t_1 \Rightarrow (t_2 \Rightarrow t_3)$	$\text{Order}(\text{IS}(t_1) \cap \text{IS}(t_2), \text{DESC}) \cup \text{Order}(\text{IS}(t_2) \cap \text{IS}(t_3), \text{DESC}) \cup$ $\text{Order}(\text{IS}(t_1), \text{DESC}) \cup \text{Order}(\text{IS}(t_2), \text{DESC}) \cup \text{Order}(\text{IS}(t_3), \text{DESC})$ $\text{Order}(\text{IS}(t_1) \cap \text{IS}(t_2), \text{DESC}) \cup \text{Order}(\text{IS}(t_2) \cap \text{IS}(t_3), \text{DESC}) \cup \text{Order}(\text{IS}(t_3)$
4	$t_1 \Rightarrow (t_2 \Rightarrow (t_3 \Rightarrow t_4))$	$\cap \text{IS}(t_4), \text{DESC}) \cup \text{Order}(\text{IS}(t_1), \text{DESC}) \cup \text{Order}(\text{IS}(t_2), \text{DESC}) \cup$ $\text{Order}(\text{IS}(t_3), \text{DESC}) \cup \text{Order}(\text{IS}(t_4), \text{DESC})$
SN: Senaryo Numarası, İK: Bilgi Kaynaklarının Sıralanmasında İzlenen Kural		
IS: Bilgi Kaynağı/Kaynakları, t: Sorgulamada Kullanılan Terim		
Order: Sıralama İşlemi, DESC (Descending): Büyükten küçüğe sıralama		

Tablo 3.3'deki örnek senaryolar aşağıda detaylı bir şekilde açıklanmıştır:

1. *Senaryo:* t_1 terimini daha iyi temsil ettiği belirlenen t_2 'nin kullanıcıya önerilmesi şeklinde gerçekleştirilecek olan bir senaryodur. Kullanıcıya sunulacak bilgi kaynakları her iki terimin ortak ve benzersiz bilgi kaynakları arasından seçilmektedir. Bunun için Tablo 3.3'deki 1. senaryonun İK sütunundaki kural izlenmektedir ($\text{Order}(\text{IS}(t_1) \cap \text{IS}(t_2), \text{DESC}) \cup \text{Order}(\text{IS}(t_1), \text{DESC}) \cup \text{Order}(\text{IS}(t_2), \text{DESC})$). Bu kural, ilk olarak t_1 ve t_2 aramasında incelenmiş olan kaynaklardan ortak olanları bulmakta ve en çok incelenen bilgi kaynağından en az incelenene doğru sıralama yaparak ($\text{Order}(\text{IS}(t_1) \cap \text{IS}(t_2), \text{DESC})$) her iki terimle ilgili sunulacak olan kaynakları bir küme olarak tutmaktadır. Daha sonra, t_1 terimiyle yapılan aramalarda incelenmiş olan kaynaklar en çok incelenenden en az incelenene doğru sıralanarak ($\text{Order}(\text{IS}(t_1), \text{DESC})$) sunulacak olan kaynaklar kümesine eklenmektedir. Son olaraksa, t_2 terimiyle yapılan aramalarda incelenmiş olan kaynaklar en çok incelenenden en az incelenene doğru sıralanarak ($\text{Order}(\text{IS}(t_2), \text{DESC})$) sunulacak olan kaynak kümesine eklenmektedir. Böylelikle, her iki terimle ilgili oluşturulmuş olunan kaynak kümesi kullanıcıya önerilebilir hale getirilmiş olmaktadır.

2. *Senaryo:* t_2 ve t_3 teriminin, t_1 terimini eşit derecede temsil edebileceğinin belirlendiği senaryodur. Bu noktada, t_1 terimini aramak isteyen kullanıcıya t_2 ve t_3 terimleri birlikte

önerilebileceği gibi farklı şekillerde de önerme gerçekleştirilebilir. Bu karar, isteğe bağlı olarak değişkenlik gösterebilir. Tablo 3.3'deki 2. senaryonun İK sütunundaki kural ($\text{Order}(\text{IS}(t_1) \cap \text{IS}(t_2), \text{DESC}) \cup \text{Order}(\text{IS}(t_1) \cap \text{IS}(t_3), \text{DESC}) \cup \text{Order}(\text{IS}(t_1), \text{DESC}) \cup \text{Order}(\text{IS}(t_2), \text{DESC}) \cup \text{Order}(\text{IS}(t_3), \text{DESC}))$), kullanıcıya önerilecek olan bilgi kaynaklarının nasıl oluşturulduğunu göstermektedir. Bu kurala göre ilk olarak, t_1 ve t_2 aramasında incelenmiş olan kaynaklardan ortak olanları bulunmakta ve en çok incelenenden en az incelenene doğru sıralama yapılarak ($\text{Order}(\text{IS}(t_1) \cap \text{IS}(t_2), \text{DESC})$) her iki terimle ilgili sunulacak olan kaynaklardan bir küme oluşturulmaktadır. Daha sonra aynı işlem t_1 ve t_3 terimleri için de gerçekleştirilmektedir ($\text{Order}(\text{IS}(t_1) \cap \text{IS}(t_3), \text{DESC})$). t_1 teriminin, t_2 ve t_3 terimleriyle ortak olan kaynakları düzenlendikten sonra her bir terimin ilgili olduğu kaynaklar da sıralanarak ($\text{Order}(\text{IS}(t_1), \text{DESC}) \cup \text{Order}(\text{IS}(t_2), \text{DESC}) \cup \text{Order}(\text{IS}(t_3), \text{DESC})$) sunulacak olan kaynak kümesine eklenmektedir.

3. *Senaryo*: t_1 teriminin t_2 tarafından, t_2 teriminin de t_3 tarafından temsil edilebildiğinin tespit edildiği senaryodur. Dolaylı olarak, t_1 terimini arayan bir kullanıcıya t_3 'ün önerilmesini içermekle beraber, t_2 terimi doğrultusunda incelenen kaynakların da kullanıldığı bir önerme şeklidir. Tablo 3.3'deki 3. senaryonun İK sütunundaki kural ($\text{Order}(\text{IS}(t_1) \cap \text{IS}(t_2), \text{DESC}) \cup \text{Order}(\text{IS}(t_2) \cap \text{IS}(t_3), \text{DESC}) \cup \text{Order}(\text{IS}(t_1), \text{DESC}) \cup \text{Order}(\text{IS}(t_2), \text{DESC}) \cup \text{Order}(\text{IS}(t_3), \text{DESC})$)) sayesinde sunulacak olan kaynak kümesi oluşturulmaktadır. Bu doğrultuda ilk olarak, t_1 ve t_2 aramasında incelenmiş olan kaynaklardan ortak olanları bulunmakta ve en çok incelenenden en az incelenene doğru sıralama yapılarak ($\text{Order}(\text{IS}(t_1) \cap \text{IS}(t_2), \text{DESC})$) her iki terimle ilgili sunulacak olan kaynaklar kümesine eklenmektedir. Daha sonra aynı işlem t_2 ve t_3 terimleri için gerçekleştirilmektedir ($\text{Order}(\text{IS}(t_2) \cap \text{IS}(t_3), \text{DESC})$). t_1 teriminin t_2 ile, t_2 teriminin de t_3 terimiyle ortak olan kaynakları düzenlendikten sonra her bir terimin ilgili olduğu kaynaklar da sıralanarak ($\text{Order}(\text{IS}(t_1), \text{DESC}) \cup \text{Order}(\text{IS}(t_2), \text{DESC}) \cup \text{Order}(\text{IS}(t_3), \text{DESC})$) sunulacak olan kaynak kümesine eklenmektedir.

4. *Senaryo*: t_1 teriminin t_2 tarafından, t_2 teriminin t_3 tarafından, t_3 teriminin de t_4 tarafından temsil edilebildiğinin belirlendiği senaryodur. Dolaylı olarak, t_1 terimini arayan bir kullanıcıya t_4 'ün önerilmesini içermekle beraber, t_2 ve t_3 terimleri doğrultusunda incelenen kaynakların da kullanıldığı bir önerme şeklidir. Tablo 3.3'deki 4. senaryonun İK sütunundaki kural ($\text{Order}(\text{IS}(t_1) \cap \text{IS}(t_2), \text{DESC}) \cup \text{Order}(\text{IS}(t_2) \cap \text{IS}(t_3), \text{DESC}) \cup \text{Order}(\text{IS}(t_3) \cap \text{IS}(t_4), \text{DESC}) \cup \text{Order}(\text{IS}(t_1), \text{DESC}) \cup \text{Order}(\text{IS}(t_2), \text{DESC}) \cup \text{Order}(\text{IS}(t_3), \text{DESC}) \cup \text{Order}(\text{IS}(t_4), \text{DESC})$))

sayesinde önerilecek olan kaynak kümesi oluşturulmaktadır. Bu doğrultuda ilk olarak, t_1 ve t_2 aramasında incelenmiş olan kaynaklardan ortak olanları bulunmakta ve en çok incelenenden en az incelenene doğru sıralama yapılarak $(\text{Order}(\text{IS}(t_1) \cap \text{IS}(t_2), \text{DESC}))$ her iki terimle ilgili sunulacak olan kaynaklar kümeye eklenmektedir. Daha sonra aynı işlem t_2 ile t_3 terimleri $(\text{Order}(\text{IS}(t_2) \cap \text{IS}(t_3), \text{DESC}))$ ve t_3 ile t_4 terimleri $(\text{Order}(\text{IS}(t_3) \cap \text{IS}(t_4), \text{DESC}))$ için gerçekleştirilmektedir. t_1 teriminin t_2 ile, t_2 teriminin t_3 ile, t_3 teriminin de t_4 terimiyle ortak olan kaynakları düzenlendikten sonra her bir terimin ilgili olduğu kaynaklar da sıralanarak $(\text{Order}(\text{IS}(t_1), \text{DESC}) \cup \text{Order}(\text{IS}(t_2), \text{DESC}) \cup \text{Order}(\text{IS}(t_3), \text{DESC}) \cup \text{Order}(\text{IS}(t_4), \text{DESC}))$ sunulacak olan kaynak kümesine eklenmektedir.

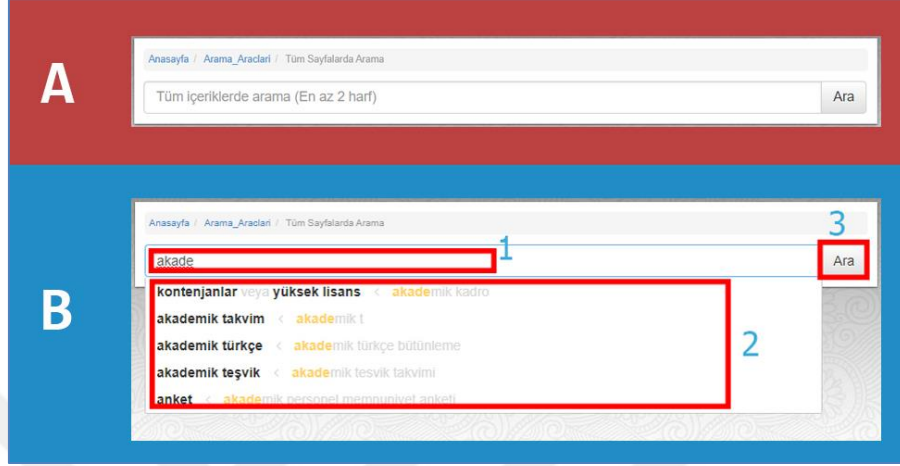
Tablo 3.3'deki senaryolar çeşitli şekillerde arttırılabilir, ancak genel itibariyle algoritma tarafından gerçekleştirilmekte olan bu aşamadaki işlemler, farklı senaryolarda da aynı mantık üzerinden ilerletilecektir. Son durumda, oluşturulan kümeler, ilgili kurala ait olan ve 4. aşamada belirlenen kimlik değeriyle sıralanmış bir şekilde dizinlenmektedir. Bu çalışma için, terim kuralları sözlüğünün oluşturulmasıyla paralel bir biçimde, günlük olarak yeni ilgili dokümanlar dizinleme işlemi gerçekleştirilmiştir.

Genel itibariyle, algoritmanın işleyişinde yer alan tüm aşamalar dikkate alındığında, kullanıcı tarafından sorgulamada kullanacak dilin (Türkçe, İngilizce, Japonca, Çince, Almanca, Arapça vb.) hiçbir şekilde algoritmanın çalışma yapısına etki etmeyeceği görülmektedir. Bu haliyle, algoritma, entegre edileceği BES hangi dilde hizmet veriyor olursa olsun, sistemde var olan dile uygun bir biçimde öneri sunabilecek yetenektedir. Bir başka ifadeyle, Analogy algoritması dilden bağımsız bir şekilde öneri sunabilen bir algoritma özelliği taşımaktadır.

3.2. ALGORİTMANIN ENTEGRASYONU

KLU bünyesinde hizmet vermekte olan toplam 170 web sitesi için, algoritmanın, arama arayüzlerinde öneri sunabilmesine yönelik güncelleme yapılmıştır. Entegrasyon, algoritmanın her bir web sitesinde birbirinden bağımsız çalışabileceği bir yapıda gerçekleştirilmiştir. Böylelikle her bir web sitesinde arama araçlarından faydalanan ziyaretçilerin ihtiyaç duydukları bilgi doğrultusunda öneri almaları sağlanmıştır. Bir başka ifadeyle, algoritma, farklı web sitelerinde popüler olan bilgi ihtiyacını temel alarak her bir web sitesinin arama arayüzlerinde farklı öneriler sunmuştur. Bu sayede, kullanıcı odaklı bir etkileşimin gerçekleştirilmesi sağlanmıştır. Kullanıcıların arama yaptıkları esnada aradıkları bilgiye yönelik öneri aldıkları

arayüz, sistemde hâlihazırda hizmet vermekte olan eski görünüm (Şekil 3.2 – A) değiştirilmeden, sadece algoritmanın öneri sunabilmesi için otomatik kelime tamamlama özelliğiyle Şekil 3.2’de yer alan B yapısına güncellenmiştir.



Şekil 3.2: Algoritmanın öneri sunma arayüzü.

Şekil 3.2’de B bölümü incelendiğinde, 1 numaralı alana bir kullanıcı “akade” yazdığında, karşısına 2 numaralı alanda yer alan öneriler getirilmiştir. Bu öneriler, otomatik kelime tamamlama özelliğiyle en az iki harf yazılması doğrultusunda sunulmuş olup, kullanıcının hatalı yazdığı terimlerde de (eğer varsa) ilişki durumuna göre sunulmuştur. Kullanıcının yazdığı terim öneri listesinin satırlarında sağ kısımda (temsilen edilen terim) turuncu renkle belirginleştirilmiş olup, sol bölümde benzer ya da farklı terimler (temsilen eden terim/terimler) siyah renkle daha belirgin olarak gösterilmiştir. Bu aşamada dikkat edilmesi gereken önemli bir nokta, her ne kadar sağ kısımda kullanıcının yazdığı terim kesinlikle bulunmasına rağmen sol kısımda bu terimi içermeyen satırların (kontenjanlar veya yüksek lisans ve anket) yer almasıdır. Bu sunum biçimi algoritma tarafından oluşturulmakta olup, algoritma, kullanıcıların geçmişte yaptıkları farklı aramalar ile yapmayı düşündükleri aramalar arasındaki ilişkileri tespit etmekte ve terimler arası ilişki durumlarına göre farklı önerileri sunabilmektedir. Bu öneriler, çalışmanın 3.1 bölümünde detaylarıyla açıklanmış olan algoritmanın oluşturduğu “ilişkili terim kuralları sözlüğü”nden çekilmektedir. Bu tarz bir öneri yapısı literatürde bulunmamakta olup, algoritma için özel olarak geliştirilmiştir. Burada kullanıcının algılaması beklenen mesaj ise, aramayı düşündüğü terimi (turuncuyla belirginleştirilen) daha iyi temsilen eden terimin siyahla belirginleştirilen kısımdaki terimler olduğudur. Bir başka ifadeyle, turuncuyla belirginleştirilen terim aslında farklı kullanıcılar tarafından daha nadir kullanılmış olup, siyah ile

belirginleştirilen kısımdaki terimler daha yoğun kullanılmıştır. Böylelikle, arama yaparken kullanıcıya siyah ile belirginleştirilen kısım vurgulanmaya çalışılmıştır.

Şekil 3.2’de B bölümündeki senaryoya göre öneri alan kullanıcının temel olarak 2 farklı eylem gerçekleştirme ihtimali bulunmaktadır. Bunlardan ilki, 2 numaralı kısımda ilgisini çeken ve ihtiyacını giderebileceğini düşündüğü öneriye tıklamak, diğeri ise önerileri dikkate almadan 3 numaralı “Ara” butonuna tıklamaktır. Birinci tercih, algoritmanın önerilerde ne kadar başarılı olduğuna yönelik veriyi, ikinci tercih ise hâlihazırda zaten geçmişten beridir tüm web sitelerinde hizmet veren normal (öneri sunmayan) arama aracının (Şekil 3.2 - A) başarı durumunu ölçmeye yönelik veriyi sağlamıştır. Genel olarak, 3 aylık süre boyunca kullanıcıların gerçekleştirdikleri bu 2 tercih üzerinden veri toplama işlemi gerçekleştirilmiştir.

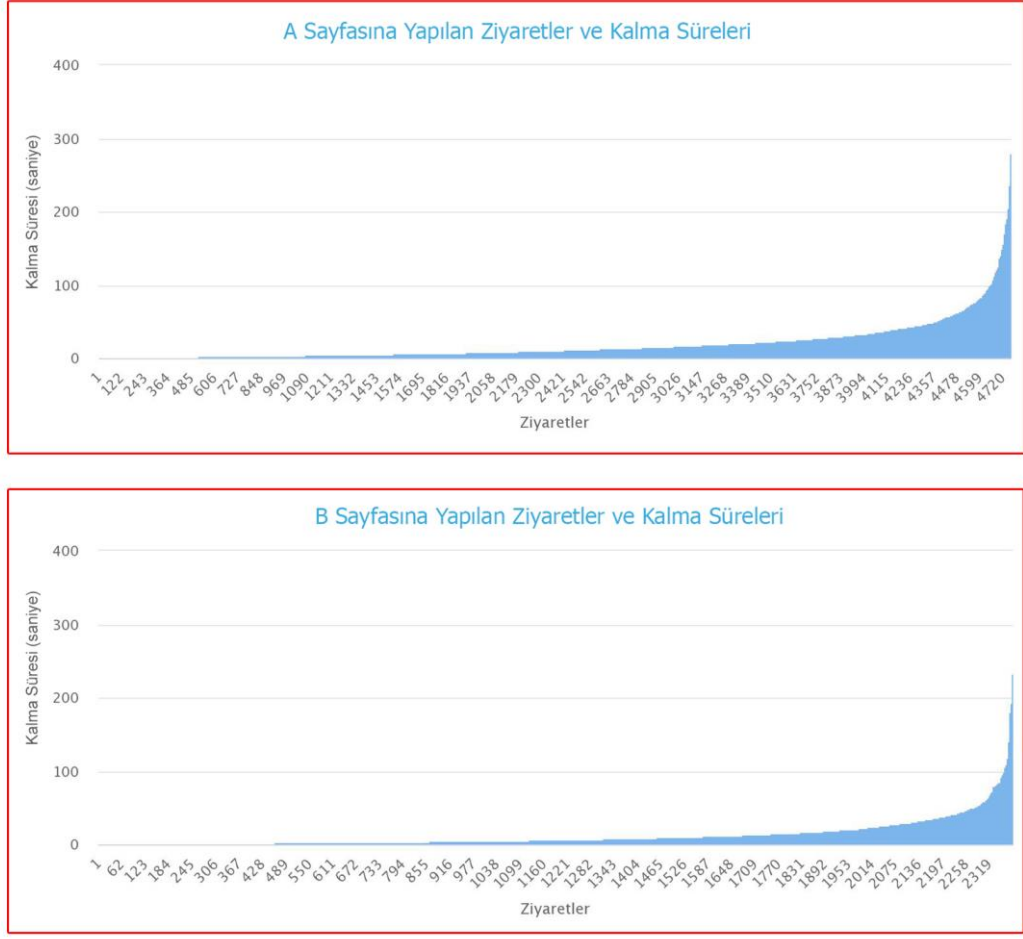
3.3. ALGORİTMANIN MAKİNE ÖĞRENMESİ İÇİN VERİ TOPLAMA SÜRECİ

Çalışmada kullanılmak için geliştirilen algoritmanın, 0-1 aralığında değişkenlik gösteren ve *DED* olarak adlandırılan bir parametreye sahip olduğu daha önce ifade edilmişti. Bu parametrenin 0’a yaklaşması, kullanıcılara arama yaptıkları esnada öneri sunarken daha esnek bir şekilde öneri sunma işlemi, 1’e yaklaşması ise daha disiplinli bir şekilde öneri sunma işlemi gerçekleştirmektedir. Bu noktada, esneklik, kullanıcıların arama yaparken karşılırlarına getirilen önerilerin, kullanıcının erişmeye çalıştığı bilgiyle ilgili olma durumunun zayıflatılması anlamına gelmektedir. Yani, kullanıcıya ihtiyacını giderme ihtimali düşük olan önerilerin de sunulması şeklinde ifade edilebilir. Disiplinlilik durumu ise bu eylemin tam tersi şeklinde gerçekleştirilmektedir.

Algoritmanın öğrenebilmesi için *DED* parametresinin haftalık periyotlar halinde değiştirilmesi sağlanmıştır. Bu etkileşim için kullanıcıların geçmiş arama davranışları analiz edilmiş ve algoritmaya geribildirim sağlanmıştır (Örtük İlgililik Geribildirimi). Bu geribildirim sayesinde de haftalık olarak kullanıcıların yaptıkları aramalarda ortaya çıkan deneyimin geliştirilmesi amaçlanmıştır. Böylelikle, haftalık olarak elde edilen geribildirimlerle birlikte algoritmanın esnek ya da disiplinli bir şekilde yönlendirme işlemi gerçekleştirilmiştir.

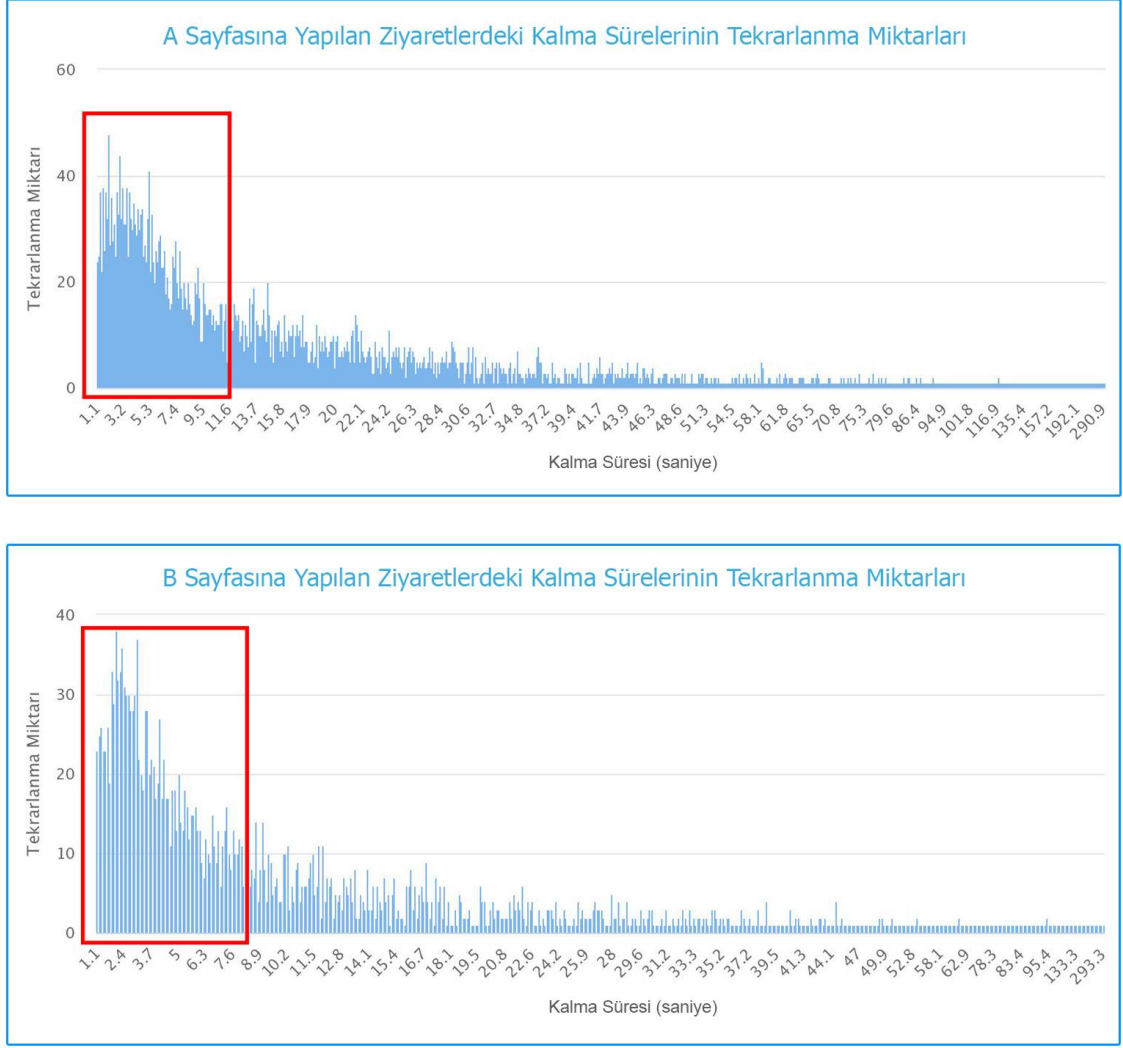
DED parametresinin 0-1 aralığında haftalık olarak alacağı değeri belirlerken, kullanıcıların geçmişte yaptıkları aramalarda hangi web sayfalarına eriştikleri ve bu sayfalarda ne kadar süre kaldıkları tespit edilmiştir. Bu süre verisi, literatürde “kalma zamanı (*dwel time*)” olarak adlandırılmakta ve kullanıcıların bir dokümanı incelemede harcadıkları zamanı ifade

etmektedir (Croft ve diğ., 2009). Bir kullanıcının ihtiyaç duyduğu bilgiyi arayarak, yararlanılan arama aracı tarafından önerilen sayfalara ziyaret gerçekleştirmesi, arama aracının önerme başarısının belirlenmesinde ve geliştirilmesinde kullanılabilecek önemli bir veridir. Bu doğrultuda, algoritmanın herhangi bir müdahale olmadan kendi kendine işlem yapabilmesi için bu veri türünün kullanılmasına karar verilmiştir. Bu verinin 0-1 aralığında bir geri bildirim sağlayabilmesi için temel olarak minimum, ortalama ve maksimum olmak üzere 3 farklı sayısal değerin ortaya çıkarılmasına çalışılmıştır. Bu değerler, kullanıcıların herhangi bir web sayfasına ziyaret gerçekleştirdikleri esnada kalma sürelerine bağlı olarak sayfanın, kullanıcının bilgi ihtiyacını karşılamada yeterli olup olmadığının belirlenmesinde kullanılmıştır. Örnek olarak; ziyaret edilen sayfanın A başlıklı bir web sayfası olduğu varsayalım. Bu sayfaya toplam 100 ziyaretin gerçekleştirildiği düşünülerek, kullanıcıların minimum 1 saniye, ortalama 15 saniye ve maksimum 25 saniye kaldıkları varsayalım. Bu durumda, minimum ve ortalama arasında kalmış olanlar için sayfanın yeterli olmadığı, ortalama ve maksimum arasında olanların da yeterli bilgiye eriştikleri varsayılmıştır. Bu tür bir ayrıştırmanın yapılması için haftalık periyotlarda aramayla erişilmiş olunan her bir web sayfası tek tek incelenmiştir. Bu incelemeler doğrultusunda da haftalık olarak kullanıcıların yeterli bilgiye erişme durumları 0-1 aralığına denk getirilecek bir biçimde hesaplanmış ve algoritmaya geri bildirim sağlanmıştır. Bu hesaplama yapılırken birbirinden farklı yöntemler denenmiştir. Bu noktada, ilk olarak, her web sayfası için ayrı olacak şekilde ziyaretlerde elde edilmiş kalma sürelerinin ortalama değeri hesaplanmış ve standart sapma kadar değişimleri aralık olarak seçilmiştir. Bu durumda, “ortalama - standart sapma” minimum olarak seçilirken, “ortalama + standart sapma” maksimum olarak seçilmiştir. Ancak bu hesaplama yöntemi sayfalarda kalma sürelerinin normal dağılmamasından dolayı kullanılamamıştır. Tek boyutlu kümeleme algoritmalarından Ckmeans (Wang ve Song, 2011) ve Jenk doğal kırılım ayrıştırma algoritması benzer şekilde denenmiş ancak yine olumlu sonuç alınamamıştır. Bu denemeler sonrasında kullanılacak yöntem belirlenirken ilk olarak, farklı web sayfalarına yapılmış olan ziyaretlere ait kalma süreleri grafikleri oluşturularak incelenmiştir. Şekil 3.3, gerçek ziyaret verisi içeren, 4500’ü (A sayfası) ve 2000’i (B sayfası) aşkın ziyaretler almış iki farklı sayfaya ait örnek bir grafikdir.



Şekil 3.3: 2 farklı web sayfalarına yapılmış olan ziyaretler ve kalma süreleri grafiği.

Şekil 3.3 incelendiğinde, birbirinden farklı kalma sürelerine sahip çok fazla ziyaretin gerçekleştiği görülmektedir. Aynı veri için kalma sürelerinin tekrarlanma miktarları yani frekansları incelendiğindeyse, Şekil 3.4’deki grafikler ortaya çıkmaktadır.



Şekil 3.4: 2 farklı web sayfasına yapılmış olan ziyaretlerdeki kalma sürelerinin tekrarlanma miktarı grafiği.

Şekil 3.4’de kırmızıyla işaretlenmiş alan tekrarlanma miktarı yüksek olan kalma sürelerinin belirtildiği alandır. Verilen örneklerde dikkat edileceği üzere, kullanıcıların 10 saniyenin üzerinde kalma miktarları gittikçe azalma göstermektedir. Bu durum, kullanıcıların aradıkları bilgiye erişme eylemlerini kısa sürede gerçekleştirmeye eğilimli olduklarını gösteren önemli bir bulgu olarak ortaya çıkmıştır. Her ne kadar burada 2 sayfa için örnek verilmiş olsada yapılan incelemelerin benzer grafiklerin farklı web sayfaları için de ortaya çıktığı gözlemlenmiştir. Böylelikle çalışmada temel alınacak kalma sürelerinin Şekil 3.4’de kırmızıyla çizilmiş alan olmasına karar verilmiştir. Bu karar sonrasında yapılan işlem ise her bir web sayfası için kırmızı alanların tespitinin nasıl yapılacağının belirlenmesi olmuştur. Bu durum, frekanslar üzerinden uç değerlerin elenmesi şeklinde de ifade edilebilir. Yani tekrarlanma sıklığı az olan kalma

süreleri uç değerlermiş gibi düşünülerek eleme işlemi yapılmaya çalışılmıştır. Bu sayede, daha önce de belirtildiği üzere normal dağılmayan veri kümesinin normalleştirilmesi/normale yakın olması sağlanmaya çalışılmıştır. Bu işlem için uç değerlerden etkilenmeyen (Orhunbilge, 2000) “kartiller arası aralık ($IQR = Q_3 - Q_1$)” değerini kullanan ve literatürde “*Tukey’s fences*” (Tukey, 1977) olarak geçen Denklem 3.4’deki formüllerden yararlanılmıştır:

$$\begin{aligned} \text{Maksimum Sınır} &= Q_1 - 1.5 \times IQR \\ \text{Minimum Sınır} &= Q_3 + 1.5 \times IQR \end{aligned} \quad (3.4)$$

Denklem 3.4’e örnek olarak, Q_1 değeri 10 saniye olan, Q_3 değeri 20 saniye olan bir veri kümesinde IQR değeri 10 olarak hesaplanır. Minimum sınırdaki formüle karşılık gelen sonuç $10 - (1.5 \times 10)$ işlemiyle -5, maksimum sınırdaki formüle karşılık gelen sonuç ise $20 + (1.5 \times 10)$ işlemiyle 35 olarak bulunur. Bu durumda minimum sınır -5, maksimum sınır ise 35 olarak alınarak, her iki sınır aralığında kalma süresine sahip veri kümesi ayrıştırılır. Kalma süresi 0’dan küçük olmayacağı için Şekil 3.4’deki kırmızı çizgide de görüldüğü gibi 0’dan başlayarak maksimum sınıra kadar olan aralık belirlenmiş olur. Böylelikle, 0-35 saniye aralığında kalma süresine sahip son veri kümesi elde edilmektedir. Bu yöntem kullanılarak farklı web sayfaları üzerinden denemeler gerçekleştirildiğinde, son veri kümesinin işlenmemiş veri kümesine oranla çalışmada faydalanabilecek yeterlilikte normal dağılım gösterdiği tespit edilmiştir. Bu aşamadan sonra geriye kalan işlem ise, kullanıcıların ziyaret ettikleri sayfalardaki minimum, ortalama ve maksimum kalma sürelerinin tespiti olmuştur.

Normal dağılım gösteren bir veri kümesinde değerlerin yarısı ortalamanın sağında, diğer yarısı ise solunda yer almaktadır (Akbulut, 2010). Denklem 3.4’ün kullanılmasıyla birlikte elde edilmiş olan veri kümesinin normal dağıldığı varsayıldığından, aritmetik ortalama değeri veri kümesini ikiye ayıran noktayı ifade edecektir. Bu durumda, kullanıcıların ziyaret ettikleri sayfada ortalama değeri ile minimum kalma süresi aralığında kalanların yetersiz, maksimum kalma süresi ile ortalama arasında kalanların ise yeterli bilgiye ulaştıkları belirtilebilir. Bu noktada cevaplanması gereken soru “ortalamanın her iki tarafında eşit veri miktarı bulunuyorsa, yeterli ve yetersiz miktarı da eşit sayılmaz mı?” şeklindedir. Cevap “Evet”tir ancak, bir web sayfasına yapılmış olan ziyaretlerin tamamı kullanıcıların arama yaparak eriştikleri ziyaretler değildir. Bir örnekle açıklamak gerekirse; 1 haftalık periyot içinde B başlıklı bir web sayfasına toplamda 200 ziyaret yapıldığı düşünölsün. Bu ziyaretlerin 50 tanesinin kullanıcıların bir arama aracı vasıtasıyla eriştikleri ziyaret olduğu, geriye kalan 150 tanesinin ise arama yapmadan

gerçekleştirilmiş olan ziyaretler olduğu düşünölsün. Denklem 3.4'deki yer alan sınırlar ile bu veri kümesinin 40 kadarının (5 tanesi arama ile yapılan, 35 tanesi ise normal ziyaret) elenerek 160 tane kalma süresine ait bir veri kümesinin elde edildiđi varsayölsün. Bu 160 veri içinde minimum 1 saniye, ortalama 15 saniye ve maksimum 20 saniye kalma süresinin belirlendiđi düşünölsün. Amacımız arama aracının başarısını tespit etmek olduğundan, bu sınırlar içinde sadece 35 tane veriden oluşan arama aracıyla yönlendirilmiş ziyaretler temel alınmaktadır. Bu durumda, 35 verinin 15 tanesinin 1-15 saniye aralığında kalarak yetersiz olduğü, 20 tanesinin de 15-20 saniye aralığında kalarak yeterli olduğünün tespit edildiđi varsayılabilir. Sonuç itibariyle, bu yöntem sayesinde 160 tane kalma süresinden oluşan veri kümesinde sadece arama aracıyla yönlendirilen sayfa ziyaretlerinin incelenmesi gerçekleştirilmiştir.

Örnekte göröldüğü üzere, *DED* için gerekli olan 0-1 aralığına sahip bir değere ihtiyaç bulunmaktadır. *DED*, Denklem 3.5, 3.6, 3.7, 3.8 ve 3.9'da yer alan formöller sayesinde hesaplanmaktadır:

$$\text{Yeterlilik Değeri (} YD_i \text{)} = Z_{i(\text{yeterli})} - Z_{i(\text{yetersiz})} \quad (3.5)$$

$$\text{Önem Katsayısı (} \ddot{O}K_i \text{)} = \frac{TZ_i}{TZ} \quad (3.6)$$

$$\text{Sayfa Değeri (} SD_i \text{)} = YD_i \cdot \ddot{O}K_i \quad (3.7)$$

$$DED_j = \sum_{i=1}^{n=12} SD_{ij} + DED_{j-1} \quad (3.8)$$

$$DED_j = \begin{cases} j < 12 \text{ and } DED_j > 1, & DED_j = 0.5 \\ j < 12 \text{ and } DED_j < 0, & DED_j = 0.5 \\ j \leq 12 \text{ and } j > 3 \text{ and } DED_j = DED_{j-1} = DED_{j-2} = DED_{j-3}, & DED_j = 0.5 \end{cases} \quad (3.9)$$

j geçilecek olan bir sonraki periyot, *i* ziyaret edilen herhangi bir web sayfası olmak üzere, *YD_i* tamamlanmış olan periyotta herhangi bir web sayfasına yapılmış olan ziyaretlerde tespit edilmiş yeterlilik ve yetersizlik miktarlarının farkını ifade etmektedir. Bu değeri, pozitif ya da negatif bir tamsayı olabilir (Denklem 3.5). *ÖK_i*, tamamlanmış olan periyot içinde arama aracı vasıtasıyla ya da normal bir şekilde herhangi bir web sayfasına yapılmış olan toplam ziyaret miktarının (*TZ_i*), tüm web sayfalarına yapılan ziyaretlere (*TZ*) oranını ifade etmektedir (Denklem 3.6). Bu katsayı sayesinde, tamamlanmış olan periyotta birbirinden farklı sayfalara ait *YD* değeriğerlerinin ne oranda dikkate alınacağına karar verilmektedir. Bu karar, aynı zamanda, Denklem 3.7'deki *SD_i* değeriğerine karşılık gelmektedir. Denklem 3.8, algoritmaya geri bildirim sağlayan ve her geçilecek yeni periyot için hesaplanan *DED* değeriğerinin nasıl oluşturulduğunu göstermektedir. Bu denklemde, tamamlanan periyotta arama aracı sayesinde yönlendirilmiş tüm

sayfalara ait *SD* değerlerinin toplanma işlemi yapılmakta ve bu sonuca yine tamamlanmış olan periyotta kullanılmış *DED* değeri eklenmektedir. Bir başka ifadeyle, yeni periyotta kullanılacak *DED* değeri kümülatif olarak değiştirilmektedir (Pilot çalışmalar esnasında algoritma öneri yapısının ilk kez aktif kullanılmış olmasından dolayı ilk haftalık periyot için *DED* değeri 0.5 olarak tayin edilmiştir, çünkü *DED*'in kümülatif hesaplanması için ön kullanım verisinin bulunması şarttır. Pilot çalışmalar sonrasında zaten hesaplanmış olan *DED* değeri ile 12 haftalık asıl çalışma sürecine girilmiştir). Hesaplanan yeni *DED* değeri bir sonraki periyotta kullanılarak algoritmanın yönlendirilmesini sağlamış ve her periyot bitişinde işlemler aynı şekilde tekrarlatılmıştır. Bu noktada, *DED* değerinin 0-1 aralığında kalmasının sağlanması ve art arda 4 periyot boyunca aynı olmaması için Denklem 3.9'da aktarılan bir kontrol yapısından da yararlanılmıştır. Denklem 3.9 incelendiğinde, tamamlanmış periyot sonrasında hesaplanmış olan *DED* değerinin 1'den büyük olduğu ve tüm periyotların tamamlanmama durumunda bir sonraki periyotta algoritmanın yönlendirilmesinde *DED* değerinin 0.5'e sabitlenmesi sağlanmıştır. Benzer şekilde, 0'dan küçük bir değer elde edildiğinde de *DED* değeri yine 0.5'e sabitlenmiştir. Bir başka kontrol de yine aynı denklemin (Denklem 3.9) son satırında görüleceği üzere, geçilen 3 haftalık periyotta aynı *DED* değerinin kullanılması durumunda ve yeni hesaplanan değer de değişmemesi durumunda *DED* değeri yine 0.5'e sabitlenmiştir. Bunun yapılmasının sebebi, farklı *DED* değerlerinin kullanıcıların arama deneyimlerinin geliştirilmesinde nasıl değişkenlik gösterdiğinin tespit edilmesinin istenmesidir.

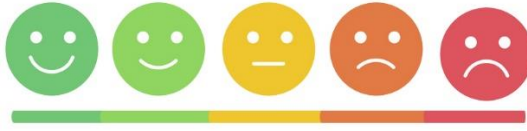
Toplamda 12 haftalık periyotlar boyunca yukarıda açıklanan işlemler, çalışma için geliştirilen algoritmanın kendi kendine veriden öğrenmesini sağlamış ve kullanıcıların arama deneyimlerinin geliştirilmesi için görev yapmıştır. Algoritma performansının ölçülmesi amacıyla ihtiyaç duyulan kullanıcı verisi daha önce de belirtildiği üzere KLU web sitelerinden elde edilmiş olup, veri kullanımına yönelik gerekli izinler alınmıştır (EK 1 ve EK 2).

3.4. DEĞERLENDİRME

Geliştirilen algoritmanın başarısının değerlendirilmesi noktasında, kullanılabilirlik tanımında yer alan üç kavram (etkililik, verimlilik ve memnuniyet) üzerinden incelemeler gruplandırılmıştır. Her bir kavram özelinde başarı değerlendirmesi yapılırken, hâlihazırda hizmet veren ve algoritmanın entegrasyonunun yapıldığı normal arama araçları ile algoritma arasında karşılaştırmalar yapılmış ve bu karşılaştırmalarda, normal arama aracının gösterdiği performansın algoritma tarafından ne derece geliştirildiğine odaklanılmıştır.

Algoritmanın etkililik ölçümünde Reciprocal Rank (RR) metriğinden faydalanılmıştır. Verimlilik ölçümleri, “Sorgu Gecikmesi (*Query Latency*)” ile birlikte çalışmanın 2.4.3 bölümünde “Tıklama Verisi Testi (*Clickthrough Data*)” başlığıyla aktarılmış bölümde geçen “İlk Tıklama Süresi (*Time to First Click*)” ile “Sorgu Terk Etme (*Query Abandonment*)” kavramları temel alınarak gerçekleştirilmiştir.

Algoritmanın memnuniyet açısından başarısının belirlenebilmesi amacıyla, normal arama aracı ve algoritma (hangi yolla arama yapıldıysa) tarafından sunulan dokümanlara tıklayan kullanıcılardan, eriştikleri sayfada “Aradığınız bilgiye yönelik bu sayfayı yeterli buluyor musunuz?” sorusuna cevap vermeleri istenmiştir. Kullanıcıların cevap vermeleri için Şekil 3.5’deki biçimde 5 dereceden oluşan bir bar kullanılmıştır.



Şekil 3.5: Memnuniyet değerlendirme barı.

Şekil 3.5’de soldan sağa doğru 5’den 1’e kadar puanlama yapılabilmektedir. Kullanıcıların sayfanın yeterlilik durumuna göre yaptıkları oylamalar, dolaylı olarak memnuniyetlerini de işaret edeceğinden, bu çalışma için sadece bu tarz bir metrik kullanılmıştır.

Üç farklı kavramın kendi içinde değerlendirilmesi dışında, algoritmanın *DED* parametresinin bu üç kavramla nasıl bir ilişki içerisinde olduğuna yönelik çeşitli incelemeler de gerçekleştirilmiştir.

Farklı açılardan ölçüm yapmak amacıyla ilgili periyotlar boyunca toplanmış verilerin analiz edilmesi ve algoritmanın başarı durumunun anlamlandırılabilmesi amacıyla çeşitli istatistik ölçüm yöntemlerinden faydalanılmıştır. Çalışmada faydalanılmış olan yöntemler genel olarak aşağıdaki gibi olup, çalışmanın 4. bölümünde detaylı olarak açıklanmıştır.

- Normallik Testleri
- Korelasyon Ölçümleri
- İşaret Testi (*Sign Test*)
- Kruskal-Wallis Varyans Analizi

- Mann-Whitney U Bağımsız İki Örneklem Testi
- Tek Örneklem Ki-Kare Testi

3.5. ÇALIŞMADA KULLANILAN YAZILIM VE SİSTEM ALTYAPISI

Bu tez çalışmasında, PHP yazılım dilinin 5.2 sürümü, MySQL veritabanı yönetim sisteminin 5.5.59 sürümü ve JQuery yazılım dilinin 2.1.0 sürümünden yararlanılmıştır. KLU bünyesindeki tüm web siteleri, Apache 2.2.22 sürümüyle hizmet veren bir web sunucusu (Debian 7.11 işletim sistemli) üzerinde barındırılmaktadır. Bu sitelerdeki kullanıcı etkileşimleri, sunucu üzerindeki kullanım izni alınmış olan web günlük (*log*) dosyalarında otomatik olarak kaydedilmektedir. Web günlük dosyalarının, işleme noktasında çok büyük boyutlu dosya olmalarından ve kullanıcı etkileşimi dışında bu çalışmada kullanılmayacak satırlar içermesinden dolayı veri kaydetme/analizi noktasında farklı bir yol izlenmiştir. Bunun için, her bir kullanıcı etkileşimi, tıpkı günlük dosyalarındaki gibi MySQL veritabanına doğrudan kaydedilerek, web günlük dosyalarındaki gereksiz veri satırlarından arındırılmıştır. Bu sayede, hem performans anlamında zamandan kazanılmış hem de sadece kullanıcı davranışlarına odaklanılan bir veri kümesinin oluşturulması sağlanmıştır.

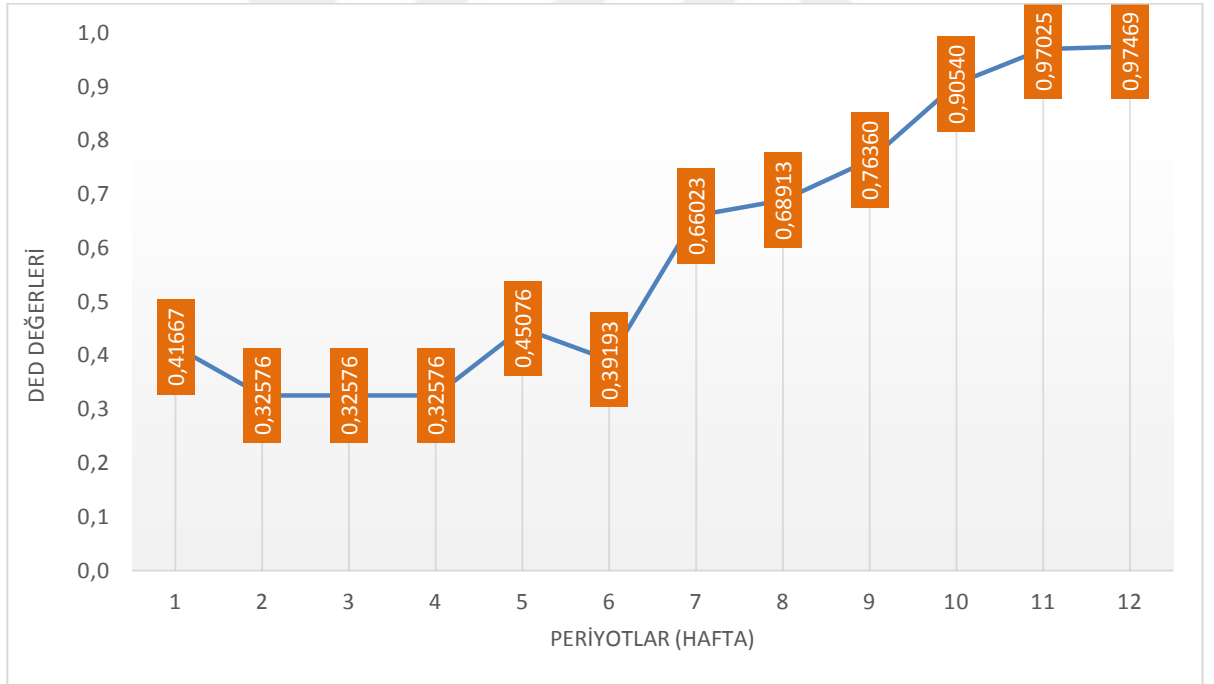
Çalışmada geliştirilen algoritmanın tasarlanması için JetBrains firması tarafından sağlanan PhpStorm 2018.2.3 uygulama geliştirme editörü kullanılmıştır. Bu editör, ücretli dağıtılmasına rağmen, öğrencilere yönelik olarak ücretsiz hizmet vermektedir. Dolayısıyla, PhpStorm 2018.2.3 uygulama geliştirme editöründen ücretsiz olarak faydalanılmıştır. Performans değerlendirme amacıyla gerçekleştirilen hipotez testleri için IBM SPSS Statistic 21 programının lisanslı hesabından yararlanılmıştır.

4. BULGULAR

Bu bölümde, ilk olarak, çalışma sürecinin genelinde ortaya çıkan bulgular paylaşılmıştır. Daha sonra, kullanılabilirlik tanımında yer alan üç farklı (etkililik, verimlilik ve memnuniyet) açıya bağlı kalınarak çıkarılmış olan bulgular paylaşılmıştır.

4.1. GENEL BULGULAR

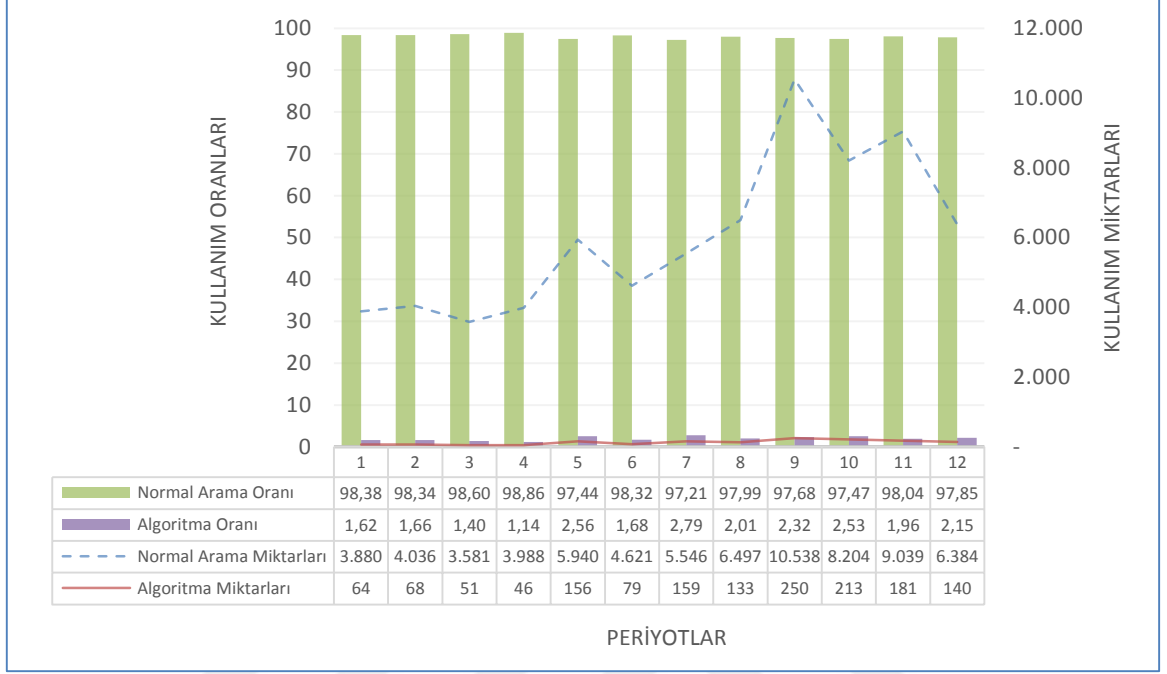
Çalışma süresince algoritmanın yönlendirilmesini sağlamak amacıyla geliştirilmiş olan geri bildirim mekanizması sayesinde, toplam 12 haftalık bir süre boyunca farklı Disiplin Eşik Değerleri (*DED*) ortaya çıkmıştır. 0-1 aralığında değerler alabilen *DED* parametresinin 0'a yaklaşması, kullanıcılara önerilerin daha esnek bir şekilde sunulmasını, 1'e yaklaşması ise daha disiplinli bir şekilde öneri sunulmasını sağlamaktadır. Çalışma süresince ortaya çıkan *DED* değerleri Şekil 4.1'de görülmektedir.



Şekil 4.1: *DED*'lerin 12 haftalık süreçteki değişim grafiği.

Şekil 4.1 incelendiğinde, algoritmanın haftalar ilerledikçe daha disiplinli bir şekilde öneri sunmaya yöneldiği görülmektedir. Bu durumda, geri bildirim mekanizmasının, kullanıcıların web sitelerinde gerçekleştirdiği aramalar üzerinden ulaştıkları kaynakları yeterli görmediğine

karar verdiği ifade edilebilir. İlgili süreçte gerçekleştirilmiş olan arama miktarlarına ait istatistikler Şekil 4.2’de aktarılmıştır.



Şekil 4.2: Arama araçlarının kullanım istatistikleri.

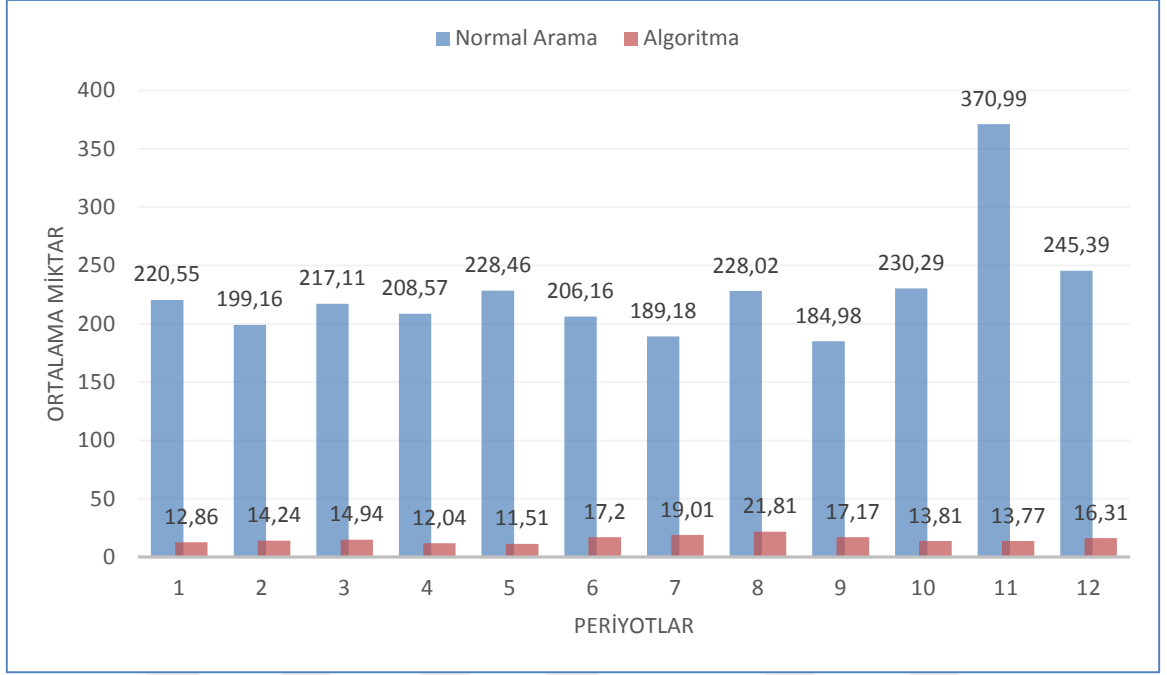
Şekil 4.2, algoritmanın kullanım oranlarının ilerleyen periyotlarda arttığını göstermektedir. Bu artışın, *DED* değerleriyle bir ilişkisinin olup olmadığının belirlenmesi amacıyla aşağıdaki şekilde hipotezler belirlenerek, veri miktarının düşük olmasından ($n=12$) ve oran değerlerinin sürekli sayısal veri türünde olmasından dolayı Spearman Korelesyon testi yapılmıştır:

H_0 : *DED* değerleri ile algoritmanın kullanım oranları arasında ilişki yoktur.

H_1 : *DED* değerleri ile algoritmanın kullanım oranları arasında ilişki vardır.

Test sonuçları 0,05 anlamlılık düzeyinde her iki değişken arasında pozitif yönde anlamlı bir ilişki olduğunu göstermektedir ($r_s=0.641$; $p=0,025<0,05$). Bu durum, “*DED* değerindeki artışın, kullanıcılara daha mantıklı gelen önerilerin ortaya çıkarılması anlamına geldiği ve bu doğrultuda kullanım oranlarının arttığı” şeklinde ifade edilebilir. Bu değerlendirmeye rağmen, kullanıcıların daha önce karşılaşmadıkları bu arayüze zamanla alışma ihtimalinin göz ardı edilmemesi gerekir. Bu sebeple, her ne kadar H_0 hipotezi çürütülse de, daha fazla çalışma yapılarak bu ilişkinin ne yönde olduğu daha net ortaya çıkarılabilir.

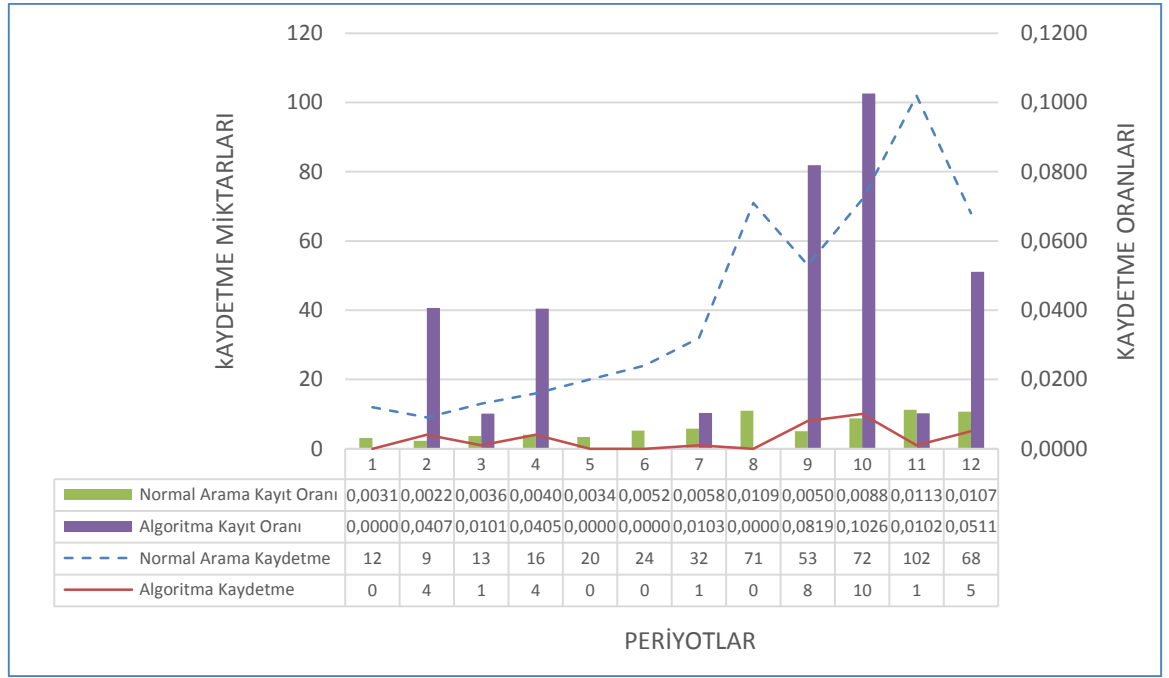
Periyotlar boyunca hem normal arama aracı hem de algoritma tarafından kullanıcıların aramalarına karşılık olarak sunulan ortalama kaynak öneri miktarlarına ait istatistik Şekil 4.3’de görülmektedir.



Şekil 4.3: Arama araçlarının ortalama kaynak öneri miktarları.

Şekil 4.3’de normal arama aracına nazaran algoritmanın çok daha az miktarda öneri sunduğu görülmektedir. Bu düşük miktarlar algoritmanın beklenildiği şekilde davranış sergilediğinin bir göstergesidir. Öyle ki, algoritmanın geliştirilmesi esnasında kullanıcıların benzer aramalarında yaptıkları ortak ziyaretleri bir araya getirmesine yönelik tasarım işlemleri gerçekleştirilmiştir. Bu eylemin getirisi, algoritma davranışlarında bariz bir şekilde görülmüştür. Bu noktada sorulması gereken soru: “öneri miktarlarındaki düşüklüğün nasıl bir etkiye sahip olacağı” şeklindedir. Bu soruya yönelik olarak verilebilecek en uygun cevap ise, kullanıcıların aradıkları bilgiye ulaşırken daha özet halde sunulan öneriler sayesinde daha etkili ve verimli bir biçimde işlemlerini tamamlayabilecekleridir. Bu cevabın doğruluğu ilerleyen kısımlarda aktarılan bulgularla da desteklenmiştir. Algoritma tarafından önerilen kaynak miktarlarının tamamıyla kullanıcı davranışlarını temel almasından ve *DED* değerinin bu davranışları değiştirme özelliği olmamasından dolayı bu istatistik ile *DED* değerleri arasında herhangi bir korelasyon ölçümü yapılmamıştır.

Çalışmada yapılmış olan bir diğer inceleme, kullanıcıların periyotlarda sunulan kaynak kaydetme durumlarının belirlenmesi şeklinde olmuştur. Bu incelemenin yapılma sebebi, algoritma tarafından sunulan kaynakların kullanıcılar tarafından ne kadar önemli bulunduğu yönelik işaretleri tespit etmeye çalışmaktır. Bu doğrultuda, kullanıcıların periyotlar boyunca kaynak kaydetme miktarlarına ve bu miktarların yapılmış olan arama miktarlarına karşılık gelen oranlarına (oran= kaydetme miktarı / arama miktarı) ait istatistikler Şekil 4.4'deki gibi ortaya çıkmıştır.



Şekil 4.4: Kullanıcıların periyodik kaynak kaydetme istatistikleri.

Şekil 4.4 incelendiğinde, miktarlar açısından normal arama aracının üstün olduğu düşünülebilir. Oransal açıdan bakıldığında, algoritmanın önerdiği kaynakları kaydetme davranışının bazı periyotlarda (1, 5, 6 ve 8) hiç görülmemesine rağmen, kaydedildiği haftaların çoğunda oransal açıdan bariz bir üstünlüğü olduğu söylenebilir. Yine de, bu istatistiğe yönelik olarak daha fazla çalışmanın yapılmasının gerektiği düşünülmektedir. Öyle ki, Spearman Korelasyonu test sonuçlarında 0,05 anlamlılık düzeyinde her iki değişken arasında anlamlı bir ilişkinin bulunamaması da bu savı destekler niteliktedir ($p=0,268>0,05$).

Çalışmada yapılmış olan bir diğer inceleme, algoritmanın 12 haftalık periyotlar boyunca her gün geçmiş bir aylık sürece yönelik yapmış olduğu analizlerde ortaya çıkardığı kural miktarlarının, *DED* değeriyle nasıl bir ilişki içerisinde olduğunun tespitine yönelik olmuştur.

Kurallar, algoritmanın kullanıcı davranışlarını analiz ederek, kullanıcıların doğru bilgiye ulaşmalarına fayda sağlayan ve farklı terimlerin bir arada tutulduğu yapılardır. Örnek olarak, farklı kullanıcılar tarafından benzer amaçla yapılmış “yapay zekâ” ile “y. zekâ” arama terimlerinin birlikte olacak şekilde kullanıcıya gösterilmesi verilebilir. Bu inceleme yapılırken, periyodik olarak oluşturulmuş olan kuralların haftalık toplam miktarlarının tek başına kullanılması yerine her bir analiz günü ortaya çıkarılmış kural miktarlarının kullanılmasına karar verilmiştir. Bunun sebebi, kuralların geçmiş bir aylık sürede yapılan kullanıcı arama davranışlarını temel alması ve haftalık toplam kural miktarları üzerinden yapılacak bir analizin geçmiş kullanıcı arama miktarlarını görmezden gelmek gibi bir hata olacağıdır. Bu doğrultuda, 12 haftalık süreçte toplam 84 defa yapılmış olan analizlerde üretilmiş olan kural miktarlarının, *DED* değerleriyle nasıl bir ilişkide olduğu belirlenmeye çalışılmıştır. Bu amaç için ilk olarak, analizin yapıldığı an ile geçmiş bir aylık süreç arasında gerçekleştirilmiş olan toplam arama miktarları (arama miktarları değişkeni, kullanıcılar tarafından arama yapılmış ve en az bir sayfa ziyareti içeren etkileşimler temel alınarak oluşturulmuştur) ve ilgili analiz gününde aktif olan *DED* değeri birlikte kullanılarak bir oran (Kural Oluşturma Oranı - *KOOR*) ortaya çıkarılmıştır. *KOOR* değeri, Denklem 4.1’deki şekilde hesaplanmıştır.

$$KOOR = \frac{\text{geçmiş ay yapılan toplam arama miktarı}}{\text{Aktif DED değeri}} \quad (4.1)$$

Algoritmanın yapısı gereği, *DED* değerinin artması durumunda oluşturulacak olan kural miktarlarının azalması, tam tersi durumda da artması, beklenen bir sonuçtur. Çünkü *DED* değeri disiplin anlamında algoritmayı yönlendiren bir parametredir. Bu açıklama dikkate alındığında, Denklem 4.1’de bölünen kısmın sabit tutularak *DED* değerinin arttırılması durumunda *KOOR* değerinin azalacağı, *DED*’in azaltılması durumunda da *KOOR*’un artacağı tahmin edilebilir. *KOOR* değeri ile ilgili analiz gününde ortaya çıkarılmış olan kural miktarları arasındaki ilişkinin incelenmesi doğrultusunda, *DED* değerinin, oluşturulan kural miktarlarına direk etkisinin olduğu tespit edilebilir. Bu tespit için hem *KOOR* hem de oluşturulan kural miktarlarından oluşan toplam 84 satırlık verinin normal dağılım gösterip göstermediği incelenmiştir (Tablo 4.1).

Tablo 4.1: KOOR ve kural miktarlarının normallik test sonuçları.

	Kolmogorov-Smirnov			Shapiro-Wilk		
	İstatistik	Serbestlik Derecesi	Anlamlılık Düzeyi	İstatistik	Serbestlik Derecesi	Anlamlılık Düzeyi
KOOR	,149	84	,000	,922	84	,000
Kural Miktar	,113	84	,010	,940	84	,001

“Dağılım normal dağılır” şeklinde kurulan H_0 hipotezinin 0,05 anlamlılık düzeyine göre çürütüldüğü Tablo 4.1’de görülmektedir. Bu sebeple, her iki değişken arasındaki ilişkinin tespiti için Spearman Korelasyon testi 0,05 anlamlılık düzeyinde olacak şekilde gerçekleştirilmiştir. Test sonuçları, belirtilen anlamlılık düzeyinde her iki değişken arasında pozitif yönde oldukça güçlü bir ilişki olduğunu göstermektedir ($r_s=0.846$; $p=0,000<0,05$). Bu durumda, Denklem 4.1 göz önüne alındığında, *DED* parametresinin artmasının kural miktarlarını azaltacağı, tam tersi durumda da arttıracığı rahatlıkla ifade edilebilir ki bu sonuç da algoritmadan beklenen bir davranış olarak görülmektedir.

Bu bölümde aktarılacak olan son inceleme ise, algoritma tarafından ortaya çıkarılmış kurallarda geçen arama terimlerinin, geçmiş bir aylık süreçte yapılmış olan tüm aramaların ne kadarlık bir kısmına tekabül ettiğinin periyodik olarak belirlenmesine ve algoritmanın, farklı terimler arasındaki ilişki tespitinde faydalandığı arama etkileşimleriyle, *DED* arasındaki ilişkiyi tespit etmeye yöneliktir. Bu doğrultuda, ilk olarak *DED* parametresinin etkisi incelenmiş ve bunun sonrasında kurallarda geçen terimlerle yapılmış aramaların, geçmiş bir aylık süreçte yapılmış tüm aramaların ne kadarlık kısmına denk geldiğine yönelik periyodik sonuçlar paylaşılmıştır.

DED etkisinin incelenme işlemi için toplam 84 günlük analiz verisi ayıklanmış ve bu ayıklama sonucu iki farklı değişken oluşturulmuştur. Değişkenler, kullanıcılar tarafından arama yapılmış ve en az bir sayfa ziyareti içeren eylemleri temel alınarak oluşturulmuştur. Bir başka ifadeyle, terk edilmemiş sorgular temel alınmıştır. Bu değişkenlerden ilki, bir önceki analizde de kullanılmış olan *KOOR* değeri, diğeri ise analiz anında ortaya çıkarılmış kurallarda geçen terimlerin geçmiş bir aylık süreçte yapılan toplam aramaların ne kadarlık miktarına denk geldiğini ifade eden bir değişkendir (Kural Oluşturmada Kullanılan Arama Miktarı - *KOKAM*). Algoritmanın yapısı gereği, *DED* değerinin artması durumunda *KOKAM* değerinin azalması, tam tersi durumda da artması, beklenen bir sonuçtur.

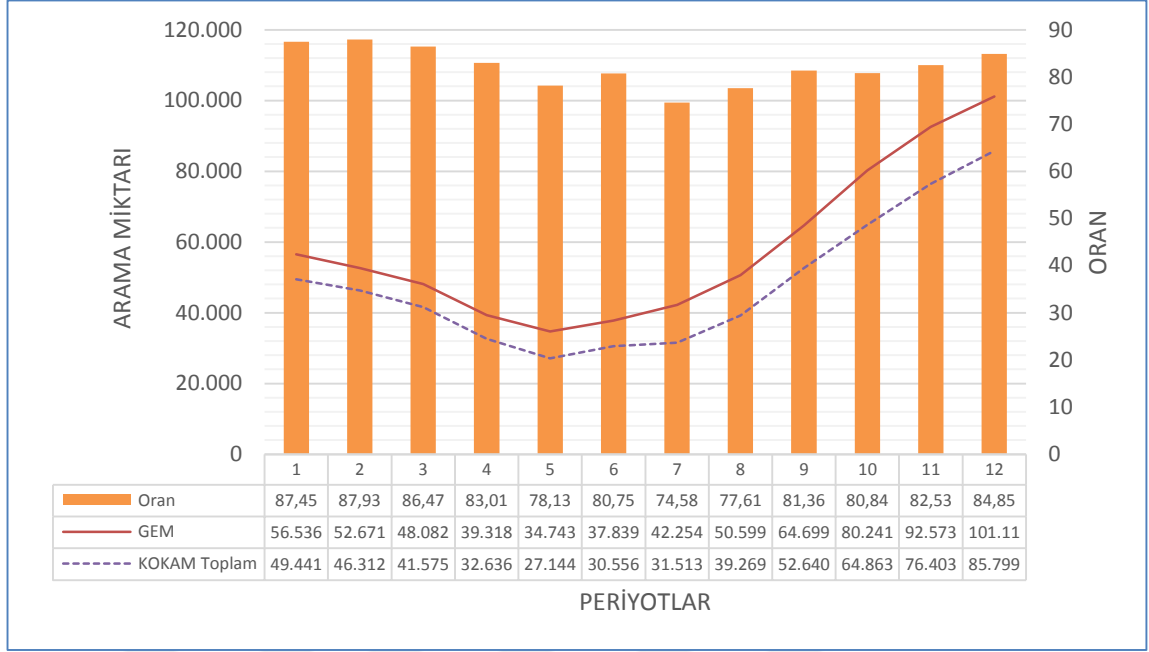
KOOR değeri ile *KOKAM* arasında yapılacak olan korelasyon testi doğrultusunda, *DED* değerinin, *KOKAM*'la olan olası ilişkisi tespit edilebilir. Bu tespit için ilk olarak hem *KOOR* hem de *KOKAM*'dan oluşan toplam 84 satırlık verinin normal dağılım gösterip göstermediği incelenmiştir (Tablo 4.2).

Tablo 4.2: *KOOR* ve *KOKAM* normallik test sonuçları.

	Kolmogorov-Smirnov			Shapiro-Wilk		
	İstatistik	Serbestlik Derecesi	Anlamlılık Düzeyi	İstatistik	Serbestlik Derecesi	Anlamlılık Düzeyi
<i>KOOR</i>	,149	84	,000	,922	84	,000
<i>KOKAM</i>	,152	84	,000	,893	84	,000

Bir önceki analize benzer şekilde H_0 hipotezinin 0,05 hata düzeyine göre çürütüldüğü Tablo 4.2'de görülmektedir. Bu sebeple, her iki değişken arasındaki ilişkinin tespiti için Spearman Korelasyon testi 0,05 anlamlılık düzeyinde olacak şekilde gerçekleştirilmiştir. Test sonuçları, belirtilen hata düzeyinde her iki değişken arasında pozitif yönde bir ilişki olduğunu göstermektedir ($r_s=0.291$; $p=0,007<0,05$). Bu durumda, Denklem 4.1 göz önüne alındığında, *DED* parametresinin artmasının *KOKAM*'ı azaltacağı, tam tersi durumda da arttıracığı rahatlıkla ifade edilebilir ki bu sonuç da algoritmadan beklenen bir davranış olarak görülmektedir.

Kurallarda geçen terimlerle yapılmış aramaların, geçmiş bir aylık süreçte yapılmış toplam aramaların ne kadarlık kısmına denk geldiğine yönelik periyodik sonuçlar hesaplanırken, *DED* etkisinin incelenme işleminde kullanılmış olan 84 satırlık veri, her bir periyot için toplanarak toplamda 12 haftalık periyoda denk gelecek şekilde birleştirilmiştir. Bu sayede, ilgili periyotta geçmişe dönük incelenmiş toplam arama etkileşim miktarı (*GEM*) ve tüm periyot boyunca oluşturulmuş olan kurallarda geçen terimlerin, *GEM* içindeki miktarı olmak (*KOKAM* Toplam) üzere 2 değişken oluşturulmuştur. Her iki değişken üzerinden Şekil 4.5'deki istatistikler ortaya çıkarılmıştır.



Şekil 4.5: Periyodik GEM ve KOKAM istatistikleri.

Şekil 4.5 incelendiğinde, periyotların hemen hemen hepsinde (9/12), oransal açıdan algoritmanın %80 değerinin üzerine çıktığı görülmektedir. Bu istatistiğin açıklaması şu şekilde yapılabilir: algoritma, kullanıcılar tarafından yapılmış olan farklı, fakat aynı amaca hitap eden aramalar arasındaki ilişkileri yüksek oranda tespit ederek kuralları oluşturmuştur.

4.2. ETKİLİLİK BULGULARI

Algoritmanın etkililik ölçümü, Reciprocal Rank (RR) değeri temel alınarak ve tüm periyotlar tek bir bütün halinde değerlendirilerek gerçekleştirilmiştir. Bu ölçüm için veri seti hazırlanırken, algoritmanın ürettiği kuralların, kullanıcılar tarafından kullanılmış olanları kayıt altına alınmıştır. Örnek olarak, bir kullanıcının arama sayfasında arama terimini yazdığı esnada algoritma tarafından çeşitli terim önerilerinin (aynı zamanda kural önerileri şeklinde de nitelendirilebilir) karşısına getirildiği varsayalım. Kullanıcı bu kurallardan herhangi birine tıkladığı anda, karşısına kuralla ilgili olan kaynak öneri listesi getirilir. Kullanıcının, bu listede tıkladığı ilk kaynağın sırası dikkate alınarak hesaplanan RR değeri, veri setinde bir kayıt olarak yer almıştır. Bu yöntem kullanılarak toplam 655 satırlık (RR değerlerinden oluşan) bir veri seti oluşturulmuştur. RR değerinin yüksek olması kullanıcıların önerilen kaynaklarda en üstte yer alan kaynakları ziyaret ettikleri anlamına gelmektedir. Bu kullanıcı davranışı, aynı zamanda algoritmanın kullanıcılara en ilgili kaynakları üstte sunması şeklinde de ifade edilebilir. 655

satırlık veri setindeki her bir tekil *RR* değeri, algoritmanın geçmişte yapılan benzer kullanıcı davranışlarını analiz ederek oluşturduğu bir kuralın, kullanıcıya sunulduktan sonraki kullanıcı tercihini ifade etmektedir. Diğer bir ifadeyle, geçmişte yapılmış benzer kullanıcı davranışlarıyla eğitilen algoritmanın bir ürünüdür. Bu durumda, tercih edilmiş olan kurallar temel alınarak, algoritmanın, kullanıcıların geçmiş davranışlarından ne derece öğrendiği ve kullanıcılara verilen hizmeti ne kadar arttırdığı tespit edilebilir. Bu doğrultuda, geçmişte normal arama aracı ile kullanıcıların gerçekleştirdikleri arama davranışlarındaki *RR* değerleri **eğitim evreni** olarak nitelendirilebilir. Kullanıcıların tercih ettiği kurallardan ortaya çıkarılmış olan 655 satırlık *RR* veri seti ise **test örnekleme** şeklinde ifade edilebilir. Algoritmanın kullanıcılara fayda sağladığının ispat edilmesi için, en az eğitim evreninde elde edilmiş sonuç kadar etkili olması beklenir. Bu etkililiğin belirlenmesi/anlamlandırılması amacıyla ilişkili örneklem testleri (sign test) gerçekleştirilmiştir. Bu testin kullanılmasının sebebi, eğitim evreninin Kolmogorov-Smirnov test sonuçları temel alındığında normal dağılım göstermemesidir ($N=135.148$; $p=0,000<0,05$; H_0 hipotezi çürütülmüştür). Alpar (2016) normal dağılım göstermeyen bir evren için işaret testinin kullanılabileceğini ifade etmiştir. Evren ortancası üzerinden değerlendirme yapmaya olanak sağlayan işaret testi, parametrik olmayan bir test türüdür (Alpar, 2016). Bu çalışma için temel alınan eğitim evreninin ortanca değeri 0,5 olarak tespit edilmiştir. Bu değer temel alınarak, aşağıdaki şekilde hipotezler oluşturulmuş ve işaret testiyle anlamlılık ölçümü yapılmıştır:

H_0 : Algoritmanın ortaya çıkardığı *RR* değerlerine ilişkin ortanca 0,5'e eşittir.

H_1 : Algoritmanın ortaya çıkardığı *RR* değerlerine ilişkin ortanca 0,5'ten farklıdır.

H_0 hipotezinin çürütülmesi, etkililik bakımından algoritmanın normal arama aracından farklı bir şekilde hizmet sunduğu anlamına gelmektedir. 0,05 anlamlılık düzeyinde gerçekleştirilmiş olan işaret testi sonuçları Tablo 4.3'de paylaşılmıştır ($z=-2,421$; $p=0,015<0,05$).

Tablo 4.3: *RR* değerleri üzerinden işaret testi sonuçları.

	N
EvrenMedyan - Algoritma	Negatif Farklılıklar ^a
	296
	Pozitif Farklılıklar ^b
	239
	Eşit Olanlar ^c
	120
	Toplam
	655

a. EvrenMedyan < Algoritma

b. EvrenMedyan > Algoritma

c. EvrenMedyan = Algoritma

Tablo 4.3'e göre algoritma üzerinden elde edilen *RR* değerlerinin, evren ortancasından farklı olduğu ispat edilerek H_0 hipotezinin çürütüldüğü ortaya çıkartılmıştır. Bu noktada, Tablo 4.3'de yer alan “*Negatif Farklılıklar*” miktarının “*Pozitif Farklılıklar*” miktarından büyük olduğu görülmektedir. Bu durum, “algoritmanın, evren ortancasına nazaran daha büyük *RR* değerlerini oluşturduğu ve dolayısıyla etkililik bakımından daha başarılı bir hizmet sunduğu” şeklinde ifade edilebilir.

DED değerinin etkililiği nasıl değiştirdiğinin tespit edilmesi amacıyla, çalışmada ortaya çıkmış olan farklı *DED* değerleri 1. periyottan son periyoda kadar tekrar numaralandırılmış (2, 3 ve 4 numaralı periyotlar aynı *DED*'e sahip olduğu için 2 ile temsil edilmiştir, bu haliyle toplam 10 farklı *DED*'den oluşan bir grup oluşturulmuştur) ve her bir periyotta ortaya çıkmış *RR* değerleri üzerinden normallik testine tabi tutulmuştur (Tablo 4.4).

Tablo 4.4: *DED* gruplarına ait normallik test sonuçları.

DED Grupları	Kolmogorov-Smirnov			Shapiro-Wilk			
	İstatistik	Serbestlik Derecesi	Anlamlılık Düzeyi	İstatistik	Serbestlik Derecesi	Anlamlılık Düzeyi	
RR	1,00	,262	32	,000	,790	32	,000
	2,00	,235	76	,000	,805	76	,000
	3,00	,280	55	,000	,798	55	,000
	4,00	,325	32	,000	,774	32	,000
	5,00	,296	64	,000	,783	64	,000
	6,00	,391	50	,000	,708	50	,000
	7,00	,296	111	,000	,800	111	,000
	8,00	,338	109	,000	,772	109	,000
	9,00	,277	57	,000	,779	57	,000
	10,00	,329	69	,000	,747	69	,000

Tablo 4.4 incelendiğinde, hiçbir grubun *RR* değerleri açısından normal dağılım göstermediği görülmektedir. Bu sebeple, farklı *DED* değerleriyle oluşturulmuş 10 grup arasında farklılığın olup olmadığının test edilmesi amacıyla, ilk olarak, bağımsız ikiden çok grubun karşılaştırılmasında kullanılan Kruskal-Wallis Varyans analizi gerçekleştirilmiş olup 0,05 anlamlılık düzeyinde “gruplar arasında fark yoktur” şeklinde oluşturulmuş olan H_0 hipotezinin çürütüldüğü belirlenmiştir ($p=0,012<0,05$). Bu aşamadan sonra hangi grubun kendi dışındaki gruba üstünlük sağladığının tespit edilmesi amacıyla tüm grupların birbiriyle kıyaslanması için Mann-Whitney U bağımsız iki örneklem testi gerçekleştirilmiştir (Toplam karşılaştırma miktarı: 45). İkili karşılaştırmaların yapıldığı bu teste ait farklılık içeren sonuçlar Tablo 4.5’de paylaşılmıştır.

Tablo 4.5: *DED* gruplarına ait etkililik karşılaştırmaları.

Karşılaştırma	Gruplar	DED	Ortanca	Ortalama	n	z	P
1	2	0,32576	0,33	0,5	76	-2,706	0,007
	3	0,45076	0,5	0,64586	55		
2	2	0,32576	0,33	0,5	76	-3,520	0,000
	6	0,68913	1	0,74246	50		
3	2	0,32576	0,33	0,5	76	-3,461	0,001
	8	0,90540	1	0,68597	109		
4	1	0,41667	0,5	0,565622	32	-2,194	0,028
	6	0,68913	1	0,74246	50		
5	5	0,66023	0,5	0,591766	64	-2,189	0,029
	6	0,68913	1	0,74246	50		

Tablo 4.5 içindeki ikili grup karşılaştırmalarında kullanılan hipotezler aşağıdaki gibidir:

H_0 : İki dağılım arasında fark yoktur.

H_1 : İki dağılım arasında fark vardır.

Toplamda 45 ayrı karşılaştırmaların yapılması sonrasında, 5 tane ikili arasında yapılan karşılaştırmaların 0,05 anlamlılık düzeyinde H_0 hipotezini çürüttüğü tespit edilmiştir. Tablo 4.5’de verilen tüm karşılaştırma sonuçlarına göre, *DED* değeri büyük olan grubun daha üstün olduğu ortaya çıkarılmıştır. Bu sonuç, ***DED* değerindeki artışın etkililiği arttırdığına** işaret eden önemli bir bulgudur. Bu değerlendirmeye ek olarak, *DED* değerlerinin *RR*’yi nasıl etkilediğinin tespit edilmesi için bir inceleme daha gerçekleştirilmiştir. Bu incelemede, toplam 12 haftalık periyottaki *RR* değerlerinin *DED* ile ilişkisinin olup olmadığına yönelik bir korelasyon testi yapılmıştır. Her bir periyodu temsil etmesi amacıyla tek bir *RR* değeri (her bir periyottaki *RR* değerlerinin kendi içinde normal dağılmamasından dolayı ortalama yerine ilgili periyottaki *RR* ortancası kullanılmıştır) seçilerek, ilgili periyotta aktif olan *DED* ile korelasyon testi gerçekleştirilmiştir. Veri miktarının $n=12$ olmasından dolayı gerçekleştirilmiş olan Spearman Korelasyon test sonuçlarında 0,05 anlamlılık düzeyinde pozitif yönde bir ilişki olduğu ortaya çıkarılmıştır ($r_s=0.586$; $p=0,045<0,05$). Bu durumda, tıpkı Mann-Whitney U test sonuçlarında olduğu gibi, *DED* değerinin artması durumunda *RR* değerlerinin de artış göstereceği ifade edilebilir. Bu korelasyon test sonuçlarına rağmen, her iki değişken arasındaki ilişkinin daha belirgin hale getirilebilmesi amacıyla daha fazla çalışma (korelasyon ölçümleri bakımından) yapılması gerektiği de düşünülmektedir.

4.3. VERİMLİLİK BULGULARI

Algoritmanın verimlilik ölçümleri aşağıdaki şekilde üç farklı başlık altında incelenmiştir:

- Sorgu Gecikmesi (*Query Latency*)
- İlk Tıklama Süresi (*Time to First Click*)
- Sorgu Terk Etme (*Query Abandonment*)

Sorgu Gecikmesi (Query Latency)

Etkililik ölçümünde yapılan işleme benzer şekilde, eğitim evreni ve test örnekleme bu ölçümde de yer almaktadır. Etkililik ölçümlerinden farkı ise, kullanıcılar tarafından yapılmış olan aramalarda sayfa ziyareti içermeyen sorgulamaların da veri setine eklenmiş olmasıdır. Bunun sebebi, kullanıcıların, sorguları terk etseler bile bir arama eylemini gerçekleştirdiklerinden dolayı sistem tarafında bir yük oluşturmuş olmalarıdır. Bu doğrultuda, mikrosaniye cinsinden kaydedilmiş olan sorgu süreleri Denklem 4.2'deki şekilde kullanılarak, Tekil Kaynak Öneri Süresi (*TÖS*) değişkeni ortaya çıkarılmıştır.

$$TÖS = \frac{\text{Sorgu Bitiş Süresi} - \text{Sorgu Başlama Süresi}}{\text{Önerilen Kaynak Öneri Miktarı}} \quad (4.2)$$

TÖS, sistemin, kaynak başına ne kadar sürede öneri yapabildiğine yönelik bir değişken olarak ifade edilebilir. Etkililik veri setinin hazırlanmasında kullanılan senaryonun aynısı bu ölçüm için oluşturulan veri setinde de kullanılmıştır. Tek fark, *RR* yerine *TÖS* değerlerinin arama başına kaydedilmesi ve terk edilen sorgulamaların da kayıt altında tutulmasıdır. Eğitim evreninin normal dağılım göstermemesi dolayısıyla bu ölçüm için de işaret testinden faydalanılmıştır ($N= 906.178$; $p=0,000<0,05$). Bu ölçümde temel alınan eğitim evreninin ortanca değeri 0,00581578 mikrosaniye olarak tespit edilmiş olup, oluşturulan hipotezler aşağıdaki şekildedir:

H_0 : Algoritmanın ortaya çıkardığı *TÖS* değerlerine ilişkin ortanca 0,00581578'e eşittir.

H_1 : Algoritmanın ortaya çıkardığı *TÖS* değerlerine ilişkin ortanca 0,00581578'ten farklıdır.

H_0 hipotezinin çürütülmesi, algoritmanın normal arama aracından farklı bir şekilde hizmet sunduğu anlamına gelmektedir. 0,05 anlamlılık düzeyinde gerçekleştirilmiş olan işaret testi sonuçları Tablo 4.6'da paylaşılmıştır ($z=-38,453$; $p=0,000<0,05$).

Tablo 4.6: *TÖS* değerleri üzerinden işaret testi sonuçları.

		N
EvrenMedyan - Algoritma	Negatif Farklılıklar ^a	15
	Pozitif Farklılıklar ^b	1525
	Eşit Olanlar ^c	0
	Toplam	1540

a. EvrenMedyan < Algoritma

b. EvrenMedyan > Algoritma

c. EvrenMedyan = Algoritma

Tablo 4.6'ya göre algoritma üzerinden elde edilen *TÖS* değerlerinin, evren ortancasından farklı olduğu ispat edilerek H_0 hipotezinin çürütüldüğü ortaya çıkartılmıştır. . Bu noktada, Tablo 4.6'da yer alan “*Negatif Farklılıklar*” miktarının “*Pozitif Farklılıklar*” miktarından küçük olduğu görülmektedir. Bu durum, “algoritmanın, evren ortancasına nazaran daha küçük *TÖS* değerlerini oluşturduğu ve dolayısıyla verimlilik bakımından daha hızlı bir hizmet sunduğu” şeklinde ifade edilebilir.

DED parametresinin sistem performansını değiştirme durumu olmadığı için farklı *DED* grupları arasında kıyaslama yapılmamıştır. Bunun yerine, algoritmanın sorgu işlem süresi açısından verimliliğe olan katkısı farklı bir şekilde incelenmiştir. Hatırlanacağı üzere, algoritmanın kullanıcıların davranışlarını özetlediği ve bunun verimliliğe fayda sağladığı belirtilmişti. Bu tespiti yönelik olarak, *TÖS* formülünde yer alan sorgulama süresi farkı (Denklem 4.2'nin bölünen kısmı) ile önerilen kaynak miktarları arasındaki ilişki incelenmiştir. Bu işlem için ilk olarak, her bir kümenin normal dağılım durumları ölçülmüştür (Tablo 4.7). Verilerin kendi içinde normal dağılım göstermemelerinden dolayı Spearman Korelasyon testi gerçekleştirilmiş olup, test sonuçları, pozitif yönde bir ilişki ortaya çıkarmıştır ($n=1540$; $r_s=0,425$; $p=0,000<0,05$). Bu sonuç, sonuç miktarının artması durumunda sorgu süresinin de artma eğiliminde olduğunu göstermektedir. Bu durumda, algoritmanın kullanıcı davranışlarını özetleyebilme yeteneğinin, sistemsel açıdan verimliliğe katkı sağlayabildiği ve bu katkının da dolaylı olarak kullanıcı verimliliğine olan katkı anlamına geldiği rahatlıkla belirtilebilir.

Tablo 4.7: Sorgu süresi ve önerilen kaynak miktarlarına ait normallik test sonuçları.

	Kolmogorov-Smirnov			Shapiro-Wilk		
	İstatistik	Serbestlik Derecesi	Anlamlılık Düzeyi	İstatistik	Serbestlik Derecesi	Anlamlılık Düzeyi
sure_farki	,379	1540	,000	,168	1540	,000
kaynak_oneri	,123	1540	,000	,931	1540	,000

İlk Tıklama Süresi (Time to First Click)

Bu ölçüm için, etkililik ölçümünde yapılan işleme benzer şekilde, eğitim evreni ve test örneklemini oluşturulmuştur. Kullanıcıların bir arama işlemini başlattıkları an ile kendilerine önerilen kaynaklardan herhangi birine ilk tıkladıkları ana kadar geçen sürenin temel alındığı bir veri seti oluşturulmuştur. Bu açıdan bakıldığında, tıpkı etkililik ölçümünde olduğu gibi terk edilmemiş sorgulamalardan oluşturulan bir test örneklemini söz konusudur. Bu doğrultuda, saniye cinsinden kaydedilmiş olan ilk tıklama süreleri Denklem 4.3'deki şekilde kullanılarak, İlk Tıklama Süresi (*İTS*) değişkeni ortaya çıkarılmıştır.

$$İTS = \text{Sorgu Başlama Süresi} - \text{İlk Tıklama Süresi} \quad (4.3)$$

İTS, kullanıcının en kısa zamanda bilgi ihtiyacını giderebileceği kaynağı bulduğuna işaret edebilme potansiyeli olan bir değişken olarak ifade edilebilir. Etkililik veri setinin hazırlanmasında kullanılan senaryonun aynısı bu ölçüm için oluşturulan veri setinde de kullanılmıştır. Tek fark, *RR* yerine *İTS* değerlerinin kayıt altında tutulmasıdır. Eğitim evreninin normal dağılım göstermemesi dolayısıyla bu ölçüm için de işaret testinden faydalanılmıştır ($N=135.148$; $p=0,000<0,05$). Bu ölçümde temel alınan eğitim evreninin ortanca değeri 17 saniye olarak tespit edilmiş olup, oluşturulan hipotezler aşağıdaki şekildedir:

H_0 : Algoritmanın ortaya çıkardığı *İTS* değerlerine ilişkin ortanca 17'ye eşittir.

H_1 : Algoritmanın ortaya çıkardığı *İTS* değerlerine ilişkin ortanca 17'den farklıdır.

H_0 hipotezinin çürütülmesi, algoritmanın, normal arama aracından farklı bir şekilde kullanıcıların bilgi ihtiyacını karşılayabildiği anlamına gelmektedir. 0,05 anlamlılık düzeyinde gerçekleştirilmiş olan işaret testi sonuçları Tablo 4.8'de paylaşılmıştır ($z=-4,936$; $p=0,000<0,05$).

Tablo 4.8: *İTS* değerleri üzerinden işaret testi sonuçları.

EvrenMedyan - Algoritma	N	
	Negatif Farklılıklar ^a	253
	Pozitif Farklılıklar ^b	378
	Eşit Olanlar ^c	24
	Toplam	655

a. EvrenMedyan < Algoritma

b. EvrenMedyan > Algoritma

c. EvrenMedyan = Algoritma

Tablo 4.8'e göre algoritma üzerinden elde edilen *İTS* değerlerinin, evren ortancasından farklı olduğu ispat edilerek H_0 hipotezinin çürütüldüğü ortaya çıkartılmıştır. Bu noktada, Tablo 4.8'de yer alan “*Negatif Farklılıklar*” miktarının “*Pozitif Farklılıklar*” miktarından küçük olduğu görülmektedir. Bu durum, “algoritmanın, evren ortancasına nazaran daha küçük *İTS* değerlerini oluşturduğu ve dolayısıyla verimlilik bakımından daha başarılı bir hizmet sunduğu” şeklinde ifade edilebilir.

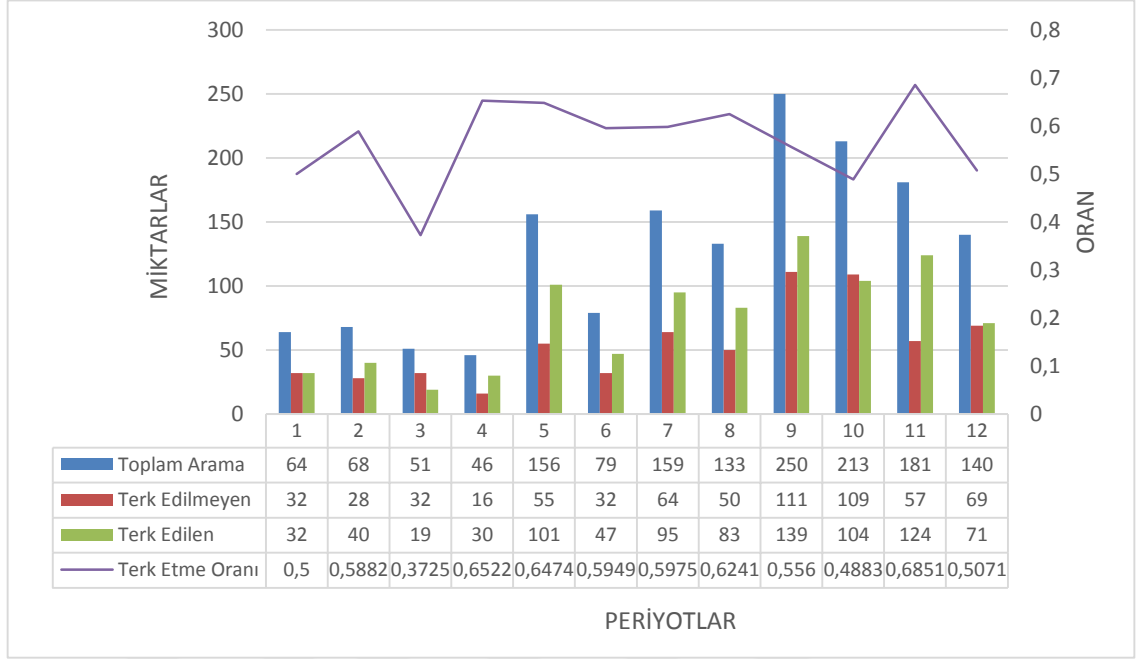
DED parametresinin *İTS*'yi değiştirebilme durumunun olup olmadığının incelenmesi amacıyla Kruskal-Wallis Varyans analizi gerçekleştirilmiş olup 0,05 anlamlılık düzeyinde “*DED* grupları arasında fark yoktur” şeklinde oluşturulmuş olan H_0 hipotezinin çürütülemediği belirlenmiştir ($p=0,341>0,05$). Buna ek olarak, etkililik ölçümünde yapılmış olan ikili karşılaştırmalara benzer şekilde, bu ölçüm için de farklı *DED* grupları birbiri arasında ikili olacak şekilde Mann-Whitney U testiyle incelenmiştir. Toplam 45 farklı karşılaştırma yapılmış olmasına rağmen, hiçbir karşılaştırmada anlamlı bir farklılık bulunamamıştır. Dolayısıyla *DED* parametresi ile *İTS* arasında belirgin bir ilişki tespit edilememiştir. Diğer taraftan, *İTS* ile önerilen kaynak miktarları arasındaki ilişki incelenmiştir. Bu işlem için ilk olarak, her bir kümenin normal dağılım durumları ölçülmüştür (Tablo 4.9). Verilerin kendi içinde normal dağılım göstermemelerinden dolayı Spearman Korelasyon testi gerçekleştirilmiş olup, test sonuçları, pozitif yönde bir ilişki ortaya çıkarmıştır ($n=655; r_s=0,218; p=0,000<0,05$). Bu sonuç, kaynak öneri miktarının artması durumunda *İTS*'nin de artma eğiliminde olacağını göstermektedir. Bu durumda, algoritmanın kullanıcı davranışlarını özetleyebilme yeteneği sayesinde, bu ölçüm açısından da verimliliğe katkı sağladığı belirtilebilir.

Tablo 4.9: *İTS* ve önerilen kaynak miktarlarına ait normallik test sonuçları.

	Kolmogorov-Smirnov			Shapiro-Wilk		
	İstatistik	Serbestlik Derecesi	Anlamlılık Düzeyi	İstatistik	Serbestlik Derecesi	Anlamlılık Düzeyi
its	,473	655	,000	,033	655	,000
kaynak_oneri	,132	655	,000	,924	655	,000

Sorgu Terk Etme (Query Abandonment)

Bir kullanıcı, algoritma tarafından önerilen kurallardan herhangi bir tanesine tıkladığında, yeni bir oturum başlatılmaktadır. Normal arama aracı içinse, kullanıcı, arama terimini yazıp “Ara” butonuna tıkladığında yeni oturum başlatılmaktadır. Açılan oturum sonrasında, kullanıcı karşısına aradığı bilgiye yönelik olarak çeşitli kaynak önerileri getirilmektedir. Kullanıcının herhangi bir kaynak önerisine tıklaması terk edilmeyen sorguyu, tıklamaması ise terk edilen sorguyu temsil etmektedir. Şekil 4.2’de aktarılmış olan algoritmaya ve normal arama aracına ait kullanım miktarları, aynı zamanda, toplam sorgu miktarlarına karşılık gelmektedir. Bu sorgular, terk edilmiş ve edilmemiş tüm arama etkileşimlerini kapsamaktadır. Algoritmanın, sorgu terk etme (*STE*) açısından verimliliğine dönük performans ölçümünde, her bir periyotta yapılmış olan aramalar (algoritmadan faydalanılarak yapılmış olanlar) ile algoritmanın kural oluşturmak için kullandığı geçmiş arama etkileşimlerinden faydalanılmıştır. Bu ölçüm için, etkililik ölçümünde yapılan işleme benzer şekilde, eğitim evreni ve test örnekleme oluşturulmuştur. Test örnekleme Şekil 4.2’de yer alan algoritmanın kullanım miktarları üzerinden oluşturulmuştur. Buna göre, algoritmanın periyotlar boyunca sorgu terk edilme ve edilmeme durumuna yönelik istatistik Şekil 4.6’daki gibi ortaya çıkarılmıştır.



Şekil 4.6: Algoritmanın periyodik sorgu durumlarına ait istatistik.

Şekil 4.6 incelendiğinde, toplam 1.540 defa kullanılmış olan algoritmanın 885 sorgusunun terk edildiği görülmektedir. Bu durumda, algoritmanın *STE* oranı %57,47 (0-1 aralığında 0,57 olarak ifade edilebilir) olarak ortaya çıkmaktadır. Algoritmanın, kuralları oluşturmada faydalandığı geçmiş kullanıcı etkileşimlerinde (normal arama aracıyla yapılan aramalar) ise bu rakamlar farklılık göstermektedir. 12 haftalık periyodun tamamında, faydalanan algoritma kurallarında geçen arama terimleri ile geçmiş bir aylık süreçte yapılmış (normal arama aracı kullanılarak) olan toplam sorgu miktarı 91.335 olarak belirlenmiştir. Bu sorgulardan 58.992 kadarı terk edilmiş olan sorguları temsil etmekte olup, %64,59'luk (0-1 aralığında 0,65 olarak ifade edilebilir) bir orana denk gelmektedir. Bu durumda, ilk bakışta, algoritma sorgularının daha az terk edildiği düşünülebilir. Bu istatistiğe rağmen, *STE* açısından algoritmanın üstünlüğünü kanıtlayabilmek için Tek Örneklem Ki-Kare Testi aşağıdaki hipotezler temel alınarak gerçekleştirilmiştir:

H_0 : Algoritmanın *STE* değeri 0,65'e eşittir.

H_1 : Algoritmanın *STE* değeri 0,65'den farklıdır.

Yapılmış olan analizler sonucunda 0,05 anlamlılık düzeyinde H_0 hipotezinin çürütüldüğü belirlenmiştir ($\chi^2 = 38,407$; $p = 0,000 < 0,05$). Bu sonuç, algoritmanın, normal arama aracından

daha az sayıda kullanıcıyı kaybettiği anlamına gelmekte ve kullanıcıların aradığı bilgiye ulaşmalarında daha verimli çalıştığını ifade etmektedir.

STE ile *DED* parametresi arasında bir ilişki olup olmadığının incelenmesi amacıyla, Şekil 4.6'da aktarılan 12 haftalık *STE* oranları ile ilgili dönemde aktif olan *DED* değerleri arasında korelasyon testi yapılmıştır. Spearman Korelasyon testi sonuçlarında, 0,05 anlamlılık düzeyinde bir ilişki bulunamamıştır ($p=0,913>0,05$).

4.4. MEMNUNİYET BULGULARI

Kullanıcıların terk etmedikleri sorgularda, ziyaret ettikleri sayfaları değerlendirmeleri amacıyla 1-5 aralığında puan verilebilecek şekilde tek soruluk bir anket uygulanmıştır (Şekil 3.5). Bu ankette, 1 değeri en düşük memnuniyeti temsil ederken, 5 değeri en yüksek memnuniyeti temsil etmiştir (Şekil 3.5). 12 haftalık süreçte, kullanıcıların oy verme konusunda çekingen davranmalarından dolayı veri miktarı yetersiz kalmıştır. Kullanıcılar, algoritmanın önerisi doğrultusunda ziyaret ettikleri toplam 655 sayfadan sadece 5 tanesini oylamış olup, bu oyların ortalaması 1,6 olarak ortaya çıkmıştır. Normal arama aracı ise, toplamda 135.148 defa ziyaret etkileşimine sebep olmasına rağmen, bu ziyaretlerden sadece toplam 162 tanesini oylanmış olup, ortalama 2,17'lik bir değer ortaya çıkmıştır. Memnuniyet açısından, hem algoritma hem de normal arama aracı vasıtasıyla ortaya çıkan ziyaret miktarları çok olmasına rağmen, kullanıcı oylamalarının yetersiz kaldığı görülmektedir. Bu sebeple, algoritma başarısının değerlendirilmesine yönelik olarak daha fazla çalışmanın yapılması gerektiği düşünülmektedir.

5. TARTIŞMA VE SONUÇ

Bilgi, her bir bireyin günlük yaşamında farklı kişilerle iletişim kurabilmesi, yapmayı düşündüğü bir eyleme başlayabilmesi veya tamamlayabilmesi için ihtiyaç duyduğu önemli bir kavramdır. Belki de, belirli bir sosyal çevre edinebilmesinde rol oynayan en kritik faktörlerden biridir. Geçmişte kütüphaneler gibi fiziksel ortamlar vasıtasıyla erişilebilen bilgi, günümüzde özellikle de internet'in desteğiyle sanal olarak daha kolay erişilebilir bir forma dönüşmüştür. Bu haliyle de, istenilen bir zamanda, istenilen kaynaktan, ihtiyaç duyduğumuz bilgiye erişebilir bir imkana kavuşmuş bulunmaktayız.

İnternet evreni ve bu evrende gezinmemizi sağlayan cihazlar, günlük yaşantımızda harcadığımız zamanın çoğunu kapsayan araçlardır. Daha önceleri belirli bir ortamda bulunan bilgisayarlar vasıtasıyla erişebildiğimiz internet, taşınabilir cihazların hayatımıza girmesiyle birlikte her yerden erişilebilir hale gelmiştir. İnternet'in bu denli erişilebilir olmasının yanında, çoğunlukla fiziki ortamlar vasıtasıyla erişebildiğimiz bilgi de, kendini daha hızlı yayabileceği ve genişletebileceği bir ortama aktarmıştır. Bu durum, insanlık için oldukça önemli bir gelişme olarak nitelendirilebilir. Bu avantaja rağmen, erişilebilir her bilginin doğru/faydalı olduğunu söylemenin hata olacağı rahatlıkla ifade edilebilir. Bunun sebebi, bilginin yanlış üretilmesi olabileceği gibi yanlış bir şekilde aktarılması da olabilir. Tüm bu avantajlar ve dezavantajlar göz önüne alındığında, “kullanışlı bilgiye erişim” kavramının, günümüzde oldukça özenle yaklaşılması gereken kritik bir araştırma alanı olduğu düşünülmektedir. Bu çalışmada da odak alınan kavram “kullanışlı bilgiye erişim” olarak yer almıştır.

Bir bilgi ihtiyacını gidermek amacıyla her gün sıklıkla faydalandığımız uygulamaların başında Google, Yandex, vb. arama motorları gelmektedir. Bu tür uygulamalar literatürde “Bilgi Erişim Sistemleri (BES)” adıyla tanınmaktadır. Çeşitli türleri olmasına rağmen, tüm BES'lerin ortak noktası, çeşitli formatlarla betimlemesi yapılan bilgi ihtiyacına karşılık en ilgili bilgi kaynaklarını ortaya çıkarmaktır. Kısacası, bilgiye ihtiyaç duyan taraf ile bu bilgiyi barındıran kaynaklar arasında bir köprü vazifesi görmektir. BES'ler, herhangi bir bilgi ihtiyacının en kullanışlı bilgi kaynaklarıyla karşılanabilmesi amacıyla, altyapılarında çeşitli modellerle hizmet veren uygulamalardır. Bu modeller, BES'lerin aranan bilgiyi analiz ederek ilgili bilgi kaynaklarıyla eşleştirebilmesinde görev yapan altyapısal bileşenlerden biridir. Sistem arkaplanında kullanıcıdan bağımsız olarak çalışan ve belirli kurallar çerçevesinde hareket eden

yapılardır. Bu çalışmada, belirli bir modelle hizmet veren bir BES altyapısına entegre edilebilecek bir algoritma (Analogy) geliştirilerek (Analogy algoritması için 09.07.2019 tarihinde 2019/10260 başvuru numarası ve “Analogy: Yoğunluk Tabanlı Bir Bilgi Erişim Optimizasyon Yöntemi” başlığıyla patent başvurusu da yapılmıştır), hem kullanıcıların değişken bilgi ihtiyaçlarına göre kendini güncelleyebilecek hem de hâlihazırda kullanılmakta olan bir BES altyapısını daha fazla kullanıcıyla bütünleştirebilecek bir ortamın sağlanmasına odaklanılmıştır. Bu doğrultuda, bilgiye erişimin etkili bir biçimde sağlanabilmesi amacıyla, kullanıcı dilinden bağımsız olarak çalışabilen bir yapay zekâ algoritması geliştirilmiş ve Kırklareli Üniversitesi (KLU) bünyesinde yer alan 170 tane web sitesinin içinde çalışan site-içi arama araçlarına entegre edilmiştir. Çalışmanın amacı, kullanıcı gerçek davranış verisi üzerinden bilgi erişim sürecinin iyileştirilmesi ve geliştirilen algoritmanın bu süreçteki başarımının değerlendirilmesi olmuştur.

Çalışma kapsamında ilk olarak, yapay zekâ olarak çalıştırılacak olan algoritma geliştirilmiş ve sonrasında bu algoritmanın KLU web sitelerindeki site-içi arama araçlarına entegrasyonu gerçekleştirilmiştir. Veri toplama sürecinin başlatılmasından önce iki haftalık bir pilot çalışmayla altyapısal eksikliklerin (web sunucusu tarafında aşırı yüklenmeye sebep olan arka plan işlemlerinin optimizasyonu, analiz sürecinde kullanılacak veri setindeki çeşitli veri tiplerinin daha sağlıklı ve tutarlı depolanmasına yönelik optimizasyon, kullanıcıların öneriyle karşılaştıkları arayüzün farklı türdeki cihazlarda oluşan görünüm hatalarının giderilmesi, vb.) giderilmesi sağlanmış ve bu süreci takiben toplam üç aylık bir periyotta kullanıcıların bilgi erişim süreçleri geliştirilmeye çalışılmıştır. Bu süreçte, tüm kullanıcı davranışları web sunucularda barındırılan günlük (log) dosyalarına benzer bir yapıyla anlık olarak MySQL veritabanında kayıt altına alınmış ve bu kayıtlar üzerinden Analogy algoritmasının öğrenme işlemi gerçekleştirmiştir. Tüm sürecin tamamlanması sonrasında, “Bilgi Erişim” ve “Kullanılabilirlik” literatürleriyle ilişkili olacak şekilde üç farklı açıdan (etkililik, verimlilik ve memnuniyet) algoritmanın bilgi erişim sürecindeki katkısı analiz edilmiştir.

Etkililik açısından ortaya çıkartılan bulgular, algoritmanın, hâlihazırda çalışan site-içi arama araçlarından daha başarılı bir şekilde performans ortaya koyduğunu göstermiştir. Reciprocal Rank (RR) metriği temel alınarak yapılmış olan bu değerlendirmelerde, 0,05 anlamlılık düzeyinde gerçekleştirilmiş olan işaret testi sonuçları, algoritmanın **daha başarılı** bir şekilde kullanıcılara hizmet ettiğini ortaya çıkarmıştır ($z=-2,421$; $p=0,015<0,05$). Algoritmanın

öğrenmesinde görev yapan ve 0-1 aralığında değer alabilen “Disiplin Eşik Değeri (*DED*)” parametresinin etkililiğe nasıl etki ettiğini tespit etmek amacıyla yapılmış olan analizlerdeyse, bu parametre ile etkililik arasında doğru orantı bulunduğu belirlenmiştir. Diğer bir ifadeyle, *DED* parametresindeki artış bilgi erişim sürecindeki etkililiği arttırmıştır. Buna rağmen, *DED* parametresinin sürekli 1’e yakın tutulmasının doğru olmayacağına ve bu parametrenin, bu çalışmadakine benzer şekilde kullanıcı davranışları üzerinden geliştirilecek olan bir yöntemle değiştirilmesi gerektiğine inanılmaktadır.

Verimlilik açısından ortaya çıkarılan bulgularda, “Sorgu Gecikmesi (*SOG*)”, “İlk Tıklama Süresi (*İTS*)” ve “Sorgu Terk Etme (*STE*)” olmak üzere üç farklı göstergeye odaklanılmıştır. *SOG*’a yönelik olarak, 0,05 anlamlılık düzeyinde gerçekleştirilmiş olan işaret testi sonuçları algoritmanın **daha hızlı bir performans** sergilediğini ortaya çıkarmıştır ($z=-38,453$; $p=0,000<0,05$). *İTS* için gerçekleştirilmiş olan işaret testi sonuçlarında, algoritma, 0,05 anlamlılık düzeyinde hâlihazırda kullanılmakta olan arama aracına kıyasla **daha hızlı bir şekilde** bilgi ihtiyacını karşılayabildiğini göstermiştir ($z=-4,936$; $p=0,000<0,05$). Ayrıca, *DED* parametresinin, Kruskal-Wallis Varyans analizi ile *İTS* ile ilişkisi incelenmiş fakat bir ilişki bulunamamıştır ($p=0,341>0,05$). Buna ek olarak, *İTS* ile önerilen kaynak miktarları arasındaki ilişkinin incelemesi yapılmış ve kaynak öneri miktarının artması durumunda *İTS*’nin de artma eğiliminde olduğu belirlenmiştir. Bu doğrultuda, kullanıcı davranışlarını özetleyebilme yeteneği olan algoritmanın, dolaylı olarak verimliliğe katkı sağladığı ifade edilebilir. *STE*’ye yönelik olarak yapılmış olan Tek Örneklem Ki-Kare Testi sayesinde, algoritmanın, site-içi arama araçlarından daha az sayıda kullanıcıyı kaybettiği belirlenmiş ($\chi^2= 38,407$; $p=0,000<0,05$) ve dolayısıyla, kullanıcıların aradığı bilgiye ulaşmalarında daha verimli çalıştığı ortaya çıkarılmıştır. Radlinski ve diğ. (2008) tarafından yapılmış olan çalışmadaki sonuca benzer şekilde, kullanıcı davranışları üzerinden öğrenebilen bir yapıyla hizmet veren bu çalışmadaki algoritmanın da, kullanıcıların sorgu terk etme davranışını azaltabildiği belirlenmiştir.

Kullanıcıların memnuniyetlerini belirleyebilmek amacıyla, herhangi bir arama davranışı sonrasında ziyaret ettikleri sayfaları değerlendirmeleri için 1-5 aralığında puan verilebilecek şekilde tek soruluk bir anket uygulanmıştır. Bu ankette, 1 değeri en düşük memnuniyeti temsil ederken, 5 değeri en yüksek memnuniyeti temsil etmiştir. Üç aylık süreçte, kullanıcıların oy verme konusunda çekingen davranışlarından dolayı veri miktarının yetersiz olduğu tespit

edilmiştir. Bu sebeple, algoritma başarısının memnuniyet bakımından ne durumda olduğunun belirlenebilmesine yönelik olarak, daha fazla çalışma yapılması gerektiği düşünülmektedir.

Bu çalışma kapsamında geliştirilmiş olan algoritmanın başarısını etkileyecek olan en önemli faktörlerden biri, entegre edilecek BES altyapısında kullanılan modeldir. Çünkü model, gereksinim duyulan bilgiyi sunma noktasında ne kadar başarılı olursa, algoritma da o kadar başarı yoğunluğunu arttıracaktır. Bununla birlikte, model ne kadar başarılı olursa olsun, kullanıcıların aradığı bilgiyi tam olarak belirleyemeyip ilgisiz dokümanları incelemesi de söz konusu olabilir. Bu durumda, algoritmanın kullanıcı kaynaklı yanlış önerme durumunun ortaya çıkması da söz konusu olabilir. Buna rağmen, bilgi erişimde en temel görevin BES'lere ve BES altyapısını geliştirebilecek algoritmalara düştüğü rahatlıkla söylenebilir. Bu sebeple, kullanıcıdan bağımsız olarak bilgi erişim sürecinin çeşitli yöntemlerle iyileştirilmesinin “kullanışlı bilgiye erişim” çerçevesinde en önemli hedef olması gerektiği düşünülmektedir.

Çalışmada eksik görülen bir diğer konu ise, algoritmanın kullanım oranlarının üç aylık sürede düşük olmasıdır. Bunun sebebi, kullanıcıların ilk kez bu çalışmadakine benzer bir yapıyla karşılaşmaları ve dolayısıyla, kullanıcıların aşinalıklarının zayıf olması olabilir. Buna ek olarak, öneri arayüzünün kullanıcı tarafından tam kavranamamış olması da bir olasılıktır. Bu sebeple, hem arayüz açısından hem de önerilerin anlaşılabilirliği açısından daha fazla çalışmanın gelecekte yapılması gerektiği düşünülmektedir.

Sonuç olarak, bilgi erişim sürecinin iyileştirilmesi amacıyla geliştirilmiş olan algoritmanın, etkililik ve verimlilik açısından başarılı olduğu rahatlıkla ifade edilebilir. Kullanıcı memnuniyeti açısından yeterli veri elde edilememesine rağmen, etkililik ve verimlilikte erişilen sonucun, memnuniyete de katkı sunduğu belirtilebilir. Bilgiye erişimin samanlıkta iğne aramaya dönüştüğü günümüzde, bu çalışmadakine benzer şekilde gelecekte yapılacak çalışmaların oldukça önemli olduğu ve bu çalışmada geliştirilen algoritmanın da bu süreçte önemli bir rol üstlenebileceği düşünülmektedir. Ek olarak, kullanıcı dilinden bağımsız olarak, kullanıcı davranışından öğrenerek aksiyon alan Analogy algoritmasının, kullanışlı bilgiye etkili ve verimli erişim amacıyla sadece arama motorlarında değil birçok alanda kullanılma potansiyeli olduğu düşünülmektedir.

KAYNAKLAR

- Aggarwal, C. C., 2016, *Recommender Systems: The Textbook*, 1st ed., Springer Science+Business Media, New York, ISBN: 978-3-319-29657-9.
- Akbari Torkestani, J., 2012, An adaptive learning to rank algorithm: Learning automata approach, *Decision Support Systems*, 54(1), 574–583.
- Akbulut, Y., 2010, *Sosyal bilimlerde SPSS uygulamaları: Sık kullanılan istatistiksel analizler ve açıklamalı SPSS çözümleri*, 1st ed., İdeal Kültür Yayıncılık, İstanbul, ISBN: 978-605-5729-05-9.
- Albishre, K., Li, Y., Xu, Y., & Huang, W., 2020, Query-based unsupervised learning for improving social media search, *World Wide Web*, 23(3), 1791–1809.
- Alonso, O., Rose, D. E. and Stewart, B., 2008, Crowdsourcing for relevance evaluation, *ACM SIGIR Forum*, 42(2), 9–15.
- Alpar, R., 2016, *Spor, Sağlık ve Eğitim Bilimlerinden Örneklerle Uygulamalı İstatistik ve Geçerlik – Güvenirlik*, 4th ed., Detay Yayıncılık, Ankara, ISBN: 978-605-5681-87-6.
- Alpaydin, E., 2014, *Introduction To Machine Learning*, 3rd ed., The MIT Press, Cambridge, ISBN: 978-0-262-02818-9.
- Alpaydin, E., 2016, *Machine Learning: The New AI*, 1st ed., The MIT Press, Cambridge, ISBN: 9780262529518.
- Baeza-Yates, R. and Ribeiro-Neto, B., 2011, *Modern Information Retrieval: The concepts and technology behind search*, 2nd ed., Addison-Wesley Publishing Company, Dorset, ISBN: 978-0-321-41691-9.
- Baxter, K., Courage, C. and Caine, K., 2015, *Understanding Your Users: A Practical Guide to User Research Methods*, 2nd ed., Morgan Kaufmann, Waltham, ISBN: 978-0-12-800232-2.
- Bhopale, A. P. and Tiwari, A., 2020, Swarm optimized cluster based framework for information retrieval, *Expert Systems with Applications*, 154, 113441 (sayfa).
- Bonaccorso, G., 2017, *Machine Learning Algorithms: A reference guide to popular algorithms for data science and machine learning*, 1st ed., Packt Publishing, Birmingham, ISBN 978-1-78588-962-2.
- Britannica Concise Encyclopedia, 2007a, *Information Retrieval*, Encyclopedia Britannica, Inc, Chicago, ISBN: 978-1-59339-293-2.
- Britannica Concise Encyclopedia, 2007b, *Artificial Intelligence*, Encyclopedia Britannica, Inc, Chicago, ISBN: 978-1-59339-293-2.

- Budak, V. Ö., 2016, *Kurumsal bir mobil web sitesinin kullanılabilirliğinin değerlendirilmesi: Kırklareli Üniversitesi örneği*, Yüksek Lisans, İstanbul Üniversitesi Fen Bilimleri Enstitüsü.
- Büttcher, S., Clarke, C. L. A. and Cormack, G. V., 2010, *Information Retrieval: Implementing and Evaluating Search Engines*, The MIT Press, Cambridge, ISBN: 978-0-262-02651-2.
- Ceri, S., Bozzon, A., Brambilla, M., Valle, E. D., Fraternali, P., & Quarteroni, S., 2013, *Web Information Retrieval*, Springer, Berlin, ISBN: 978-3-642-39313-6.
- Chapelle, O., Metzlzer, D., Zhang, Y., & Grinspan, P., 2009, Expected reciprocal rank for graded relevance, *Proceedings of the 18th ACM conference on Information and knowledge management*, 2-6 Kasım 2009 Hong Kong-China, New York, Association for Computing Machinery, ISBN: 978-1-60558-512-3, 621–630.
- Chifu, A.-G., Hristea, F., Mothe, J., & Popescu, M., 2015, Word sense discrimination in information retrieval: A spectral clustering-based approach, *Information Processing & Management*, 51(2), 16–31.
- Croft, B., Metzler, D. and Strohman, T., 2009, *Search Engines: Information Retrieval in Practice*, 1st ed., Pearson, Boston, ISBN: 9780136072249.
- Djenouri, Y., Belhadi, A., Fournier-Viger, P., & Lin, J. C.-W., 2018, Fast and effective cluster-based information retrieval using frequent closed itemsets, *Information Sciences*, 453, 154–167.
- Djenouri, Y., Belhadi, A. and Belkebir, R., 2018 Bees swarm optimization guided by data mining techniques for document information retrieval, *Expert Systems with Applications*, 94, 126–136.
- Dumas, J. S. and Loring, B. A., 2008, *Moderating Usability Tests: Principles and Practices for Interacting*, 1st ed., Morgan Kaufmann, Burlington, ISBN: 978-0-12-373933-9.
- Ertel, W., 2017, *Introduction to Artificial Intelligence*, 2nd ed., Springer International Publishing, Cham, ISBN: 978-0-12-373933-9.
- Flach, P., 2012, *Machine Learning: The Art and Science of Algorithms that Make Sense of Data*, 1st ed., Cambridge University Press, New York, ISBN: 978-1-107-09639-4.
- Göker, A. and Davies, J. (eds), 2009, *Information Retrieval: Searching in the 21st Century*, 1st ed., Wiley, Chichester, ISBN: 978-0-470-02762-2.
- Grossman, D. A. and Frieder, O., 2004, *Information Retrieval: Algorithms and Heuristics*, 2nd ed., Springer, Dordrecht, ISBN: 978-1-4020-3004-8.
- Harman, D., 2011, *Information Retrieval Evaluation*, 1st ed., Morgan & Claypool Publishers, San Rafael, Calif., ISBN: 978-1-59829-971-7.
- Hofmann, K., Li, L. and Radlinski, F., 2016, Online Evaluation for Information Retrieval, *Foundations and Trends in Information Retrieval*, 10(1), 1–117.

- Howe, J., 2006, *The Rise of Crowdsourcing*, <https://www.wired.com/2006/06/crowds/>, [Ziyaret Tarihi: 19 Ekim 2020].
- Huang, Y.-F. and Hsu, C.-H., 2008, PubMed smarter: Query expansion with implicit words based on gene ontology, *Knowledge-Based Systems*, 21(8), 927–933.
- Hull, D., 1993, Using statistical testing in the evaluation of retrieval experiments, *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, 27 Haziran – 01 Temmuz 1993 Pittsburgh-USA, New York, Association for Computing Machinery, ISBN: 978-0-89791-605-9, 329–338.
- ISO, 1998, 9241-11:2018(en)- *Ergonomics of human-system interaction — Part 11: Usability: Definitions and concepts*, <https://www.iso.org/obp/ui/#iso:std:iso:9241:-11:ed-2:v1:en>, [Ziyaret Tarihi: 6 Temmuz 2020].
- Järvelin, K. and Kekäläinen, J., 2000, IR evaluation methods for retrieving highly relevant documents, *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, 24-28 Temmuz 2000 Athens-Greece, New York, Association for Computing Machinery, ISBN: 978-1-58113-226-7, 41–48.
- Järvelin, K. and Kekäläinen, J., 2002, Cumulated gain-based evaluation of IR techniques, *ACM Transactions on Information Systems*, 20(4), 422–446.
- Joachims, T., 2002, Optimizing search engines using clickthrough data, *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, 23-26 Temmuz 2002 Edmonton-Alberta-Canada, New York, Association for Computing Machinery, ISBN: 978-1-58113-567-1, 133–142.
- Joachims, T. and Radlinski, F., 2007, Search Engines that Learn from Implicit Feedback, *Computer*, 40(8), 34–40.
- Joshi, A. V., 2020, *Machine Learning and Artificial Intelligence*, Springer International Publishing, Cham. ISBN: 978-3-030-26621-9.
- Jung, S., Herlocker, J. L. and Webster, J., 2007, Click data as implicit relevance feedback in web search, *Information Processing & Management*, 43(3), 791–807.
- Kelly, D., 2007, Methods for Evaluating Interactive Information Retrieval Systems with Users, *Foundations and Trends® in Information Retrieval*, 3(1-2), 1–224.
- Khalifi, H., Elqadi, A. and Ghanou, Y., 2018, Support Vector Machines for a new Hybrid Information Retrieval System, *Procedia Computer Science*, 127, 139–145.
- Khennak, I., Drias, H., Kechid, A. and Moulai, H., 2019, *Clustering Algorithms for Query Expansion Based Information Retrieval*, Computational Collective Intelligence, In: Nguyen, N. T., Chbeir, R., Exposito, E., Aniorté, P. and Trawiński, B. (ed.), Chapter 23, Springer International Publishing, Cham, ISBN: 978-3-030-28373-5, 261–272.
- Kowalski, G., 2011, *Information Retrieval Architecture and Algorithms*, Springer US, Boston, ISBN: 978-1-4419-7715-1.

- Larson, R. R., 2011, *Information Retrieval Systems*, Understanding Information Retrieval Systems: Management, Types, and Standards, In: Bates, M. J. (ed.), Chapter 2, CRC Press, Taylor & Francis Group, Boca Raton., ISBN: 978-1-4398-9199-5, 15–30.
- Latha, K., 2018, *Experiment and Evaluation in Information Retrieval Models*, CRC Press, Taylor & Francis Group, Boca Raton, ISBN: 978-1-315-39262-2.
- Laudon, K. C. and Laudon, J. P., 2012, *Essentials of Management Information Systems*, 10th ed., Pearson, Boston, ISBN: 978-0-13-266855-2.
- Liu, T.-Y., 2009, Learning to Rank for Information Retrieval, *Foundations and Trends in Information Retrieval*, 3(3), 225–331.
- Lucci, S. and Kopec, D., 2015, *Artificial Intelligence in the 21st Century*, 2nd ed., Mercury Learning & Information, Dulles, Virginia, ISBN: 978-1-942270-00-3.
- Malizia, A., Olsen, K. A., Turchi, T., & Crescenzi, P., 2017, An ant-colony based approach for real-time implicit collaborative information seeking, *Information Processing & Management*, 53(3), 608–623.
- Manning, C. D., Raghavan, P. and Schütze, H., 2008, *Introduction to Information Retrieval*, Cambridge University Press, New York, ISBN: 978-0-521-86571-5.
- Meadow, C. T. *et al.*, 2007, *Text Information Retrieval Systems*, 3rd ed., Elsevier, Amsterdam, ISBN: 978-0-12-369412-6.
- Melucci, M., 2015, *Introduction to Information Retrieval and Quantum Mechanics*, Springer Berlin Heidelberg, New York, ISBN: 978-3-662-48312-1.
- Metzler, D., 2011, *A Feature-Centric View of Information Retrieval*, Springer, Heidelberg, New York, ISBN: 978-3-642-22897-1.
- Moffat, A. and Zobel, J., 2008, Rank-biased precision for measurement of retrieval effectiveness, *ACM Transactions on Information Systems*, 27(1), 2:1–2:27.
- Mourão, A., Martins, F. and Magalhães, J., 2015, Multimodal medical information retrieval with unsupervised rank fusion, *Computerized Medical Imaging and Graphics*, 39, 35–45.
- Mueller, J. and Massaron, L., 2018, *Artificial Intelligence For Dummies*, 1st ed., John Wiley & Sons, Inc., Hoboken, New Jersey, ISBN: 978-1-119-46765-6.
- Mueller, J. P. and Massaron, L., 2016, *Machine Learning For Dummies*, John Wiley & Sons, Inc., Hoboken, New Jersey, ISBN: 978-1-119-24551-3.
- Nasir, J. A., Varlamis, I. and Ishfaq, S., 2019, A knowledge-based semantic framework for query expansion, *Information Processing & Management*, 56(5), 1605–1617.
- Neapolitan, R. E. and Jiang, X., 2018, *Artificial Intelligence: With an Introduction to Machine Learning*, 2nd ed., CRC Press, Taylor & Francis Group, Boca Raton, ISBN: 978-1-138-50238-3.

- Nielsen, J., 1993, *Usability Engineering*, 1st ed., Morgan Kaufmann, Amsterdam, ISBN: 978-0-12-518406-9.
- Nordlie, R., 1999, User revealment - a comparison of initial queries and ensuing question development in online searching and in human reference interactions, *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, 15-19 Ağustos 1999 Berkeley-USA, New York, Association for Computing Machinery, ISBN: 978-1-58113-096-6, 11–18.
- Orhunbilge, N., 2000, *Tanımsal İstatistik Olasılık ve Olasılık Dağılımları*, 1st ed., İstanbul Üniversitesi, İstanbul, ISBN: 978-975-404-548-2.
- Peters, C., Braschler, M. and Clough, P., 2012, *Multilingual Information Retrieval: From Research To Practice*, Springer, Heidelberg, New York, ISBN: 978-3-642-23007-3.
- Poole, D. L. and Mackworth, A. K., 2017, *Artificial Intelligence: Foundations of Computational Agents*, 2nd ed., Cambridge University Press, New York, ISBN: 978-1-107-19539-4.
- Radlinski, F., Kleinberg, R. and Joachims, T., 2008, Learning diverse rankings with multi-armed bandits, *Proceedings of the 25th international conference on Machine learning*, 5-9 Temmuz 2008 Helsinki-Finland, New York, Association for Computing Machinery, ISBN: 978-1-60558-205-4, 784–791.
- Radlinski, F., Kurup, M. and Joachims, T., 2008, How does clickthrough data reflect retrieval quality?, *Proceedings of the 17th ACM conference on Information and knowledge management*, 26-30 Ekim 2008 Napa Valley-California, New York, Association for Computing Machinery, ISBN: 978-1-59593-991-3, 43–52.
- Van Rijsbergen, C. J., 1979, *Information Retrieval*, 2nd ed., Butterworths, Boston, ISBN: 978-0-408-70929-3.
- Ritter, F. E., Baxter, G. D. and Churchill, E. F., 2014, *Foundations for Designing User-Centered Systems: What System Designers Need to Know about People*, Springer, London, ISBN: 978-1-4471-5133-3.
- Robertson, S., 2006, On GMAP: and other transformations, *Proceedings of the 15th ACM international conference on Information and knowledge management*, 6-11 Kasım 2006 Arlington-USA, New York, Association for Computing Machinery, ISBN: 978-1-59593-433-8, 78–83.
- Rosenzweig, E., 2015, *Successful User Experience: Strategies and Roadmaps*, 1st ed., Morgan Kaufmann, Waltham, Massachusetts, ISBN: 978-0-12-800985-7.
- Rowley, J., 2007, The wisdom hierarchy: representations of the DIKW hierarchy, *Journal of Information Science*, 33(2), 163–180.
- Rubin, J. and Chisnell, D., 2008, *Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests*, 2nd ed., Wiley, Indianapolis, ISBN: 978-0-470-18548-3.

- Ryan, J., 2010, *A History of the Internet and the Digital Future*, 1st ed., Reaktion Books, London, ISBN: 978-1-86189-777-0.
- Salton, G., Wong, A. and Yang, C. S., 1975, A vector space model for automatic indexing, *Communications of the ACM*, 18(11), 613–620.
- Schuchmann, S., 2019, *History of the first AI Winter*, Medium, <https://towardsdatascience.com/history-of-the-first-ai-winter-6f8c2186f80b>, [Ziyaret Tarihi: 30 Ekim 2020].
- Searle, J. R., 1980, Minds, brains, and programs, *Behavioral and Brain Sciences*, 3(3), 417–424.
- Tombros, A., Villa, R. and Van Rijsbergen, C. J., 2002, The effectiveness of query-specific hierarchic clustering in information retrieval, *Information Processing & Management*, 38(4), 559–582.
- Tukey, J., 1977, *Exploratory Data Analysis*, 1st ed., Reading, Mass, Pearson, ISBN: 978-0-201-07616-5.
- Turing, A. M., 1950, Computing machinery and intelligence, *Mind*, LIX(236), 433–460.
- Türk Dil Kurumu, 2020a, *Bilgi nedir?*, <https://sozluk.gov.tr/?kelime=bilgi>, [Ziyaret Tarihi: 15 Nisan 2020].
- Türk Dil Kurumu, 2020b, *Bilgi Erişim nedir?*, [https://sozluk.gov.tr/?kelime=bilgi erişim](https://sozluk.gov.tr/?kelime=bilgi%20erişim), [Ziyaret Tarihi: 30 Eylül 2020].
- Voorhees, E. M., 2004, Overview of the TREC 2004 Robust Retrieval Track, *Proceedings of the Thirteenth Text REtrieval Conference*, 16-19 Kasım 2004 Gaithersburg-Maryland-USA, NIST, 10 (sayfa).
- Wang, H. and Song, M., 2011, Ckmeans.1d.dp: Optimal k-means Clustering in One Dimension by Dynamic Programming, *The R journal*, 3(2), 29–33.
- Wang, J., Pan, M., He, T., Huang, X., Wang, X., & Tu, X., 2020, A Pseudo-relevance feedback framework combining relevance matching and semantic matching for information retrieval, *Information Processing & Management*, 57(6), 102342 (sayfa).
- White, R. W., Jose, J. M. and Ruthven, I., 2006, An implicit feedback approach for interactive information retrieval, *Information Processing & Management*, 42(1), 166–190.
- Yang, G. H., Sloan, M. and Wang, J., 2016, *Dynamic Information Retrieval Modeling*, Morgan & Claypool Publishers, San Rafael, California, ISBN: 978-1-62705-525-3.
- Zhai, C. and Massung, S., 2016, *Text Data Management and Analysis: A Practical Introduction to Information Retrieval and Text Mining*, ACM Books, New York, ISBN: 978-1-970001-16-7.

EKLER**EK 1. Veri Kullanım İzin Dilekçesi.**

11.01.2019

TC.
Kırklareli Üniversitesi Rektörlüğü
Bilgi İşlem Daire Başkanlığına

Kırklareli Üniversitesi Bilgi İşlem Daire Başkanlığında Öğretim Görevlisi olarak görev yapmaktayım. "Bilgiye Erişimde Kullanılabilirliğin Yeni Bir Yapay Zekâ Yöntemiyle Geliştirilmesi" başlığıyla İstanbul Üniversitesinde yürütmekte olduğum doktora tez çalışmam ve bu tez çalışmamdan gelecekte çıkarmayı planladığım diğer yayın çalışmalarım için Kırklareli Üniversitesi web günlük dosyası (web access log) verisinin tarafımca kullanılmasına yönelik gerekli iznin verilmesini istiyorum.

Gereğini arz ederim.

Öğretim Görevlisi

Veli Özcan Budak



EK 2. Veri Kullanım İzin Dilekçesi Cevabı.



**T.C.
KIRKLARELİ ÜNİVERSİTESİ REKTÖRLÜĞÜ
Bilgi İşlem Daire Başkanlığı**

Sayı : 28940837-705.01-E.984
Konu : Kırklareli Üniversitesi Web Günlük
Dosyası

15/01/2019

Sayın Öğr. Gör. Veli Özcan BUDAK

İlgili dilekçenizde bahsi geçen Kırklareli Üniversitesi web günlük dosyası (web access log) verisinin tarafınıza kullanılarak bir mahsur görülmemiş olup, veri kullanımı için izin verilmiştir. Gereğini bilgilerinize rica ederim.

e-imzalıdır

Öğr. Gör. Gökhan DOĞAN
Bilgi İşlem Daire Başkan Vekili

Adres: Kırklareli Üniversitesi Rektörlüğü Kayalı Kampüsü /KIRKLARELİ
Telefon: 0 288 2129681 Faks: 0 288 2129679
e-posta: bidb@klu.edu.tr Elektronik Ağ: <http://www.klu.edu.tr/>

Barış ULUŞAN
Dahili: 1105

5070 sayılı Elektronik İmza Kanunu'na uygun olarak Güvenli Elektronik İmza ile üretilmiştir.
Evrak teyidi <https://ebys.klu.edu.tr/sorgu/sorgula.aspx> adresinden E667-TN3Z-8717 kodu ile yapılabilir.

ÖZGEÇMİŞ

Kişisel Bilgiler	
Adı Soyadı	Veli Özcan Budak
Doğum Yeri	Gaziantep
Doğum Tarihi	06.10.1988
Uyruğu	☒ T.C. ☐ Diğer:
Telefon	0 507 794 76 06
E-Posta Adresi	veliozcanbudak@gmail.com
Web Adresi	http://personel.klu.edu.tr/veliozcanbudak



Eğitim Bilgileri	
Lisans	
Üniversite	Çanakkale Onsekiz Mart Üniversitesi
Fakülte	Eğitim Fakültesi
Bölümü	Bilgisayar ve Öğretim Teknolojileri Eğitimi
Mezuniyet Yılı	2009

Yüksek Lisans	
Üniversite	İstanbul Üniversitesi
Enstitü Adı	Fen Bilimleri Enstitüsü
Anabilim Dalı	Enformatik Anabilim Dalı
Programı	Enformatik Programı

Doktora	
Üniversite	İstanbul Üniversitesi
Enstitü Adı	Fen Bilimleri Enstitüsü
Anabilim Dalı	Enformatik Anabilim Dalı
Programı	Enformatik Programı

Makale, Kitap Bölümleri ve Bildiriler	
Makaleler Budak, V. Ö., Selçukcan Erol, Ç., 2020, Web Kullanıcılarının Bilgi Erişim ve Ziyaret Desenlerinin Web Madenciliği ile Keşfi: Kırklareli Üniversitesi Örneği, <i>AJIT-e: Bilişim Teknolojileri Online Dergisi</i> , 11(43), 37-55. Budak, V. Ö., Akadal, E., & Gülseçen, S. (2020). Research on user behaviors and tolerance of faulty web interactions. <i>Acta Informatica Pragensia</i> , 9(2), 108–131. Emre, İ. E., Selçuk, M., Budak, V. Ö., Bütün, M., & Şimşek, İ. (2019). Eğitim Amaçlı Sanal Gerçeklik Uygulamalarında Kullanılan Cihazların Daldırma Açısından İncelenmesi. <i>Bilişim Teknolojileri Dergisi</i> , 12(2), 119–129.	

- Bütün, M., Budak, V. Ö., Selçuk, M., Emre, İ. E., & Şimşek, İ. (2019). Eğitimde Sanal Gerçeklik Uygulamalarında Erişebilirlik ve Uyumluluk. *Eğitim Teknolojisi Kuram ve Uygulama*, 9(1), 251–275.
- Budak, V. Ö., Kartal, E., & Gülseçen, S. (2018). Site içi Aramalar ve Apriori Algoritması Kullanılarak Web Sitesi Ziyaretçilerinin İhtiyaç Tespitine Yönelik Bir Örnek Olay İncelemesi. *Bilişim Teknolojileri Dergisi*, 11(2), 211–222.
- Budak, V. Ö., Erol, Ç., & Gezer, M. (2017). Kurumsal Bir Mobil Web Sitesinin Kullanılabilirliğinin Geliştirilmesi. *Electronic Journal of Vocational Colleges*, 7(3), 15–26.
- Budak, V. Ö., & Gezer, M. (2016). Farklı Ekran Çözünürlükleri İçin Esnek Web Arayüz Yapıları Tasarlanması. *Electronic Journal Of Vocational Colleges*, 6(1), 10–24.
- Budak, V. Ö., Gezer, M., & Erol, Ç. (2016). The power of mobile web Kırklareli University case. *İnternet Uygulamaları ve Yönetimi Dergisi*, 7(2), 39–50.

Kitap Bölümleri

- Budak V. Ö., Aydın Çakırel Ö., Erol Ç. (2020), *Web Sitelerinde Kullanıcıların Bilgiye Erişim Davranışlarındaki Etkileşimlerinin İncelenmesi*, İnsan Bilgisayar Etkileşimi: Araştırma ve Uygulamalar: Gülseçen, S. Rızvanoğlu, K., Tosun, N., Akadal, E. (ed.), İstanbul University Press, ISBN:978-605-07-0726-7, 59-80.
- Budak V. Ö., Gezer M., Erol Ç. (2018), *Web Sitelerinde Kullanıcı Deneyiminin Geliştirilmesine Yönelik Bir Eklenti Önerisi*, Yönetim Bilişim Sistemlerinde Gündem: Gülseçen S., Erol Ç., Ayvaz Reis Z., Gezer M. (ed), Çağlayan Kitap, Yayıncılık, Eğitim A. Ş., ISBN: 978-975-436-152-0, 23-37.

Bildiriler

- Budak, V. Ö., & Emre, İ. E. (2019). Web Sayfalarının Yüklenme Performansları Üzerine Bir İnceleme. Presented at the 13. İstanbul Bilişim Kongresi “Verinin Evrimi: Ticaretin Devrimi”, İstanbul.
- Budak, V. Ö., & Erol, Ç. (2018). A Case Study on Improving the Usability of Web Interfaces by Mouse Tracking Method. Presented at the 1st International Symposium on Graduate Research in Science Focus on Entrepreneurship and Innovation (ISGRS 2018), İstanbul.
- Budak, V. Ö., & Erol, Ç. (2018). Navigation Behavior Analysis of Users on A Distance Education Website KLUDEC Sample. Presented at the 7th International Future-Learning Conference On Innovations In Learning For The Future: Digital Transformation in Education, İstanbul.
- Aslan, B., Yıldırım, E., Budak, V. Ö., Doğan, G., Güner, E. S., & Yavuzer Aslan, F. (2018). Mobil Uygulama Geliştirme Süreci Analizi Kırklareli Üniversitesi Örneği. Presented at the International Congress on Entrepreneurship, Technology, Innovation and Design 2018, Kırklareli.

- Bütün, M., Selçuk, M., Budak, V. Ö., Emre, İ. E., & Şimşek, İ. (2018). Eğitim Amaçlı Sanal Gerçeklik Uygulamalarında Kullanılan Cihazların Daldırma Açısından İncelenmesi. Presented at the 4.Uluslararası Eğitim Bilimleri Sempozyumu, Alanya.
- Emre, İ. E., Bütün, M., Budak, V. Ö., Selçuk, M., & Şimşek, İ. (2018). Eğitimde Sanal Gerçeklik Uygulamalarının Geliştirilen Yazılımlar ve Platformlar Açısından İncelenmesi. Presented at the 27. Uluslararası Eğitim Bilimleri Kongresi, Antalya.
- Budak, V. Ö., & Doğan, G. (2017). Kurumsal E Posta İşlemleri İçin Pratik Bir Çözüm. Presented at the 22. Türkiye de İnternet Konferansı, İstanbul.
- Budak, V. Ö., & Beyazgül, G. (2017). Web Uygulamalarda Cihaz Ekran Çözünürlüklerinin Kullanıcı Deneyimine Etkisi. Presented at the 22. Türkiye de İnternet Konferansı, İstanbul.
- Budak, V. Ö., Erol, Ç., & Gezer, M. (2017). An Add On Suggestion for Improving User Experience on Websites. Presented at the 4th International Management Information Systems Conference 2017, İstanbul.
- Budak, V. Ö., & Gezer, M. (2016). Esnek Web Ara Yüzü Tasarlanması Üzerine Bootstrap İle Bir Uygulama. Presented at the 18. Akademik Bilisim Konferansı, Aydın.
- Budak, V. Ö., Doğan, G., Deniz, G., Aslan, B., & Yavuzer Aslan, F. (2015). Kırklareli Üniversitesinin BYS Planlama Tasarlama ve Uygulama Süreci. Presented at the 20. Türkiye de İnternet Konferansı, İstanbul.