Cumhuriyet Üniversitesi Fen Fakültesi
Fen Bilimleri Dergisi (CFD), Cilt 38, No. 1 (2017)
ISSN: 1300-1949

Cumhuriyet University Faculty of Science
Science Journal (CSJ), Vol. 38, No. 1 (2017)
ISSN: 1300-1949

http://dx.doi.org/ 10.17776/csj.40741

Hadoop ve Mapreduce Teknolojisi aracılığıyla Gıda-tabanlı Mobil Uygulamaları için bir Arama Hizmeti

Mehmet Akif ÇİFÇİ[1*], Duygu ÇELİK ERTUĞRUL[2]

[1]*Bilgisayar Mühendisliği Bölümü, İstanbul Aydın Üniversitesi, 34295, İstanbul, Türkiye*

[2]*Bilgisayar Mühendisliği Bölümü, Doğu Akdeniz Üniversitesi, Mersin-10, G.Mağusa, KKTC.*

Received: 12.04.2016; Accepted: 22.01.2017

Özet. Son zamanlarda güvenli gıda tüketimi ve e-sağlık üzerine birçok mobil uygulama geliştirilmiştir. Sağlık bilinciyle hareket eden kullanıcılar, özellikle zararlı gıda ve katkı maddelerinden kaçınarak, güvenli gıda tüketimi için bu alana yönelik uygulamaları son derece önemsemektedir. Günümüzde bu tür uygulamaları destekleyen yapılandırılmış veya yapılandırılmamış verileri içeren kapsamlı bir veritabanı eksikliği bulunmaktadır. Bu makalede mobil uygulamalar için sağlıklı bir gıda tüketimi arama hizmeti sunan Hadoop ve Mapreduce (MR) yaklaşımından yararlanan Mobile Apps Search Sevice (MSS) önerilmektedir. MSS, gıda ve gıda katkı maddeleri alanına yönelik hizmet vermekte ve mobil kullanıcıların sorgularını ele alarak bilgi sunma hizmetini kapsamaktadır. MSS herhangi bir mobil uygulamanın arkasındaki işlem olarak çalışabilir. Çünkü MSS, bir arama motoru ile aynı mantıkla çalışır; mobil uygulamalarda tıklamalarla oluşan kullanıcı sorgularına yanıt aramak ve kullanıcıya bilgi sunmak için bağlantılı bilgileri kataloglar ve web kaynakları üzerinde tarama yapar. MSS'in tasarımı ve geliştirilmesi, sistem mimarisi, sorgu anlayışı, Hadoop-MR ortamında ve Action Script kullanımı ile vurgulanmaktadır. Makalenin içinde, bir örnek olay incelemesi ile MSS'in genel özellikleri, işleyişi ve mevcut faydaları ortaya konulmuştur.

Anahtar Kelimeler: Mobil Sağlık Sistemleri, Gıda-Tabanlı Mobil Uygulamalar, Hadoop ve MR, Mobil Sistemler İçin Büyük Veri, Mobil Bilgi Servisi

A Search Service for Food Consumption Mobile Applications via Hadoop and Mapreduce Technology

Abstract. Many mobile applications on safe food consumption and e-health have been developed recently. Health conscious users highly regard such applications for safe food consumption, especially avoiding offending foods and additives. However, there is the lack of a comprehensive database containing structured or unstructured data to support such applications. In this paper we propose the Mobile Apps Search Sevice (MSS), a healthy food consumption search service for mobile applications utilizing Hadoop (a file system) and Mapreduce (MR- a framework). The MSS may work as a process behind any mobile application to provide a service to search for information on food and food additives. The MSS works by the same logic as a search engine (SE); it crawls over Web sources cataloguing relevant information for eventual use in responding to queries from mobile applications. MSS design and development are highlighted through its system architecture, query understanding, its use of the Hadoop/MR Environment, and action scripts. A case study helps displaying the virtues of MSS.

Keywords: Mobile Medical Systems, Food - Based Mobile Applications, Hadoop and MR, Big Data for Mobile Systems, Mobile Data Service

1. GİRİŞ

"Arama motorları, veri metni alımının birincil araçlarıdır. Standart bir arama motorunun şu üç temel görevi yerine getirmesi gerekir: Web'de tarama, taranılan içeriği indeksleme ve indeksi kullanarak sorguları işleme [1]." İnternet üzerinde -çoğu yapılandırılmamış- çok büyük miktarda veri bulunmaktadır. Bu veriler; insanlar, araçlar ve makineler tarafından oluşturulmuştur. Büyük veri dinamik, çok boyutlu ve birbirinden farklı hacimlerde olabilir. "Büyük Veri; büyük hacimli, karmaşık

* Corresponding author. *Email address:* wwwakif@msn.com

 ©2016 Faculty of Science, Cumhuriyet University

ve büyüyen verilerle ilgilenmektedir. Hızlı ağ oluşumu ve veri depolama özelliği sayesinde büyük veri -fiziksel, biyolojik ve biyomedikal bilimler de dâhil olmak üzere- tüm bilim ve mühendislik alanlarında hızla büyümektedir [2].” Tüm bu yapılandırılmamış verilerle başa çıkmak için işlevsel ve ölçeklenebilir yeni bir teknoloji gerekmektedir. Bu ihtiyaçtan ötürü Hadoop çerçeve programı geliştirilmiştir. Hadoop sıradan sunuculardan oluşturulan kümeler üzerinde büyük hacimli verilerin dağıtık dosya sistemi ile incelenmesine olanak sağlayan açık kaynaklı bir yapıdır. Hadoop ve MR çok büyük miktardaki verileri işlemek için tasarlanmıştır. Hawking (2011)’in bulguları da bu çalışmayı desteklemektedir. “Hadoop, çok büyük verileri işlemek için tasarlanmış bir platformdur. Hadoop, sıradan sunucuların bir araya gelerek oluşturdukları küme yapısıyla çalışmaktadır. Sunucular, küme yapısına dinamik olarak eklenip çıkarılabilmektedir. Hadoop kendi kendini tedavi edebilen bir mimari yapıya sahiptir [3].” Hadoop oluşumları dört tür işlem içerir: Hadoop üzerindeki tüm dosyalar hakkındaki bilgileri saklayan NameNode (master), dağıtılan iş parçacıklarının çalışmasından sorumlu olan JobTracker, görevi blokları saklamak olan DataNode (slave) ve tamamlamak üzere iş parçacığı talep eden TaskTracker’dır. Hadoop, büyük miktarda veri içeren binlerce düğümle, sistemlerdeki uygulamaların çalıştırılmasını sağlar. Hadoop işlemi, her türlü veri için büyük çapta depolama sağlayarak kesintisiz ve limitsiz bir şekilde devam etmektedir. Hadoop internetin hemen hemen her alanında yer aldığı gibi normal ve dikey arama motorlarında sıklıkla kullanılmaktadır. Herodotu ve Lim (2011) çalışmalarında bunu desteklemişlerdir. Ayrıca Hadoop Google, Facebook, DuckDuckGo gibi arama motorları için vazgeçilmezdir.

İlaveten günümüzde mobil uygulamalar zaman ve mekândan bağımsız olarak bilgiye erişim olanağı sağlamaktadır. Hoopmann, Moritz, Deutsch ve Boyle (2012)’un belirttikleri gibi Hadoop ve MR arama motorları için büyük öneme sahiptir. Hoopmann vd., yaptıkları çalışmalarla bu öngörüye katkı sağlamaktadırlar. Hadoop esnek, ölçeklenebilir ve güvenilirdir. İhtiyaç duyulduğunda, verinin kendisini, formatını, yerini değiştirmeden çalışan, işlerin ve uygulamaların nasıl yazıldığını dikkate almadan yeni düğüm noktası ekleyebilen özelliğine sahiptir. Esnek bir yapıya sahip olan Hadoop, farklı kaynaklardan gelen farklı veri tiplerini birbirleriyle birleştirip, özetlenebilir ve işlenebilir hale getirir. Ayrıca veri noktalarından birisine ulaşılamaz olduğunda, sistem, gelen yükü diğer veri noktalarına paylaştırarak kesintisiz hizmete devam ettiği için güvenilirdir. Söz konusu yaklaşım sadece veriyi işlemeyi değil; veriyi anlamlı hale getirip kullanıcıya sunmayı da amaçlamaktadır. Lam (2010)’in çalışmaları da bu yaklaşımı desteklemektedir. Bunlara ek olarak, bütün mobil platformlarda çalışılabilir olup; ayrıca açık kaynaklı bir proje olması özelliği de MSS’i yenilikçi bir yaklaşım olarak göstermektedir.

Bu makalede, gıda tüketimine yönelik, mevcut mobil uygulamaların arka planında çalışacak olan ve ürünlerdeki gıda katkı maddeleri hakkında bilgi sunan bir arama servisi ele alınmıştır. Bu mobil uygulamalar, kullanıcılarının tükettikleri/tüketecekleri ürünlerin içinde bulunan gıda katkı maddeleri ile ilgili çeşitli sorgular yapma ihtiyacı duyar. Yapılan sorgulara doğru sonuçlar sağlamak için mobil uygulamaların kendi gıda katkı maddeleri hakkında veri tabanı olmalıdır; aksi takdirde bilgi döndürülememektedir. Bu nedenle, gıda tabanlı mobil uygulamalar için özel amaçlı gıda katkı maddeleri hakkında bilgi sağlama servisi olan Mobile Apps Search Sevice (MSS) düşünülmüştür. Hadoop temelli bir MSS'in sağlıklı gıda tüketimi ile ilgili verileri taraması, toplaması, indekslemesi, kataloglaması ve hizmet vermesi süreçleri önerilmiştir. Bu makalenin geri kalanında ele alınacak olan MSS, verilerin ve hesaplamaların farklı bilgisayarlar arasında dağıtılarak, arama sonuçlarının daha hızlı geri dönmesinde yardımcı olur. Böylece birden fazla görev aynı anda yapılabilir.

Bu makale aşağıdaki sıra ile sunulacaktır: Bölüm 2’de problem tanımı ve MSS sisteminin mimarisi anlatılacaktır. Bölüm 3’te önerilen sistemde kullanıcı arama terimlerini ve/veya başka parametreleri yazıldığında girdi sorgunun oluşturulması hakkında bilgi sunulmaktadır. Bölüm 4’te sistemin geliştirilmesi için kullanılacak araç ve yöntemlerden bahsedilmektedir. Bölüm 5’te önerilen MSS

sistemi için iş akış diyagramı ve sisteme genel bakış incelenecektir. Bölüm 6'da sistemin geliştirilmesine yönelik uygulama algoritmaları anlatılacaktır. Bölüm 7'de örnek mobil uygulama üzerinde MSS'in çalıştırılması ve değerlendirilmesi gösterilmiştir. Son olarak Bölüm 8'de ilgili sonuçlar verilmiştir.

2. SİSTEM MİMARİSİ

MSS sistem mekanizması, bir arama hizmeti sağlayıcısının neredeyse aynısıdır. MSS, yalnızca anahtar kelimeleri ya da anahtar cümleleri tanımlamak yerine, yapılandırılmış veriler arasındaki bağlantıyı anlar. MSS bir anahtar kelime eşleştirme hizmeti değildir. Ayrıca MSS arayan kişinin maksadının ne olabileceğini anlamaya çalışarak arama doğruluğunu artırır. Genel arama motorlarının aksine MSS özellikle gıda tüketimi işlevine odaklanmaktadır. Bu işlevselliği elde etmek için birbirleri ile etkileşimde olan yedi farklı modül tasarlanmış ve sistem mimarisine dahil edilmiştir. Bilindiği üzere arama motorları; web tarama, indeksleme, sorgulama ve bilgi çekmeyi gerçekleştiren bilgi erişimi uygulamalarıdır. MSS ise Hadoop ve MR kullanarak gıda ile ilgili verileri aramak için özellikle tasarlanmış mobil uygulamalara yönelik özel bir arama hizmetidir. "MR, büyük veri kümeleri işlemeye ve oluşturmaya yarayan uygulama ve programlama modelidir. Kullanıcılar, bir ara anahtar/değer çifti seti oluşturmak için bir anahtar/değer çiftini işleyen eşleşme fonksiyonu ve aynı ara anahtarla ilişkili tüm ara değerleri birleştiren bir indirme fonksiyonu belirler. Birçok görev bu modelde anlatılabilir [4]." MR, çalışma sisteminin görevlerini azaltmak için girdi-veri eşleşmesini bölümler. Ayrıca girdi/çıktı verilerini makinelere transfer eder.

Tablo 1. Birbirleri ile etkileşimli MSS'in modülleri

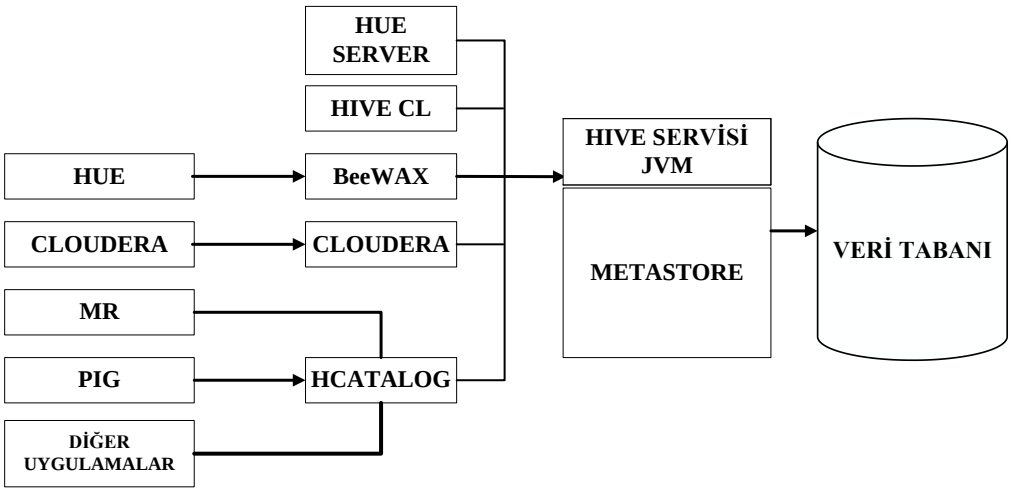
Modüller		
1	Veritabanı	* Veri aktarmak için kullanılan RDBMS
2	Veri havuzu	* Depolama için yönetilen bir yer.
3	Websitesi Ekleme Sihirbazı	* Websiteleri Crawler'a ekleme sihirbazı
4	Crawler/Spider/bot	* Websitelerini indekslemek için gerekir
5	Ürün Entegrasyon Servisi	* Cep telefonlarını bağlama entegresi
6	Websitesi	* Ağ üzerinden erişilebilen bir sayfa
7	Hive/sqoop/MR	* Hadoop ortamındaki araçlar
8	Mobil Uygulamalar için Bağlantı Hizmeti	* Herhangi bir mobil uygulama için hizmet

Ağ transferlerini azaltmak amacıyla girdi verilerinin hâlihazırda depolandığı makineler kullanılarak öncelikle hesaplama yapılmıştır. MR'ın bir başka özelliği ise Hadoop üzerinde çalışabilen ve dağıtık mimari üzerinde çok büyük verilerin kolay bir şekilde analiz edilebilmesini sağlayan sistemlerden bir tanesi olmasıdır. Bünyesindeki MAP (haritalama) ve REDUCE (indirgeme) fonksiyonlarını kullanarak verileri işler. Map aşamasında, analiz edilen veri içerisinden alınmak istenilen veriler alınır. Reduce aşamasında ise bu çekilen veri üzerinde istenilen analiz ve diğer işlemler gerçekleşir. Yine Reduce aşamasında tamamlanan işler işin mantığına göre birleştirilerek istenilen sonuç elde edilir. Map aşamasındaki işlemler birbirinden bağımsız olarak gerçekleşebildiği için aynı anda birden çok işlem yapılabilir. Bu sayede büyük miktardaki veri, Cluster (küme) içerisindeki Node (düğüm)'lar tarafından hızlı bir şekilde okunabilir. Cluster'da yer alan Node sayısı arttıkça işlemlerin hızı da aynı oranda artar. Reduce aşamasında ise aynı anahtara (Key) sahip veriler paralel olarak işlenebilir.

Hadoop ve MR sayesinde MSS sistemi, internet taraması yapıp; taramaları indeksleyerek ve girdi / çıktı veriler arasında bir ilişki oluşturarak mobil uygulama kullanıcılarına sağlıklı gıda tüketimi ile ilgili sorgu sonuçları sağlayabilmektedir. Hadoop'un yüksek hata toleransına sahip olması nedeniyle MSS kesintisiz bir şekilde çalışabilir. Hadoop başarısız görevleri yeniden çalıştırarak hataların üstesinden gelir. Bu tür başarısız görevler, MSS'in işini kolaylaştıran başarılı ve hızlı sonuçlar elde etme olasılığını arttırmak için birden çok düğümde uygulanabilir.

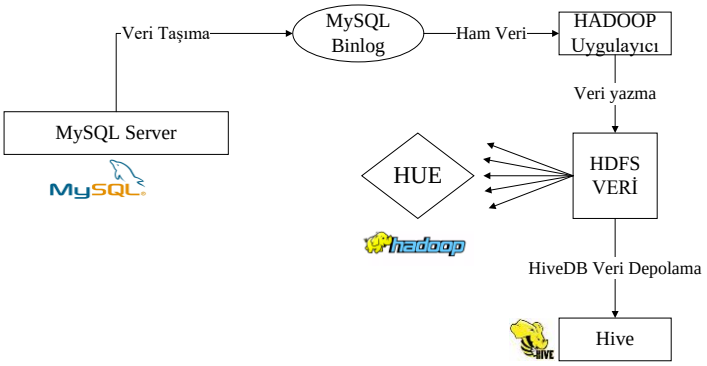
MSS özellikle çeşitli web kaynaklarından bilgi toplama adımı olan web taramasına odaklanmaktadır. Crawler(arama motoru robotu) ilgili belgeleri bulur ve indeksleme işlemi için hazırlar. Crawler, internetteki bilgileri tarayıp kendi belirlediği kurallara göre dizinlerine kaydeder. Tam manasıyla taranmayan web sayfaları için genellikle web sayfalarının açıklamalarını daha doğru bir şekilde sağlayan anchor (bağlayıcılar) kullanılır. Binlerce web sitesi indekslendikten sonra, içerik İlişkisel Veri Tabanı Yönetim Sistemi (RDBMS) içine transfer edilecektir. Daha sonra, veri RDBMS'ten bazı basit SQL kodları kullanarak sqoop (veri işleme CL) yoluyla, sorgu ve analiz sağlayan bir veri ambarı altyapısı olan Hive'a transfer edilir.

MSS'in çalışma mekanizması Şekil 1'de gösterilmektedir. Hadoop kullanıcı arayüzü (HUE)' nün RDBMS'e Hadoop ile çalışmayı kolaylaştıran bazı basit SQL kodları ile bağlı olduğu görülmektedir. Crawler'lardan yapılandırılmamış veri, dağıtık dosya sistemi Hadoop Distributed File System (HDFS) ortamında MR yardımıyla yapılandırılmış formata dönüştürülür. Tarama verisi MR algoritmaları ve görevleri üzerinde işlenir. Şekil 1'de görüldüğü gibi HUE ana çerçeve Cloudera HUE'nun üzerine kurulu olduğu sanal sistem, MR, Pig ise Hadoop üzerinde çalışan programlardır ve veri kataloglama işleminden sorumludurlar. Verileri analiz ederek düzenleme yaparlar. Neticede bu programların hepsi Hive JVM'den yararlanırlar ve MSS'in veri tabanı ile sürekli iletişim içerisindedirler.



Şekil 1. HUE ve RDBMS İş akışı.

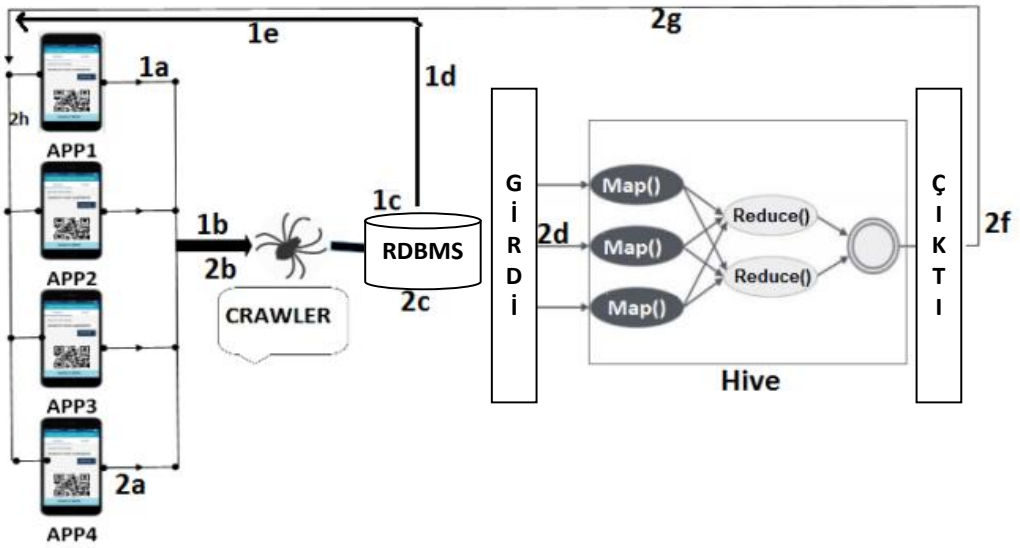
HUE ve RDBMS arasındaki bağlantı Şekil 2'de açıkça gösterilmektedir. Öncelikle, gerçek zamanlı veri MySQL'den getirilir. Daha sonra, ham veri haline gelir ve son olarak MSS için işlenecek veri HiveDB'nin içine yerleşir.



Şekil 2. MySQL'den HDFS Tümlevleme

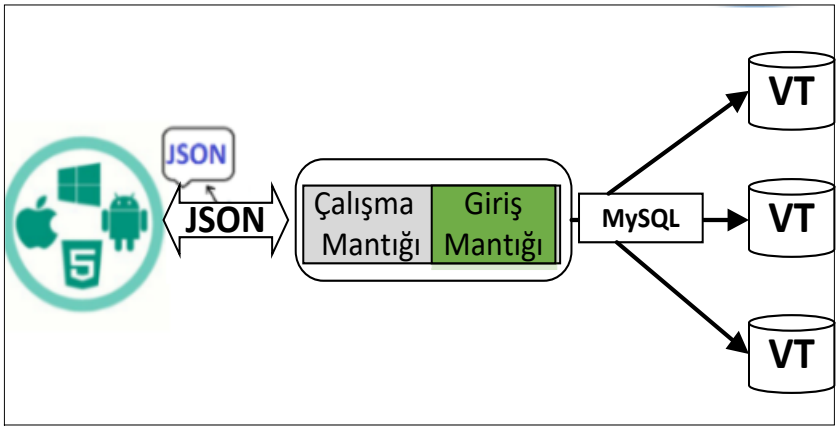
MSS'teki bilgi servisi destek akışı Şekil 3' te gösterilmektedir. MSS sistem mekanizması şu şekilde çalışmaktadır: Öncelikle kullanıcı bir sorgu yapar, yani arayüze bir anahtar kelime girer (bkz. 1a). Girilen anahtar kelime doğrudan bir arama robotu olan Crawler'a ulaşır (bkz. 1b) istenilen sonuç veritabanında (RDBMS) bulunursa (bkz. 1c) sonuç doğrudan kullanıcıya döndürülür (bkz.1d) ve böylelikle bilgi kullanıcıya ulaşmış olur (bkz. 1e).

Şayet istenilen sonuç veritabanında bulunamazsa yapılan sorgu (bkz. 2a) doğrudan Crawler'a ulaşır (bkz. 2b) oradan RDBMS (bkz. 2c) üzerinden büyük ölçekli hiveDB'ye yönlendirilir (bkz. 2d) ardından bilgiyiye ulaşılırsa veri kullancıya (bkz. 2f) taşınmış olur. Son olarak ilgili sonuç kullanıcılara ulaşmış olur (bkz. 2g).



Şekil 3. MSS Bilgi Hizmeti Desteği İş Akışı.

MSS ve mobil uygulamalar arasındaki bağlantı mekanizması Şekil 4'te gösterilmektedir. Bu bağlantı JSON (Hem insanlar, hem de makineler tarafından okunabilen bir veri değişimi biçimidir) programlama dili aracılığıyla yapılmaktadır. MSS, verilere doğrudan erişim için ya da verileri kullanarak bazı görevlerin çalıştırılması amacıyla mobil uygulamalardan gelen istekleri destekler.



Şekil 4. Veri Tabanına Bağlanan e-Sağlık Mobil Uygulamaları.

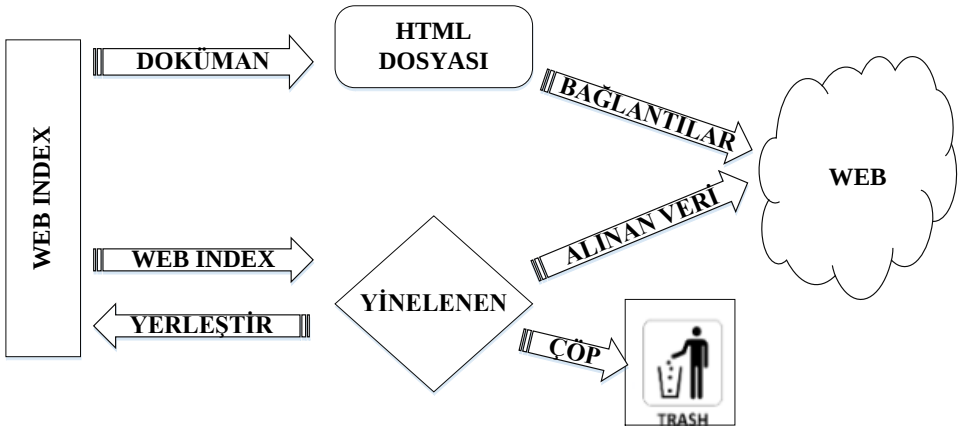
3. SORGU OLUŞTURULMASI

Kullanıcı; arama terimlerini ve/veya başka parametreleri yazıp; mobil uygulama üzerinde tıkladığında işletim sisteminin arka planında bir Spider (Crawler) bu terimleri her bir arama hizmetinin işleyebileceği sorgulara çevirir. Sonrasında sorguları gönderir ve daha iyi işlenmesi için sonuçlar standart bir formata dönüştürülür. Bu fotmata Tuple denir. Tuple (veri kümesi), çok değişik verilerin bir araya gelmesiyle oluşmuş veri gruplarıdır. Tuple, verileri birden çok değere çevirebilir ve çok sayıda değeri hafızasında tutabilir. Tuple belge referanslarını tanımlayan Crawler tarafından kullanılmaktadır. Tuple parantezler kullanır (). Her bir Tuple kendi özel imzasını taşır. (Tuple1, Tuple2, …) gibi. Tuple'larda belge referansı bulunur. Bir belge referansı, verilen belge için URL, başlık, parça ve güven puanıdır. Genel olarak bakıldığında işin büyük bir bölümü, farklı arama hizmetlerinin sorunlarıyla başa çıkan Wrapper tarafından yapılır. Wrapper bilgisayar veya yazılım kütüphanesinde bulunan ve aramaları daha akıllı hale getirmek için kullanılan bir altyordamdır (subroutine). Ayrıca Wrapper genel programlama da sağlar.

3.1. Harmanlama ve Yinelenen Algılama

Ganesh (2004)'in belirttiği gibi ortalama bir Crawler başlangıç sayfası P0 ile başlar ve başlangıçta bu sayfaya erişim sağlanır. Daha sonra bu sayfadaki bütün URL'leri ayıklar ve onları incelemek için URL sırasına ekler. Ardından sırasıyla URL'leri işler ve bu işlemi tekrar eder. Crawler taradığı her sayfayı kaydeder, sayfalar için indeksleme yapar veya sayfa içeriğini analiz edip özetler [5]. İlk adım, taramanın yapılandırma ayarlaraını yapmaktır. Sayfaların uyup uymadığını ve Crawler'ın ne yaptığını görebilmek için taranacak sayfaları bir şekilde sınırlamak önemlidir.

Bir web indeksinin MSS'te yinelenen bir dosyayı silmek için Crawler'a belgeleri kontrol etme isteği göndermesi Şekil 5'te gösterilmektedir. Crawler işi kontrol ettiğinde şayet benzer dosya bulursa bunun bir kopyasını saklar, diğer kopyasını ise çöpe yollar. Şekil 5'te de açıkça görüldüğü gibi saklanan uygun yolla web'e gönderilir.



Şekil 5. Crawler Yinelemeleri Nasıl Temizler.

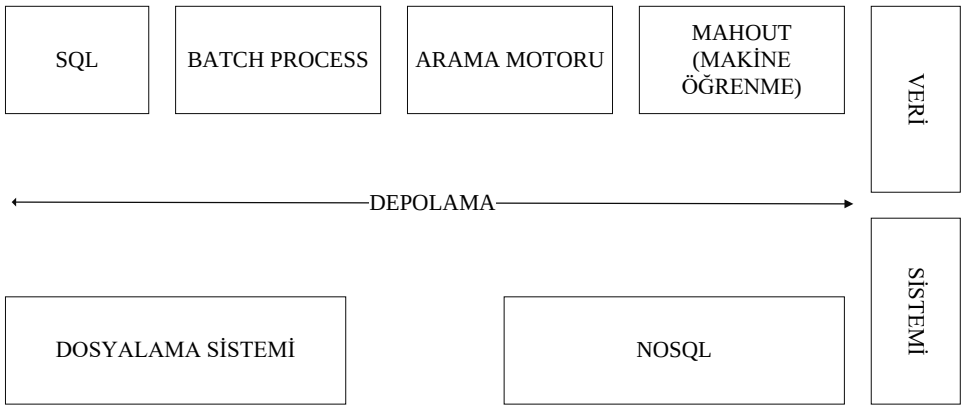
3.2. Veri Seçimi ve Toplama

Veri toplamanın ilk yöntemi çeşitli web sitelerinden gıda ile ilgili indirilen verileri veri tabanına eklemektir. Bu nedenle verinin nerede ve nasıl alındığı ya da gıda ile bağlantılı verinin formatı önemlidir. Çünkü yüzlerce web sitesi aranabilir ve binlerce gigabayt ücretsiz veri bu amaçla indirilebilir. En önemli kaynaklardan bir tanesi Westbury laboratuvarıdır. Westbury laboratuvar kitaplığı "konular ve kelime kullanımları yelpazesiyle ve konuya göre belgenin düzgün organize edilmesi nedeniyle korpus olarak Vikipedi seçmiştir. Buna ek olarak Vikipedi'de en sık kullanılan, küçük harfle yazılmış 30.000 kelime MSS veri tabanında kullanılmıştır [6]." Örneğin Westbury laboratuvarı Vikipedi kitaplığından yaklaşık 2 milyon serbest nanoyapılı veri makalesi almıştır. Bu yapılandırılmamış veriler, RDBMS'e (MySQL) kolayca yüklemeyi mümkün kılan TXT formatındadır. Veri toplamanın ikinci tekniği ise veriyi araştırmak ve sonrasında web sayfalarının içeriğini indekslemek için web taraması yapmaktır. Web Crawler, bir arama motoru tarafından daha sonra işlenmesi ve indirilen sayfaların indekslenmesi için ziyaret ettiği tüm sayfaları kopyalarak kullanıcıların çok daha etkin bir şekilde arama yapmasını sağlayabilir. MSS, işlemleri tipik veya geleneksel arama motorlarıyla aynı olmayan bir bilgi-tabanlı arama hizmetidir. Belgeleri, anahtar kelimeler olarak indekslemek yerine MSS, kullanıcılara bağlantılı sonuçlar vermek için Web Ontoloji Dili (OWL), Kaynak Tanımlama Çerçevesi Şeması (RDFS) veya Kaynak Tanımlama Çerçevesi (RDF) gibi modellerde büyük miktarda ontoloji toplayacaktır. "Bir arama motoru için bağlantılılık, doğru kelimelerle bir sayfayı bulmaktan daha fazlası demektir [7]." Bu nedenle sorgu sonuçları anlamlı olmalıdır.

4. KULLANILAN ARAÇLAR

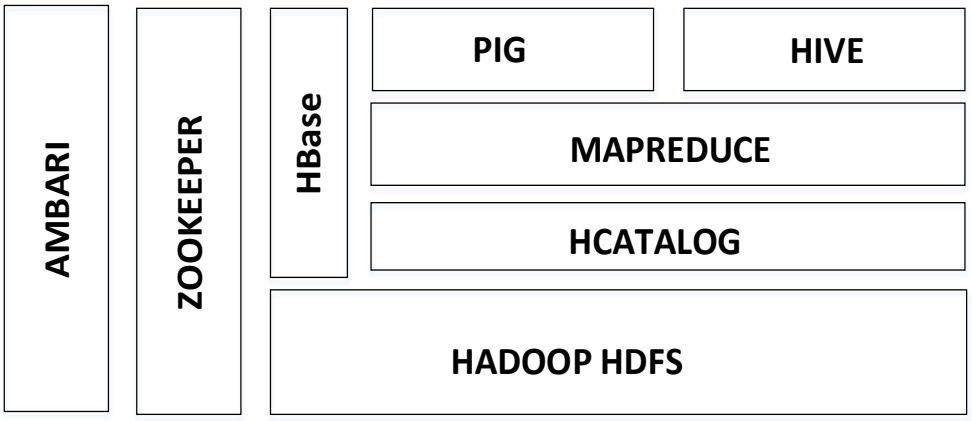
MSS, arama motoru bileşenlerinin geliştirilmesi için birçok araç kullanmaktadır. Hadoop ve MR en önemlileri olmakla beraber, açık kaynak kodlu veri tabanı olan Hbase (veri tabanı) ve RDBMS ile birlikte Hive (veri tabanı), sqoop ve MSS oluşturmak için PHP ve SQL gibi araçlardan yararlanılmıştır. Şekil 7'de, Cloudera Hub'da bir SQL ve NOSQL belleğinin de olduğu gösterilmektedir. Odak noktası SQL'dir, çünkü MSS çoğunlukla PHP ve SQL komutlarında geliştirilmiştir. Bu nedenle, Hadoop'u RDBMS'e bağlamak için SQL'i seçmek, hayati önem taşımaktadır. MSS'i daha akıllı hale getirerek kullanıcı sorgularından kullanıcının ne istediğini öğrenmeye yarayan Hadoop üzerinde bir program olan makine öğrenimi ise diğer bir odak noktasıdır.

MSS, anlamı kavrayarak kullanıcıların sorgularına daha doğru cevaplar vermeyi hedefleyen semantik arama üzerine yoğunlaşmaktadır. Semantik aramanın buradaki amacı, kullanıcıların belirli bir bağlamda ne istediklerini anlamak için kelimeler ya da cümlelerin 'statik' sözlük anlamının ötesine geçmesini sağlamaktır (bkz. Şekil 7).



Şekil 7. Cloudera Data Hub.

Cloudera CDH (Hadoop için gerekli olan sanal sistem)' in kurulması, tüm bu araçlara sahip olmak için daha kolay bir yoldur. Kurulum bittikten sonra HUE kullanıma hazırdır. Şekil 8'de gösterildiği gibi, HUE arayüzünde Hive ve Hbase gibi tüm araçlara sahip olunabilir. Ambari HDFS dosyalarını gözetleme, görüntüleme ve yönetme işleminden sorumludur. Zookeeper, dağıtık yapıda olan servislerin koordine edilmesi için geliştirilmiş -kendisi de dağıtık yapıda çalışan- bir uygulamadır. Hbase çok büyük boyutlara sahip verilere gerçek zamanlı okuma/yazma erişimi yapmak gerektiği zamanlarda kullanırken; aynı ortamda çalışan Pig ise MR ile büyük verileri dağıtık sistemlerde analiz eder. Hive, SQL biçimde sorgular geliştirmemizi sağlayan Hadoop tabanlı veri ambarı (datawarehouse) kütüphanesidir. MR Hadoop projesinde, verileri birden fazla makinede paralel olarak işlememizi sağlayan bir kütüphanedir. Hcatalog verilerin Pig ve MR yoluyla daha kolay okunmasına, yönetilmesine yardımcı olur (bkz. Şekil 8). Semantik temelli MSS oluşturmak için, Hadoop üzerinde semantik arama eklentisi olan Redlink Solr (semantik için eklenti) (bkz. Tablo 2) eklentisi kullanılmaktadır.



Şekil 8. HADOOP Ortamı

Redlink Solr Eklentisi Hadoop üzerinde çalışabilen MSS'in sonuçlarını anlamsal kılan bir eklentidir. Bu eklenti Tablo 2'de gösterildiği gibi sunucu üzerine kurulmuştur.

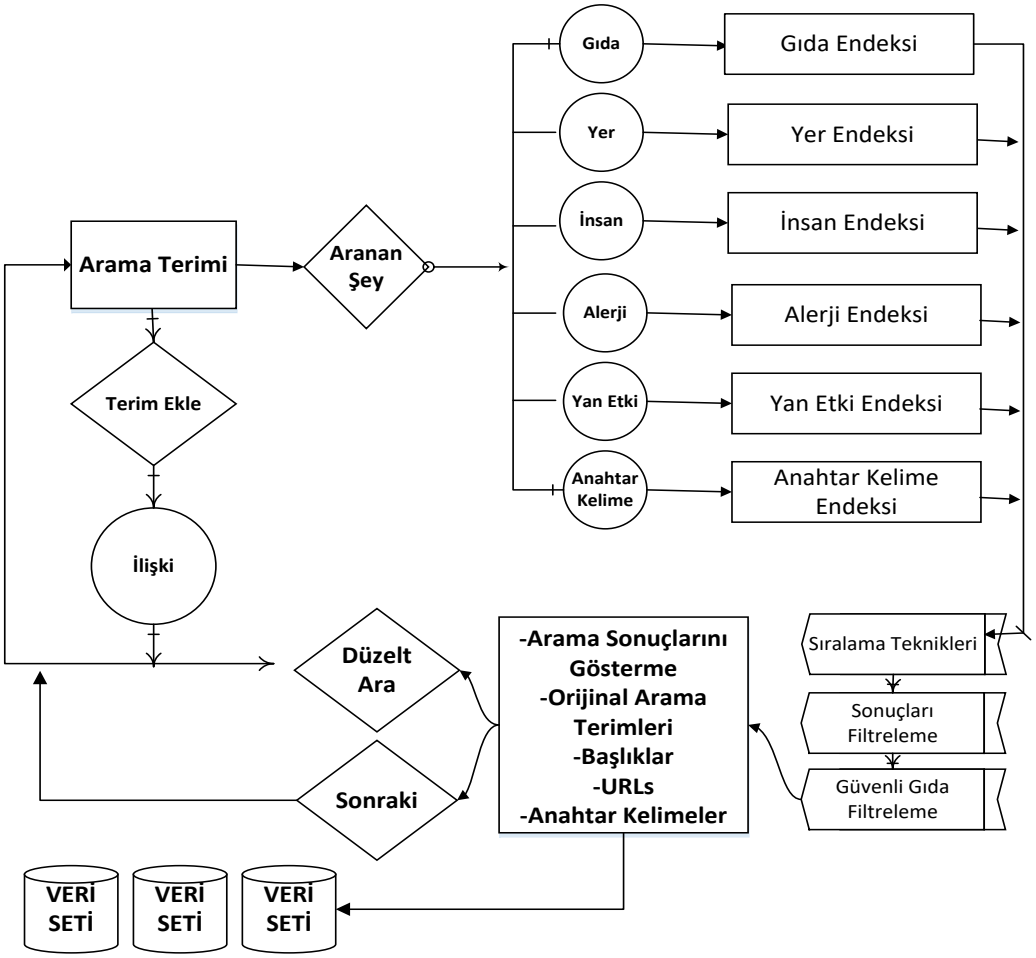
Tablo 2. Redlink Solr Eklentisi Kurulumu

Prosedür 1: Redlink eklentisi algoritması
1. Create lib directory for your core.
2. Enable it by adding <lib dir="./lib" /> to the solrconfig.xml file.
3. Copy the plugin (solr-plugin-1.0.0.jar) into the lib directory of the core

5. AKIŞ ŞEMASI

Kullanıcının gıda güvenliği ile ilgili herhangi bir arama terimini veya sorgusunu girdiği arama motorunun akış şeması Şekil 9'da gösterilmektedir. Sonrasında işlem gerektiğinde, yeni arama terimlerinin eklenebildiği arama bölümü ile devam edilmektedir. Ardından bir bağlantı kurulur, veri filtrelenir ve depoları geçtikten sonra, veri daha anlamlı hale gelir. Yinelenenler tarandıktan sonra silinir. Veri her zaman yinelenenlerden arındırılır ve bir kopyası Hadoop ile bağlantılı veri tabanında depolanır. Hadoop'ta veriyi depolamak için kümeler kullanılır. MR kullanılarak bir Hadoop kümesi içine toplanan veriler çıkarıldıktan sonra belge türleri tespit edilerek metaveri olarak bu belgelerden aktarılır. Son olarak, metaveri indekslenir, böylece arama yapıldığında tüm indeksler aranabilir. Bu sayede, sonuçlar daha bağlantılı ve anlamlı hale gelmektedir.

Bilgi çekme aşamasında, anahtar kelimelerle arama yapan bir kullanıcı indeks veri tabanına uyan dizeleri arar ve böylece en iyi eşleşen sayfaların listesi kısa bir açıklama ile karşısına çıkar. Binlerce sayfa arama dizesiyle eşleşir. Bu nedenle kullanıcı sorgusuyla bağlantılı sayfaları belirlemek için, arama motoru-MSS- kendine ait bir algoritma bulundurur. Bu algoritma Şekil 9'daki, MSS'in akış şemasından da anlaşılmaktadır. Öyleki MSS'teki herhangi bir sorunun çözümü için izlenmesi gereken aritmetik ve mantıksal adımların bütünü Şekil 9'da anlatılmıştır. "Kuchiki, Tsukada vd., (2008) çalışmalarında belirttikleri gibi akış şeması MSS algoritmasının adımlarının mantıksal sırasını, adımların birbiri ile bağlantısını, bir işlemden diğerine nasıl bir yol izleyeceğini belirten kontrol mekanizmasıdır. Girilen anahtar kelimeler belirli ve mantıklı yollar izlemektedir [8]."



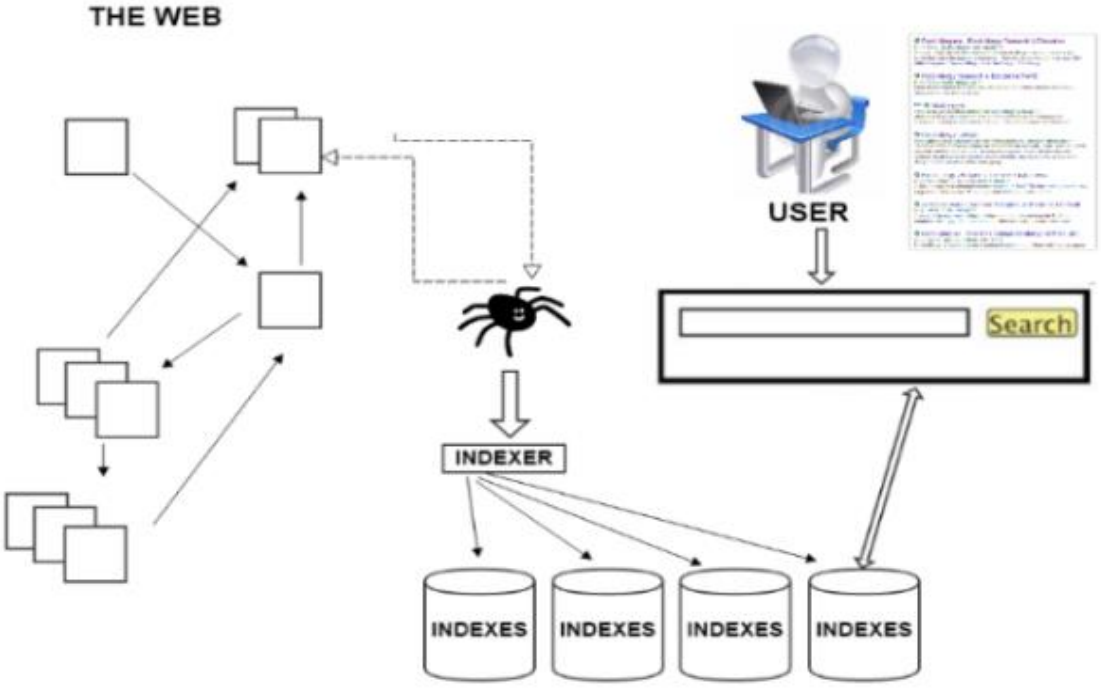
Şekil 9. Arama Motoru Akış Şeması.

Gıda katkı maddeleri ile ilgili sonuç sayfasının nasıl görüntülendiği Şekil 10'da gösterilmektedir. "Kitchen" anahtar kelimesi girildiğinde, MSS bir bilgi tabanlı arama motoru olduğu için kullanıcının gıda güvenliği ve gıda katkı maddelerini aramakta olduğunu anlar. MSS, bilgi temelli olduğundan konuyla ilgili sonuçları aramaktadır. Böylece sonuçlar daha akıllıca ve anlamlı olacaktır. Görüldüğü gibi, Şekil 10'da bir arama sonucunun sıralama sistemi gösterilmektedir. En bağlantılı sonuçlar, sıralama algoritmasına göre en tepede bulunmaktadır. "PageRank, web sitesinin ne kadar önemli olduğu hakkında kabaca bir tahmin belirlemek için bir sayfaya giden bağlantıların kalitesini ve sayısını hesaplayarak çalışır. Altta yatan varsayım, daha önemli web sitelerinin diğer web sitelerden daha fazla bağlantı almasıdır [9].” MSS içerik tabanlı olacaktır. Kullanıcının niyetini anlayacak ve verinin işlendiği HUE içinde veriler arasındaki bağlantıları yapabilecektir. MSS, gıda tüketimi üzerine mobil uygulamalar için dikey tip bir arama hizmet sağlayıcısı olduğu için bağlantılı sorguyu bulabilir ve sorgu Şekil 10'da gösterildiği gibi anlamlı olur.



Şekil 10. MSS Sonuçlar Sayfası.

MSS'in genel bir bakışı Şekil 11'de gösterilmektedir. Bir kullanıcı istek gönderdikten sonra MSS bağlantılı sonuç için interneti tarar ve bağlantılı sonuç bulunursa gönderilir. Aksi takdirde yapılandırılmış veri, veritabanı ile eşleştirilir ve bir eşleşme bulunursa sonucu kullanıcıya gönderir. Bir site için tüm eşsiz bağlantılar tıklandığında ve ziyaret edildiğinde Crawler'ın işi tamamlanır. Elbette, sitede kendilerine bir bağlantı olmayan sayfalar da bulunabilir. Bu durumda "Sayfaya başka bir sitede başvurulmadığı sürece, Crawler bu sayfa hakkında bilgi edinemeyecektir [10]." Bu engelle başa çıkmak için, MSS veritabanına binlerce gigabayttan fazla veri karşıdan yüklenir.



Şekil 11. MSS'e Genel Bakış.

6. UYGULAMADA KULLANILAN ALGORİTMALAR

MSS'teki Haritalama ve İndirgeme Algoritması Tablo 3'te gösterilmektedir. Öncelikle Haritalama işlemi gerçekleşir. Bunu takiben veriler türüne göre bileşenlerine ayrılır. Sonuç olarak indirgeme işleminde kategorize edilmiş verilerden sonuca varılır. MSS'in MySQL ve PHP programlama dillerinden yararlandığını bilinmektedir. MSS; Korsanlık, SQL enjeksiyonu ya da başka bir Web saldırısına karşı güçlü şifrelerle korunacaktır.

Tablo 3. MR Algoritması.

Prosedür 2: Haritalama ve İndirgeme Algoritması	
1.	MAPPER(record):
2.	my number = record ['a number'] value = {1, record ['a value']}
3.	emit (my number, value)
4.	JOINER (my number, value sequence): record number = 0; value sum = 0;
5.	for each (value: value_ sequence) {
6.	record number += value [0] value sum += value [1]value out = {record number, value sum}
7.	emit (my number, value out)
8.	REDUCER (my number, value_ sequence)
9.	record number = 0 value sum = 0
10.	for each (value: value_ sequence)
11.	record number += value [0] value sum += value [1] total = value sum / record number;
12.	emit (my number, total)

Web Sayfalar arası ilişkiler Tablo 4'te açıkça gösterilmiştir. Tabloda görüldüğü üzere Sayfa 1 öncelikle Sayfa 2' yi takip eder. Bu, ortak olasılığın nasıl çalıştığını göstermek için basit bir senaryodur; ancak olasılıklar, yapılan sorgu ile tahminler ilişkilendirilmeye çalışıldığı için tüm olasılıklar göz önünde bulundurulmalıdır.

Tablo 4. Sayfalar Arası İlişkiler.

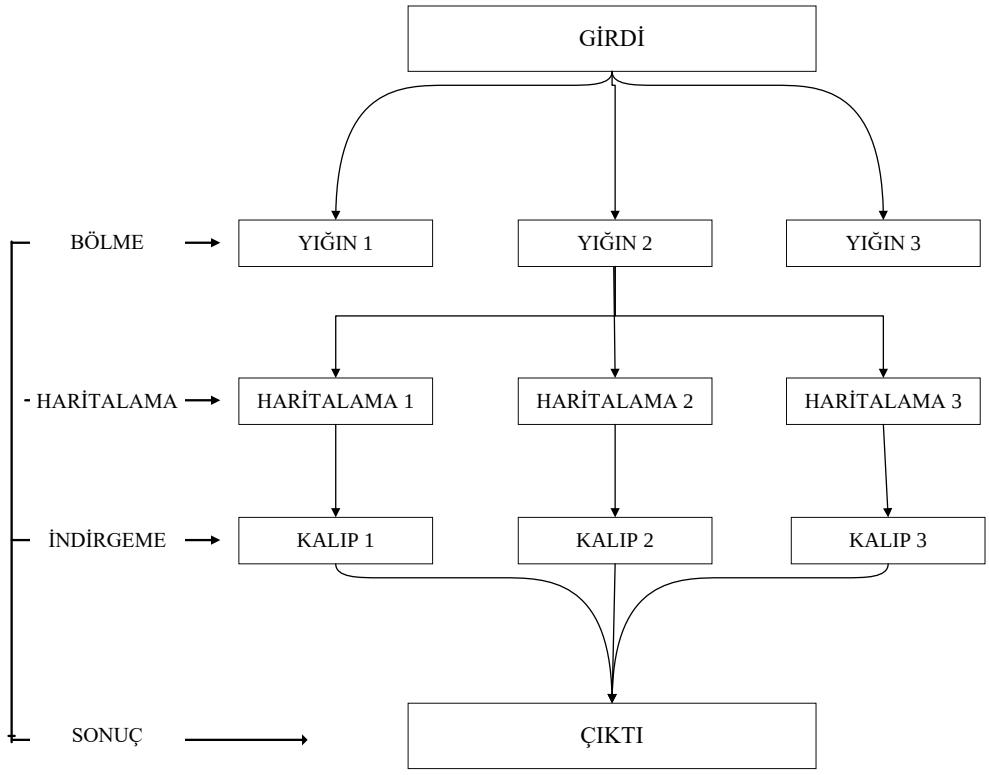
Sayfa1(s1)	Sayfa2(s2)
İlişki Olasılıkları: S(ij, s) = δij / ηij = Tij S(12, s1) = 1 / 2 = 0.5 S(13, s1) = 1 / 2 = 0.5 Birleşme Olasılıkları: S(Q,p1) = (S(12, s1)) ∩ (S(13, s1)) = ((1/2) ∩ (1/2)) = (0.5 *) = 0.25	İlişki Olasılıkları: S(ij, s) = δij / ηij = Tij S(12, s2) = 2 / 2 = 1 S(13, s2) = 0 / 2 = 0 Birleşme Olasılıkları: S(Q,s2) = (S(12, s2)) ∩ (S(23, s2)) ∩ (S(13, s2)) = ((2/2) ∩ (1/2) ∩ (1/2)) = (1

Ayrıca, sıfırdan farklı bir puanı, ilgili olmayan kavramların bulunduğu sayfalara atamanın bir yolu belirlenmelidir. Her bir kavram en azından sorgudaki başka bir kavramla alakalıysa; bu, olası tüm yayılım oranlarını, grafik altındaki bağlı bileşenlerin bir koleksiyonunu, sayfa alt-grafiği "s" sorgusuna göre dikkate almakla eşdeğerdir.

7. ÖRNEK OLAY İNCELEMESİ

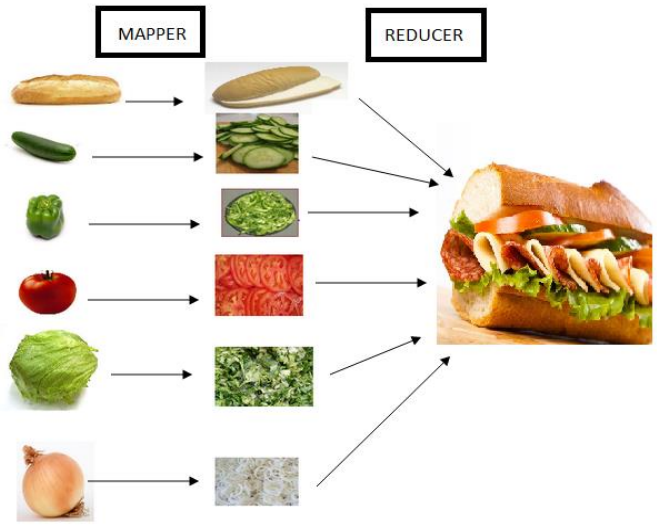
MSS'in önerilmesinin nedeni, “InFood” veya “FoodWiki” gibi gıda odaklı mobil uygulamaların yanısıra kalp hastalığı uygulamaları, diyet uygulamaları vs. için Hadoop kullanan kapsamlı bir bilgi tabanlı arama hizmeti sunmasıdır. Ayrıca, "FoodWiki sistemi, müşterilerin farklı risk grupları için uygun olmayan ambalajlı gıdaların içindeki yan etkili bileşenleri ya da gıda katkı maddelerini incelemek için bir web hizmeti olarak mobil uygulama arayüzünü kullanmalarını sağlayacaktır [11].” Bu uygulamaların sayısı artmaktadır; ancak gıda güvenliği için çok kapsamlı bir veri tabanı eksikliği olması, bilgi tabanlı ve bu uygulamalarda daha anlamlı sorgulamalar sunan, aranabilir, ölçeklenebilir ve güçlü bir veri tabanı olması MSS'in önerilmesini sağlamıştır.

Girdi, bölme, haritalama, indirgeme ve çıktıdan oluşan MR iş akışı Şekil 12'de gösterilmektedir. Öncelikle Girdi (input) yapıldıktan sonra yığınlara (chunk) bölünmektedir. Daha sonra yığınlardan haritalama (map) işlemine geçilir ve işler burada ayrıştırılır. Sonuçta kalıplarlara (pattern) indirgenmiş olan veri çıktı olarak (output) karşımıza çıkar.



Şekil 12. MapReduce (MR) Akışı.

Haritalama, İndirgeme Algoritması Şekil 13'te temsili olarak gösterilmektedir. Veriler haritalanıp indirgeme işleminden sonra anlamlı hale gelip kullanıcya ulaşır.



Şekil 13. MapReduce (MR) temsili.

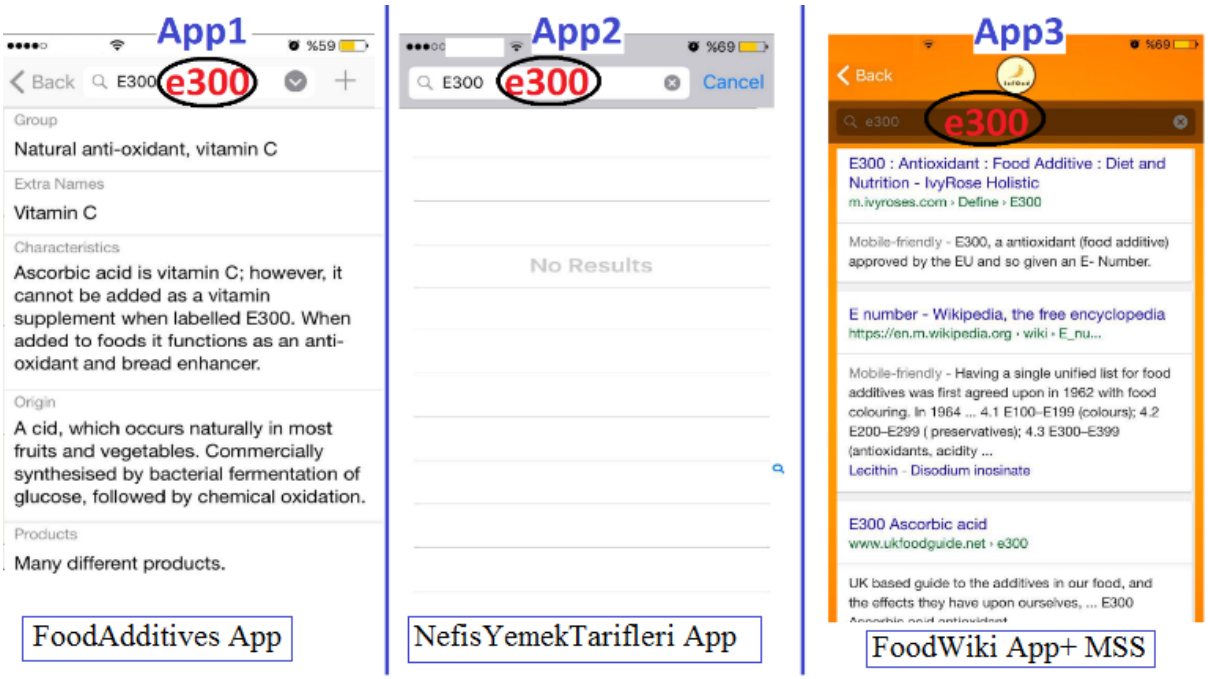
7.1. Değerlendirme

Bu çalışmada indeksleme ve kataloglama için gıda ile ilgili veriyi geliştirmek adına yapılan uğraşıların ardından, arka planda bir süreç olarak çalışabilen, mobil uygulamalar için hizmet sağlayıcı olan MSS'in kullanımı sunulmaktadır. Özetle birçok araçtan yararlanılmıştır. MR programlama çerçevesinin kullanılmasının başlıca nedeni, bağlantılı verileri işlemek için görevleri hızlandırılmasını sağlamasıdır. Yukarıda belirtildiği gibi, Hadoop platformu kısa sürede büyük miktarda veriyi işleyebilir. MSS, Hadoop ve MR olmadan kolayca erişilemeyen birçok bilgi işlem düğümü arasında paylaşılacak verilere ihtiyaç duyar. Değerlendirme incelemesi için odaklanılan noktalardan biri bağlantı düzeyidir. Geleneksel arama motorlarında, bağlantılılık basit bir adımdan başka bir şey değildir. Örneğin; eğer "A" ve "B" aranıyor ve belgede tam olarak "A" ve "B" terimleri bulunuyorsa, bu bağlantılı anlamına gelir. Oysaki akıllı arama motorları, semantik tonları da dikkate alarak, arayan kişilerin sorgularıyla sonuçları eşleştirir. Bağlantıyı etkileyen yüzlerce faktör olduğu için, "A" ile "B"nin eşleşmesinden daha fazlasını vermektedir. Semantik arama sayesinde, artık arama motorları sorgulara daha akıllıca yanıtlar verebilmektedir.

MSS, güvenli gıda ile ilgili her türlü arama isteğini bulabilmektedir. Şekil 14'te, üç farklı mobil uygulamada aynı arama isteği "E300" gösterilmektedir. Bu örnekte gösterilen uygulamaların karşılaştırılma nedeni MSS'in diğer arama motorlarına göre daha anlamlı ve geniş sonuçlara erişiyor olmasıdır. Açıkça görüldüğü gibi, sonuçlar birbirinden farklıdır. App1 (Foodadditives App) gıda katkı maddeleri için geliştirilen ve halkın kullanımına sunulan bir mobil uygulamadır. Sorgu yapan kullanıcıya gıda katkı maddeleri hakkında bilgi sağlamak için uygulama kendi veri tabanından faydalanmaktadır. Ancak MSS gibi bilgi tabanlı bir web servisi hizmetinden herhangi bir şekilde yararlanmadığından kullanıcıya döndürülen sonuçlar sınırlı olabilmektedir.

App2 ise sadece yemek tarifleri üzerine odaklanan bir mobil uygulamadır. MSS'ten yararlanmayan bu uygulama gıda katkı maddeleri hakkında herhangi bir bilgiye ulaşamamaktadır (bkz. Şekil 14, App2). App3 ise gıda katkı maddeleri hakkında arama yapan bir mobil uygulama olan FoodWiki uygulamasıdır. Bu uygulama MSS bilgi tabanlı web servisi hizmetinden yararlanmaktadır. MSS FoodWiki uygulaması arkasında bir fonksiyon olarak işlev görmektedir. Gıda katkı maddeleri ile ilgili gönderilen sorgularda mesela E300, kullanıcıya anlamlı sonuçlar gösterilecektir. FoodWiki kendine gelen anahtar kelimeleri

doğrudan, bilgi tabanlı bir web servisi hizmeti olan MSS'e göndermektedir. MSS ise Şekil 3' teki çalışma prensibine göre kendi bünyesinde bulunan tüm ilgili web sayfalarını FoodWiki uygulamasına yeni bir arayüz üzerinden göndermektedir. Sonuç olarak FoodWiki gibi uygulamalar MSS gibi web servisi hizmetlerini kullanarak bilgiyi doğrudan alıcılara sunmak gayesini taşırlar.



Şekil 14. Üç mobil uygulamanın karşılaştırılması.

8. SONUÇ

Bu makalede mobil uygulamalar için bir gıda tüketim arama hizmeti olan MSS tanımlanmıştır. Tasarımı, gelişimi ve kullanımı vurgulanmıştır. Örnek olay incelemesinin bulguları gösterilmiştir. MSS incelemesi sonucunda MSS'teki arama terimlerinin ve verinin çoğunun mobil uygulama geliştiricileri tarafından aranan güvenli gıda, gıda katkı maddeleri ve zararlı gıdalar ile ilgili olduğunu göstermiştir. MSS'in yapılandırılmış verileri içeren çok büyük bir veritabanı sağlayarak e-sağlık mobil uygulaması için oldukça yararlı olduğu düşünülmektedir. MSS tarafından çıkarılan sonuçların oldukça adil ve tarafsız olduğu öne sürülebilir. MSS'in gizliliği önemseyen, bilinçli bir arama hizmeti olduğu vaat edilmektedir. MSS; insanları kandırmaya yönelik bir sakınca, reklam, korsan dağıtım ya da spam bağlantılar içermez. Gelecekte MSS'i geliştirmek için kullanıcı davranışlarının analiz edilmesi ve buna bağlı gerekli adımların atılması muhakkaktır. "Standart Web arama hizmetleri yararlı olsa da ideallikten uzaktır [12]". MSS'in Hadoop'un da gücüyle e-sağlık konusunda en iyi olması umut vericidir. İnsanlar, gıda ve gıda katkı maddelerinin içeriğini merak ettiklerinde bu gibi e-sağlık mobil uygulamalarını kullanmaktadır.

Bu çalışmada Hadoop ve MR'ın büyük ölçekli bilgi alımı için nasıl kullanılacağı gösterilmiştir. MSS'in öncülük ettiği yeni bir yaklaşım ile veriyi analiz etme, işleme ve kullanıcıya döndürme, hedefi ortaya konulmaktadır. Bundan sonra yapılacak çalışmalarda, dikey arama motorları ve diğer gıda tabanlı uygulamalar için akıllı sorgular yapan, kullanıcıya anlamlı sonuçlar veren ve Hadoop mimarisinden yararlanan bir arama hizmetinin diğer bütün çalışmalar için ön ayak olması hedeflenmiştir. Bu yönüyle

literatürdeki diğer çalışmalardan ayrılmaktadır. MSS bu özelliklerinden ötürü içerik temelli yöntemler kullanıldığı için ileriki çalışmalarda umut ışığı anlamına gelmektedir.

KAYNAKÇA

[1] Doğan, Mustafa. Büyük Veri'nin kişiler ve kurumlar üzerindeki etkileri. Diss. İstanbul Bilgi Üniversitesi, 2014.
[2] Suel, T., et al. ODISSEA: A Peer-to-Peer Architecture for Scalable Web Search and Information Retrieval. In WebDB. 2003.
[3] Wu, X., et al., Data mining with big data. Knowledge and Data Engineering, IEEE Transactions on, 2014. 26(1): p. 97-107.
[4] Fitzgerald, J., et al., Validated designs for object-oriented systems. 2005: Springer Science & Business Media.
[5] Ganesh, S., et al. Ontology-based web crawler. In Information Technology: Coding and Computing, 2004. Proceedings. ITCC 2004. International Conference on. 2004. IEEE.
[6] Huang, Eric H., et al. "Improving word representations via global context and multiple word prototypes." Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1. Association for Computational Linguistics, 2012.
[7] Brin, S. and L. Page, Reprint of: The anatomy of a large-scale hypertextual web search engine. Computer networks, 2012. 56(18): p. 3825-3833.
[8] Kuchiki, A. and H. Tsukada, Flowchart Approach to Industrial Cluster Policy: Guangzhou's Automobile Industry Cluster, In The Flowchart Approach to Industrial Cluster Policy. 2008, Springer. pp. 41-70.
[9] Ciganovic-Jankovic, D., T. Banek, and D. Milicic, Link analysis algorithms (HITS and PageRank) in the information retrieval context. Text Analysis and Retrieval 2014: p. 24.
[10] Cho, J., H. Garcia-Molina, and L. Page, Efficient crawling through URL ordering. 1998.
[11] Ertuğrul, D.Ç., FoodWiki: a Mobile App Examines Side Effects of Food Additives Via Semantic Web. Journal of medical systems, 2016. 40(2): pp. 1-15.
[12] Jansen, B.J., A. Spink, and T. Saracevic, Real life, real users, and real needs: a study and analysis of user queries on the web. Information processing & management, 2000. 36(2): p. 207-227.