Büyük Veri Yaklaşımıyla Birden Çok Bilgi Erişim Merkezinin Kolektif Kullanımı

Hidayet TAKCI[1], Nuriye BAKTIR[2]

[1]Bilgisayar Mühendisliği, Cumhuriyet Üniversitesi, Sivas, Türkiye
[2]Yönetim Bilişim Sistemleri, Cumhuriyet Üniversitesi, Sivas, Türkiye
htakci@cumhuriyet.edu.tr, nuriyebaktir@gmail.com
(Geliş/Received: 04.07.2017; Kabul/Accepted: 23.02.2018)
DOI: 10.17671/gazibtd.324869

***Özet*—**Gelişen bilgisayar sistemleri üretilen veri miktarını artırırken değerli veriye erişimi daha zorlu hale getirmiştir. Örneğin, veri erişimde klasik katalog taramanın modası geçmek üzeredir. Mevcut durumun doğası gereği hem veri kaynakları bir araya getirilmeli hem de bu veriler üzerinde performansı yüksek yöntemlerle bilgiye erişim sağlanmalıdır. Veri kaynaklarını bir araya getirme için çözüm önerimiz bilgi erişim merkezlerinin ortak bir çatı altında toplanmasıdır. Yüksek performanslı tarama için önerdiğimiz yöntem ise bibliyografik verilerin özetlenmesi ve dağıtık yapıda paylaşılmasıdır. Bugüne kadar teknik nedenlerle mümkün olmayan böylesi bir yaklaşım büyük veri yardımıyla mümkün olabilecektir. Çalışmamız; bilgi kaynaklarının özetlendiği ve paylaşıldığı bir mimari önerir. Bu yaklaşıma göre her bir bilgi erişim merkezinin bibliyografik verisi Hadoop mimarisinde dağıtık olarak bir veri düğümü ile bütün bilgi erişim merkezlerinin özet verisini tutan ana merkez ise bir isim düğümü ile eşleştirilecektir. Veri düğümlerinde bibliyografik veri, isim düğümünde ise bilgi erişim merkezi bilgileri ve karakter n-gramlara dayalı özetler yer alacaktır. Sistemden yararlanmak isteyen bir kullanıcı önce isim düğümü üzerinde sorgulamasını yapacak, sorgusu ile en iyi eşleşen veri düğümünü bulacak ve daha sonra veri düğümü üzerinde detay sorgusunu yapabilecektir. Bu çalışma kapsamında kişilerin eskiden beri kullandığı bilgi erişim yöntemleri büyük veri yaklaşımıyla modernize edilmiş olup ortaya bir öneri konmuştur.

Anahtar kelimeler **—**Büyük veri, Hadoop mimari, MapRaduce, Apache Lucene, N-Gram

Collective Use of Multiple Information Access Centers with Big Data Approach

Abstract **—**Developing computer systems have increased the amount of data generated and accessing valuable data has become more difficult. For example, the fashion of the classic catalog search for information retrieval is about to pass. The nature of the current situation requires both data sources should be collected together and access to information by means of high performance on these data. Our proposal for collecting data sources is to bring information access centers together under a data center. The method we recommend for high-performance screening is to summarize and distribute bibliographic data. This approach, which is not possible for technical reasons up to now, would be able to possible with the aid of large data. Our study; suggests an architecture in which the sources of information are summarized and shared. According to this approach, the bibliographic data of each information access center will be mapped to a name node distributed in a Hadoop architecture with a data node and the main center holding the summary data of all information access centers. Bibliographic data in the data nodes will be summarized based on the information access center information and character n-grams in the name node. A user who wishes to utilize the system will first find the data node that best matches the query to query on the name node, and then it will be able to query the data node for the detail query. In this study, information access methods that people used to date have been modernized with a large data approach and a proposal has been made.

Keywords **—**Big data, Hadoop architecture, MapRaduce, Apache Lucene, N-Gram

1. GİRİŞ (INTRODUCTION)

Olguları, gerçekleri veya ölçüm sonuçlarını sunan, sayısal veya mantıksal işaretler topluluğuna veri denir. Verinin değeri ondan elde edilen anlamla ilgilidir. Veriden anlamlı bilginin elde edilmesi veya başka bir ifadeyle veriden değer oluşturma tarihsel gelişim içerisinde farklı aşamalardan geçerek mümkün hale gelmiştir. Sürecin en başında üretilen verilerin kaydedilmesi bulunur. Özellikle yedekleme ünitelerinin yeteneklerinin artması ve maliyetlerinin düşmesi bu adımı desteklemiştir. İkinci aşama kayıt altına alınan verilerin çeşitli arama yöntemleriyle açığa çıkarılması şeklinde kendini göstermiştir. Bu aşama dosyalama sistemleri üzerinde aramaların yapıldığı dönem olarak görülebilir. Üçüncü aşama veri tabanı sistemlerinin gelişimi ve SQL sorguları yardımıyla değerli bilgilerin elde edilmesidir. SQL'den OLAP analizlerine ve oradan veri madenciliğine giden aşama dördüncü aşama ve şu anda şahit olduğumuz büyük veri ve bulut bilişim ise beşinci aşama olarak görülebilir.

Büyük veri geleneksel veri depolama alanları ile veri işleme teknolojilerinin yetersiz kaldığı durumu ifade etmek için kullanılan bir terimdir. Büyük veri kavramı sadece veri ve onun saklandığı ortamı değil aynı zamanda veriden değer çıkarmaya odaklanmış ileri seviye analiz tekniklerini de kapsar. Büyük veri analizi; iş zekâsı uygulamaları, tıbbi analiz, askeri suçlar ve buna benzer birçok alanda kullanılmaktadır. Sadece bilim adamlarının değil aynı zamanda işletmelerin, tıp çalışanlarının ve diğerlerinin de ilgi odağı durumundadır [1], [2]. Büyük verinin bu denli ilgi odağı olmasının nedeni hemen her sektörün ihtiyacı olan verilerin toplanması ve o veriler üzerinde analitik işlemler yapılmasına yardımcı olmasıdır. Akıllı cihazlar, uzaktan algılayıcılar, kameralar, mikrofonlar ve kablosuz algılayıcı ağlar büyük veri için veri kaynaklarından sadece bir kaç tanesidir [3].

Büyük veri yaklaşımı düşünme ve iş yapış şekillerimizi modernize etmiştir. Daha önceden çözülememiş problemler çözülmeye başlamış ayrıca önceden çözülmüş problemlerin daha akılcı çözümleri elde edilebilmiştir. Bu konulardan birisi de bilgi erişim merkezlerinin yeniden yapılanması olacaktır. Ülkemizde ve dünyada çok sayıda bilgi erişim merkezi kendine ait basılı ve elektronik dokümana sahiptir. Basılı dokümanların bibliyografik verileri, elektronik dokümanların ise hem üst verileri hem de içerik verileri sayısal olarak tutulmaktadır. Bilgi erişim merkezlerine ait kaynakların bir birlik mantığında bir araya getirilmesi 2016 yılında tarafımızdan yapılan bir çalışmada önerilmiştir [4]. Bu öneri katmanlı bir yapıda bilgi erişim merkezlerinin nasıl yönetilebileceği ile ilgilidir. Yöntem olarak; her bir kütüphanenin kataloğu özetlenmiş ve birlik içerisinde bulunan kütüphaneler bu özetler yardımıyla daha hızlı şekilde kayıtlara erişebilmiştir. 2016 yılında yapılan ilgili çalışma bu çalışmada büyük veri imkânları ile yeniden ele alınmış ve modernize edilmiştir.

Bu çalışmada önerilen model; büyük veri olanakları ile bilgi erişimin yeniden organize edilmesidir. Bu kapsamda; Hadoop File System (HDFS) ve n-gram analizi kullanılarak birden çok bilgi erişim merkezinin kaynağının nasıl bir araya getirileceği ve ortak kullanımın nasıl sağlanacağı gösterilecektir. Önerilen model yardımıyla hem hiyerarşik yapı kurulacak hem de hızlı arama imkânı sağlanacaktır. HDFS yapısı isim düğümü adı verilen bir merkez ve veri düğümleri adı verilen alt düğümler şeklinde organize edilmektedir. Her bir veri düğümü önerdiğimiz modelde bir bilgi erişim merkezini, isim düğümü ise bilgi erişim merkezlerinin özet katalog verilerini tutacaktır. Düğümlerde bilgi erişim merkezlerinde yer alan kaynakların bibliyografik verilerine ait n-gram özetleri yer alacaktır. Üst veri ile eşleşme yapıldıktan sonra derinlemesine analizde Apache Lucene veya başka bir arama tekniği kullanılabilecektir.

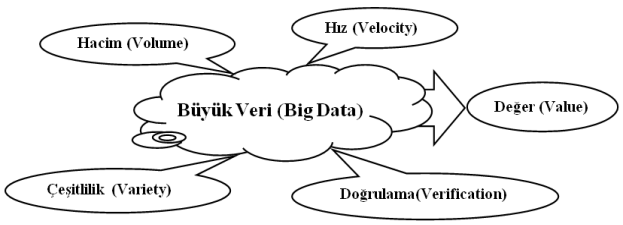
2. BÜYÜK VERİ (BIG DATA)

Büyük veri genel olarak kullanılan programların saklama, yönetme ve işleme kapasitesinin ötesindeki veri kümelerini anlatmak için kullanılan bir terimdir. Bu kavram ilk kez Ağustos 2000'de 8. Dünya Ekonometri Kongresi'nde ortaya atılmıştır [5]. Büyük veri kaynakları GPS, ağ konumları gibi mobil veriler, kullanıcı kimlik verileri, çevrimiçi erişim için kullanılan tek seferlik şifre (OTP), biyometrik kimlik saptama verileri, kimlik doğrulamada kullanılan dijital sertifikalar gibi verileridir [6]. Büyük verinin devasa boyutları ile bundan fayda sağlamak için gereken analizlerin karmaşıklığının birleşmesi, yeni sınıf teknolojilerin ve bunları yönetecek araçların gelişmesine neden olmuştur. Aslında büyük veri, hem yönetilen verinin türünü, hem de onu depolamak ve işlemek için kullanılan teknolojiyi anlatmaktadır. Bu teknolojilerin büyük bir kısmı, Google, Amazon, Facebook ve LinkedIn gibi şirketlerin inanılmaz büyüklükteki sosyal medya verileri ile uğraşırlarken, kendileri için geliştirdikleri teknolojiden doğmuştur. Bu şirketler, doğası gereği düşük maliyetli hazırda bulunan donanım ve açık kaynaklı yazılımlara önem vermektedirler [7].

İlk başlarda bilgi hacminin çok büyümesi durumunda, incelenen veri miktarının, bilgisayarların işlem için kullandıkları belleğe uygun olmadığı düşünülmekteydi. Mühendislerin verinin tümünü analiz etmek için kullandıkları araçları yenilemeleri gerekiyordu. Günümüzde ise Google MapReduce ve Hadoop gibi yeni işlem teknolojilerinin kaynağı olan bu araçlar, günümüzde eskisinden çok daha fazla verinin yönetilmesine olanak tanımaktadır. Bu teknolojilerde verinin düzenli sıralara ya da klâsik veri tabanı tablolarına yerleştirilmesi gerekmemektedir [8].

2.1. Büyük Veri Bileşenleri (Big Data Components)

Büyük veriyi anlamak için onu oluşturan beş bileşen doğru şekilde anlaşılmalıdır. Bu bileşenler; Şekil 1 de görüleceği üzere, kısaca 5V (Volume, Velocity, Variety, Verification, Value) olarak adlandırılır.



Şekil 1. Büyük veri 5V gösterimi (5V display for Big Data)

1.Hacim (Volume): Hacim, üretilen verinin yüksek boyutunu ifade etmekte kullanılır [9].

2.Hız (Velocity): Hız, büyük verinin çok hızlı şekilde üretildiğini ifade eder. Daha hızlı üretilen veri, o veriye muhtaç olan işlev sayısının ve çeşitliliğinin de aynı hızda artması sonucunu doğurmuştur [10].

3.Çeşitlilik (Variety): Çeşitlilik, toplanan verilerin kaynaklarının fazlalığına vurgu yapar. Büyük veri sistemleri farklı kaynaklardan beslendiği için heterojen yapıya sahiptir.

4.Doğrulama (Verification): Bilgi yoğunluğu içinde akan verinin "güvenli" kalması ile ilgili bir özelliktir.

5.Değer (Value): En önemli bileşen ise verinin bir değer yaratmasıdır. Karar verme süreçlerine anlık olarak etki etmesi ve doğru kararın zamanında verilmesi için hazır olması gerekir.

3. HADOOP MİMARİSİ (HADOOP ARCHITECTURE)

Büyük veri altyapısını anlayabilmek için dağıtık mimarinin anlaşılması elzemdir. Bu kapsamda karşımıza en sık çıkan mimari Hadoop mimarisidir. Hadoop, sıradan sunuculardan oluşan bir küme üzerinde büyük verileri işlemek amacıyla çalıştırılan ve HDFS olarak adlandırılan bir dağıtık dosya sistemi ile MapReduce özelliklerini bir araya getiren açık kaynak kodlu bir kütüphanedir. Dolayısıyla bilgi erişim merkezlerine ait n-gram özetleri HDFS sistemindeki sıradan sunucular üzerine dağıtılacaktır.

HDFS sayesinde sıradan sunucuların diskleri bir araya gelerek büyük, tek bir sanal disk oluştururlar. Bu sayede çok büyük boyutta birçok dosya bu sistemde saklanabilir. Bu dosyalar bloklar halinde birden fazla ve farklı sunucu üzerine dağıtılarak RAID benzeri bir yapıyla yedeklenir. Bu sayede veri kaybı önlenmiş olur. Ayrıca HDFS çok büyük boyutlu dosyalar üzerinde okuma işlemi imkânı sağlamaktadır. Ancak rasgele erişim özelliği bulunmaz. HDFS, NameNode ve DataNode işlem süreçlerinden oluşmaktadır [11]. Apache Hadoop Dağıtılmış Dosya Sistemi Java ile yazılmıştır ve farklı işletim sistemlerinde çalışabilir.

4. BİLGİ ERİŞİM MERKEZLERİ VE UYGULAMALAR (INFORMATION ACCESS CENTERS AND APPLICATIONS)

Bilgi erişim merkezleri üzerine uzun yıllardır çok sayıda çalışma yapılmıştır. Klasik dönem çalışmalarından modern dönem çalışmalarına kadar bir özet aşağıda verilmiştir.

4.1. Mevcut Uygulamalar (Current Studies)

Bilgi erişim merkezleri konusundaki araştırmalar temel olarak iki başlıkta ele alınmıştır. Bunlardan birincisi bilgi erişim merkezlerinin kullanımıyla ilgili çalışmalar diğeri ise bilgi erişim merkezlerinden bilgi almanın kolaylaştırılması üzerine çalışmalardır.

Bilgi erişimi kolaylaştıran çalışmalardan birinde Nicholson [12] yönlü aramayı (faceted search) desteklemek üzere bir yöntem geliştirmiş ve bu yöntemi; web üzerinde arama yapan öğrenciler için akademik makaleleri diğer makalelerden ayırt etmede kullanmıştır. Bu çalışmada veri madenciliği yöntemleri filtre mekanizması olarak uygulanmıştır.

Bir başka çalışmada kütüphane kullanıcıların bilgi erişim kayıtları incelenerek birlikte kitap alma örüntüleri tespit edilmiştir. Dwivedi ve Bajpai [13] birliktelik kuralları madenciliği yardımıyla bilgi erişimin bir boyutu olan birbirine benzeyen yayınları tespit etmiş ve bilgi erişim performansı için önemli bir çalışma yerine getirmişlerdir.

Takcı ve Soğukpınar [14] 2002 yılında kütüphane kullanıcılarının erişim örüntülerinin keşfi üzerine bir çalışma yapmış ve bu çalışmada web sunucu günlükleri ile kütüphane kullanıcılarının davranışlarını tespit edebilmişlerdir. Ayrıca aynı makalede bir puanlama yöntemi sayesinde kullanıcıları davranış tiplerine göre iki farklı kümeye ayırmışlardır. Takcı ve Soğukpınar tarafından yapılan bir başka çalışmada [15] ise yine web sunucu verileri üzerinde, bu sefer Statistica Data Miner uygulaması yardımıyla kümeleme yapılmış ve kütüphane kullanıcıları davranışlarına göre kümelere ayrılmıştır.

Mevcut uygulamalar daha çok klasik veri madenciliği yöntemlerinin kütüphanede kullanımı üzerinedir, yeni bir yöntem olarak metin madenciliği kütüphanelerde bilgi erişim için faydalı olacaktır. Çünkü metin madenciliği özellikle bilgi erişim konusunda faydalı teknikler içerir. Metin madenciliği; veri madenciliği, makine öğrenmesi, istatistik ve hesaplamalı dilbilim alanlarından yararlanan disiplinler arası bir alandır. Metin madenciliği kapsamında bugüne kadar; metin sınıflandırma, soru cevaplama, metin özetleme ve görselleştirme gibi işler yapılmıştır [16]. Metin madenciliği kullanılarak yapılan bir çalışmada kullanıcı tarafından belirtilen internet sitelerinin içeriği analiz edilerek site üzerinde elektronik ticaret yapılıp yapılmadığı tespit edilmiştir [17].

Klasik bilgi erişim sorgu kelimeleri ile bibliyografik verilerin ikili model gibi basit modellerle karşılaştırılmasını ifade ederken metin madenciliği

ilişkililik gibi parametrelere dayalı bilgi erişimi destekler. Klasik bilgi erişimde erişim kalitesini artırabilmek için, sorgu metinleri katalog kaydının başında, ortasında veya sonunda aranmıştır. Ayrıca birden fazla sorgu kelimesi girişine izin veren çoklu sorgular AND ve OR bağlaçlarıyla gerçeklenmiştir. Bunlara ek olarak doğal dil işleme tabanlı faceted search çalışmaları mümkündür [18]. Yapılan başka bir yönlü arama çalışmasında kütüphane katalogları ile araştırmacılar arasındaki etkileşimin geliştirilip geliştirilmediği araştırılmıştır [19]. Son dönemde kavramsal arama (concept search) konusu [20] gündemdedir. Kavramsal aramada kullanıcı sorgu olarak bir kavram sunmakta, karşılığında o kavramla ilişkili kayıtlar geri dönmektedir. Bütün bu çalışmalar bilgi erişim kalitesini artırıcı çalışmalar olup bu alandaki yeni çalışmalar devam etmektedir. Çalışmalardan görüleceği üzere metin madenciliği alanı özellikle de bilgiye erişim anlamında kütüphanecilerin hizmetindedir.

Bilgiye erişim performansını artırabilmenin yollarından birisi kayıtların indekslenmesidir. Bugüne kadar kavramsal dizin gibi yapılarla kütüphaneler kayıtlarını indekslemiş ve kullanıcılar kelime tabanlı olarak katalog veritabanlarına erişim sağlamıştır.

Kelime tabanlı bilgi erişime ek olarak bir diğer yöntem n-gram tabanlı bilgi erişimdir. N-gram yöntemi [21] metin madenciliğinde sıklıkla kullanılan bir yöntemdir. Bugüne kadar birçok metin madenciliği probleminde başarıyla kullanılmıştır [22], [23]. Ayrıca, n-gram dizileri sıklıkla birlikte geçen ikilileri yakalama ve semantik bilgi bulma anlamında da faydalı bir tekniktir. N-gram tabanlı bilgi erişim konusunda TELLTALE isimli bilgi erişim sistemi önemlidir [24]. Bu sistem karakter seviyeli n-gramları kullanır. Kelime tabanlı bilgi erişimin sorunlarına bir çözüm olarak geliştirilmiştir. N-gram tabanlı yaklaşım kelime ve harf seviyesinde çalışmaya uygundur. Hata toleransı, dilden bağımsız oluşu ve semantik bilgiyi yakalama becerisi dolayısıyla n-gram tabanlı bilgi erişim kelime tabanlı bilgi erişimden daha üstün özelliklere sahiptir [23]. Ayrıca metinlerin kelimelere dayalı olarak sunumu yoğun doğal dil işleme çabalarına ihtiyaç duyduğu için n-gramların kullanımı daha uygundur. N-gram yöntemi için bir dezavantaj özellik seti boyutunun n>3 iken büyük olmasıdır, bu problemin çözümü için genellikle özellik seçimi algoritmaları kullanılır. N-gram yöntemi sadece bir indeksleme yöntemi olmayıp veri özetleme amacıyla da kullanılabilir.

Son dönemde, özellikle de büyük veri yardımıyla yapılan bilgi erişim çalışmaları kapsamında bazı yayınlara rastlanmıştır. Bu çalışmalardan birinde Hadoop kullanılarak yararlı ve özet bilgiler elde edebilmek için günlük dosyaları analiz edilmiştir. Önerilen yaklaşım, belirli bir alan içerisinde yer alan farklı araştırma makalelerine uygulanarak bilgi erişim kalitesi artırılmaya çalışılmıştır [25].

5. N GRAM YÖNTEMİ (N GRAM METHOD)

N-gram algoritması, verilen bir dizideki alt ifadelere ait kombinasyonları elde ederek, bu kombinasyonların karşılaştırılacak olan dizi içerisindeki tekrar oranını bulmaya yarayan bir yöntemdir. Metin madenciliğinde sıkça kullanılan n-gram algoritması dizgi halindeki bir ifadeyi kullanıcı tarafından belirlenmiş olan N birim uzunluğunda kelime veya karakterlere ayırarak çalışmaya başlar. N değeri 1 iken algoritmadan elde edilen parçacıklara uni-gram, 2 ise bi-gram, 3 ise tri-gram vb. isimler verilir. Bir örnek ile açıklamak gerekir ise; *"Milli sınırlar içinde bulunan yurt parçaları bir bütündür; birbirinden ayrılamaz."* kelime grubunu ele alalım. Bu ifadenin 2-gram algoritmasının girdisi olarak kullanıldığını varsayarsak, alt ifadeler Tablo 1 de olduğu gibi sıralanabilir.

Tablo 1. Bi-gram ve frekansları (Bi-gram and frequencies)

Bi- Gram	N gram	Frekansı
	milli sınırlar	1
	sınırlar içinde	1
	içinde bulunan	1
	…	…
	bütündür birbirinden	1
	birbirinden ayrılamaz	1

N-gram yöntemi sayesinde bir metin içerisindeki N uzunluklu parçalar bulunduktan sonra yapılacak ikinci işlem onların sayılmasıdır. Metinde yer alan bütün n-gramlar elde edilip sıklıkları bulunduktan sonra elde edilen bilgi dokümanın n-gram profili olarak isimlendirilir ve kaydedilir. Dolayısıyla; bir kütüphanede yer alan kitapların katalog bilgileri de metin bilgisi olarak düşünüldüğünde katalog verisi için de n-gram özetleri bulmak mümkündür. Her bir kütüphane için n-gram profili bulunduktan sonra kullanıcının sunduğu sorgu teriminin de n-gram profili elde edilir ve profiller birbiriyle karşılaştırılır. Bu karşılaştırma Şekil 2 de sözde kodu verilen mantığa göre yerine getirilmektedir. Sözde kod her iki profilin de vektör uzayı modelinde sunulduğu varsayıma göre yazılmıştır.

```
x, benzerliği bulunacak kayıt için bir profil vektörü
cj, kütüphanede yer alan kayıtlardan biri için profil vektörü
m, her birinin özellik adedini vermek üzere;
pay=0;
x_boyut=0
cj_boyut=0
for i:=1 to m do
begin
pay=pay+xi*cji
x_boyut=x_boyut+xi2
cj_boyut=cj_boyut+cji2
end
benzerlik=pay/(x_boyut)1/2*(cj_boyut)1/2
"iki vektör arasındaki benzerlik=" benzerlik
```

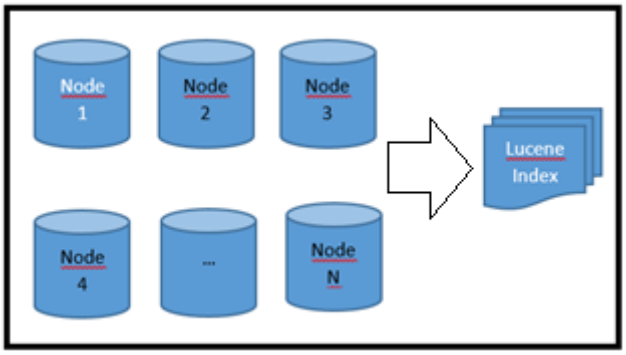
Şekil 2. N-gram tabanlı benzerlik (N-gram basedsimilarity)

N-gram algoritmasındaki önemli noktalardan birisi hangi seviyede çalışma yapılacağıdır. Bu çalışmada karakter seviyeli n-gramlar kullanılacaktır. İkinci husus N değerinin ne olacağıdır. Dilimizde en küçük kelime birimleri ortalama olarak 3 karakterden başladığı için uygun değer 3 olarak belirlenmiştir. Bu konuyla ilgili son husus ise profil karşılaştırmanın nasıl yapılacağıdır. Profil karşılaştırma için uygun yöntem sorgu teriminde yer alan n-gram değerleri ile katalog verilerinden elde edilen n-gram profillerinin karşılaştırılmasında en yüksek sıra değerini veren profilin seçimidir. Örneğin aradığımız ifade "data mining" ise ve 7 numaralı kütüphanede bu ifadeden türeyen "dat", "ata"… gibi değerler daha yüksek sırada ise 7 numaralı kütüphane aksi takdirde en iyi benzeşen kütüphane seçilecektir.

Çalışmamızda önemli konulardan birisi de dilden kaynaklanan problemlerin nasıl çözüleceğidir. Doğal dil işleme çalışmalarında sıklıkla dile ait problemler ve karmaşa ile çalışma ihtiyacı ortaya çıkmaktadır. Bununla birlikte n-gram istatistiksel bir yaklaşım olup dilden kaynaklanan problemler bu yöntemde yer almamaktadır. Ayrıca; n-gram yöntemi gürültülü verilerle çalışmaya uygundur. Bu nedenlerden dolayı dilden kaynaklanan bir zorluğun olmayacağı öngörülebilir.

6. BİLGİ ERİŞİM SİSTEMLERİ İÇİN BÜYÜK VERİYE DAYALI BİR ÇÖZÜM ÖNERİSİ (A BIG DATA BASED SOLUTION FOR INFORMATION ACCESS SYSTEMS)

Büyük veri çözümleri kapsamında bilgi erişim merkezlerinin kullanımıyla ilgili akla gelen ilk çözüm; bilgi erişim merkezlerinde yer alan dokümanların HDFS yardımıyla organize edilmesi ve organize edilen veriler üzerinde Apache Lucene yardımıyla indeksleme ve sorgulama yapılmasıdır. HDFS sistemi kullanılmasının en önemli nedeni üretilen verinin miktarı dolayısıyla veri işlemenin geleneksel yöntemlerle çözümünün zor hale gelmesidir. Hadoop mimarisinde, dağıtılmış veri üzerinde işlem yapabilmek ve sonuçları bir araya getirebilmek için MapReduce yöntemi ortaya konmuştur. MapReduce sayesinde analizler yapabilmek için uygun ortam sağlanmaktadır. MapReduce paralel işlem yeteneği ile verilerin hızlı bir şekilde analiz edilmesini sağlar. Apache Lucene ise dokümanlardan bilgi çıkarımında kullanılan açık kaynak kodlu bir yazılımdır. HDFS ile Apache Lucene arasındaki ilişki Şekil 3'de de görüleceği üzere; HDFS alt yapı, Lucene veri içeriği ve indeksle ilgili bilgi tutar.

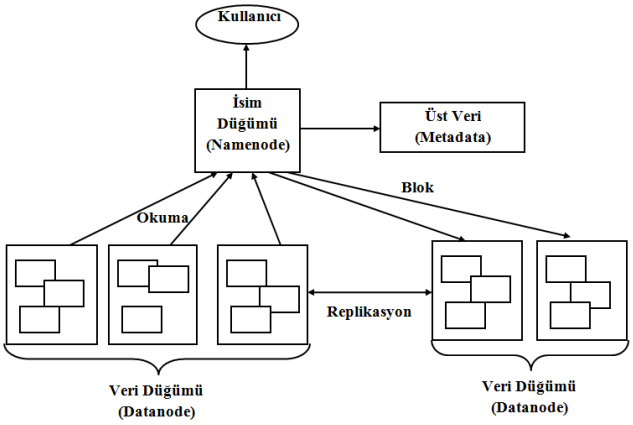


Şekil 3. HDFS ve Lucene (HDFS and Lucene)

Apache Lucene kabaca verileri indeksler ve veriler üzerinde aramalar yapar. İstenildiği kadar alan üzerinde indeksleme imkânı sağlar. Apache Lucene için çeşitli sürümler geliştirilmiştir. Bunlar; Apache Lucene Java, ApacheSolr, Apache Lucene.Net, ApachePyLucene, Apache Open Relevance Project, ApacheLucy ve ApacheDroids adıyla bilinir. Apache Lucene temel olarak indeksleme ve arama görevlerini destekler.

5.1. Önerilen Mimari (Suggested Architecture)

Bu çalışmada, büyük veri yetenekleri ile birden çok kütüphanenin kaynaklarına eş zamanlı olarak erişen ve böylece daha kaliteli sonuç elde etme imkânı sağlayan bir model ortaya konmuştur. Bu işlemi gerçekleştirebilmek için Hadoop mimarisinden faydalanılmış ve bu sayede çok sayıdaki kaynağa hızlı erişim ve sonuç birleştirme imkânı elde edilebilmiştir. Önerilen dağıtık yapıda her bir kütüphane bir düğüm olarak planlanmış ve böylece eş zamanlı sorgu imkânı sağlanabilmiştir.



Şekil 4. Önerilen yapının çalışma şematiği (Suggested work workings chematic)

Şekil 4'deki modele göre; kütüphane katalog bilgileri veri düğümleri (datanode) üzerinde bloklar halinde tutulur. Veri düğümleri arasında replikasyon işlemleri yapıldığı için her bir veri düğümünde verilerin tamamı saklanır. Böylece, herhangi bir veri düğümüne zarar geldiği zaman veri kaybının önüne geçilmiş olur.

Bu sistemden bilgi almak isteyen kullanıcılar, sorgu terimlerini önerilen dağıtık yapının arayüzü kanalıyla sisteme girerler. Daha sonra sorgu terimleri isim düğümleri (namenode) üzerinde aranır. İsim düğümleri üzerinde tutulan, veri düğümleri üzerinden elde edilmiş yayınlara ait bibliyografik veriler üst veri formatındadır (metadata). Özet veriler sayesinde veriye hızlı bir şekilde erişim sağlanır ve kullanıcı sorgusuna karşılık olarak ilişkili sonuçlar döndürülür.

Apache Lucene yardımıyla yapabileceğimiz indeksleme ve sorgulama için bu çalışmada n-gram profillerine göre indeksleme ve n-gram profillerinin saklanması planlanmaktadır. Önerdiğimiz yapıda kütüphanelere ait katalog verileri n-gram özetleri şeklinde isim düğümü üzerinde tutulacaktır. İsim düğümü birden çok kütüphaneye ait üst verileri barındıracaktır. Bu üst verilerin formatı kütüphanelerde yer alan bibliyografik verilerin n-gram özeti şeklinde olacaktır. Kurulan yapı sayesinde sisteme gönderilen sorguda n-gram yöntemiyle özetlenecektir. Daha sonra kullanıcı sorgusu ve isim düğümündeki n-gram özetleri karşılaştırılacak ve en ilişkili olan sonuç kullanıcıya gönderilecektir. Bunun sonucunda kullanıcı detay sorgu için ilgili düğüme (kütüphaneye) yönlendirilecektir.

MapReduce dağıtık mimari üzerinde çok büyük verilerin kolay bir şekilde analiz edilebilmesini sağlayan bir sistemdir. Veriler işlenirken map ve reduce olarak temel iki fonksiyon kullanılmaktadır. Map aşamasında ana düğüm verileri alıp daha ufak parçalara ayırarak işçi düğümlere dağıtırken işçi düğümler işleri tamamladıkça sonucunu ana düğüme geri göndermektedir. Reduce aşamasında ise tamamlanan işler birleştirilerek sonuç elde edilmektedir. Map aşamasında analiz edilen veri içerisinden almak istediğimiz veriler çekilmekte, Reduce aşamasında ise bu çektiğimiz veri üzerinde istediğimiz analiz gerçekleşmektedir [26]. Kullanıcı sorguları ile isim düğümündeki n-gram profillerinin ilişkilendirilmesi ve ilişkisi yüksek olanların seçimi hadoop mimarisindeki map yöntemi ile gerçekleştirilecektir. İlişkinin yüksekliği hesabı ise reduce yöntemi ile yapılacaktır.

Kullanıcının sorgulama yapacağı, isim düğümü üzerinde yer alan n-gram profili veri yapısı;

MerkezID	nGram	Seviye	Frekans	Sıra

Şekil 5. Veri tabanı gösterimi (Database display)

Şekil 5 de gösterilen veri yapısında;

MerkezID: konsorsiyumla bir anlaşması olan her bir kütüphaneye, merkez tarafından verilen, tekil, tanıtıcı bir anahtar değer olacaktır.

nGram: ilgili kütüphanenin bibliyografik verileri arasında sıklıkla rastlanan kelime n-gram dizisini sunar. Örneğin bu değer "digital library" gibi bir 2-gram dizisi olabilir.

Seviye: nGram alanında yer alan n-gram dizisi için n değerini verir. Örneğin bir önceki örnek için bu değer 2 olacaktır, çünkü ifade iki kelimeli bir ifadedir. Bu bilgi sorgu eşleştirme aşamasında gerekli olacaktır.

Frekans: katalog veritabanında ilgili nGram'dan kaç adet ortaya çıktığı bilgisi tutulur. Bu bilgi yine sorgu eşleştirme aşamasında kullanılacaktır.

Sıra: bir n-gram dizisinin sıklığı kadar bütün n-gramlar içerisindeki yeri de önemlidir. Bu alan, n-gram dizisinin diğer n-gramlar arasındaki yerini sunar.

7. SONUÇ VE ÖNERİLER (CONCLUSIONS AND RECOMMENDATIONS)

Büyük veriden daha önemli bir şey varsa o da muhakkak büyük değerdir. Veri değere dönüştüğü zaman anlamlı olacaktır. Değer, büyük verinin 5 temel başlığından sadece biri değil onun sonucudur. O nedenle büyük veri çalışmalarında en başta dikkat edilmesi gereken konu değer elde etme olmalıdır. Bu çalışmanın amacı da aslında büyük veriden değer elde etme ile ilgilidir.

Değer elde etme sürecini hızlandırmak için zaten değerli olduğunu bildiğimiz verilerin bulunduğu bilgi erişim merkezleri üzerinde çalışma yapılmıştır. Bilgi erişim merkezlerinde yer alan ama bir türlü yerel kalmaktan kurtulamayan verilerin büyük veri yaklaşımı ile bir araya getirilmesinin faydalı olacağı düşünülmektedir. Dolayısıyla; hem bilgi erişim merkezlerinin bir çatı altında toplanması, hem daha değerli verilerin bir araya getirilmesi hem de kolektif çalışma anlamında bir öneri ortaya konmuştur. Büyük veri mimarisi ile benzerlik kurulmuş; dağıtık yapıdaki sıradan sunucular kütüphanelerin katalog veri tabanı sunucuları olarak, ana merkez ise isim düğümü olarak düşünülmüştür. Çok sayıda kütüphanenin bütün verileriyle çalışmak her şeye rağmen uzun süreler alacağı için de katalog veri tabanlarında yer alan veriler n-gram yöntemine göre özetlenmiştir.

Ortaya konan çalışma uygulamaya geçebilecek bir model önerisi şeklindedir. Bilgi erişim merkezlerinin günümüze uyarlaması anlamında değerli bir çalışma olacağı düşünülmektedir. Yapının kabul görmesi halinde katmanlı büyük veri mimarisi, yük dengeleme ve buna benzer konular için model kolayca güncellenebilir haldedir. Büyük veri amaçlarına hizmet edecek olan bilgi erişim merkezlerinin modernizasyonu bir ihtiyaç olup bu amaç için çalışmamızda önerilen yapı kullanılabilir haldedir.

KAYNAKLAR (REFERENCES)

[1] A. Bozkurt, "Öğrenme analitiği: e-öğrenme, büyük veri ve bireyselleştirilmiş öğrenme", *Açık Öğretim Uygulamaları ve Araştırmaları Dergisi (AUAd), 2*(4), 55-81, 2016.

[2] İnternet: G. Utkun, Microsoft Türkiye Blog, http://blog.microsoft.com.tr/buyuk-veri-nedir.html, 30.06.2017.

[3] F. X. Diebold, "Big Data Dynamic Factor Models for Macro economic Measurement and Forecasting", **In Advances in Economics and Econometrics: Theory and Applications, Eighth World Congress of the Econometric Society**, Editörler: Dewatripont, M., Hansen, L.P., Turnovsky, S., 115-122, 2003.

[4] H. Takcı, H. Çetin, "N-Gram Tabanlı Katmanlı Bir Bilgi Erişim Modeli Geliştirilmesi", **International Multidisciplinary Conference (IMUCO 2016)**, Antalya, Türkiye, 503-512, 21-22 Nisan 2016.

[5] V. M. Schönberger, K. Cukier, **Büyük Veri - Yaşama, Çalışma ve Düşünme Şeklimizi Dönüştürecek Bir Devrim**, Çeviren: B. Erol,,Paloma Yayınları, İstanbul, Türkiye, 2013.

[6] C. Eyüpoğlu, M. A. Aydın, A. Sertbaş, A. H. Zaim, O. Öneş, "Büyük Veride Kişi Mahremiyetinin Korunması", *Bilişim Teknolojileri Dergisi*, 10(2), 177-184, 2017.

[7] İnternet: C. Göksu, Datawarehouse Türkiye, http://datawarehouse.gen.tr/big-datanedir-geleneksel-veri-yonetimine-etkisi-ne-olur, 30.06.2017.

[8] B. Hoy, "Bigdata: An introductionforlibrarians". *Medical Reference Services Quarterly*, 33(3), 320-326, 2014.

[9] D. M. Schaeffer, P. C. Olson, "Big data options for small and medium enterprises", *Review of Business Information Systems*, 18 (1), 41-46, 2014.

[10] E. Dumbill, "Making sense of bigdata", *Big Data*, 1(1), 1-2, 2013.

[11] L. Aysan, İ. G. Özbilgin, "Tek Kart Bilgisayarlar ile Bulut Oluşturarak MapReduce İşlemleri Denemesi", *Bilişim Teknolojileri Dergisi*, 8(3), 179-191, 2015.

[12] S. Nicholson, "Bibliomining for automated collection development in a digital library setting: Using data mining to discover web-based scholarly research Works", *Journal of the American Society for Information Science and Technology*, 54(12), 1081-1090, 2003.

[13] R. K. Dwivedi, R. P. Bajpai, "Data Mining Techniques For Dynamically Classifying And Analyzing Library Database", **5th International CALIBER-2007**, Panjab University, Chandigarh, 477-485, 08-10 February 2007.

[14] H. Takçı, İ. Soğukpınar, "Discovery of Access Patterns of Library Users", *Information World Journal*, 3(1), 12-26, 2002.

[15] H. Takçı, İ. Soğukpınar, "Web Kullanıcıların Kümelenmesi ile Nüfuz Tespiti", **TBD 21. Ulusal Bilişim Kurultayı**, ODTÜ, Ankara, 4-6 Ekim 2004.

[16] A. Visa, "Technology of Text Mining", **Machine Learning and Data Mining in Pattern Recognition**, Editör: Perner, P., Springer, 1–11, 2001.

[17] T. Kaşıkçı, H. Gökçen, "Madenciliği ile E-Ticaret Sitelerinin Belirlenmesi", *Bilişim Teknolojileri Dergisi,* 7(1), 25-32, 2014.

[18] M. G. Armentano, D. Godoy, M. Campo, A. Amandi, "NLP-based faceted search: Experience in the development of a science and technology search engine", *Expert Systems with Applications*, 41, 2886-2896, 2014.

[19] X. Niu, B. Hemminger, "Analyzing the Interaction Patterns in a Faceted Search Interface", *Journal Of The Associatıon For Informatıon Scıence And Technology*, 66(5), 1030–1047, 2015.

[20] F. Giunchiglia, U. Kharkevich, I. Zaihrayeu, **Concept Search: Semantics Enabled Information Retrieval**, University of Trento, 2010.

[21] W. B. Cavnar, J. M. Trenkle, "N-gram-based text categorization", **3rd AnnualSymposium on Document Analysis and Information Retrieval**, Nevada, A.B.D., 161-175, 11-13 Nisan 1994.

[22] F. Peng, V. Keselj, N. Cerconey, C. Thomasy, "*N-Gram-Based Author Profiles For Authorship Attribution*", Faculty of Computing Science, Dalhousie University, Canada, 2003.

[23] E. Miller, D. Shen, J. Liu, C. Nicholas, T. Chen, "Techniques for Gigabyte-Scale N gram Based Information Retrieval on Personal Computers", **International Conference on Paralleland Distributed Processing Techniques and Applications**, Las Vegas, A.B.D., 1410-1416, Haziran 1999.

[24] C. Pearce, E. Miller, "The TELLTALE Dynamic Hypertext Environment: Approaches to Scalability", **Advances in Intelligent Hypertext**, Editörler: Mayfieldand, J., Nicholas, C., Lecture Notes in Computer Science, Springer-Verlag, 109 – 130, 1997.

[25] D. Motwani, M.L. Madan, "Information Retrieval Using Hadoop Big Data Analysis", *Advances in Optical Science and Engineering, Part of the Springer Proceedings in Physics book series (SPPHY)*, 166, 409-415, 2015.

[26] R. Lammel, "Google's MapReduce programming model — Revisited", *Science of Computer Programming*, 70, 1-30, 2008.