GAZİOSMANPAŞA ÜNİVERSİTESİNİN KURUMSAL WEB SAYFASI ZİYARETLERİNİN WEB MADENCİLİĞİ İLE ANALİZİ

2011
YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ

Turgut ÖZSEVEN

GAZİOSMANPAŞA ÜNİVERSİTESİNİN KURUMSAL WEB SAYFASI ZİYARETLERİNİN WEB MADENCİLİĞİ İLE ANALİZİ

Turgut ÖZSEVEN

Karabük Üniversitesi

Fen Bilimleri Enstitüsü

Bilgisayar Mühendisliği Anabilim Dalında

Yüksek Lisans Tezi

Olarak Hazırlanmıştır

KARABÜK

Şubat 2011

Turgut ÖZSEVEN tarafından hazırlanan "GAZİOSMANPAŞA ÜNİVERSİTESİNİN KURUMSAL WEB SAYFASI ZİYARETLERİNİN WEB MADENCİLİĞİ İLE ANALİZİ" başlıklı bu tezin Yüksek Lisans Tezi olarak uygun olduğunu onaylarım.

Yrd.Doç. Dr. Muharrem DÜĞENCİ
Tez Danışmanı, Bilgisayar Mühendisliği Anabilim Dalı

Bu çalışma, jürimiz tarafından oy birliği ile Bilgisayar Mühendisliği Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir. 03/02/ 2011

Ünvanı, Adı SOYADI (Kurumu)	İmzası
Başkan : Prof. Dr. Abdullah ÇAVUŞOĞLU (KBÜ)	.....
Üye : Yrd. Doç. Dr. Muharrem DÜĞENCİ (KBÜ)	.....
Üye : Yrd. Doç. Dr. Resul KARA (DÜ)	.....

...../..../2011

KBÜ Fen Bilimleri Enstitüsü Yönetim Kurulu, bu tez ile Yüksek Lisans derecesini onamıştır.

Doç. Dr. Nizamettin KAHRAMAN
Fen Bilimleri Enstitüsü Müdürü

"Bu tezdeki tüm bilgilerin akademik kurallara ve etik ilkelere uygun olarak elde edildiğini ve sunulduğunu; ayrıca bu kuralların ve ilkelerin gerektirdiği şekilde, bu çalışmadan kaynaklanmayan bütün atıfları yaptığımı beyan ederim."

Turgut ÖZSEVEN

ÖZET

Yüksek Lisans Tezi

GAZİOSMANPAŞA ÜNİVERSİTESİNİN KURUMSAL WEB SAYFASI ZİYARETLERİNİN WEB MADENCİLİĞİ İLE ANALİZİ

Turgut ÖZSEVEN

Karabük Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Tez Danışmanı:
Yrd. Doç. Dr. Muharrem DÜĞENCİ
Şubat 2011, 111 sayfa

Günümüzde internet yaşamın her aşamasında kullanılan önemli bir bilgi kaynağıdır. Ayrıca internet üzerinde kullanıcılar ve web sayfası sahipleri için keşfedilmeyi bekleyen önemli bilgiler bulunmaktadır. Web madenciliği internet üzerinde bulunan verilerin veri madenciliği teknikleri ile analiz edilerek önemli bilgilerin keşfedilmesini sağlar. Web madenciliği, web sitelerini ziyaret eden kullanıcıların davranışlarını inceleyerek web sitelerinin güncellenmesi veya geliştirilmesi, müşterilerin ilgi alanları, reklam alma, pazarlama stratejileri oluşturma, sayfa kullanım dağılımlarını belirleme gibi birçok konuda karar verilmesini sağlayan bilgileri sunar.

Bu çalışmada, Gaziosmanpaşa Üniversitesi web sayfasına ait altı aylık erişim kayıtlarının web kullanım madenciliği ile analiz edilmesi için web kullanım madenciliğinin tüm aşamalarını kapsayan yazılım oluşturulması, oluşturulan yazılım

ile web sayfasına ait birlikte ziyaret edilen sayfaların tespit edilmesi ve web sitesinin kullanımına ait çeşitli istatistiki bilgilerin elde edilmesi amaçlanmıştır. Bu bilgiler site yöneticilerine sitenin güncelleme ve tasarım aşamalarında karar vermesine yardımcı olacaktır.

Anahtar Sözcükler : Veri madenciliği, web madenciliği, web kullanım madenciliği, birliktelik kuralları, web log madenciliği.

Bilim Kodu : 902.1.014

ABSTRACT

M.Sc. Thesis

ANALYZING GAZIOSMANPASA UNIVERSITY VISITED WEB SITE WITH WEB MINING

Turgut OZSEVEN

Karabük University
Graduate School of Natural and Applied Sciences
Department of Computer Engineering

Thesis Advisor:
Asst. Prof. Dr. Muharrem DUGENCI
February 2011, 111 pages

Today, internet is an important source of information used in every stage of life. In addition, users and owners of the web page for important information on the internet are waiting to be discovered. Web mining allows the discovery of important information by analyzing data on the internet with data mining techniques. Web mining offers information that allows deciding many issues such as update or development of web sites, interests of customers, ad-making, creating marketing strategies, distribution of page usage by examing the behavior of websites visited by users.

In this study, a software is creation the covering all phases of web usage mining in that six-month user access logs belonging to the web site of Gaziosmanpasa University were analyzed with web usage method, with created software was aimed to be determine together pages visited by the web site and a variety of statistical

information retrievel on how to use the web site. This information will help you decide of the web site administrators in update and design stages of web sites.

Key Words : Data mining, web mining, web usage mining, association rules, web log mining.

Science Code : 902.1.014

TEŞEKKÜR

Bu tez çalışmasının planlanmasında, yürütülmesinde ve hazırlanmasında ilgi ve desteğini esirgemeyen, bana yol gösteren danışman hocam Yrd. Doç. Dr. Muharrem DÜĞENCİ 'ye içtenlikle teşekkür ederim.

Çalışmalarda kullanılan verileri sağlayan Gaziosmanpaşa Üniversitesi Bilgi İşlem Daire Başkanlığı yönetici ve personellerine ve Bilgi İşlem Daire Başkanlığı internet grubu personeli Alperen DÜN 'e teşekkürü bir borç bilirim.

Sevgili aileme manevi hiçbir yardımı esirgemeden yanımda oldukları için tüm kalbimle teşekkür ederim.

İÇİNDEKİLER

Sayfa

Sayfa

Sayfa

Sayfa

ŞEKİLLER DİZİNİ

Sayfa

Sayfa

Sayfa

ÇİZELGELER DİZİNİ

Sayfa

SİMGELER VE KISALTMALAR DİZİNİ

SİMGELER

$\|D\|$	: yapılan tüm alışverişlerin sayısı
h-1	: oturum süresi temelli sezgisel yaklaşım
h-2	: sayfada kalma süresi temelli sezgisel yaklaşım
h-ref	: referans temelli sezgisel yaklaşım
p	: sayfa isteği
q	: p isteğinden sonraki sayfa isteği
θ	: eşik değeri
S	: oturum
t_0	: oturum başlangıç zamanı
t_1	: bir sonraki sayfaya geçiş zamanı
t_n	: oturum bitiş zamanı
Δ	: bekleme süresi
t_p	: p isteğinin gerçekleşme zamanı
t_q	: q isteğinin gerçekleşme zamanı
X	: değişken
Y	: değişken
$\|X\|$	: X ürünü içeren alışverişlerin sayısı
$\|X.Y\|$	: X ve Y ürünlerini içeren destek
$X \Rightarrow Y$	: kural

KISALTMALAR

CLF	: Common Log Format
CRISP-DM	: Cross Industry Standard Process for Data Mining:
ECLF	: Extended Common Log Format
HTML	: Hyper Text Markup Language
HTTP	: Hyper Text Transfer Protocol
IIS	: Internet Information Server
KDD	: Knowledge Discovery in Databases (Veritabanlarında Bilgi Keşfi)
NCSA	: National Center for Supercomputing Applications
OLAP	: Online Analytical Processing
OLTP	: Online Transaction Processing
SQL	: Structured Query Language (Yapısal Sorgulama Dili)
VA	: Veri Ambarı
VM	: Veri Madenciliği
VTYS	: Veritabanı yönetim sistemleri
WWW	: World Wide Web
XML	: eXtensible Markub Language

BÖLÜM 1

GİRİŞ

Teknolojinin gelişmesi ile birlikte işlem gören bilgi miktarı artmış ve her geçen gün artmaya devam etmektedir. Bu artış sahibi olan kurum veya kuruluş için depolama gibi ek sorunlar getirmesine karşın herhangi bir katkı sağlamamaktadır. Oluşan bu büyük veri yığınlarının anlamlı hale getirilmesi, karar mekanizmalarında kullanılması kısacası sahibi olan kurum veya kuruluş için katkı sağlaması için bu verilerin incelenmesi, analiz edilmesi, önemli örüntülerin keşfedilmesi yani veri madenciliği gerekmektedir.

Veri madenciliği (VM) büyük miktardaki veriden anlamlı bilgilerin çıkartılmasını amaçlamaktadır.

VM'nin bir diğer uygulama alanı da WWW (world wide web) üzerinde bulunan verilerdir. WWW üzerinde bulunan veriler üzerinde işlem yapan VM yöntemi web madenciliği olarak adlandırılır.

Web madenciliği, WWW üzerinden kullanışlı bilgiyi keşfetme ve analiz etme işlemi şeklinde tanımlanır. Bu tanım, kullanılan veri ve uygulama alanına göre ilk olarak web içerik madenciliği ve web kullanım madenciliği olmak üzere iki gruba ayrılmıştır. Web içerik madenciliği, temel olarak internette saklı bilgiyi bulma üzerine yoğunlaşmıştır. Web'deki metin, görsel, ses, video ve çevrimiçi veritabanlarından verilerin otomatik olarak aranması ve elde edilmesi işlemidir. Web kullanım madenciliği, ziyaretçi trafik bilgilerinin web sunucu veya vekil sunucu log dosyalarından yararlanılarak raporlanmasını sağlamaktadır. Bu iki web madenciliği tekniğine daha sonradan web yapı madenciliği de eklenmiştir. Web yapı madenciliği, web sitesi ve sayfalarının yapısal olarak özelliklerini belirler ve sitenin yapısal tasarımını iyileştirmek için kullanılır.

Bu tez çalışmasının amacı, web kullanım madenciliği teknikleri ile yapılan çalışmaları incelemek ve Gaziosmanpaşa Üniversitesi altı aylık erişim kayıtları incelerek sitenin analizini yapmaktır. Kullanılacak veriler http://www.gop.edu.tr adresine ait altı aylık sunucu erişim kayıtlarıdır. Web kullanım madenciliğinin ilk aşaması olan ön işlem aşamasını veriler üzerinde gerçekleştirmek için NET tabanlı yazılım geliştirilmiştir.

Bu tez çalışması altı bölümden oluşmaktadır. İkinci bölümde veri madenciliği, üçüncü bölümde web madenciliği, dördüncü bölümde web kullanım madenciliği, beşinci bölümde Gaziosmanpaşa Üniversitesi web sitesinin analiz uygulaması ve son olarak altıncı bölümde sonuçlar ve öneriler sunulmuştur.

BÖLÜM 2

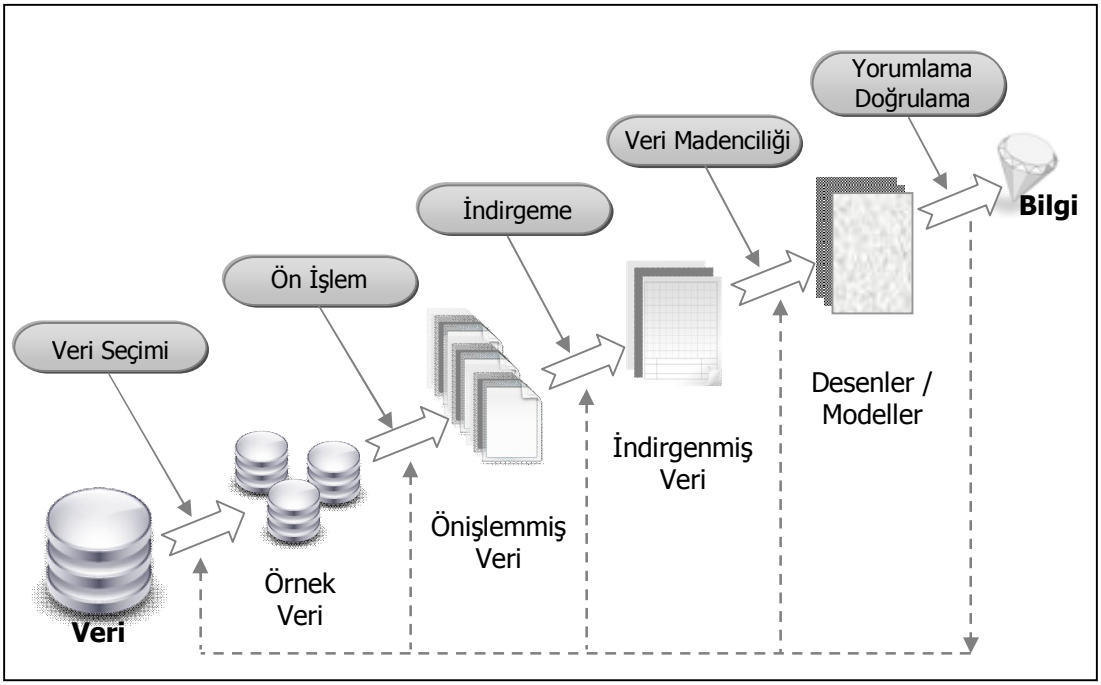
VERİ MADENCİLİĞİ

Teknolojinin gelişmesi ve ucuzlamasıyla birlikte işlem gören veya elde edilen verilerin saklanması kolaylaşmıştır. Saklanan veri bir alışveriş merkezi, kredi kartı kullanım ve telefon kayıtları gibi insan faktörü olan veriler olabileceği gibi moleküler veritabanları veya tıbbi kayıtlar olabilir. Bu veriler anlamlandırılmadan sadece depolandığı sürece verinin sahibi olduğu kurum veya kuruluş için çok fazla faydası bulunmamaktadır. Verilerin işlenmesi, analiz edilmesi ve anlamlandırılması geleneksel veritabanı ve sorgulama yaklaşımıyla mümkün olmamakta, yeni tekniklere ve araçlara ihtiyaç duyulmaktadır.

Depolanan veri miktarının artması ve gittikçe karmaşıklaşması verilerin anlamlandırılmasını zorlaştırmaktadır. VM ve veritabanlarında bilgi keşfi bu sorunun ortadan kaldırılmasını amaçlamaktadır. Verilerin anlamlandırılması veri sahibi için kullanışlı bilgiler elde edilmesini ve karar süreçlerinin kısaltılmasını sağlayacaktır.

Başka bir ifadeyle VM, büyük miktarda veri içinden gelecekle ilgili tahmin yapmamızı sağlayacak bağıntı ve kuralların bilgisayar programları kullanarak aranmasıdır (Ian and Eibe, 2005).

VM ve veritabanlarından bilgi keşfi kavramları birbirine karıştırılmamalıdır. Veritabanlarında bilgi keşfi kavramı ilk olarak 1995 yılında Montreal'de KDD (Knowledge Discovery in Databases) konferansında, veriden bilgi elde edilmesi için gerekli olan tüm süreci ifade etmek amacıyla kullanılmıştır. VM ise, bu süreçteki önemli adımlardan bir tanesidir (Gürsoy, 2009). KDD sürecinde yer alan adımlar Şekil 2.1'de gösterilmiştir.



Şekil 2.1. KDD sürecinde yer alan adımlar (Fayyad et al., 1996).

Veri seçimi adımı seçim ölçütlerinin belirtildiği adımdır. Veri kümesi birbirleriyle ilişkili verileri barındırmasına rağmen, içerdiği verilerin önemli olan bölümlerinin seçilmesi gerekir. Bu adım birkaç veri kümesini birleştirerek, sorguya uygun örnek veri kümesi elde etmeyi sağlar. Örneğin, bir siteye ait ziyaret kayıtlarından son iki aylık kısmının alınması mevcut verilerden seçim yapılmasıdır.

Elde edilen veri yanlış girişler, eksik veya geçersiz veri değerleri içerebilir. Veri temizleme ve ön işlem aşaması veriden geçersiz değerlerin yani gürültülerin silinmesidir. Veriye bağlı olarak KDD sürecinin en uzun süren adımı olabilir (Susan, 2001). Gürültü içermeyen ancak sorgu için gereksiz olan ve sorguyu yavaşlatan verilerin temizlenmesi de sağlanır. Örneğin, bir veri kümesi içerisinde evlilik tarihi ve evlilik süresi aynı anda bulunuyorsa bu verilerden birisi temizlenebilir. Ayrıca veri tutarlılığı da sağlanmalıdır. Belirlenen biçime uymayan değerler biçimlendirilmelidir. Örneğin, bir veri kümesi içerisinde cinsiyet için E ve K kısaltması kullanılırsa, veri içerisinde 0 ve 1 şeklindeki girişler E ve K şeklinde düzenlenmelidir.

İndirgeme adımı kullanılacak verideki bilgi türlerini azaltmak için kullanılır. Örnek veriden ilgisiz niteliklerin atıldığı ve tekrarlı verilerin ayıklandığı adımdır. Bu aşama ile seçilen VM sorgusunun çalışma zamanı iyileştirilir.

VM, KDD sürecindeki bir adımdır ve örnek veri bu adımda kullanılır. İlgili desenler için gerçek arama bu adımda gerçekleştirilir ve VM görevi için uygun algoritmaya karar verilir. VM işi sınıflandırma, lineer regresyon veya kümeleme analizi olabilir (Susan, 2001).

Yorumlama adımı sonuç raporlarının içerdiği bilgiyi yorumlar. KDD süreci başlamadan önce beklenen bilgi ile yeni bilgi uyumsuz olabileceği için keşfedilen bilgi ile olası uyumsuzlukları yorumlama ve değerlendirme adımı çözebilir. Elde edilen bilgi, insanların karar vermesine yardımcı olacak ve daha sonra kullanılabilecek şekilde düzenlenir. Gerekli yorumlama kullanıcıyı tatmin ettiği zaman bilgi raporlanır (Susan, 2001).

VM çok büyük veri yığınlarından kritik bilgileri elde etmeyi sağlar. Böylelikle normal şartlar altında uzun zaman süren araştırmalarla doğruluğu kesin olmayacak şekilde elde edilen bilgi VM ile kısa sürede ve kesin olarak elde edilir. Elde edilen bu bilgi objektif değerlendirmeler yapılmasında ya da stratejik kararlar almada kullanılır. Bu bilgiler kurumsal veri kaynaklarının iyi analiz edilmesine ve iş dünyasındaki yaklaşımlara ilişkin tahminlerde bulunulmasına yardımcı olur. Kısaca VM sayesinde şirketler stratejik adımlar atarken çok büyük veri yığınları arasından kendilerine yol gösterecek kritik verileri ayıklayarak analiz edebilir (Alpaydın, 2000).

VM'nin yoğun olarak kullanılmasının temel sebebi çok fazla miktardaki verinin istenildiği gibi kullanılabilmesidir. VM'de kullanılan yöntemler esasında uzun yıllardır istatistikte kullanılmaktadır. VM'nin getirmiş olduğu avantaj bilişim teknolojisini kullanarak yapılan analizlerin çok kısa sürede ve daha az maliyetle yapılabilmesidir (Özmen, 2001).

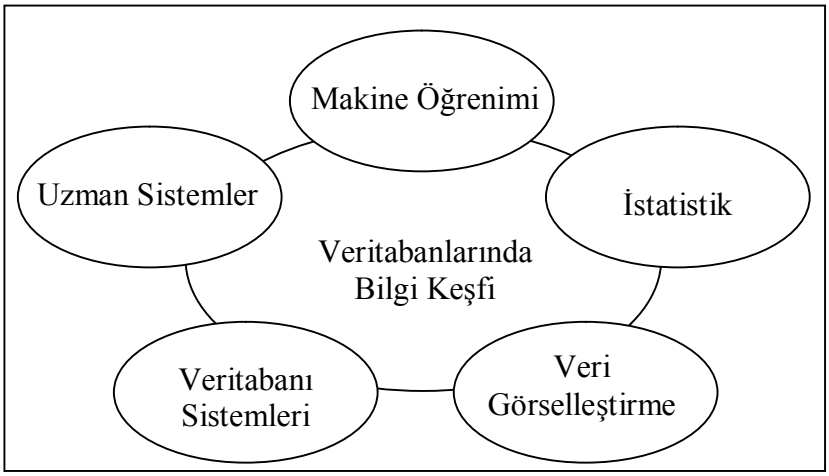
2.1. VERİ MADENCİLİĞİNİN TARİHÇESİ

VM teknikleri üzerine matematikçiler 1950'li yıllarda mantık ve bilgisayar bilimleri alanlarında çalışarak yapay zekâ ve makine öğrenmeyi oluşturmuşlardır. 1960'lı yıllarda regresyon analizi, en büyük olabilirlik kestirim, sinir ağları vb. metotlar VM'nin ilk adımlarını oluşturmuştur. Ayrıca veritabanı sistemleri gelişerek büyük sayıda metin dokümanlarının saklanması ve bilginin geri kazanılması sağlanmıştır.

1970 ve 1990'lı yıllar arasında yeni programlama dilleri ve yeni bilgisayar tekniklerinin geliştirilmesi genetik algoritmalar gibi algoritmaları da içermiştir.

1990 yılıyla beraber veritabanında bilgi keşfinin ilk adımları oluşturulmuş ve büyük veritabanları için veri ambarı geliştirilmiştir. Ayrıca, aynı zaman içerisinde yeni teknolojilerle beraber VM değiştirilerek yaygın olarak kullanılan standart bir işin parçası olmuştur (Kaya ve Köymen, 2008).

KDD, yeni bir teknik değildir. Araştırmanın birden fazla disiplin kullanılarak yapılmasıdır. Makine öğrenimi, istatistik, veritabanı sistemleri, uzman sistemler ve verilerin görüntülenmesi işlemlerinin hepsinin birlikte kullanıldığı bir tekniktir (Gürsoy, 2009). Amaç, büyük miktarda veri içerisindeki düşük seviye verilerden yüksek seviye bilgiler çıkarmaktır (Fayyad et al., 1996).



Şekil 2.2. KDD sürecinin içerdiği disiplinler.

Veritabanı sistemleri, günlük işlem gören verilerin elektronik olarak depolanması için sürekli olarak kullanılmaktadır. VM için kaynak teşkil eden veriler çoğu zaman veritabanı sistemlerinden temin edilmektedir.

İstatistik ve VM verinin yapısını keşfetmeyi amaçlayan iki disiplindir. Örtüşen amaçlar söz konusu olduğu için VM, istatistiğin bir alt dalıymış gibi düşünülmektedir. VM ile istatistiği birbirinden ayıran en belirgin özellik, VM'nin veritabanı sistemleri ve makine öğrenimi gibi pek çok alanla ilişkisinin olmasıdır (Gürsoy, 2009).

Makine öğrenimi, bilgisayarların veri türlerine dayalı öğreniminin sağlanmasıdır. Makine öğrenimi örnekler yardımıyla sağlanmaktadır. Öğrenilen bilgiler ile benzer olaylar yorumlanarak kararlar verilir veya problemler çözülür.

VM'de girdi ve çıktı kolay, anlaşılır ve kullanıma uygun olmalıdır. Görselleştirme teknikleri verideki dağılımları, örüntüleri, kümeleri ve sınır dışı öğelerin gücünü daha çekici ve etkin bir hale getirebilmek amacıyla kullanılır. Bunu için grafikler, veri dağılım haritaları, eğriler, üç boyutlu şekiller ve yüzeyler kullanılır (Gürsoy, 2009).

VM'ye ilginin artması aşağıdaki faktörlerle açıklanabilir (Adriaans and Zantinge, 1996);

- 1980'lerde şirketler sahip oldukları müşteriler, rakipleri ve ürünleri hakkında veriler içeren veritabanları oluşturmuşlardır. Potansiyel bir altın madeni gibi olan bu veritabanları sayısı milyonları geçen veriler ve gizli bilgiler içerirler. Bu verilere SQL (Structured Query Language) veritabanı sorgulama dili ya da başka yüzeysel sorgulama dilleri kullanılarak kolaylıkla erişilebilir. SQL mevcut veriler içerisinde belirlenen sınırlamalar içeren sorgulamalar yapmaya yardımcı olur. VM algoritmaları ise tipik olarak, veritabanının alt gruplarında veya belirlenen veri kümelerinde belirginleşir. Çoğu kez tekrarlanabilen SQL sorguları kullanılır ve ortalama sonuçlar elde edilir.

- Bilgisayarlarda ağ kullanımı gelişmeye devam etmektedir. Bu durumda veritabanı ile bağlantı kurmak kolaylaşır. Böylece demografik verili dosya ile müşteri dosyası arasında bağlantı kurulabilir ve belirli popülasyon gruplarının kimliklerinin belirlenmesi sağlanabilir.
- Son birkaç yılda makine öğrenimi teknikleri oldukça gelişmiştir. Sinir ağları, genetik algoritmalar ve diğer basit uygulanabilir öğrenme teknikleri veritabanlarıyla bağlantılar kurmayı kolaylaştırır.
- Müşteri ile hizmet veren arasındaki ilişki, kişisel bilgileri hizmet verenin masasındaki bilgisayardan merkezi bilgi sistemlerine gönderir. Pazarlamacılar ve sigortacılar da bu yeni kazanılan teknikleri kullanmak isterler.

2.2. VERİ MADENCİLİĞİNİN AMACI

Otomatik veri toplama araçları ve veri tabanı teknolojilerindeki gelişme, veritabanlarında, veri ambarlarında ve diğer bilgi depolarında çok miktarda bilgi depolanması sonucunu doğurmuştur. Çok fazla veri var, ancak bilgi yok. Veri ambarları içindeki gizli örüntüler geleneksel çözümleme araçlarıyla bulunamaz. Toplanan veri miktarı büyüdükçe ve toplanan verilerdeki karmaşıklık arttıkça, daha iyi çözümleme tekniklerine olan gereksinim de artmaktadır. Bu tür bilgiler, KDD ya da VM olarak bilinen teknikler yardımıyla çözümlenebilir.

VTYS (Veritabanı yönetim sistemleri) büyük miktardaki verilerin saklanmasını ve istenilen şekilde erişimi sağlamaktadır. VTYS belirlenen özelliklerde verilerin elde edilmesi veya veritabanına girilmesi için kullanılmaktadır. Bu yüzden veritabanlarında var olan bilgilerin yorumlanması ve yeni bilgi keşfi görevini yerine getiremez. VTYS veritabanın içerdiği gereksiz verilerin temizlenmesi, örnekleme yapılması ve transfer gibi işlemleri gerçekleştirerek verinin VM'ye uygun hale getirilmesini sağlar.

2.3. VERİ MADENCİLİĞİ KULLANIM ALANLARI

VM'nin temel amacı büyük bir veri topluluğu içerisinden anlamlı bilgiler elde etmek ve bu bilgileri karar almada kullanmaktır. Bu yüzden, çoğunlukla işletmeler müşterilerinin davranışlarını tahmin etmek için kullanmaktadır. VM pazarlama, perakendecilik, bankacılık, telekomünikasyon, tıp ve daha birçok alanda kullanılmaktadır.

Aşağıda veri madenciliğinin uygulama alanlarına göre kullanım şekillerine örnekler verilmiştir.

Perakendecilik alanındaki uygulamalar;

- Satış terminalleri ve kodlama sistemleri sayesinde toplanan verilerin analizi yapılarak rekabet ortamında avantaj elde etmek için kullanılabilir (Gürsoy, 2009).
- Satın alma ve stok yönetimi uygulamalarında kullanılabilir (Gürsoy, 2009).
- Pazar sepeti analizlerinde kullanılabilir (Silahtaroğlu, 2008).
- Mevcut müşterilerin elde tutulması, yeni müşterilerin kazanılması için kullanılabilir (Silahtaroğlu, 2008).

Telekomünikasyon alanındaki uygulamalar;

- Kaybedilecek müşteriler daha önceden belirlenerek bu müşterileri elde tutma amaçlı kampanyalar düzenlenebilir (Gürsoy, 2009).
- İletişim hatlarının hangi dönem ve aralıklarda yoğun olarak kullanıldığı belirlenebilir.
- Müşterilerin iletişim zaman dilimleri ve iletişim kurduğu diğer hatlar belirlenerek müşterilere yönelik kampanyalar düzenlemek için kullanılabilir.
- Servis kalitesinin arttırılmasında kullanılabilir.
- Müşteri kaybına neden olan faktörleri belirlemek için kullanılabilir.

Web alanındaki uygulamalar;

- Kullanıcıların profilleri çıkarılabilir ve zaman içindeki değişimleri takip edilebilir, sitedeki beğenilen ya da beğenilmeyen köşeler tespit edilebilir (Bing, 2007).
- Müşterilerin çoğunlukla ilgilendiği ürünler tespit edilebilir.
- Site ziyaretçilerinin gezindiği sayfalar üzerinden çoğunlukla ne tür ziyaretçi geldiği belirlenebilir.
- Siteye çoğunlukla hangi kaynaktan ulaşıldığı tespit edilerek ilgili kaynaklara reklamlar verilebilir.
- Kullanıcıların gezinti şekli/hızı sitenin içerik, yapılandırma ve altyapı açısından performansı hakkında fikir verir (Bing, 2007).
- Kullanıcı profillerine uygun ürünlerin reklam kampanyaları en çok ziyaret ettikleri sayfalara koyulabilir (Güvenç, 2001).
- En sık beraber ziyaret edilen çift sayfalar belirlenebilir (Güvenç, 2001).
- Kötü niyetli kullanıcı istekleri belirlenip bunlara karşı alınması gereken önlemler belirlenebilir.

İşletme alanındaki uygulamalar;

- Bir işletme, kendi müşterisiyken rakibine giden müşterilerle ilgili analizler yaparak rakiplerini tercih eden müşterilerinin özelliklerini elde edebilir ve bundan yola çıkarak gelecek dönemlerde kaybetme olasılığı olan müşterilerin kimler olabileceği yolunda tahminlerde bulunarak onları kaybetmemek, kaybettiklerini geri kazanmak için strateji geliştirebilir.
- Ürün veya hizmette hangi özelliklerin ne derecede müşteri memnuniyetini etkilediği, hangi özelliklerinden dolayı müşterinin bunları tercih ettiği ortaya çıkarılabilir.
- Müşterilerin kredi riskleri hesaplanarak hangi müşterilerin kredi riskinin yüksek olduğu, hangi müşterilerin geri ödemesini zamanında yapmayacağı kestirilebilir.

- Kredi kartı ödemelerini aksatan, gecikmeli olarak yapan veya hiç yapmayanların özelliklerinden yola çıkılarak bundan sonra aynı duruma düşebilecek muhtemel kişiler saptanabilir.
- Ürün talebi bazında müşteri görünümlerini belirleyerek, müşteri segmentasyonuna gitmek ve çapraz satış olanakları yaratmakta kullanılabilir.
- Piyasada oluşabilecek değişikliklere mevcut müşteri portföyünün vereceği tepkinin firma üzerinde yaratabileceği etkinin tespitinde kullanılabilir.
- En kârlı mevcut müşteriler saptanarak, potansiyel müşteriler arasından en kârlı olabilecekler belirlenebilir. Kârlı müşteriler tespit edilerek onlara özel kampanyalar uygulanabilir. En masraflı müşteriler daha masrafsız müşteri haline dönüştürülebilir. Örneğin en çok bankacılık işlemi yapanlar ortaya çıkarılıp bunlar şube bankacılığı yerine daha masrafsız internet bankacılığına yönlendirilebilir.
- Bir ürün veya hizmetle ilgili bir kampanya programı oluşturmak için hedef kitlenin seçiminden başlayarak bunun hedef kitleye hangi kanallardan sunulacağı kararına kadar olan süreçte veri madenciliği kullanılabilir.
- Kurum teknik kaynaklarının en uygun şekilde kullanılmasını sağlamakta kullanılabilir.
- Geçmiş ve mevcut yapı analiz edilerek geleceğe yönelik tahminlerde bulunulabilir. Özellikle ciro, kârlılık, pazar payı, gibi analizlerde veri madenciliği çok rahat kullanılabilir.

Tıp alanındaki uygulamalar;

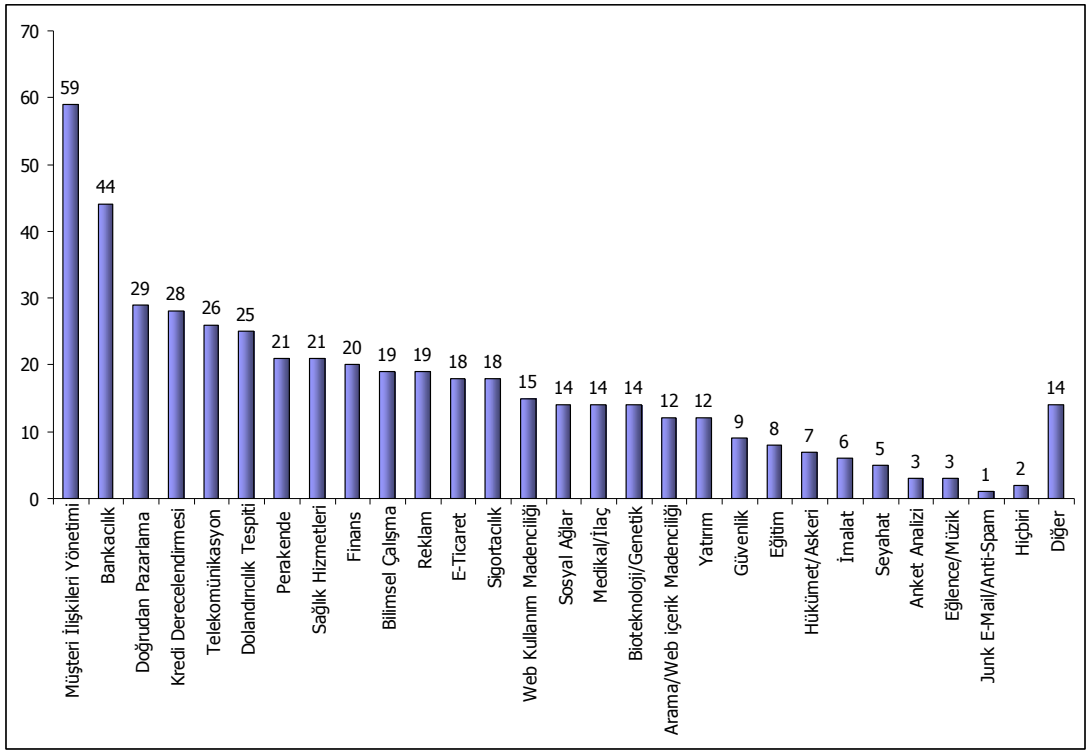
- Makine öğrenmesi sağlanarak çeşitli hastalıkların risk tahmininde kullanılabilir.
- Hastanedeki servislerin ve programların başarısının belirlenmesinde kullanılabilir (Gürsoy, 2009).
- Benzer hastalıklar için daha önce uygulanan tedavi süresine göre mevcut hasta için tedavi süresinin belirlenmesinde kullanılabilir.
- Ameliyat riski taşıyan ancak, ameliyat olması gerektiği tam olarak anlaşılamayan hasta ve hastalıklar için kullanılabilir (Silahtaroğlu, 2008).

- Magnetik rezonans verileri ile sinir sistemi bölge ilişkilerinin belirlenmesinde kullanılabilir.

KDNuggets tarafından 2009 yılında toplamda 180 oy veren arasında yapılan bir araştırma sonucuna göre veri madenciliğinin sektörler bazında kullanımına ilişkin sonuçlar Çizelge 2.1'de ve bu verilerin grafiksel gösterimi Şekil 2.3'de verilmiştir.

Çizelge 2.1. Veri madenciliğinin sektörler bazında kullanımı (http://www.kdnuggets.com/polls/2009/industries-data-mining-applications.htm, 2010).

180 Kişiden Toplam 486 oy			
Sektör	Oy	Sektör	Oy
Müşteri İlişkileri Yönetimi	59	Medikal/İlaç	14
Bankacılık	44	Bioteknoloji/Genetik	14
Doğrudan Pazarlama	29	Arama/Web içerik Madenciliği	12
Kredi Derecelendirmesi	28	Yatırım	12
Telekomünikasyon	26	Güvenlik	9
Dolandırıcılık Tespiti	25	Eğitim	8
Perakende	21	Hükümet/Askeri	7
Sağlık Hizmetleri	21	İmalat	6
Finans	20	Seyahat	5
Bilimsel Çalışma	19	Anket Analizi	3
Reklam	19	Eğlence/Müzik	3
E-Ticaret	18	Junk E-Mail/Anti-Spam	1
Sigortacılık	18	Hiçbiri	2
Web Kullanım Madenciliği	15	Diğer	14
Sosyal Ağlar	14		



Şekil 2.3. Veri madenciliğinin sektörler bazında kullanımı.

2.4. VERİ MADENCİLİĞİNDE KARŞILAŞILAN PROBLEMLER

Az miktardaki veriler üzerinde hatasız çalışan ve beklenen sonuçları veren bir veri madenciliği sisteminde, veri miktarı veya veri içerisindeki gürültü arttıkça sonuçlarda sapmalar meydana gelebilir.

Aşağıda VM sistemlerinin karşı karşıya olduğu problemler verilmiştir (Sever ve Oğuz, 2002).

2.4.1. Veritabanı Boyutu

VM sistemlerinin karşı karşıya olduğu en önemli sorunlardan biri veritabanı boyutunun çok büyük olmasıdır. Kullanılan VM yöntemi küçük veri kümelerini üzerinde oluşturulmuşsa fazla miktardaki verileri analiz ederken daha dikkatli olmak gerekir. Dolayısıyla VM yöntemleri ya sezgisel bir yaklaşımla arama uzayını taramalıdır ya da örnekleri yatay/dikey olarak indirgemelidir.

2.4.2. Gürültülü Veri

Büyük miktardaki veriler kullanıcı kaynaklı veya yanlış hesaplama sonucu oluşan hatalar içerebilir. Veri kümesi içerisinde bu şekilde bulunan ve istenmeyen veriler gürültülü veri olarak adlandırılır.

Kullanılan VTYS'ler kullanıcı veri girişi veya hesaplama sırasında yapılan hataları giderme özelliğine sahip olmadığı için bu tür veriler de veritabanı içerisinde bulunacaktır. VM yöntemi veri kümesi içerisinde bulunan gürültüleri tespit etmeli ve bu verileri analize dahil etmemelidir.

2.4.3. Boş Değerler

VM için kaynak teşkil eden veriler bir VTYS'den alınıyorsa veritabanı içerisinde birincil anahtar haricindeki niteliklerin boş değer içerebileceği her zaman hesaba katılmalıdır. Boş değer, herhangi bir değere sahip olmayan değerdir. Boşluk karakterinin boş değer olarak algılanmadığı unutulmamalıdır. Kullanılan veri kümesi boş değerler içeriyorsa ya boş değer ihmal edilmeli ya da olası en yakın değer atanmalıdır.

2.4.4. Eksik Veri

VM için kullanılacak veri kümesi bir gruba özel değil genele hitap edecek verileri içermelidir. Aksi durumda elde edilecek sonuçlar sadece belirli bir grup için geçerli olacaktır. Örneğin, hastalığın tanısını koymak için kurallar sadece çok yaşlı insanların belirtilerinin bulunduğu bir veri kümesi kullanılarak üretilseydi, bu kurallara dayanarak bir çocuğa tanı koymak pek doğru olmazdı.

2.4.5. Artık Veri

VM için kullanılacak veri kümesi, eldeki probleme uygun olmayan veya artık nitelikler içerebilir. Örneğin, eldeki problem ile ilgili veriyi elde etmek için iki ilişkiyi ortak nitelikler üzerinden birleştirirsek, sonuç ilişkide kullanıcının farkında

olmadığı artık nitelikler bulunur. Artık nitelikleri elemek için geliştirilmiş algoritmalar özellik seçimi olarak adlandırılır.

2.4.6. Dinamik Veri

Kurumsal çevrim içi veritabanları dinamiktir, yani içeriği sürekli olarak değişir. Bu durum, VM için önemli sakıncalar oluşturmaktadır. Bunlardan ilki, veritabanından bilgi okunduktan sonra analizi yapılır ve VM'ye bu şekilde devam edilirse mevcut veritabanı hem VM için hem de mevcut uygulama için kullanılacaktır. Buda ciddi performans düşüşüne neden olacaktır. Diğer bir sakınca ise, veritabanında bulunan verilerin kalıcı olduğu varsayılıp, veri üzerinde VM uygulanırsa değişen verinin elde edilen sonuçlara yansıması mümkün olmayacaktır. Bu nedenle VM uygulanacak verinin mevcut uygulamadan ayrılması gerekir. Bu amaçla da veri ambarları kullanılır.

2.4.7. Farklı Tipteki Veriler

Gerçek hayattaki uygulamalar, makine öğreniminde olduğu gibi yalnızca sembolik veya kategorik veri türleri değil, aynı zamanda tamsayı, kesirli sayılar, çoklu ortam verisi, coğrafi bilgi içeren veri gibi farklı tipteki veriler üzerinde işlem yapılmasını gerektirir. Veri tipi çeşitliliğinin fazla olması bir VM algoritmasının tüm veri tiplerini ele alabilmesini olanaksızlaştırmaktadır. Bu yüzden veri tipine özgü VM algoritmaları geliştirilmektedir.

2.5. VERİ AMBARLARI VE OLAP

Günümüzde gelişen yeni tekniklerle büyük miktardaki veriler kısa sürede toplanmakta ve analiz edilebilmektedir. Veri miktarı arttıkça, veriden anlamlı bilgilerin çıkarılması daha karmaşık bir hal almaktadır. VA(Veri Ambarı) bu karmaşıklığı giderebilecek çözümlerden bir tanesidir.

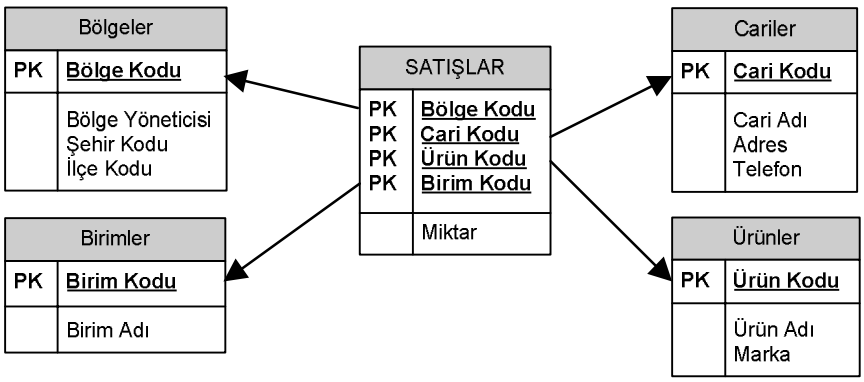
Veri ambarları son kullanıcıların sorgular yapabildiği, raporlar oluşturabildiği, analizler yapabildiği ortamlardır (Gürsoy, 2009).

Belirli bir döneme ait, yapılacak çalışmaya göre konu odaklı olarak düzenlenmiş, birleştirilmiş ve sabitlenmiş işletmelere ait veritabanlarına VA denilir (Silahtaroğlu, 2008).

VA, VM yapılacak veri ve veritabanını sağlamaktadır. İşletmelerin gerçek zamanlı olarak kullandığı veritabanları ön işlemden geçirilmeden VM için kullanılamaz.

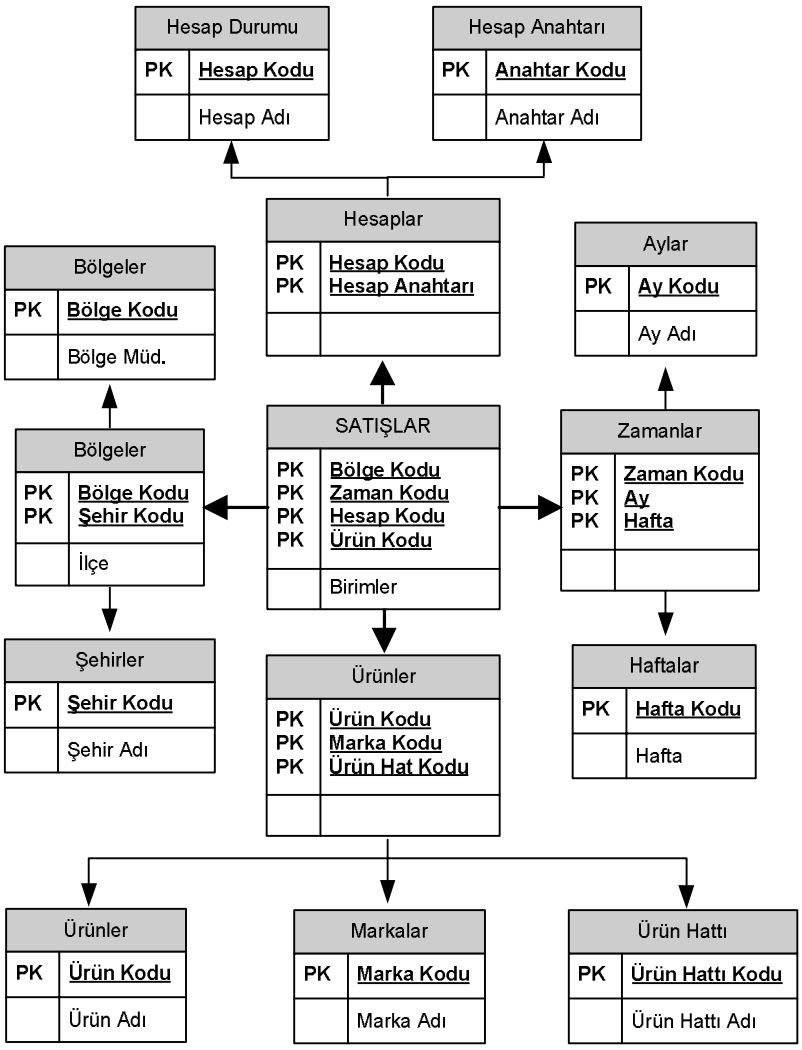
Elde edilen VA içerisindeki analiz ve sorgulama işlemlerine OLAP (Online Analytical Processing) denilir. VA'yı oluşturan veritabanları üzerinde yapılan sorgulamalar ile OLAP birbirinden farklıdır. İşletmelerin günlük kayıt giriş çıkış işlemleri için kullanılan ve VA'ya kaynak teşkil eden işlemsel veritabanları OLTP (Online Transaction Processing) sistemler olarak adlandırılır. Bu veritabanları içerisinde günlük satış, stok işlemleri, alım ve faturalama gibi kayıtları barındırır; dolayısıyla yapılabilecek sorgulama da ona göre olacaktır. İşlemsel veritabanlarında günlük veya haftalık satışlar, alım satım oranı ve stok durumu gibi sorgulamalar yapılabilir. Oysa OLAP sorgulamaları, hafta sonları belirli bir ürünün satışlarının belirli bir değeri aşma olasılığı, belirli bir meslek grubundaki belirli bir ürünü satın alan müşterilerin başka bir ürünü alma olasılığı gibi sorgulamaları içerecektir.

VA mimarisi temel olarak üç değişik şema kullanır. Bunlar yıldız, kartanesi ve anatablo birliğidir. Yıldız şema türünde, ortada bir ana tablo ve etrafında VA'nın boyutlarını oluşturan ana tabloyla ilişkili tablolar bulunur. Örnek yıldız şema Şekil 2.4'de verilmiştir.



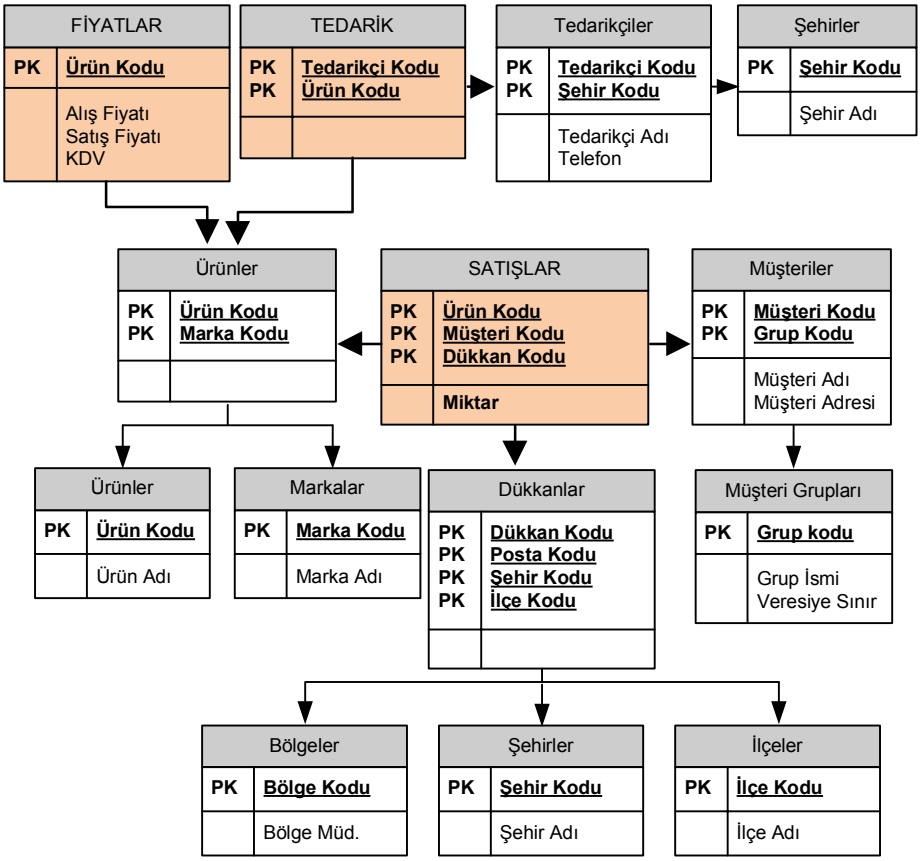
Şekil 2.4. Örnek yıldız şema.

Kartanesi şema türünde, yıldız şemadan farklı olarak ana tabloya bağlı tablolara da bağlı ilişkili tablolar bulunur. Yine ortada ana tablo bulunmaktadır. Örnek kartanesi şema Şekil 2.5'de verilmiştir.



Şekil 2.5. Örnek kartanesi şema.

Anatablolar birliğinde, birden fazla anatablo mevcut tabloları ortak olarak kullanır ve birden fazla yıldız şema iç içe monte edilmiş gibi görüntüye sahiptir. Örnek anatablolar birliği şema Şekil 2.6'da verilmiştir.



Şekil 2.6. Örnek anatablolar birliği şema.

VA mimarisinin daha iyi anlaşılması için aşağıda örnek bir veritabanının bir parçası verilmiş ve bu parçanın VA'ya nasıl yansıdığı gösterilmiştir (Silahtaroğlu, 2008).

Müşteri ID	Adı	Soyadı	Doğum Tarihi	…
123	Haluk	Atalay	11/02/1967	…
333	Onurhan	Turkkan	15/01/1971	…
…	…	…	…	…

Şekil 2.7. Örnek müşteriler tablosu.

Ürün ID	Marka	Tip	Miktar	…
45600	Aaa	Tip 1	300 g	…
45601	Bbb	Tip 2	275 g	…
…	…	…	…	…

Şekil 2.8. Örnek ürün tablosu.

Müşteri ID	Ürün ID	İşlem No	Miktar	…
123	45600	1	1	…
123	45602	1	1	…
123	45601	1	2	…
Kayıtsız	45602	2	1	…
333	45601	3	3	…
…	…	…	…	…

Şekil 2.9. Örnek satışlar tablosu.

Burçlar	Marka	Alışveriş Günü	Alışveriş Miktarı	…
Kova	Aaa	Pazartesi	255	…
Oğlak	bbb	Pazartesi	523	…
…	…	…	…	…

Şekil 2.10. Örnek VA parçası.

Şekil 2.10'da görüldüğü gibi VA veritabanının istenilen konuya göre yeniden düzenlenmiş halidir.

VA ile OLTP arasındaki önemli bir fark da OLTP kasiyer, veritabanı yöneticisi ve sistem uzmanları gibi kullanıcılar tarafından kullanılırken, VA ve OLAP uygulamaları analistler ve yöneticiler gibi bilgi üreten kullanıcılar tarafından kullanılır. Veritabanı ile VA arasındaki benzerlikler ve farklılıklar Çizelge 2.2'de verilmiştir.

Çizelge 2.2. Veritabanı ile Veri Ambarı arasındaki benzerlikler ve farklılıklar (Gürsoy, 2009).

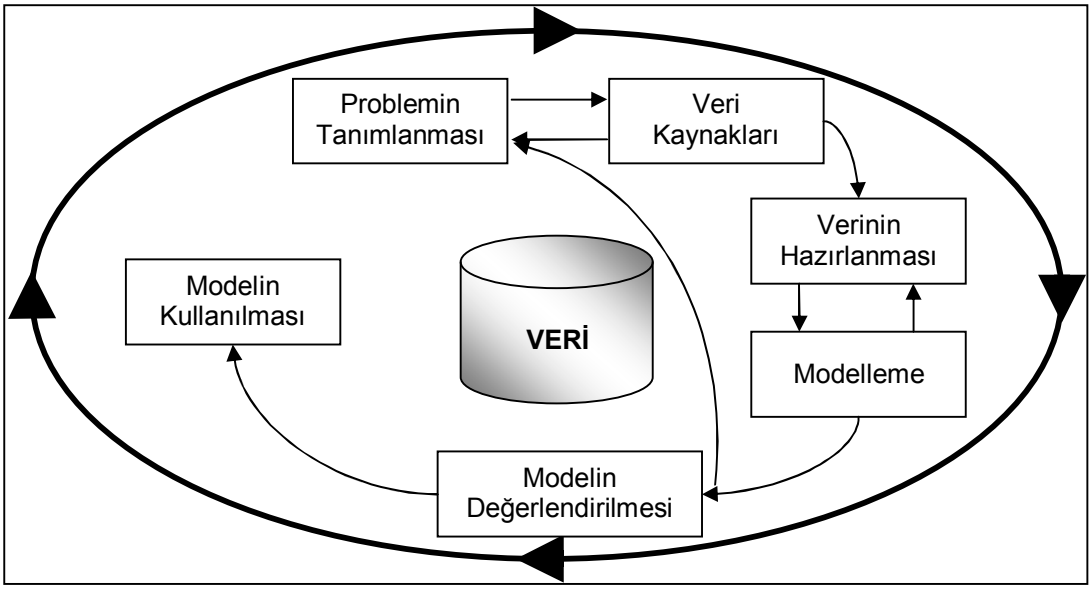
	Veritabanı	**Veri Ambarı**
Amaç	Günlük kayıtlar tutulur	Veri analiz edilir
Kullanıcı Sayısı	>1000	<100
Yapı	Veritabanı sistemleri	Veritabanı sistemleri
Veri Modeli	Normal	Çok boyutlu
Giriş	SQL	SQL+Veri analiz yöntemleri
Veri Tipi	En eski veri günlük 90 günlük	En eski veri yıllık
Verinin Durumu	Değişken, tamamlanmamış	Tanımlayıcı, tarihsel veri
Tablolar	Küçük boyutlu tablolar	Geniş boyutlu tablolar
Güncelleme	Sürekli	Daha uzun aralıklı

2.6. VERİ MADENCİLİĞİ SÜRECİ

VM kısaca gizli bilgilerin keşfi sürecidir. VM'nin bir süreç olarak tanımlanabilmesi için sürecin her bir aşamasının dikkatle izlenmesi gerekmektedir. Bir aşamanın sonucu, diğer bir aşamanın girdisidir. Bu sebeple her aşama bir önceki aşamanın sonuçlarına bağımlıdır (Gürsoy, 2009).

VM süreci belirli standartlar çerçevesinde gerçekleştirilir ve bu süreç matematiksel ve bilimsel metotların karışımı şeklindedir. Bu süreçler çeşitli ama benzer şekillerde belirlenir (Nisbet et al., 2009). Burada, VM süreci en çok kullanılan süreç standardı olan CRISP-DM (Cross Industry Standard Process for Data Mining) standardına göre açıklanacaktır.

CRISP-DM endüstriyi geliştirmek için oluşturulan bir projedir. 1996 yılında Daimler Chrysler, SPSS ve NCR tarafından oluşturulmuş ve birkaç yıl içerisinde belirli kullanıcı gereksinimleri ve endüstriyel deneyler için geliştirilmiştir. CRISP-DM, amaca ulaşmak için VM projelerini daha hızlı ve daha ucuz bir şekilde gerçekleştirmektedir (Hornick et al., 2007). CRISP-DM süreci altı aşamadan oluşmaktadır. Bu aşamalar Şekil 2.11'de gösterilmiştir.



Şekil 2.11. CRISP-DM metoduna göre veri madenciliği süreci.

Problemin tanımlanması VM sürecinin en önemli adımıdır. Bu adımdaki amaç projenin hedefini yani ne tür sonuçların alınacağının belirlenmesidir. Projenin amacı ve proje sonucu elde edilen değerlerin nasıl ölçüleceği tanımlanmalıdır (Soares, 2008).

Veri kaynakları, proje için VM'de kullanılacak veri kaynaklarının tanımlandığı aşamadır. Mevcut verilerin kalitesi değerlendirilebilir ve istenilirse işlemeye uygun desenler içeren alt veri kümeleri tanımlanabilir.

Verinin hazırlanması, modelin kurulması aşamasında ortaya çıkacak sorunlar, bu aşamaya sık sık geri dönülmesine ve verilerin yeniden düzenlenmesine neden olacaktır. Bu durum verilerin hazırlanması ve modelin kurulması aşamaları için, bir analistin veri keşfi sürecinin toplamı içerisinde enerji ve zamanının % 50 - % 85'ini harcamasına neden olmaktadır (Aynekin, 2006).

Verilerin hazırlanması aşaması kendi içerisinde toplama, değer biçme, birleştirme ve temizleme, seçme ve dönüştürme adımlarından meydana gelmektedir (Aynekin, 2006).

Toplama, tanımlanan problem için gerekli olduğu düşünülen verilerin ve bu verilerin toplanacağı veri kaynaklarının belirlenmesi adımıdır. Verilerin toplanmasında kuruluşun kendi veri kaynaklarının dışında, nüfus sayımı, hava durumu, merkez bankası kara listesi veya veri pazarlayan kuruluşların veri tabanlarından faydalanılabilir (Aynekin, 2006).

Değer biçme, VM'de kullanılacak verilerin farklı kaynaklardan toplanması doğal olarak veri uyumsuzluklarına neden olacaktır. Bu uyumsuzluklar, farklı zamanlara ait olmaları, kodlama farklılıkları (örneğin bir veri tabanında cinsiyet özelliğinin e/k, diğer bir veri tabanında 0/1 olarak kodlanması), farklı ölçü birimleridir. Ayrıca verilerin nasıl, nerede ve hangi koşullar altında toplandığı da önem taşımaktadır. Bu nedenlerle iyi sonuç alınacak modeller ancak iyi verilerin üzerine kurulabileceği için toplanan verilerin ne ölçüde uyumlu oldukları bu adımda incelenerek değerlendirilmelidir (Aynekin, 2006).

Birleştirme ve temizleme, farklı kaynaklardan toplanan verilerde bulunan ve bir önceki adımda belirlenen sorunlar mümkün olduğu ölçüde giderilerek veriler tek bir veri tabanında toplanır. Ancak, basit yöntemlerle ve baştan savma olarak yapılacak sorun giderme işlemlerinin, ileriki aşamalarda daha büyük sorunların kaynağı olacağı unutulmamalıdır (Aynekin, 2006).

Seçme, kurulacak modele bağlı olarak veri seçimi yapılır. Örneğin tahmin edici bir model için, bu adım bağımlı ve bağımsız değişkenlerin ve modelin eğitiminde kullanılacak veri kümesinin seçilmesi anlamını taşımaktadır. Sıra numarası, kimlik numarası gibi anlamlı olmayan ve diğer değişkenlerin modeldeki ağırlığının azalmasına da neden olabilecek değişkenlerin modele girmemesi gerekmektedir. Bazı VM algoritmaları konu ile ilgisi olmayan bu tip değişkenleri otomatik olarak elese de, pratikte bu işlemin kullanılan yazılıma bırakılmaması daha akılcı olacaktır. Verilerin görselleştirilmesine olanak sağlayan grafik araçları ve bunların sunduğu ilişkiler, bağımsız değişkenlerin seçilmesinde önemli yararlar sağlayabilir. Genellikle yanlış veri girişinden veya bir kereye özgü bir olayın gerçekleşmesinden kaynaklanan verilerin, önemli bir uyarıcı enformasyon içerip içermediği kontrol edildikten sonra veri kümesinden atılması tercih edilir (Aynekin, 2006).

Dönüştürme, kredi riskinin tahmini için geliştirilen bir modelde, borç/gelir gibi önceden hesaplanmış bir oran yerine, ayrı ayrı borç ve gelir verilerinin kullanılması tercih edilebilir. Ayrıca modelde kullanılan algoritma, verilerin gösteriminde önemli rol oynayacaktır. Örneğin bir uygulamada bir yapay sinir ağı algoritmasının kullanılması durumunda kategorik değişken değerlerinin evet/hayır olması; bir karar ağacı algoritmasının kullanılması durumunda ise örneğin gelir değişken değerlerinin yüksek/orta/düşük olarak gruplanmış olması modelin etkinliğini artıracaktır (Aynekin, 2006).

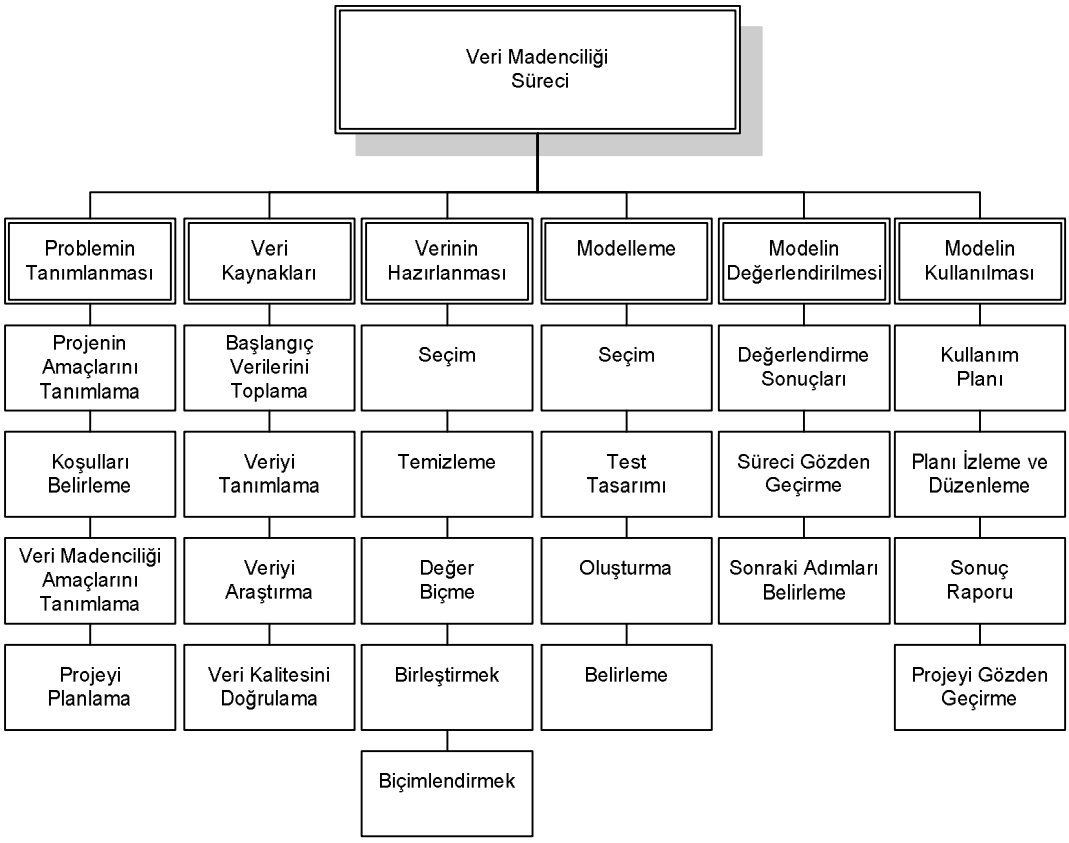
Modelleme adımında önceki adımın uygulanması sonucu elde edilen veriden gerekli bilginin çıkartılması için kullanılacak VM modeli belirlenir. Seçilen model verilerden en fazla verimin alınması için çok önemlidir. Eğer doğru model seçilmezse veriler içerisinde gizli olan bilgilere tam anlamıyla ulaşılamayabilir.

VM'de işlenecek veri miktarı çok fazla olacağı için model seçiminde verinin tamamı üzerinde birkaç model test etmektense, veriler içerisinden örnekleme yaparak çok fazla modelin test edilmesi ve en iyi sonucun alındığı modelin seçilmesi daha uygundur. Örnekleme yapılan veriler verinin tamamını temsil etmek zorundadır.

Tanımlanan problem için en uygun modelin bulunabilmesi, olabildiğince çok sayıda modelin kurularak denenmesi ile mümkündür. Bu nedenle veri hazırlama ve model kurma aşamaları, en iyi olduğu düşünülen modele varılıncaya kadar yinelenen bir süreçtir.

VM sürecinin son aşaması, kurulan ve geçerliliği kabul edilen modelin kullanılmasıdır. Bu doğrudan bir uygulama olabileceği gibi bir başka modelin alt parçası olarak da kullanılabilir. Kurulan modeller risk analizi, kredi değerlendirme, dolandırıcılık tespiti gibi işletme uygulamalarında doğrudan kullanılabileceği gibi, tahmin edilen envanter düzeyleri yeniden sipariş noktasının altına düştüğünde otomatik olarak sipariş verilmesini sağlayacak bir uygulamanın içine gömülebilir. Kullanılan modelin zaman içerisinde izlenip ortaya çıkan değişikliklerin modele yansıtılması, yaşayan bir süreç olması açısından vazgeçilmez bir koşuldur (Gürsoy, 2009).

CRISP-DM metodunun aşamalara göre özellikleri Şekil 2.12'de gösterilmiştir.



Şekil 2.12. CRISP-DM metodunun aşamalara göre özellikleri(http://www.crisp-dm.org/CRISPWP-0800.pdf, 2010).

2.7. VERİ MADENCİLİĞİ MODELLERİ

VM'de kullanılan modeller temel olarak tahmin edici ve tanımlayıcı modeller olmak üzere ikiye ayrılır.

2.7.1. Tahmin Edici Modeller

Tahmin edici modellerde, sonuçları bilinen veriler ve önceki tecrübelerden hareket ederek bir model geliştirilmesi ve bu model ile sonuçları bilinmeyen veri kümeleri için sonuçların tahmin edilmesi amaçlanmaktadır (Özekes, 2003). Tahmin edici modellerde bağımlı ve bağımsız değişken adı altında iki adet değişken bulunmaktadır. Eldeki var olan yani bilinen veriler bağımsız değişken olarak adlandırılırken, istenilen soruların cevapları ise bağımlı değişken olarak adlandırılmaktadır.

Tahmin edici modeller özellikle karar alma süreçlerinde kullanılmaktadır. Örneğin, bir firma müşterilerinin daha önce almış olduğu ürünlerin bilgisine sahiptir. Burada, alınan ürünler bilinen veriler olduğu için bu veriler bağımsız değişkenlerdir. Bağımlı değişken ise müşterinin alabileceği diğer ürünlerdir. Bu veriler doğrultusunda kurulan bir model müşterinin alabileceği ürünlerin tahmin edilmesini sağlayabilir.

Tahmin edici modellerin temel iki türü sınıflandırma ve regresyondur. Sınıflandırma büyük ölçekli problemlerin çözümünde ve kategorik değerlerin(evet/hayır değişkenleri veya çok seçmeli değer içeren değişkenler) tahmininde kullanılır. Regresyon ise süreklilik gösteren değerlerin(kişilerin yaşları, kan basınçları veya ürünlerin günlük satış miktarları) tahmin edilmesinde kullanılır. Örneğin, bir sınıflandırma modeli banka kredi uygulamalarının güvenli veya riskli olmalarını kategorize etmek amacıyla kurulurken, regresyon modeli geliri ve mesleği verilen potansiyel müşterilerin bilgisayar ürünleri alırken yapacakları harcamaları tahmin etmek için kurulabilir (Nisbet et al., 2009; Hornic et al., 2007; Özekes, 2003).

Sınıflama ve regresyon modelinde kullanılan başlıca teknikler şunlardır (Akpınar, 2000);

- Karar Ağaçları (Decision Trees)
- Yapay Sinir Ağları (Artificial Neural Networks)
- Genetik Algoritmalar (Genetic Algorithms)
- K-En Yakın Komşu (K-Nearest Neighbor)
- Bellek Temelli Nedenleme (Memory Based Reasoning)
- Naive-Bayes
- Lojik Regresyon (Logistic Regression)

2.7.2. Tanımlayıcı Modeller

Tanımlayıcı modeller, mevcut veriler içerisinde daha önce bilinmeyen ve karar vermeye rehberlik etmede kullanılacak gizli kalmış ilişkileri tespit etmektedir. X/Y aralığında geliri ve iki veya daha fazla arabası olan çocuklu aileler ile çocuğu olmayan ve geliri X/Y aralığından düşük olan ailelerin satın alma örüntülerinin

birbirlerine benzerlik gösterdiğinin belirlenmesi tanımlayıcı modellere bir örnektir (Özekes, 2003; Gürsoy, 2009; Silahtaroğlu, 2008).

Tanımlayıcı modellerin temel iki türü kümeleme ve ilişki analizidir. Birliktelik kuralları ve ardışık zamanlı örüntüler ilişki analizi kapsamındadır.

BÖLÜM 3

WEB MADENCİLİĞİ

WWW (World Wide Web) günümüzdeki en hızlı ve en ucuz bilgi paylaşım aracıdır. Kişisel kullanımlardan kurumsal kullanımlara kadar her alanda hayatın vazgeçilmez bir parçası haline gelmiştir.

Günümüzde insanların büyük çoğunluğu başta iletişim olmak üzere, e-ticaret, reklam, bilgi/belge paylaşımı, bankacılık işlemleri, kurumsal işlemler ve eğitim gibi birçok işlemi internet üzerinden yani kısa adıyla web üzerinden gerçekleştirmektedir.

WWW her geçen gün daha da büyümekte ve hayatımızda daha da fazla yer almaktadır. İnternetin herkese açık olması, içerdiği bilgilerin her geçen gün daha düzensiz olmasına ve daha da artmasına neden olmaktadır. Web ortamındaki bu verilerin büyük olması kadar düzensiz olması da web madenciliğine ayrı bir önem kazandırmaktadır (Gürcan ve Köse, 2008).

Web madenciliği ilk olarak 1996 yılında Oren Etzioni tarafından ortaya atılmıştır. Bu bildiride Etzioni'ye göre (1996) web madenciliği, VM tekniklerini kullanarak WWW'de bulunan dosya ve servislerden otomatik olarak bilginin ayıklanması, ortaya çıkartılması ve analiz edilmesidir. Web 'de bulunan verilerin sürekli olarak güncellenmesi, silinmesi ve yeni bilgilerin eklenmesi web'den bilgi çıkarım işleminde karşılaşılan en önemli zorluklardan birisidir.

Web madenciliği ile web sitesinin sahibi olan kurum veya kuruluşa site güncellemesi, reklam veya karar alma süreçlerinde yardımcı olacak yararlı bilgiler sunulabilir.

Web madenciliği ile işlenecek olan veri web sitesinin içerdiği bilgiler, sitenin yapısı, kullanıcıların ziyaretleri esnasında sunucu tarafından toplanan veriler ve ziyaretçilerin üyelik işleminde vermiş olduğu bilgilerden oluşmaktadır.

3.1. WEB TERİMLERİ

Web madenciliği süreci içerisinde geçen ve W3C konsorsiyumunca tanımlanan bazı terimler aşağıda açıklanmıştır (http://www.w3.org/1999/05/WCA-terms,2010; Gezer vd., 2007; Daş vd., 2008).

Kaynak (Resource), W3C'nin Değişmez kaynak tanımlayıcısı tarifine göre (Uniform Resource Identifier - URI) özdeşliği olan her şey olabilir.

URI, kaynağın fiziksel adresini tanımlayan karakter kümesi olarak açıklanabilir. Örneğin, http://www.gop.edu.tr/default.aspx.

Web kaynağı, HTTP protokollerinden (Örneğin, HTTP 1.0) herhangi bir sürümüne ulaşabilen kaynaktır.

Web sunucusu, bir veritabanı içeren ve internet üzerinde belgelere erişim hizmetlerini sunan bilgisayardır.

Web sayfası, URI tarafından tanımlanan bir veya birden fazla web kaynağının veri kümesidir.

Web sitesi, bir web sunucusu tarafından web 'de sunulan veritabanları, ilgili belgeler ve dosyalar. Bir web sitesindeki belgeler birbirleriyle ilgili birkaç konuyu kapsayıp, aralarında üst metin linkleri ile bağlantılar kurulur.

Sayfa görüntüleme, bir web tarayıcısının (web browser) belli bir zamanda bir web sayfasında bulunması.

Web tarayıcı, internet üzerinde bilgi kaynaklarını aramaya elverişli, bağlantılı, metin ve ortamların olanaklarını kullanan istemci yazılımı. Başka bir ifade ile istenen URI 'yi görüntüleyen yazılım (IE, Mozilla, Opera, vb.).

Web isteği, istemcinin bir web kaynağına yapmış olduğu istek. Bunlar açık (kullanıcı taraflı, explicit) ya da dolaylı (web istemci taraflı, implicit) olarak ikiye ayrılır. Açık web istekleri (aynı zamanda tıklama – click olarak da adlandırılır) kendi içersinde iki sınıfa ayrılır; gömülü (embedded) ve kullanıcı girişli şeklinde adlandırılır. Gömülü web isteğine örnek olarak bir web sayfası içerisinde bulunan bağlantılardan yapılan istekler verilebilir. Kullanıcı girişli web istekleri ise kullanıcının web tarayıcısı üzerinden yazarak ya da seçerek yapacağı isteklerdir. Dolaylı web istekleri çağrılmış olan web sayfası içerisinde gömülmüş olan öğelerin (örneğin sayfa içerisindeki resim, betik (script) dosyaları vb.) getirilmesini sağlayan isteklerdir.

Kullanıcı oturumu, bir kullanıcının bir veya daha fazla web sunucusu üzerinde yapmış olduğu sınırlandırılmış sayıda kullanıcı taraflı web istekleridir.

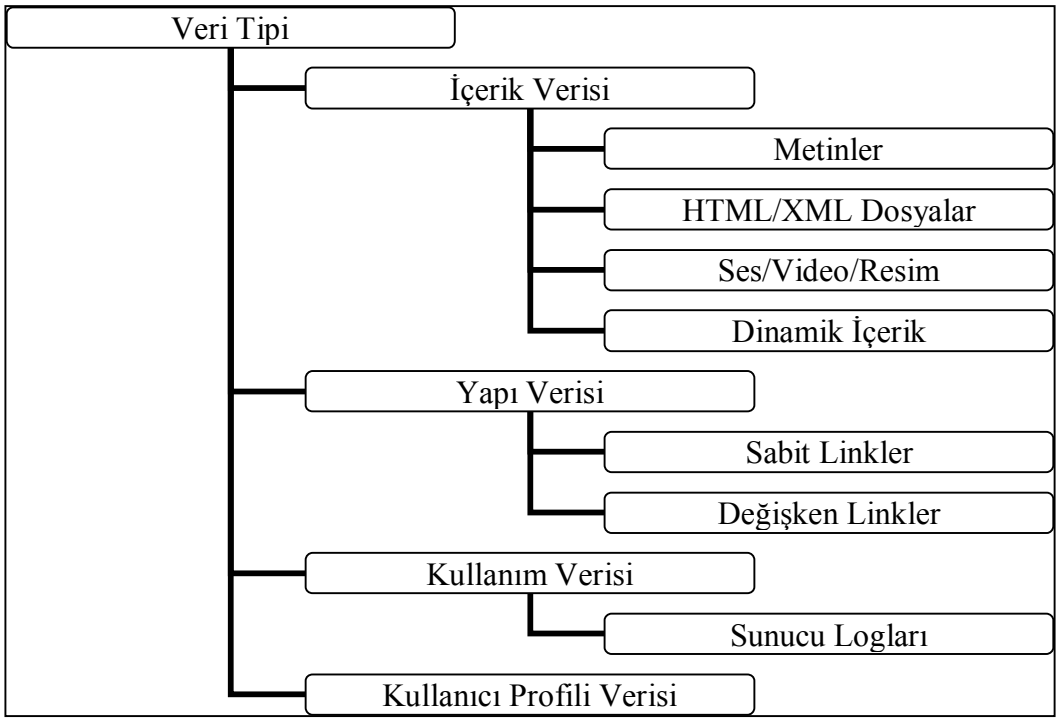
Ziyaret, belli bir zaman süresince kullanıcı oturumu esnasında yapılmış olan sayfa görüntülemesi eylemidir.

Oturum tanımlama, kullanıcının bir siteye oturum açmasından kapattığı zamana kadarki yapılan işlemlerinin belirli zaman aralığına göre sınıflandırılmasıdır.

Kullanıcı tanımlama, siteye erişen kullanıcıların tarayıcı ve kullanım özelliklerine göre sınıflandırılmasıdır.

3.2. WEB VERİ TİPLERİ

WWW çeşitli kaynaklarda bulunan farklı tiplerdeki verileri içerir. Web madenciliği için kullanılacak olan veri de bu kaynaklar üzerinden toplanır. WWW, verinin kaynağına göre üç gruba ayrılmaktadır. Şekil 3.1'de veri tipleri kategorik olarak gösterilmiştir.



Şekil 3.1. Web veri tipleri.

3.2.1. İçerik Verisi

Web sayfalarının içermiş olduğu ve kullanıcılara sunulan verilerdir. Web içerik verisi metinler, HTML sayfalar, XML sayfalar, dinamik olarak oluşturulan sayfalar veya içerik ile ilgili veritabanından alınan bilgilerden oluşmaktadır. Sayfalar içerisinde bulunan resimler, videolar, ses veri tipleri, tanımlayıcı kelimeler, doküman özellikleri ve sayfa içerisindeki etiketler de içerik verisini oluşturmaktadır (Gezer vd., 2007).

3.2.2. Yapı Verisi

Bir web sitesinden diğer bir web sitesine ya da bir web sayfasından diğer bir web sayfasına yapılan bağlantı yapısının kesin ve açık olarak belirtilmesidir. Yani, web bağlantılarının organizasyonunu gösteren bilgilerdir. Bu bilgiler, web tasarımcısının siteye bakış açısını göstermektedir. Web sitesi yapı verisi, site haritalama araçları ile otomatik olarak oluşturulan sitenin harita bilgisidir (Daş vd., 2008).

3.2.3. Kullanım Verisi

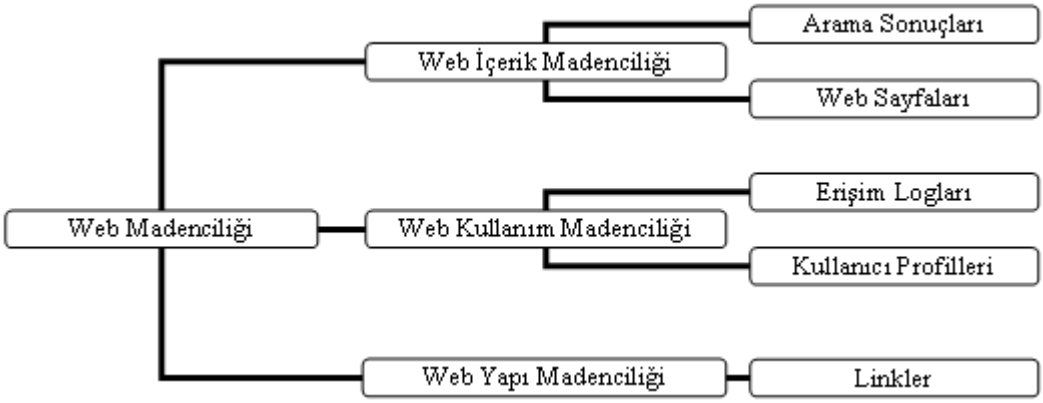
Kullanıcıların web sitesine erişimi sonucu oluşturulan verilerdir. Bu veriler web sunucu, vekil sunucu veya web tarayıcısının geçmiş bilgisi içerisinde tutulan cookie'ler yardımıyla tutulur. Kullanım verisinin önemli bir bölümünü sunucular üzerinde tutulan kayıt dosyaları (log files) oluşturmaktadır. Bu dosyalar içerisinde kullanıcının ve sunucunun IP adresi, ziyaret edilen adres, referans adres, bağlantı yapılan tarih ve saat, kullanıcı tarayıcı ve sistem bilgisi ve yapılan veri transferi miktarı gibi bilgiler tutulmaktadır. Kullanıcının bir web sitesi üzerinde yapmış olduğu her ziyaret kayıt dosyaları üzerinde tutulmaktadır. Kayıt dosyaları analiz amacına uygun olarak dönüştürülmeli ve bir araya getirilmelidir (Daş vd., 2008; Gezer vd., 2007; Srivastava et al., 2000).

3.2.4. Kullanıcı Profili Verisi

Web sitesine kayıt olan kullanıcılar hakkında demografik bilgilerin sağlandığı verilerdir. Bir siteye kayıt olmak isteyen kullanıcı ya da müşterilerden alınan bilgiler, bu veriler içerisinde yer almaktadır. Bu tür verilerin elde edilebilmesi için internet kullanıcısının web sitesine kayıt yaptırması gerekmektedir (Daş vd., 2008; Gezer vd., 2007; Srivastava et al., 2000).

3.3. WEB MADENCİLİĞİ SINIFLANDIRMASI

Web madenciliği çalışma alanlarının kapsamlı ve detaylı olması bu alanda düzenli bir sınıflandırmayı da gerektirmektedir. Web madenciliği ilk ortaya atıldığı dönemlerde Web İçerik Madenciliği (Web Content Mining) ve Web Kullanım Madenciliği (Web Usage Mining) olmak üzere iki sınıfa ayrılmaktaydı. Web madenciliğinin yaygınlaşması ile birlikte Web Yapı Madenciliği de (Web Structure Mining) üçüncü bir sınıf olarak eklenmiştir (Etzioni, 1996; Kosala and Blockeel, 2000).



Şekil 3.2. Web madenciliği sınıflandırması.

Web içerik madenciliği, www 'de bulunan içerik verisinden bilgi çıkarım işlemini gerçekleştirir (Kantardzic, 2003). Web yapı madenciliği, web sayfaları ve web siteleri arasındaki bağlantıları yani web yapı verisini inceleyerek bilgi çıkarım işlemini gerçekleştirir (Kantardzic, 2003; Belen vd., 2008). Web log mining olarak da bilinen web kullanım madenciliği ise sunucu üzerinde tutulan ziyaret kayıt dosyalarından bilgi çıkarım işlemini gerçekleştirir.

3.3.1. Web İçerik Madenciliği

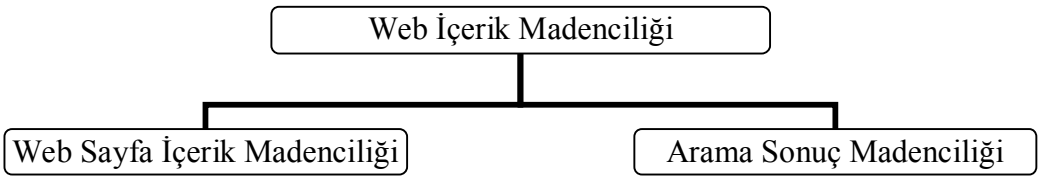
Bazı kaynaklarda metin madenciliği olarak da geçen web içerik madenciliği, internette saklı olan bilgiyi bulmak için kullanılmaktadır. Bu işlem için kullanacağı veri kaynağı sitenin kendisi veya arama motorlarıdır. Yani web sitelerinin sahip olduğu içeriklerinden yararlı bilginin elde edilmesini sağlar.

Web sayfalarının sahip olduğu içerikler çoğunlukla düz metinlerden oluşmaktadır. Düz metinler ile birlikte resim, ses, video gibi çoklu ortam öğeleri de veri içeriğini oluşturmaktadır. Veri içeriğinin bu şekilde farklı türlerde olması verinin analizini zorlaştırmaktadır. Bu yüzden farklı türlerdeki verilerin daha iyi analiz edilmesi için bilgiye erişim ve veritabanı yaklaşımları geliştirilmiştir (Gündüz ve Adalı, 2004; Kosala and Blockeel, 2000).

Bilgiye erişim yaklaşımları bilgiye erişim, bilginin analiz edilmesi ve analiz edilen bilginin amaca uygun sınıflandırılması için kullanılır.

Veritabanı yaklaşımları elde edilen verilerin veritabanına kaydedilerek, veritabanı üzerinden sorgulanması ve filtrelenmesi için kullanılır. Sorgulama, filtreleme ve sınıflandırma veritabanı üzerinde daha kolay yapılabilmektedir.

Web içerik madenciliği kullanılan veriye göre Şekil 3.3 'de görüldüğü gibi web sayfa içerikleri ve arama motorları sonuçlarına göre iki alt gruba ayrılır.



Şekil 3.3. Web içerik madenciliği sınıflandırması.

Web sayfa içerik madenciliği, metin, resim, ses ve video gibi web sayfalarının içeriklerini kaynak veri olarak kullanarak bilgi çıkarım için kullanılır.

Arama sonuç madenciliği, arama motorlarından elde edilen sonuçların sınıflandırılmasını sağlar. Bu sayede arama sonuçlarından, aranan hedefe daha yakın olanların saptanması mümkündür (Gürcan ve Köse, 2008).

3.3.2. Web Yapı Madenciliği

Web yapı madenciliği, web sitesi ve sayfalarının yapısal olarak özelliklerini belirler. Bu özellikleri belirlerken sayfa bağlantılarını ve doküman yapısını kullanmaktadır (Gürsoy, 2009). Web içerik madenciliği web sayfasının içeriği ile ilgilenirken, web yapı madenciği web sayfaları arası bağlantıları incelemektedir (Daş ve Türkoğlu, 2010).

Web yapı madenciliğinin amacı web sitesi ve web sayfaları içerisindeki ilgili bağlantı verisine bakarak istenilen bilgiyi keşfetmektir. Ayrıca, web yapı madenciliği sayfaların bağlantı (link) tasarımlarını ortaya çıkarmaya yardımcı olur. Web sayfalarında bulunan site haritası bu bağlamda sıklıkla kullanılmaktadır. Elde edilen

bağlantı yapısını kullanarak web sayfalarını sınıflandırır ve farklı web siteleri arasındaki benzerlik ve ilişki gibi sonuçları üretir.

Web yapı madenciliği, kullandığı verinin tipine bağlı olarak iki gruba ayrılmaktadır (Daş ve Türkoğlu, 2010).

3.3.2.1. Sayfa Bağlantıları

Aynı web sayfası içerisindeki bir noktaya yada farklı bir web sayfasına bağlantı sağlayan sayfaların fiziksel kaynağını tanımlayan karakterler zinciridir. Aynı sayfa içerisindeki bir noktaya yapılan bağlantıya iç doküman bağlantısı, farklı sayfalara yapılan bağlantıya dış doküman bağlantısı denilmektedir.

3.3.2.2. Doküman Yapısı

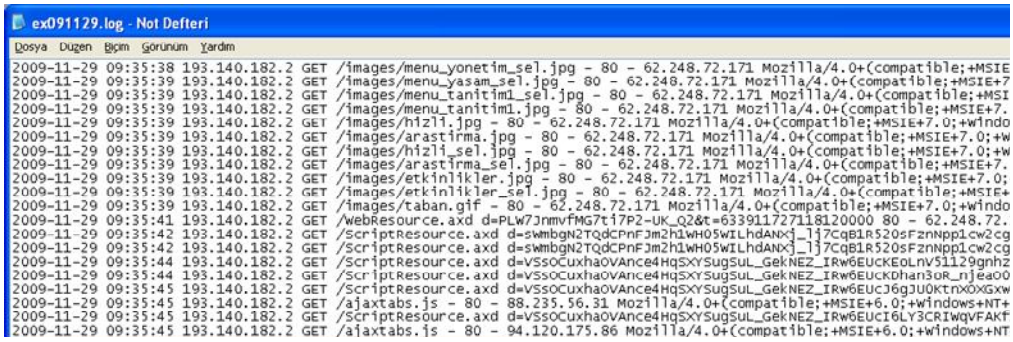
HTML veya XML biçimindeki bir web sayfası içerisindeki etiketlerin analizi ve tanımlaması için ağaç yapıları kullanılır. Bir web sitesinin organizasyonu ve dokümanlarının düzenlenmesi bu alana girmektedir. Özellikle, bir web sitesindeki dokümanların yapısını otomatik olarak çıkarmak için kullanılır.

3.3.3. Web Kullanım Madenciliği

Ziyaretçilerin bir web sitesi üzerinde yapmış olduğu her türlü işlem kayıt altına alınmaktadır. Bu kayıtlar web sunucusuna ait erişim kayıtları, uygulama sunucusu ait kayıtlar, çerezler ve kullanıcı profillerinden oluşmaktadır. Web kullanım madenciliğinde çoğunlukla web sunucusuna ait erişim kayıtları(log) veri kaynağını oluşturmaktadır (Srivastava et al., 2000; Srivastava et al., 2005).

Web kullanım madenciliği, web kullanım verilerinden ilginç desenleri keşfetmek için kullanılan VM tekniklerinin uygulama sürecidir. Kullanıcının siteyi kullanırken gerisinde bıraktığı erişim verilerinden bilgi üretmeyi amaçlar. Web yapı madenciliği ve web içerik madenciliğinden farklı olarak web üzerindeki doğrudan erişilebilen veriyi kullanmak yerine, kullanıcıların web'de dolaşırken hareketlerinden oluşturulan

veriden bilgi üretir. Bu veriler ikinci sınıf verilerdir yani bir yere girilmiş, bir yerde yazılan ya da kullanıcın isteğiyle oluşan veri değildir. Tamamen kullanıcıdan bağımsız oluşur ve çok ciddi boyutlardadır. Bu veriler istemcilerde, sunucularda ve proxy sunucularda depolanır.



Şekil 3.4. Örnek sunucu kayıt(log) dosyası.

Web kullanım madenciliği ile web yöneticisi için, web sunucusuna gelen taleplerin zamana, kullanıcılara ve URL tiplerine göre dağılımları, başarılı ve başarısız erişimler, gelinen kaynağın belirlenmesi, ziyaretçi tiplerinin belirlenmesi, kurum içi erişim dağılımlarının belirlenmesi, sık ve birlikte ziyaret edilen sayfaların belirlenmesi gibi birçok bilgi sağlanmaktır. Bu bilgiler yardımıyla web yöneticisi site üzerinde gerekli güncelleştirme ve düzenlemeleri yapabilir, kurum veya kuruluşlar müşterilerine yönelik reklam kampanyaları düzenleyebilir ve ziyaretçilere ürün tavsiyesinde bulunabilir.

BÖLÜM 4

WEB KULLANIM MADENCİLİĞİ

Web kullanım madenciliği, web sitesinin kullanım analizi için web kullanım verilerinden en yoğun ve en ilginç kullanıcı erişim örüntülerini keşfetmek ve anlamlı verileri çıkartmak için VM tekniklerini uygulama sürecidir. Web ve vekil sunucularda tutulan kullanıcı erişim kayıtları, tarayıcı kayıtları, kullanıcı profilleri, çerezler ve kullanıcıların web ile olan etkileşimlerinden oluşan tüm kayıtlar web kullanım verisini içermektedir. Web kullanım madenciliğinde kullanılan verilerin sınıflandırılması kullanım verilerinin türüne bağlıdır.

Web sunucu verisi, web sitesinin bulunduğu web sunucu tarafından tutulan erişim kayıtlarıdır. Veriler metin dosyaları içerisinde tutulmaktadır ve site üzerinde yapılan her bir işlem metin dosyasına yeni satır olarak eklenmektedir. Oluşturulan erişim kayıtlarının yapısı kullanılan web sunucu ve web sunucu konfigürasyonuna göre farklılık göstermektedir.

Uygulama sunucu verisi, elektronik ticaret uygulamalarında kullanılan ticari uygulama sunucularında tutulan çok önemli verilerdir. Yani, uygulama sunucusunda bulunan müşteri özelliklerine ait kayıtların ve iş olaylarına ait izlerin tutulduğu önemli bilgilerdir.

4.1. WEB KULLANIM VERİSİ

Web kullanım madenciliğinin ana veri kaynağını oluşturan web kullanım verisi web ve uygulama sunucusu üzerinde otomatik olarak toplanır. Veriler sunucu üzerinde metin tabanlı log dosyalarına kaydedilir ve kullanıcıların sunucudan her bir isteği log dosyaları içerisine yeni bir satır olarak eklenir.

Oluşturulan log dosyaları kullanılan servis türüne göre access log, mail log, error log ve ftp log şeklinde sınıflandırılır. Oluşturulan her bir veri dosyası günlük olarak tutulur ve bir sonraki güne geçildiğinde o güne ait yani bir metin belgesi oluşturulur. Ayrıca kullanılan sunucuya ve konfigürasyonuna bağlı olarak tutulan veriler değişiklik gösterebilir. Genel olarak her bir satırda erişim tarihi, saat, sunucu IP adresi, istemci IP adresi, istekte bulunulan web adresi, sayfa referansları, tarayıcı ve işletim sistemi bilgilerini içeren user-agent bilgisi tutulur. Tutulan parametre sayısı sunucu üzerinde yapılan konfigürasyon ile artırılabilir veya azaltılabilir. Ayrıca, kullanım verileri sunucu üzerinde kullanılan işletim sistemine bağlı olarak farklılık gösterebilir. Örneğin, Linux işletim sistemi üzerinde çalışan Apache web sunucusu ile Windows Server 2003 işletim sistemi üzerinde çalışan IIS (Internet Information Server) sunucusunun oluşturduğu kullanım verileri biçimi birbirinden farklı olacaktır.

Microsoft IIS web sunucusunda log dosyaları CLF (Common Log Format), ECLF (Extended Common Log Format) ve NCSA (National Center for Supercomputing Applications) olmak üzere üç farklı biçimde tutulmaktadır. Bu çalışmada kullanılacak olan ECLF biçimindeki log dosyalarına ait örek bir satır Şekil 4.1'de ve ECLF biçimindeki log dosyalarında kullanılan alanların açıklamaları Çizelge 4.1'de verilmiştir.

```
2010-03-05 00:22:31 193.140.180.4 GET /Default.aspx - 80 - 212.154.80.164
Mozilla/4.0+(compatible;+MSIE+6.0;+Windows+NT+5.1;+SV1;+GTB6.4) - 200
0 0 67049 428 31
```

Şekil 4.1. ECLF biçimindeki log dosyalarından örnek bir satır.

Çizelge 4.1. ECLF biçimindeki log dosyalarında kullanılan alanların açıklaması.

Alan Adı	Örnek Değer	Açıklama
date	2010-03-05	Aktivitenin meydana geldiği tarih.
Time	00:22:31	Aktivitenin meydana geldiği saat.
c-ip	212.154.80.164	İstekte bulunan kullanıcının IP adresi.
Cs-username	-	Aktivite kimlik denetimi ile gerçekleşmişse kullanılan kullanıcı adı.
s-ip	193.140.180.4	Web sunucunun IP adresi.

Çizelge 4.1. (devam ediyor).

s-port	80	Aktivitenin sunucu üzerinde kullandığı port numarası.
cs-method	GET	Kullanılan web isteği metodu.
cs-uri-stem	/Default.aspx	İstekte bulunulan web adresi.
cs-uri-query	-	Özel tanımlamalar yapılmadığı sürece bilgi bulunmaz. Sadece dinamik sayfalar için geçerlidir. Dinamik sayfalar ile gönderilen bilgileri içerir.
sc-status	200	İsteğe verilen cevabın durum kodunu içerir. 200 nolu kod isteğin başarılı olduğunu, 404 nolu kod bağlantının sağlanamadığını gösterir. Çizelge 4.2 ve Çizelge 4.3 'de durum kodları ve açıklamaları verilmiştir.
sc-win32-status	0	Windows durum kodu.
sc-bytes	67049	Sunucu tarafından gönderilen verinin boyutu.
cs-bytes	428	Sunucu tarafından alınan verinin boyutu.
time-taken	31	Aktivitenin milisaniye cinsinden gerçekleşmesi için harcanan zaman.
cs(user-agent)	Mozilla/4.0+(compatible ; +MSIE+6.0; +Windows+NT+5.1;)	İstemci tarafından kullanılan tarayıcının tipi ve diğer özellikler.
cs-referrer	-	Aktif sayfaya hangi kaynaktan gelindiğini gösterir. Bir önceki ziyaret edilen sayfa veya site.
sc-substatus	0	Protokol alt durum kodu.

Çizelge 4.2. sc-status durum kodları ve açıklamaları (Çetiner vd., 2000).

Durum Kodu	**Açıklama**
1xx	İsteğin sunucuya geldiği anda devam eden bir işlemin olduğunu belirten geçici bir yanıt içerir.
2xx	İsteğin başarılı bir şekilde gerçekleştiğini belirtir.
3xx	İsteği tamamlamak için ek işlem yapılması gerektiğini belirtir.
4xx	İsteğin hatalı olduğunu ya da tamamlanamadığını gösterir.
5xx	Sunucunun tamamen geçerli olan bir isteği kendisindeki bir hatadan dolayı gerçekleştiremediğini gösterir.

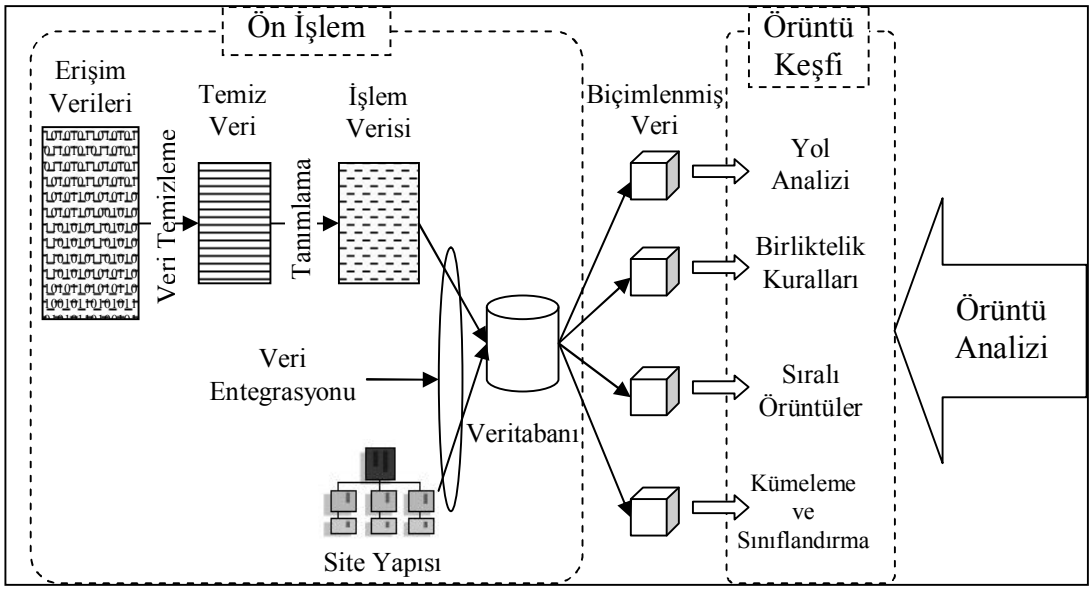
Çizelge 4.3. Genel durum kodları (Çetiner vd., 2000).

Durum Kodu	Sonuç	Açıklama
200	OK	İstek başarılı sonuçlandı.
202	Kabul edildi	İstek işlem için kabul edildi fakat işlem tamamlanamadı.
302	Bulunamadı	İstenen belge yeni bir yere taşınmış durumdadır.
304	Değişiklik yok	İstemci, sunucu tarafından yerel diskteki belge sunucu tarafındaki belge ile aynı oldugu (değişiklik olmadığı) için yerel diskteki belgeyi kullanıyor.
400	Kötü istek	İstek, hatalı sözdizimi nedeniyle sunucu tarafından anlaşılamadı.
401	Yetkisiz	İstek kullanıcıdan doğrulama bekliyor, fakat istemci geçerli bir kullanıcı adı veya şifre sağlamadı.
403	Yasak	Sunucu isteği anladı fakat tamamlamayı reddediyor (istenen belgeye erişim yasak).
404	Bulunamadı	İstenen belge sunucuda yok (adres yanlış olabilir ya da belge taşınmış olabilir). Ayrıca bu kod erişim izni olmayan istemcilere belgenin olmadığı mesajını vererek belgeyi korumak amacıyla kullanılabilir.
500	Dahili Sunucu Hatası	Sunucu beklenmedik bir durumla karşılaştı ve isteği tamamlayamadı. Bu durumlarda sunucu yöneticisi hata kütüğünü kontrol ederek hatanın neden kaynaklandığına bakmalıdır.

4.2. WEB KULLANIM MADENCİLİĞİ UYGULAMA SÜRECİ

Web kullanım madenciliği, web sunucusu üzerinde tutulan verilerden kullanıcı desenlerinin keşfinin ve analizinin yapıldığı veri madenciliğidir. Kısaca, web üzerinden elde edilen verilere VM tekniklerinin uygulanmasıdır.

Web kullanım madenciliği ön işlem, örüntü keşfi ve örüntü analizi olmak üzere 3 aşamada gerçekleştirilir (Kosala and Blockeel, 2000). Bu aşamalar Şekil 4.2'de gösterilmiştir.



Şekil 4.2. Web kullanım madenciliğinin uygulama adımları.

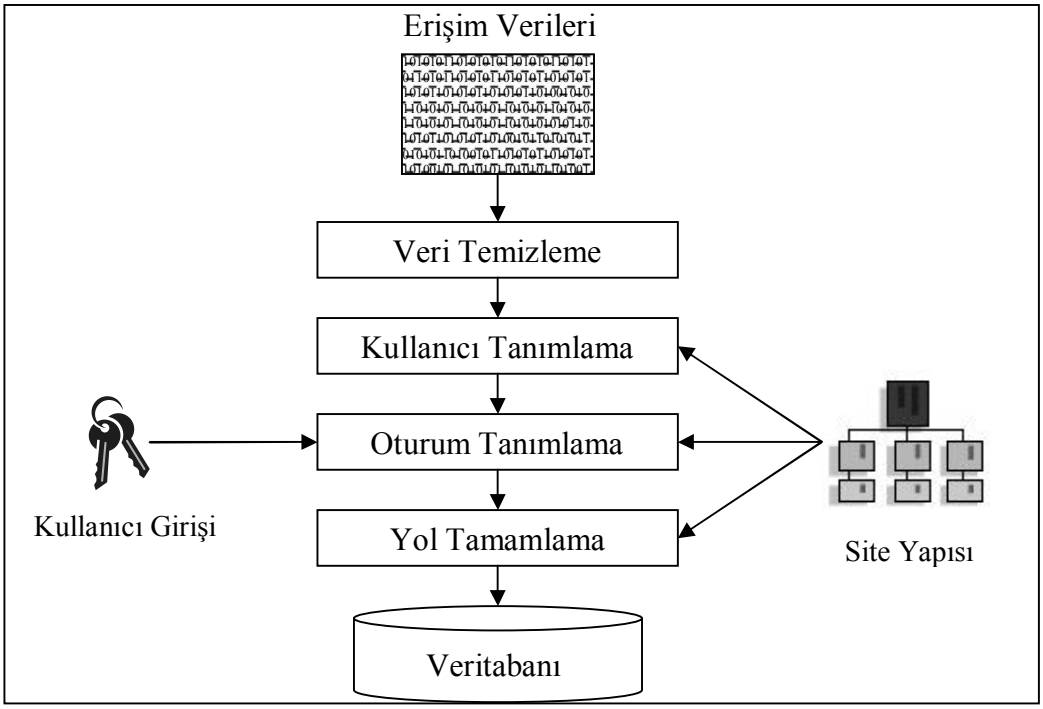
4.2.1. Ön İşlem Süreci

Web kullanım madenciliği uygulama sürecinin en önemli aşamalarından birisi VM ve istatistiksel algoritmaların uygulanabileceği uygun hedef veri kümesinin oluşturulmasıdır. Web sunucu üzerinde tutulan kullanıcı erişim dosyaları(log files) karmaşık, düzensiz ve herhangi bir anlam ifade etmeyen şekilde tutulmaktadır. Web sunucusu üzerinde tutulan log dosyalarından sağlıklı bilgi çıkarımı yapabilmek için gereksiz verilerden temizlenmesi ve belirli bir düzene sokulması gerekmektedir. Sunucular üzerinde karmaşık ve düzensiz bir şekilde tutulan log dosyalarındaki verilerin analiz değeri olmayan ilişkisiz verilerden temizlenmesi, belirli bir biçime getirilmesi ve veritabanına aktarılması işlemi ön işlem sürecidir.

Ön işlem süreci web kullanım madenciliğinin en önemli ve en uzun süren basamağıdır. Bu süreç sonrasında veri örüntü keşfi için uygun hale getirilmektedir. Bu süreçte önemli olan verinin orijinalliğinin korunmasıdır.

Ön işlem süreci veri temizleme, kullanıcı tanımlama, oturum tanımlama, yol tamamlama ve biçimlendirme olmak üzere dört adımda gerçekleşir. Verilerin temizlenmesi, kullanıcı ve oturum tanımlama aşamalarında sezgisel(heuristic) teknikler kullanılmaktadır (Cooley et al., 1999). Web kullanım verisine VM

tekniklerinin başarılı bir şekilde uygulanması, ön işlem sürecindeki işlemlerin doğru uygulanmasına büyük oranda bağımlıdır. Ön işlem sürecinin adımları Şekil 4.3'de gösterilmektedir.



Şekil 4.3. Web kullanım madenciliği ön işlem süreci adımları.

4.2.1.1. Veri Temizleme

Veri temizleme ön işlem sürecinde uygulanması gereken ilk adımdır. Elde edilen erişim kayıtlarının tamamı madencilik süreci için gerekli veriler değildir. Bu nedenle, erişim kayıtları içerisindeki geçerli ve gerekli olan veriler alınmalı diğerleri temizlenmelidir (Liu and Keselj, 2007). Temizliğe ihtiyaç duyulan gereksiz veya alakasız üç tür veri vardır. Bunlar HTML dosya içerisine gömülü kaynaklar, robot istekleri ve başarısız isteklerdir.

HTTP (Hyper Text Transfer Protocol) protokolü bağlantısız bir protokol olduğu için bir kullanıcının sayfa görüntüleme isteği erişim kayıtlarında birden fazla yer alacaktır. Bunun nedeni, sayfa içerisinde kullanılan resim dosyaları, stil (css) dosyaları, script dosyaları ve sayfa içerisinde kullanılan diğer dosyaların da erişim kayıtları içerisinde ayrı satırlar halinde yer almasıdır. Erişim kayıtları içerisinde yer

alan bu tür satırlar gömülü kaynakları göstermektedir ve silinmelidir. Bir log dosyası içerisinde yer alan sayfa isteği ile birlikte kaydedilen satırları içeren örnek Şekil 4.4'de gösterilmiştir.

```
2010-02-19 07:49:59 193.x.x.x GET /Default.aspx - 80 ……..
2010-02-19 07:50:00 193.x.x.x GET /ajaxtabs.js - 80 ……..
2010-02-19 07:50:00 193.x.x.x GET /yenimenu.js - 80 ……....
2010-02-19 07:50:00 193.x.x.x GET /anaDuyuruGrid.css - 80 ……..
2010-02-19 07:50:00 193.x.x.x GET /modaldbox.js - 80 ……....
2010-02-19 07:50:00 193.x.x.x GET /modaldbox.css - 80 ……
2010-02-19 07:50:00 193.x.x.x GET /images/menu_tanitim2.jpg - 80 ….....
2010-02-19 07:50:00 193.x.x.x GET /images/menu_tanitim2_sel.jpg - 80 .....
2010-02-19 07:50:00 193.x.x.x GET /images/menu_yonetim_sel.jpg - 80 ……
2010-02-19 07:50:00 193.x.x.x GET /images/menu_yonetim.jpg - 80 ……...
```

Şekil 4.4. Bir log dosyası içerisinde yer alan kayıtlar.

Web robotları (spider-crawler) web sitesi içerisindeki linkleri otomatik olarak çıkaran yazılımlardır. Google gibi arama motorları bir web sitesine ait tüm sayfaları ve linkleri tespit etmek için periyodik olarak bu tür araçları kullanılır. Bu tür araçlar tarafından yapılan sayfa istekleri robot istekleri oluşturmaktadır ve kullanıcı isteğinde olduğu gibi erişim kayıtları içerisinde yer alacaktır. Erişim kayıtları içerisinde yer alan bu tür kayıtlar da temizlenmelidir. Bir log dosyası içerisinde yer alan robot isteğine ait örnek satırlar Şekil 4.5'de gösterilmiştir.

```
2010-03-11 12:51:02 193.140.180.4 GET /Default.aspx - 80 - 123.125.66.91
Baiduspider+(+http://www.baidu.com/search/spider.htm) - 200 0 64 0 192 562
```

Şekil 4.5. Bir log dosyası içerisinde yer alan robot kayıtları.

Erişim kayıtları içerisindeki her bir istek için durum kodu (sc-status) tutulmaktadır. Bu durum kodu isteğin başarılı olup olmadığını tutmaktadır. Başarısız istekler madencilik süreci için gereksiz olabilir. 200 ile 299 arasındaki durum kodları başarılı istekler olduğu için istenilirse bunlar dışında kalan istekler silinebilir. Örneğin, 404 durum kodu istekte bulunulan kaynağın var olmadığını göstermektedir. Erişim kayıtları içerisinde yer alan başarısız istekler istenilirse silinebilir. Ancak, hatalı istekler, kırık linkler veya engelli girişler gibi analiz işlemleri yapılacaksa durum kodları dikkate alınacağı için başarısız erişimler silinmemelidir.

4.2.1.2. Kullanıcı Tanımlama

Web kullanım madenciliği analizi için bir kullanıcının doğrulanmasına ihtiyaç yoktur. Fakat farklı kullanıcıları ayırt etmeye ihtiyaç duyulur. Bir kullanıcı bir siteyi birden daha fazla kez ziyaret ettiği için erişim kayıtları her kullanıcı için çoklu oturumları kaydeder.

Kullanıcı tanımlama, benzer kullanıcılara ait olan aktiviteleri belirlemek için kullanılır. Kimlik doğrulama mekanizması kullanılmadığında benzersiz ziyaretleri ayırt etmek için çoğunlukla kullanıcı taraflı çerezler kullanılmaktadır. Fakat çerezler güvenlik amacıyla kullanıcı tarafından devre dışı bırakılabileceği veya silinebileceği için IP adresleri de kullanılabilir. Tek başına IP adresi benzersiz ziyaretleri tespit etmek için yani kullanıcı tanımlama için yeterli değildir. Çünkü vekil sunucu kullanılan sistemlerde tüm bilgisayarlar internete tek bir IP adresi ile çıkacağı için tüm kullanıcılar için aynı IP adresi görünecektir. Kimlik doğrulama veya kullanıcı taraflı çerezler olmaksızın kullanıcıları tanımlamak için IP adresi ile birlikte tarayıcı ve işletim sistemi bilgilerini tutan user-agent bilgisi de kullanılabilir. IP adresi ve user-agent bilgisi kullanılarak kullanıcı tanımlama örneği Çizelge 4.4'de gösterilmiştir.

Çizelge 4.4. IP adresi ve user-agent bilgisi kullanılarak kullanıcı tanımlama. a) Örnek erişim kayıtları. b) Oluşturulan kullanıcılar.

Time	IP	URL	Referrer	User-agent
00:01	1.2.3.4	A	-	IE5;Win2K
00:09	1.2.3.4	B	A	IE5;Win2K
00:10	2.3.4.5	C	-	IE6;WinXP;SP1
00:12	2.3.4.5	B	C	IE6;WinXP;SP1
00:15	2.3.4.5	E	C	IE6;WinXP;SP1
00:19	1.2.3.4	C	A	IE5;Win2K
00:22	2.3.4.5	D	B	IE6;WinXP;SP1
00:22	1.2.3.4	A	-	IE6;WinXP;SP1
00:25	1.2.3.4	E	C	IE5;Win2K
00:25	1.2.3.4	C	A	IE6;WinXP;SP1
00:33	1.2.3.4	B	C	IE6;WinXP;SP1
00:58	1.2.3.4	D	B	IE6;WinXP;SP1
01:10	1.2.3.4	E	D	IE6;WinXP;SP1
01:15	1.2.3.4	A	-	IE5;Win2K
01:16	1.2.3.4	C	A	IE5;Win2K
01:17	1.2.3.4	F	C	IE6;WinXP;SP1
01:26	1.2.3.4	F	C	IE5;Win2K
01:30	1.2.3.4	B	A	IE5;Win2K
01:36	1.2.3.4	D	B	IE5;Win2K

(a)

Kullanıcı 1	00:01	1.2.3.4	A	-
	00:09	1.2.3.4	B	A
	00:19	1.2.3.4	C	A
	00:25	1.2.3.4	E	C
	01:15	1.2.3.4	A	-
	01:16	1.2.3.4	C	A
	01:26	1.2.3.4	F	C
	01:30	1.2.3.4	B	A
	01:36	1.2.3.4	D	B
Kullanıcı 2	00:10	2.3.4.5	C	-
	00:12	2.3.4.5	B	C
	00:15	2.3.4.5	E	C
	00:22	2.3.4.5	D	B
Kullanıcı 3	00:22	1.2.3.4	A	-
	00:25	1.2.3.4	C	A
	00:33	1.2.3.4	B	C
	00:58	1.2.3.4	D	B
	01:10	1.2.3.4	E	D
	01:17	1.2.3.4	F	C

(b)

4.2.1.3. Oturum Tanımlama

Bir oturum kullanıcının siteye girişi ile çıkışı arasındaki sürede gerçekleştirdiği aktiviteler grubu olarak tanımlanabilir. Bu nedenle oturum tanımlama işlemi, web oturumları içerisindeki her bir kullanıcının davranış ve aktivite kayıtlarının kümelenmesidir (Cooley et al., 1999). Oturum tanımlamadaki amaç oturumlar içerisindeki her kullanıcının sayfa erişimlerini birbirinden ayırt etmektir. Kimlik doğrulama sistemi bulunmayan web sitelerinde oturum tanımlama işlemi için sezgisel yaklaşımlar kullanılmaktadır. Oturum süresi temelli (session-duration-based), sayfada kalma süresi temelli (page-stay-time-based) ve referans temelli (referrer-basic huristic) olmak üzere üç sezgisel yaklaşım bulunmaktadır (Cooley at al., 1999; Spiliopoulou et al., 2003; Berendt et al., 2001).

Oturum süresi temelli sezgisel yaklaşımda (h1) bir oturumun süresi eşik değeri (θ) ile belirlenmektedir. Her bir oturum belirlenen eşik değerini aşmamalıdır. Catledge ve Pitkow (1995) ziyaretçilerin siteden ayrılma sürelerini ölçmüştür ve bu değer 1.5 standart sapma ile 25.5 dakikadır. Bu eşik değeri birçok çalışmada ortalama 30 dakika olarak kullanılmaktadır. Herhangi bir kullanıcı için oturumun başlangıç zamanı t_0 ve bitiş zamanı t_n olarak düşünüldüğünde, t_n - $t_0 \leq \theta$ şartını sağlayan tüm erişim aktiviteleri aynı oturum içerisinde değerlendirilir (Liu, 2006). Zamana yönelik oturum süresi temelli yaklaşıma göre oturum oluşturma Çizelge 4.5'de gösterilmiştir.

Çizelge 4.5. Zamana yönelik oturum süresi temelli yaklaşıma göre oturum oluşturma.

	Zaman	IP	URL	Referrer
Kullanıcı 1	00:01	1.2.3.4	A	-
	00:09	1.2.3.4	B	A
	00:19	1.2.3.4	C	A
	00:25	1.2.3.4	E	C
	01:15	1.2.3.4	A	-
	01:16	1.2.3.4	C	A
	01:26	1.2.3.4	F	C
	01:30	1.2.3.4	B	A
	01:36	1.2.3.4	D	B
Kullanıcı 2	00:10	2.3.4.5	C	-
	00:12	2.3.4.5	B	C
	00:15	2.3.4.5	E	C
	00:22	2.3.4.5	D	B
Kullanıcı 3	00:22	1.2.3.4	A	-
	00:25	1.2.3.4	C	A
	00:33	1.2.3.4	B	C
	00:58	1.2.3.4	D	B
	01:10	1.2.3.4	E	D
	01:17	1.2.3.4	F	C

Kullanıcı 1 Oturum 1	00:01	1.2.3.4	A	-
	00:09	1.2.3.4	B	A
	00:19	1.2.3.4	C	A
	00:25	1.2.3.4	E	C
Kullanıcı 1 Oturum 2	01:15	1.2.3.4	A	-
	01:16	1.2.3.4	C	A
	01:26	1.2.3.4	F	C
	01:30	1.2.3.4	B	A
	01:36	1.2.3.4	D	B
Kullanıcı 2 Oturum 1	00:10	2.3.4.5	C	-
	00:12	2.3.4.5	B	C
	00:15	2.3.4.5	E	C
	00:22	2.3.4.5	D	B
Kullanıcı 3 Oturum 1	00:22	1.2.3.4	A	-
	00:25	1.2.3.4	C	A
	00:33	1.2.3.4	B	C
Kullanıcı 3 Oturum 2	00:58	1.2.3.4	D	B
	01:10	1.2.3.4	E	D
	01:17	1.2.3.4	F	C

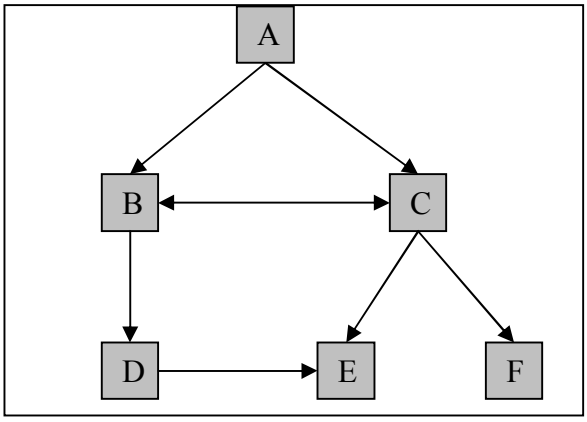
Sayfada kalma süresi temelli sezgisel yaklaşımda (h2) bir sayfada harcanan toplam süre eşik değeri (θ) ile belirlenmektedir. Her bir sayfada harcanan süre belirlenen eşik değerini aşmamalıdır. Catledge and Pitkow (1995) bir site içerisindeki ortalama hareketsizlik zamanını ölçmüştür ve bu değer 9.3 dakikadır. Yapılan birçok çalışmada bu eşik değeri 10 dakika olarak belirlenmiştir. Herhangi bir kullanıcı için bir sayfaya giriş zamanı t_0 ve bir sonraki sayfaya giriş zamanı t_1 olarak düşünüldüğünde, $t_1 - t_0 \leq \theta$ şartını sağlayan tüm erişim aktiviteleri aynı oturum içerisinde değerlendirilir (Liu, 2006). Zamana yönelik sayfada kalma süresi temelli yaklaşıma göre oturum oluşturma örneği Çizelge 4.6'da gösterilmiştir.

Çizelge 4.6. Zamana yönelik sayfada kalma süresi temelli yaklaşıma göre oturum oluşturma.

	Zaman	IP	URL	Referrer
Kullanıcı 1	00:01	1.2.3.4	A	-
	00:09	1.2.3.4	B	A
	00:19	1.2.3.4	C	A
	00:25	1.2.3.4	E	C
	01:15	1.2.3.4	A	-
	01:16	1.2.3.4	C	A
	01:26	1.2.3.4	F	C
	01:30	1.2.3.4	B	A
	01:36	1.2.3.4	D	B
Kullanıcı 2	00:10	2.3.4.5	C	-
	00:12	2.3.4.5	B	C
	00:15	2.3.4.5	E	C
	00:22	2.3.4.5	D	B
Kullanıcı 3	00:22	1.2.3.4	A	-
	00:25	1.2.3.4	C	A
	00:33	1.2.3.4	B	C
	00:58	1.2.3.4	D	B
	01:10	1.2.3.4	E	D
	01:17	1.2.3.4	F	C

Kullanıcı 1 Oturum 1	00:01	1.2.3.4	A	-
	00:09	1.2.3.4	B	A
	00:19	1.2.3.4	C	A
	00:25	1.2.3.4	E	C
Kullanıcı 1 Oturum 2	01:15	1.2.3.4	A	-
	01:16	1.2.3.4	C	A
	01:26	1.2.3.4	F	C
	01:30	1.2.3.4	B	A
	01:36	1.2.3.4	D	B
Kullanıcı 2 Oturum 1	00:10	2.3.4.5	C	-
	00:12	2.3.4.5	B	C
	00:15	2.3.4.5	E	C
	00:22	2.3.4.5	D	B
Kullanıcı 3 Oturum 1	00:22	1.2.3.4	A	-
	00:25	1.2.3.4	C	A
	00:33	1.2.3.4	B	C
Kullanıcı 3 Oturum 2	00:58	1.2.3.4	D	B
Kullanıcı 3 Oturum 3	01:10	1.2.3.4	E	D
	01:17	1.2.3.4	F	C

Referans temelli sezgisel yaklaşımda (h-ref) oturumları oluşturmak için Şekil 4.6'da gösterildiği gibi web sayfaları arasındaki linkleri içeren site haritası, ziyaret edilen sayfa bilgisi (cs-uri-stem) ve referans adres bilgisi (cs-referrer) kullanılır. İstekte bulunulan sayfaya önceden ziyaret edilen sayfalardan erişilemiyorsa farklı bir oturum olarak tanımlanmalıdır (Nadjarbashi and Ghorbani, 2004).



Şekil 4.6. Örnek web sitesi haritası.

p ve q ardışık iki istek olmak üzere t_p ve t_q erişim zamanlarıdır ve p ile başlayan bir S oturumu bulunmaktadır. Eğer q için referans, S oturumu içerisinde önceden ziyaret edilmiş sayfa ise veya referans belirsiz ve belirli bir Δ bekleme süresi için (t_q - $t_p \leq \Delta$) ise S oturumu içerisine q isteği de dahildir. Aksi durumda q yeni bir oturum olarak oluşturulur (Berendt et al., 2001).

Erişim kayıtları içerisindeki referans adresin belirsizliği (-) çeşitli durumlarda meydana gelir. Bu durumlar aşağıda açıklanmıştır (Berendt et al., 2001; Nadjarbashi and Ghorbani, 2004);

- Eğer farklı bir siteden gelinmişse veya gezintiye başlangıç sayfasından başlanmışsa referans adres belirsiz olacaktır. Ayrıca, mevcut siteye arama motorları veya link veren sitelerden gelindiğinde gelinen sitenin bilgisi referans adres olarak görünecektir.
- Belirli bir adrese doğrudan erişilirse veya tarayıcıya ait yer imleri aracılığı ile ulaşılmışsa referans adres belirsiz (-) olacaktır.
- Frameset içeren sayfalar tekrar yüklendiğinde referans adres belirsiz olacaktır.
- Tüm sayfalar için tarayıcı üzerinde bulunan ileri ve geri butonu yardımıyla yapılan gezintiler referans bilgisinin belirsiz olmasına neden olacaktır.
- Birden fazla frameset içeren sayfalarda framesetlerin içerdiği sayfalar sırasıyla yükleneceği için referans bilgisi belirsiz olacaktır.

Bekleme süresi Δ referans adresi belirsiz olan istekler için kullanılmaktadır. Yapılan çalışmalarda bu süre 10 saniye olarak kullanılmıştır (Berendt et al., 2001; Nadjarbashi and Ghorbani, 2004; Spiliopoulou et al., 2003; Berendt et al., 2002).

Aynı kullanıcıya ait aktivite kayıtlarına uygulanan referans temelli sezgisel yaklaşım ile oturum oluşturma örneği Çizelge 4.7'de gösterilmiştir.

Çizelge 4.7. Referans temelli sezgisel yaklaşım ile oturum oluşturma.

	Zaman	IP	URL	Referrer
Kullanıcı 1	00:01	1.2.3.4	A	-
	00:09	1.2.3.4	B	A
	00:19	1.2.3.4	C	A
	00:25	1.2.3.4	E	C
	01:15	1.2.3.4	A	-
	01:16	1.2.3.4	C	A
	01:26	1.2.3.4	F	C
	01:30	1.2.3.4	B	A
	01:36	1.2.3.4	D	B

Kullanıcı 1 Oturum 1	00:01	1.2.3.4	A	-
	00:09	1.2.3.4	B	A
	00:19	1.2.3.4	C	A
	00:25	1.2.3.4	E	C
Kullanıcı 1 Oturum 2	01:15	1.2.3.4	A	-
	01:16	1.2.3.4	C	A
	01:26	1.2.3.4	F	C
	01:30	1.2.3.4	B	A
	01:36	1.2.3.4	D	B

4.2.1.4. Hangi Oturum Oluşturma Yaklaşımı Seçilmelidir?

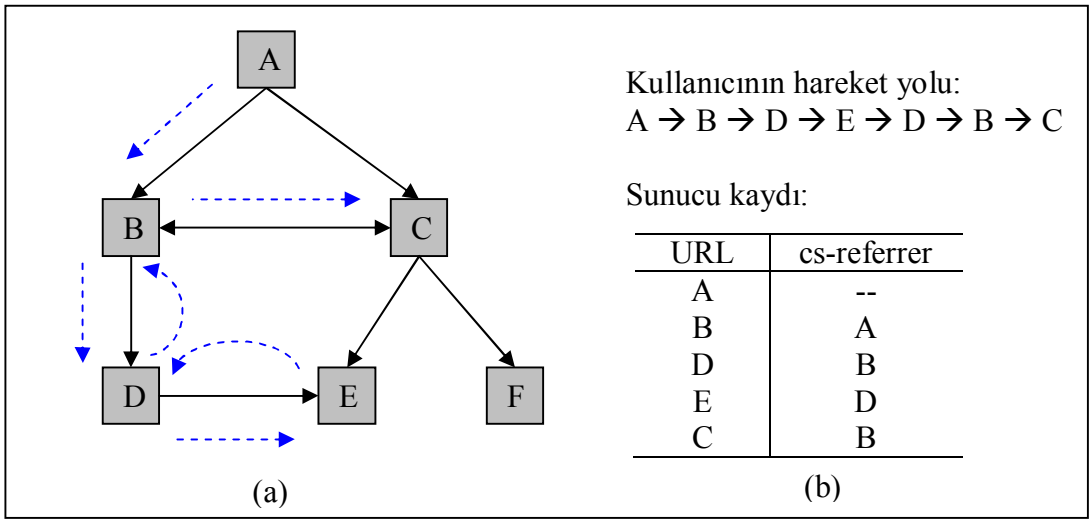
Oturum yaklaşımının seçimi yapılacak analizin amacına ve veri karakteristiğine bağlıdır. Bu durumlar aşağıda açıklanmıştır;

- Çok kısa oturumlarda h-ref oturum oluşturmak için daha iyi ama genel olarak h1 ve h2 daha güçlüdür.
- Bireysel kullanıcı site ziyaretleri daha kısa süreli oturumları daha yoğundur. Bu tür durumlar için *h2* yaklaşımı *h1*'den daha iyi performans verebilir.
- Erişim kayıtlarındaki zaman bilgisi güvenilir olmadığı zaman h-ref yaklaşımı daha iyi bir seçimdir.
- Referans tabanlı yaklaşımlar frame tabanlı sitelerde kötü performans göstermektedir.
- Yapılan deney sonuçları h-ref ve h2 kombinasyonu istenilen sonuçları verebileceğini gösteriyor.

4.2.1.5. Yol Tamamlama

Erişim kayıtları vekil sunucuda tutuluyorsa veya site gezintisi esnasında ön bellekten sayfa ziyaretleri gerçekleşiyorsa log dosyaları içerisine kaydedilmeyen önemli erişimler vardır. Örneğin, site içerisinde gezinti yapan bir kullanıcı tarayıcı üzerinden geri düğmesi ile gezinti yaptığında bu işlem erişim kayıtları içerisinde yer almayacaktır veya vekil sunucudan istekte bulunulan sayfa sunucunun ön belleğinden gösterilirse yine bu erişimde log dosyası içerisinde yer almayacaktır. Yol tamamlamanın görevi erişim kayıtları içerisinde bulunan bu eksik referansları tamamlamaktır. Kullanıcı tanımlama için kullanılan yöntemlere benzer yöntemler yol tamamlama için de kullanılabilir. Bir sayfanın hangi istek sayfasından geldiğini görmek için kullanıcı referans kayıtlarını (cs-referrer) kullanmak gerekir. Bir sayfa isteği kullanıcının istekte bulunduğu son sayfaya doğrudan bağlı değilse yani link içermiyorsa, log içerisinde tutulan cs-referrer alanı isteğin geldiği sayfayı görmek için kontrol edilebilir. Eğer cs-referrer alanı log içerisinde tutulmuyorsa web sitesine ait site topolojisi kullanılabilir (Chaofeng, 2006).

Eksik referansların gösterildiği örnek Şekil 4.7'de verilmiştir.



Şekil 4.7. Yol tamamlama. a) Kullanıcının hareket yolu. b) Sunucu kayıtları.

Şekil 4.7 (a)'da görüldüğü üzere A → B → D → E → D → B → C yollarını izleyen bir kullanıcının E sayfasından sonra C sayfasını ziyaret etmiştir ama E sayfasından C

sayfasına link yoktur. Kullanıcı E sayfasından sonra tarayıcı üzerinden geri tuşu ile D sayfasına oradan da tekrar geri tuşu ile B sayfasına geçmiştir. Sunucu kaydında C sayfasının referansı B sayfası olduğu için ve B sayfasından C sayfasına geçilebildiği için B sayfasından C sayfasına geçmiştir. Şekil 4.7 (b)'deki sunucu kayıtları incelendiğinde E sayfasından B sayfasına geri tuşu ile geçişler log içerisinde tutulmamıştır. Bu nedenle referans alanı eksik ve belirsiz bilgilerle kayıtlara geçmiştir. Bu problemin çözülebilmesi için sayfalardan gidilebilecek yollar dikkate alınmalı ve geri referansların gerekli olanlarından en az biri seçilmelidir (Liu, 2006; Daş, 2008).

4.2.2. Örüntü Keşfi

Örüntü keşfi aşamasında ön işlem sürecinden sonra elde edilen düzenli ama anlamsız olan verilerden, VM yöntemlerini kullanarak istenilen faydalı ve gerekli bilgilerin ortaya çıkarılması gerçekleştirilmektedir.

Örüntü keşfi istatistik, VM, makine öğrenme ve örüntü tanıma gibi çeşitli alanlarda geliştirilen yöntem ve algoritmaları kullanmaktadır (Srivastava et al., 2000).

Web kullanım madenciliği örüntü keşfi aşaması için istatistiksel analiz, birliktelik kuralları, yol analizi, kümeleme, sınıflandırma ve sıralı örüntüler yaygın olarak kullanılmaktadır.

4.2.2.1. İstatistiksel Analiz

İstatistiksel analiz bir web sitesi ziyaretçileri hakkında bilgi çıkartmak için kullanılan bir yöntemdir. Oturum dosyaları analiz edilerek sayfa görüntülemeler, görüntüleme süresi ve izlenen yolun uzunluğu gibi değişkenler üzerinde farklı tanımlayıcı istatistiksel analizler gerçekleştirilebilir. Web trafik analiz araçlarının çoğu en sık erişilen sayfalar, bir sayfanın ortalama erişim süresi, site içerisindeki hatalı sayfaların ve kırık köprü bağlantılarının tespit edilmesi veya bir site içerisindeki yolun ortalama uzunluğu gibi istatistiksel bilgileri içeren dönemsel rapor üretir. Bu rapor yetkisiz giriş yapılmak istenen noktaları tespit veya çoğunlukla istekte bulunulan adresleri

bulma gibi düşük seviye hata analizlerini içerebilir. Ayrıntılı analizlerde eksiklik olmasına rağmen bu tür bilgiler sistem performansını geliştirmek, sistem güvenliğini artırmak, site düzenleme işlemini kolaylaştırmak ve pazarlama kararları için destek sağlamak için yararlı olabilir (Srivastava et al., 2000).

4.2.2.2. Birliktelik Kuralları

Genellikle alışveriş uygulamalarında kullanıldığı için market sepet analizi olarak da bilinmektedir. Bu yöntemdeki amaç bir küme içerisindeki nesnelerin birbirleri ile olan bağlarının tespit edilmesidir.

Toplanan ve depolanan verinin her geçen gün artması, şirketlerin kendi veritabanlarındaki öğelerin birliktelik kurallarını ortaya çıkarmaya itmektedir. Birliktelik kurallarının çıkarımı katalog tasarımı, müşterilerin satın alma alışkanlıklarına göre sınıflandırılması, mağaza ürün yerleşim planı gibi pek çok uygulama alanında kullanılabilir (Daş, 2008).

Örneğin, A ürününün alınması ile B ürününün veya C ürünün alınması arasındaki işlemlerde bir bağlantı olup olmadığının tespit edilmesi ve eğer bağlantı var ise bu bağlantılar arasındaki kuvvet veya önem derecesinin ortaya çıkartılması sağlanabilir. Bu analizin amacı A ürününü alan kişilerin B veya C ürünleri alımları ile ilgili olarak kuvvetli bir bağın bulunup bulunmadığını kontrol etmek eğer var ise bununla ilgili olarak örneğin müşterilere promosyonlar veya ürünlerin raflarının daha yakın yerlere yerleştirilmesini sağlamak olabilir. Bu işlem bir web sitesi içerisinde sayfaların yapılandırılması amacı ile de kullanılabilir (Gezer vd., 2007).

Birliktelik kuralları eş zamanlı olarak gerçekleşen ilişkilerin tanımlanmasında kullanılır. Keşfedilen örüntüler uygulamada sıklıkla birlikte geçen nitelik değerleri arasındaki ilişkiyi gösterir. Şampuan ve saç kremi satın alan müşterilerin %20 ihtimalle saç jölesi de almaları, kola satın alan müşterilerin, %75 ihtimalle patates cipsi de almaları ya da düşük yağlı peynir ve yağsız yoğurt satın alan müşterilerin, %85 ihtimalle diyet süt de almaları birliktelik kurallarına örnek olarak verilebilir. Bu

tür birliktelik örüntüleri ancak, örüntüde yer alan öğelerin işlemleri birden fazla tekrarlandığında potansiyel olarak bu kuralın geçerliliği sağlanabilir.

Birliktelik kurallarının kullanıldığı en tipik örnek market sepeti uygulamasıdır. Bu işlem, müşterilerin yaptıkları alışverişlerdeki ürünler arasındaki birliktelikleri bularak müşterilerin satın alma alışkanlıklarını analiz eder. Bu tip birlikteliklerin keşfedilmesi, müşterilerin hangi ürünleri bir arada aldıkları bilgisini ortaya çıkarır ve market yöneticileri de bu bilgi ışığında daha etkili satış stratejileri geliştirebilirler. Örneğin, bir marketin müşterilerinin süt ile birlikte ekmek satın alma oranı yüksekse, market yöneticileri süt ile ekmek raflarını yan yana koyarak ekmek satışlarını arttırabilirler.

Sepet analizinde mallar arasındaki bağıntı, destek ve güven değerleri aracılığıyla hesaplanır. Destek veri içerisinde bu bağıntının ne kadar sık olduğunu, güven de X ürününü almış bir kişinin hangi olasılıkla Y ürününü alacağını ifade eder. Bağıntının önemli olabilmesi için her iki değerin de olabildiğince büyük olması gerekir.

X ve Y farklı ürünler olmak üzere, X ürünü için destek tüm alışverişler içinde X ürününün oranıdır. |X|, X ürünü içeren alışverişlerin sayısını, |D| yapılan tüm alışverişlerin sayısını göstermek üzere;

$$\text{Destek (X)} = \frac{|X|}{|D|} \tag{4.1}$$

X ve Y ürünleri için destek, X ve Y ürünlerini içeren alışveriş sayısı olmak üzere;

$$\text{Destek } (X \Rightarrow Y) = \frac{|X.Y|}{|D|} \tag{4.2}$$

X ve Y ürünleri için güven ise;

$$\text{Güven } (X \Rightarrow Y) = \frac{\text{destek}(X.Y)}{\text{destek}(X)} \tag{4.3}$$

Örneğin, bir X ürünü satın alan müşteriler aynı zamanda Y ürünün de satın alıyorlarsa, bu durumun birliktelik kuralı ile gösterimi;

$$X \Rightarrow Y \text{ [destek=\%2, güven=\%60]} \quad (4.4)$$

Buradaki destek ve güven ifadeleri, kuralın ilginçlik ölçüleridir. Sırasıyla, keşfedilen kuralın kullanışlığını ve doğruluğunu gösterir. Yukarıdaki bağıntı için %2 oranındaki bir destek değeri, analiz edilen tüm alışverişlerden %2'sinde X ile Y ürünlerinin birlikte satıldığını belirtir. %60 oranındaki güven değeri ise X ürününü satın alan müşterilerinin %60'ının aynı alışverişte Y ürününü de satın aldığını ortaya koyar. Kullanıcı tarafından minimum destek eşik değeri ve minimum güven eşik değeri belirlenir ve bu değerlere eşit veya bu değeri aşan birliktelik kuralları dikkate alınır.

4.2.2.3. Sınıflandırma

Örüntü keşfi uygulamalarında en çok kullanılan yöntemlerden birisi olan sınıflandırma, bir veriyi daha önceden tanımlanmış sınıflara dağıtma tekniğidir.

Sınıflandırma, daha önceden belirlenmiş ölçütlere göre, örneğin yaşa, cinsiyete, gelir durumuna, eğitim düzeyine ve müşterinin kredi borcunu zamanında ödeyip ödememesine, bir kampanyaya olumlu cevap verip vermemesine, hedeflenen değerlerin üzerinde bulunup bulunmamasına yani ilgilenilen herhangi bir özelliğe veya birkaç ölçüte göre yapılır (Daş, 2008). Web etki alanında, sınıflandırma tekniği kullanarak müşterilerinin hangi sınıf veya kategoride bir profile sahip olduğu belirlenebilir. Örneğin; internet bankacılığında yaptıkları elektronik fon transferi sıklıklarına göre sınıflandırmada internet müşterileri, “seyrek” kullanıcı, “orta sıklıkta” kullanıcı ve “sık” kullanıcı olarak sınıflandırılabilir. Müşteriler bu şekilde gruplandıktan sonra amaç, her bir grubun özelliklerini analiz etmek, profilini ortaya çıkarmak ve bu grupların özelliklerini, tutum ve davranışlarını içeren bir davranış geliştirebilmektir. Sınıflandırma işlemi; karar ağaçları, bayes sınıflayıcıları, en yakın komşu ve destek vektör makineleri gibi denetlenen tümevarımsal öğrenim algoritmaları kullanılarak yapılabilir.

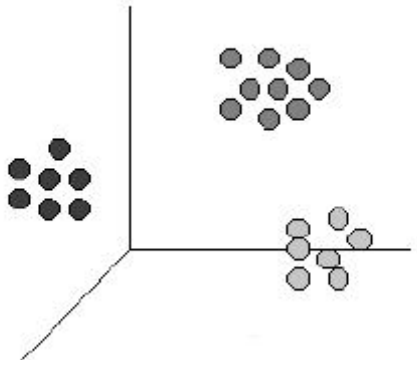
Sınıflama algoritması bir sınıfı diğerinden ayıran örüntüleri keşfeder. Sınıflama algoritmaları iki şekilde kullanılır (Daş, 2008);

- Karar değişkeni ile sınıflandırmada seçilen bir niteliğin aldığı değerlere göre sınıflandırma işlemi yapılır. Seçilen nitelik karar değişkeni adını alır ve veri tabanındaki çoklular karar değişkeninin değerlerine göre sınıflara ayrılır. Bir sınıfta yer alan çoklular karar değişkeninin değeri açısından özdeştir.
- Örnek ile sınıflandırmada veri tabanındaki çoklular iki kümeye ayrılır. Kümelerden biri pozitif, diğeri negatif çokluları içerir. Yaygın kullanım alanları, banka kredisi onaylama işlemi, kredi kartı sahteciliği tespiti ve sigorta risk analizidir.

4.2.2.4. Kümeleme

Verilerin bir grupta toplanması kümeleme olarak adlandırılır. Mevcut veriler incelenerek benzer ya da birbiriyle ilişkili olan veriler aynı kümeyi oluştururken farklı ya da ilişkili olmayan veriler başka bir kümeyi oluşturur (Larose, 2005). Dolayısıyla elde edilen kümede ki veriler homojendir. Kümelemenin temel hedefi, dağınık bir halde bulunan verileri benzerliklerine göre bir araya getirip sınıflandırarak işlenebilir hale getirmektir (Kaya ve Köymen, 2008; Özekes, 2003). Çoğunlukla yapay sinir ağları ve istatistiksel metotlardan yararlanılır ve örüntü tanıma, görüntü işleme, ekonomi bilimi (özellikle market araştırma), internet üzerinde doküman sınıflandırılması, benzer ortak arkadaş grupları keşfetme, veri madenciliği, istatistik, biyoloji ve makine öğrenmesi gibi pek çok alanda kullanılır (Kaya ve Köymen, 2008). Kümeleme modeli sınıflandırmaya benzer bir modeldir ama aralarındaki en büyük fark kümeler önceden belirlenmemiştir. Kümeleme modelinde, sınıflama modelinde olan veri sınıfları yoktur. Verilerin herhangi bir sınıfı bulunmamaktadır. Sınıflama modelinde, verilerin sınıfları bilinmekte ve yeni bir veri geldiğinde bu verinin hangi sınıftan olabileceği tahmin edilmektedir. Oysa kümeleme modelinde, sınıfları bulunmayan veriler gruplar halinde kümelere ayrılırlar (Özekeş, 2003; Ramkumar, 1998).

Kümeleme modellerinde amaç, Şekil 4.8'de görüldüğü gibi küme üyelerinin birbirlerine çok benzediği, ancak özellikleri birbirlerinden çok farklı olan kümelerin bulunması ve verilerin bu farklı kümelere bölünmesidir.



Şekil 4.8. Kümeleme modeli.

Kümeleme, benzer özelliklere sahip nesneler kümesini gruplamak için kullanılan bir tekniktir. Web kullanım alanında kullanıcı kümeleri ve sayfa kümeleri olmak üzere iki türe ayrılır. Kullanıcıların kümelenmesi, benzer tarama desenleri sergileyen kullanıcıların gruplandırılmasını sağlar. Bu bilgiler özellikle e-ticaret uygulamaları veya kullanıcılar için web içeriğini kişiselleştirmede pazar bölümlemesi yapmak amacı ile kullanıcı demografik bilgilerini çıkarmak için kullanışlıdır. Örneğin, bir portal içerisinde oyun ve spor sayfasına girenleri bir grup içerisine alıp bir sonraki bağlantıyı yaptıklarında bu konuda reklamların sayfalarda gelmesini sağlamak gibi (Gezer vd., 2007). Diğer taraftan, sayfaların kümelenmesi, ilgili içeriğe sahip sayfa gruplarını keşfedecektir. Bu bilgi arama motorları ve hizmet sağlayıcılar için kullanışlıdır. Her iki uygulama da kullanıcı sorguları veya ihtiyaç duyulan geçmiş bilgilere göre kullanıcılar için ilgili linkleri önermek için kullanılabilir (Srivastava et al., 2000).

4.2.2.5. Sıralı Örüntüler

Sıralı (ardışık) örüntülerin keşfi, belirli bir zaman içerisinde olaylar ya da oturumlar kümesindeki bir öğeden diğer bir öğeyi takip eden örüntüleri bulmaya çalışmaktadır. Sıralı örüntü bulma işleminde, belirli zaman aralıklarında oturumlar incelenir ve karşılaştırmalar yapılır. Sıralı örüntülerin bulunması, gelecekteki eğilimi tahmin

edecek web pazarlamacıları için oldukça anlamlıdır. Böylece, bir web sitesinde yapılan ilanlar ya da ürün satışları belirli kullanıcı gruplarına yönlendirilebilecektir (Cooley et al., 1997).

/bilgisayarlar/ yolunu izleyerek ürün satın alan müşterilerin %50'si, 15 gün içerisinde /aksesuarlar/ yolunu izleyerek sipariş verebilir, X ameliyatı yapıldığında, 15 gün içinde %45 ihtimalle Y enfeksiyonun oluşması, çekiç satın alan bir müşterinin ilk üç ay içerisinde %15, bu dönemi izleyen diğer üç ay içerisinde de %10 ihtimalle çivi satın alacak olması sıralı örüntülere örnek olarak verilebilir.

4.2.3. Örüntü Analizi

Örüntü analizi web kullanım madenciliğinin son adımıdır. Örüntü analizinin amacı bulunan örüntülerden ilginç olmayan kuralları, istatistikî bilgileri ya da örüntüleri elemektir (Srivastava et al., 2000; Cooley et al., 1997). Genellikle örüntü analiz işlemi web madenciliği uygulamaları tarafından elde edilir. SQL, MySQL gibi veritabanı uygulamaları ve On-Line Analytical Processing (OLAP) yaygın olarak kullanılan bilgi sorgulama mekanizmalarıdır.

Görsel teknikler olarak daha çok grafiksel örüntüler, farklı değerlerle yoğun ve dikkat çeken örüntüler, işaretlenmiş renkler göz önünde bulundurulmaktadır (Cooley et al., 1999). Örüntü analizi konusunda yapılmış birçok çalışma ve uygulamalar mevcuttur. Örneğin; Iocchi (1999) makale çalışmasında, web kullanım madenciliği uygulaması ile kullanımı kolay ve anlaşılır kullanıcı ara yüzü sayesinde kullanıcının istekleri ve seçimleri doğrultusunda örüntü analizi yapılabilmektedir. Örüntü analizinde önemli olan konulardan biri de ilginç örüntülerinin nasıl öğrenileceğidir. Web kayıt dosyalarının temizlenip, istatistikî bilgilerin elde edilmesini sağlayan Nihuo (http://www.nihuo.com, 2010), Sarg (http://sarg.sourceforge.net, 2010), eWebLog (http://www.esoftys.com, 2010), NetIQ (http://www.netiq.com, 2010), WebTrends (http://www.webtrends.com, 2010) gibi birçok farklı program bulunmaktadır.

4.3. APRIORI ALGORİTMASI

Apriori algoritması birliktelik kuralı çıkarım algoritmaları içerisinde en fazla bilinen ve kullanılan algoritmadır. Sık geçen öğeleri bulmak için birçok kez veritabanını taramak gerekir, bu taramalar aşamasında apriori algoritmasının birleştirme, budama işlemleri ve minimum destek ölçütü yardımı ile birliktelik ilişkisi olan öğeler bulunur. Apriori algoritması k+1 adet sık geçen öğe kümesini bulmak için k adet sık geçen öğe kümesine ihtiyaç duyar (Han and Kamber, 2001).

Sık geçen öğe kümelerini bulmak için veritabanını ilk taramada bir elemanlı minimum destek metriğini sağlayan sık geçen öğe kümeleri bulunur ve L_1 şeklinde gösterilir. L_1, L_2'yi ve L_2 ise L_3'ü bulmak için kullanılır. İzleyen taramalarda bir önceki taramada bulunan sık geçen öğe kümeleri aday kümeler (C_k) adı verilen yeni potansiyel sık geçen öğe kümelerini üretmek için kullanılır. Aday kümelerin destek değerleri tarama sırasında hesaplanır ve aday kümelerden minimum destek metriğini sağlayan kümeler o geçişte üretilen sık geçen öğe kümeleri olur. Sık geçen öğe kümeleri bir sonraki geçiş için aday küme olurlar. Bu süreç yeni bir sık geçen öğe kümesi bulunmayana kadar devam eder. Her bir L_k'yı bulmak için veritabanını tamamen taramak gerekir. Apriori'nin verimliliğini artırmak ve arama zamanını kısaltmak için kullanılan önemli bir özeliği vardır (Han and Kamber, 2001). Bu özellik;

- Bir sık geçen öğenin bütün boş olmayan alt kümeleri de sık geçmektedir. Bu özellik şu gözleme dayanmaktadır: eğer bir I öğe kümesi minimum destek değerini sağlamıyorsa (P(I)<min_des) I sık geçen öğe değildir. Eğer I öğe kümesine yeni öğeler eklenirse, oluşan I $\cup$ A kümesi de sık geçen değildir (P(I $\cup$ A)<min_des). Yani, eğer bir öğe kümesi minimum destek değerini sağlamıyorsa o kümenin bütün süper kümeleri de destek değerini sağlayamaz.

Apriori algoritmasının kullanımı ve L_k bulunurken L_{k-1}'den nasıl yararlanıldığı birleştirme ve budama adı verilen iki aşamadan oluşmaktadır (Han and Kamber, 2001).

4.3.1. Birleştirme Adımı

L_k'yı bulmak için L_{k-1} öğe kümesi kendisi ile birleştirilerek C_k aday kümesi oluşturulur. I_1 ve I_2, L_{k-1}'de bulunan öğe kümeleri olsun. $I_i[j]$ gösterimi, I_i'deki j'nci öğeyi temsil eder (örneğin; $I_1[k-2]$, I_1'deki sondan 2. öğeyi ifade eder). Geleneksel olarak apriori, bir işlem veya öğe kümedeki öğelerin sözlüksel sıraya göre dizildiğini kabul eder. L_{k-1}'in ilk öğelerinin (k-2) ortak olması gerekir. L_{k-1}'in üyeleri olan I_1 ve I_2 , $(I_1[1]=I_2[1] \wedge I_1[1]=I_2[1] \wedge \ldots \wedge I_1[k-2] = I_2[k-2] \wedge I_1[k-1] < I_2[k-1])$ şartı sağlanıyorsa birleştirilebilir. $I_1[k-1] < I_2[k-1]$ şartı kopyaların oluşmasını engeller. I_1 ve I_2'nin birleştirilmesiyle oluşan sonuç küme $I_1[1]I_1[2]\ldots I_1[k-1]I_2[k-2]$ 'dir (Han and Kamber, 2001).

4.3.2. Budama Adımı

Aday küme C_k, L_k'nın süper kümesidir. C_k'nın içerdiği öğeler sık öğe de olabilir sık geçmeyen öğe de, fakat sık geçen k-öğe kümelerinin tamamı C_k içerisinde yer alır. Veritabanının taraması ile C_k'nın içerdiği öğelerden minimum destek değerini sağlayan öğeler L_k'nın öğelerini oluşturmaktadır. C_k öğe kümesinin çok büyük boyutlarda olduğu durumlarda tarama işlemi çok yavaş gerçekleşecektir. C_k'nın boyutunu küçültmek için apriori özelliği şu şekilde kullanılır: destek değerini sağlamayan herhangi bir (k-1)-öğe k-öğe kümenin alt kümesi olamaz. Bundan dolayı eğer aday k-öğe kümenin herhangi bir (k-1) alt kümesi L_{k-1}'in içinde değilse, aday da destek değerini sağlamayacağından C_k 'nın içinden çıkarılabilir. Bu çıkarma işlemi sonucunda C_k aday kümesinin öğeleri azalır dolayısıyla da boyutu azalmış olur. Bu altküme testi hash ağacı yöntemi ile daha hızlı bir şekilde yapılabilir (Han and Kamber, 2001).

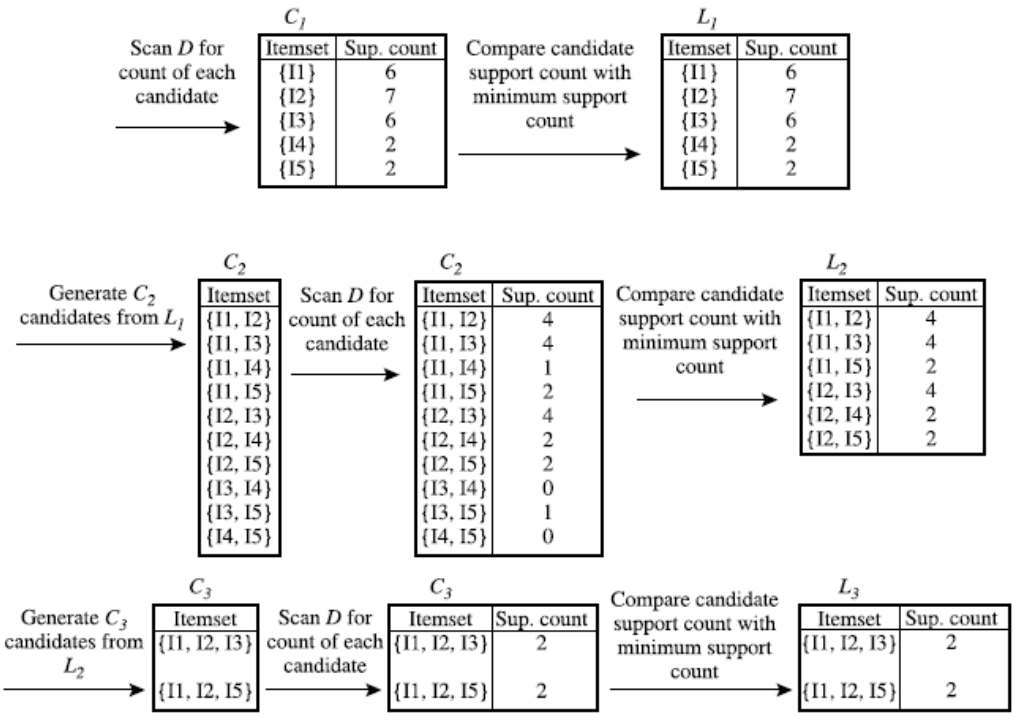
4.3.3. Apriori Algoritması Örnek Uygulama

Apriori algoritması ile sık geçen öğe kümelerinin bulunması için aşağıda bir örnek verilmiştir (Han and Kamber, 2001). İşlem veritabanı olarak Çizelge 4.8'de verilen işlem veritabanı (D) kullanılmıştır.

Çizelge 4.8. İşlem verileri.

TID	Item_ID Listesi
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3

Çizelge 4.8'de verilen işlem veritabanında 9 adet işlem vardır (|D|=9). TID sütunu işlem kodunu, Item_ID sütunu ise işlemde yer alan öğeleri göstermektedir. Apriori algoritmasının D veritabanındaki izlediği adımlar Şekil 4.9'da verilmiştir.



Şekil 4.9. Minimum destek değeri 2 ile apriori adımları.

- Algoritmanın ilk adımında 1-öğe kümelerinin tamamı C1 aday kümesini oluşturur. Algoritma, C1 aday kümesinin her bir elemanının destek değerini bulmak için veritabanındaki işlemler tarar.

- C1 aday kümesinin içerisinde minimum destek değeri olan 2'yi sağlayan öğe kümelerinin tamamı L1 öğe kümesini oluşturur.
- 2-öğe kümesi olan L2'yi bulmak için algoritma L1⋈L1 kartezyen çarpımını kullanarak C2 aday kümesini oluşturur. C2 aday kümesi L1'in ikili kombinasyonlarından elde edilir.
- C2 aday kümesindeki öğelerin destek değerleri D taranarak bulunur.
- C2 aday kümesinde minimum destek değeri 2'yi sağlayan öğe kümeleri L2 'yi oluşturur.
- 3-öğe kümesi olan L3'ü bulmak için algoritma L2⋈L2 kartezyen çarpımını kullanarak C3 aday kümesini oluşturur. Şekil 4.10'da C3 'ün oluşturulması ayrıntılı olarak verilmiştir. $C_3 = L_2 \bowtie L_2$ ={{I1,I2,I3},{I1,I2,I5}, {I1,I3,I5}, {I1,I3,I4},{I2,I3,I5},{I2,I4,I5}} öğelerini içerecektir. Apriori özeliğine göre son dört adayın sık geçen öğe olmayacağı belirlenir ve aday küme içerisinden çıkarılır. Böylece L_3 'ü oluşturmak için D veritabanı taramasında gereksiz taramaların önüne geçilmiş olur.
- C3 aday kümesindeki öğelerin destek değerlerini bulmak için D veritabanı tekrar taranır.
- C3 aday kümesinde minimum destek değerini sağlayan adaylar L3 kümesine aktarılır.
- 4-öğe kümesi olan L4'ü bulmak için algoritma L3⋈L3 kartezyen çarpımını kullanarak C4 aday kümesini oluşturur. Birleşim sonucu {{I1, I2, I3, I5}} olmasına rağmen, alt kümesi olan {{I2, I3, I5}} sık geçen olmadığı için bu öğe kümesi budanır. Böylece C4 aday kümesi boş küme olur ve algoritma sonlanır.

Join: $C_3 = L_2 \bowtie L_2$ ={{I1,I2},{I1,I3}, {I1,I5}, {I2,I3}, {I2,I4}, {I2,I5}} ⋈ {{I1,I2}, {I1,I3}, {I1,I5}, {I2,I3}, {I2,I4}, {I2,I5}}={{I1,I2,I3}, {I1,I2,I5}, {I1,I3,I5}, {I2,I3,I4}, {I1,I3,I5}, {I2,I4,I5}}
Apriori budama özelliği:
♦ {I1,I2,I3}'ün 2 öğeli alt kümeleri {I1,I2}, {I1,I3} ve {I2,I3}. 2 öğeli alt kümelerin hepsi L_2'nin üyesidir. Onun için {I1,I2,I3} C_3'ün içinde tutulur.
♦ {I1,I2,I5}'ün 2 öğeli alt kümeleri {I1,I2}, {I1,I5} ve {I2,I5}. 2 öğeli alt kümelerin hepsi L_2'nin üyesidir. Onun için {I1,I2,I5} C_3'ün içinde tutulur.

Şekil 4.10. C_3'ün oluşturulması.

♦ {I1,I3,I5}'ün 2 öğeli alt kümeleri {I1,I3}, {I1,I5}ve {I3,I5}. {I3,I5} L_2'nin üyesi değildir. Sık geçen de değildir. Onun için {I1,I3,I5} C_3'den çıkartılır.
♦ {I2,I3,I4}'ün 2 öğeli alt kümeleri {I2,I3}, {I2,I4}ve {I3,I4}. {I3,I4} L_2'nin üyesi değildir. Sık geçen de değildir. Onun için {I2,I3,I4} C_3'den çıkartılır.
♦ {I2,I3,I5}'ün 2 öğeli alt kümeleri {I2,I3},{I2,I5} ve {I3,I5}. {I3,I5} L_2'nin üyesi değildir. Sık geçen de değildir. Onun için {I2,I3,I5} C_3'den çıkartılır.
♦ {I1,I3,I5}'ün 2 öğeli alt kümeleri {I1,I3}, {I1,I5} ve {I3,I5}. {I3,I5} L_2'nin üyesi değildir. Olağan da değildir. Onun için {I1,I3,I5} C_3'den çıkartılır.
♦ {I2,I4,I5}'ün 2 öğeli alt kümeleri {I2,I4}, {I2,I5} ve {I4,I5}. {I4,I5} L_2'nin üyesi değildir. Olağan da değildir. Onun için {I2,I4,I5} C_3'den çıkartılır.
Budama işleminden sonra C_3={{I1,I2,I3}, {I1,I2,I5}}

Şekil 4.10. (devam ediyor).

Sık geçen öğe kümeleri bulunduktan sonra birliktelik kuralları oluşturulur. Örneğin, sık geçen öğe olan {I1,I2,I5} için boş olmayan bütün alt kümeler {I1,I2}, {I1,I5}, {I2,I5}, {I1}, {I2} ve {I5}'dir. Bu kümelerden aşağıda verilen birliktelik kuralları çıkarılabilir.

$$I1, I2 \Rightarrow I5, \text{güven} = \frac{2}{4} = \%50,$$

$$I1, I5 \Rightarrow I2, \text{güven} = \frac{2}{2} = \%100,$$

$$I2, I5 \Rightarrow I1, \text{güven} = \frac{2}{2} = \%100,$$

$$I1 \Rightarrow I2, I5, \text{güven} = \frac{2}{6} = \%33,$$

$$I2 \Rightarrow I1, I5, \text{güven} = \frac{2}{7} = \%29,$$

$$I5 \Rightarrow I1, I2, \text{güven} = \frac{2}{2} = \%100,$$

eğer minimum güven eşik değeri %70 olarak belirlenmişse sadece (I1, I5 $\Rightarrow$ I2), (I2, I5 $\Rightarrow$ I1) ve (I5 $\Rightarrow$ I1, I2) kuralları dikkate alınır.

4.3.4. Apriori Algoritması Pseudocode

Apriori algoritmasına ait pseudocode Şekil 4.11'de ve pseudocode içerisinde kullanılan yordamlar Şekil 4.12'de verilmiştir (Han and Kamber, 2001).

Giriş: D, veritabanı hareketleri. *min_sup,* minimum destek eşiği.
Çıkış: L, D'de yer alan sık geçen öğe kümeleri
Metod:

(1) L_1 = find_frequent_1-itemsets(d);
(2) for(k =2; $L_{k-1} \neq \varnothing$; k++) {
(3) C_k = aprirori_gen(L_{k-1});
(4) for each transaction $t \in D$ { *//sayılar için D'yi tara*
(5) C_t = subset(C_k, t); *//aday olan t altkümelerini al*
(6) for each candidate $c \in C_t$
(7) c.count++;
(8) }
(9) $L_k = \{ c \in C_k \mid c.count \geq min\_sup\}$
(10) }
(11) return $L = \cup_k L_k$;

Şekil 4.11. Apriori algoritması pseudocode.

1.adımda L_1 bulunmaktadır. 2-10. adımlarda, L_{k-1} kullanılarak aday C_k ve L_k bulunmaktadır. Aprirori_gen() yordamı aday kümenin elemanlarını oluşturup sık geçmeyen alt kümelere ait elemanları elemek için apriori özelliğini kullanmaktadır. İlk olarak adaylar oluşturulmuş, veritabanı taranmıştır (4. adım). Her bir işlemin üye altkümelerini bulmak için subset() fonksiyonu kullanılmıştır (5. adım). Daha sonra bu adayların her birinin sayısı toplanmıştır (6-7. adım). Sonuç olarak tüm bu minimum support noktasına ulaşan adaylar, sık geçen öğe kümesi olan L'yi oluşturmaktadır. Daha sonra sık geçen öğe kümelerden ilişkisel kuralları çıkaracak bir yordam çağrılır.

Procedure apriori_gen(L_{k-1}:frequent(k – 1)-itemsets)

(1) for each itemset $I_1 \in L_{k-1}$
(2) for each itemset $I_2 \in L_{k-1}$
(3) if $(I_1[1]=I_2[1]) \wedge (I_1[2]=I_2[2]) \wedge \ldots \wedge (I_1[k-2]=I_2[k-2]) \wedge (I_1[k-1]=I_2[k-1])$ then {
(4) $c = I_1 \bowtie I_2$; *//birleştirme adımı:adayları oluştur*
(5) if has_infrequent_subset(c, L_{k-1}) then
(6) delete c; *//budama adımı*

Şekil 4.12. Pseudocode içerisinde kullanılan yordamlar**.**

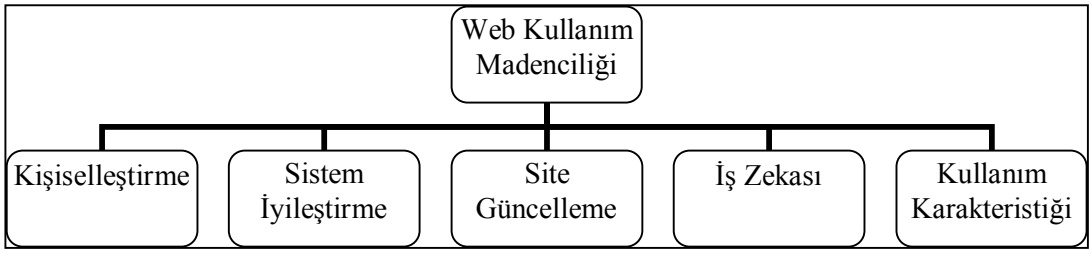
```
(7)                 else add c to Ck;
(8)             }
(9)     return Ck;
Procedure has_infrequent_subset( c:candidate k-itemset;
                                 Lk-1: frequent(k -1)-itemsets);
(1)     for each (k-1)-subset s of c
(2)             if s ∉ Lk-1 then
(3)                     return TRUE;
(4)     return FALSE;
```

Şekil 4.12. (devam ediyor).

Apriori_gen() yordamı, birleştirme ve budama isimli iki işlemi yerine getirir. Birleştirme kısmında, potansiyel adayları bulmak için L_{k-1}'i kendisiyle birleştirmektedir (1-4. adım). Budama kısmı ise sık geçmeyen alt kümelere sahip adayları elemek için apriori özelliğini çalıştırmaktadır (5-7. adım). Sık geçen alt küme olup olmadığını anlamak için kullanılan test has_infrequent_subset() yordamı tarafından yerine getirilmektedir.

4.4. WEB KULLANIM MADENCİLİĞİ UYGULAMA ALANLARI

Srivastava et al.(2000) yaptıkları çalışmada; web kullanım madenciliğinin uygulama alanlarını Şekil 4.13'de gösterildiği gibi kişiselleştirme, sistem iyileştirme, site güncelleme sistemleri, iş zekâsı ve kullanım karakteristiği başlıkları altında toplamıştır.



Şekil 4.13. Web kullanım madenciliğinin başlıca uygulama alanları.

4.4.1. Kişiselleştirme

Kullanıcıların ziyaret ettiği web sitesi üzerinde kullanım davranışları ve kullanıcı profili gibi bilgilerin tespit edilmesi ve bunların sınıflandırılması sonucu kullanıcıların sonraki davranışlarının tahmin edilmesi sağlanabilir. Tahmin edilen davranışlar doğrultusunda da kullanıcıya öneriler sunabilir. Örneğin, elektronik ticaret yapan bir web sitesi üzerinde alışveriş yapan kayıtlı bir kullanıcının aldığı ürünlere ve site üzerindeki davranışlarına göre bir sonraki ziyarette bu kullanıcının davranışlarına yönelik tahminler yapılabilmektedir. Çizelge 4.8'de görüldüğü gibi kişiselleştirilme amacıyla WebWatcher (Joachhims et al., 1997), Letizia (Leieberman, 1995), Krishnapuram (Moore et al., 1997), Analog (http://www.analog.cx, 2010), WebPersonalizer (Mobasher et al., 1999), SiteHelper (Ngu et al., 1997) gibi çeşitli projeler geliştirilmiştir. Bu yazılım projelerinde, kullanıcıların sahip olduğu benzer erişim örüntülerinin kümelemelerini keşfetmek için web sunucu kayıtları kullanılmıştır.

4.4.2. Sistem İyileştirme

Web kullanıcı memnuniyetini ve web kullanım aktivitelerini yüksek kaliteye çıkarmak için web sunucu başarımını ve diğer servis özelliklerini arttırmak gerekmektedir. Web kullanım madenciliği ile ortaya çıkan eksikler göz önünde bulundurularak yazılım ve donanımsal olarak sistemin ve diğer bileşenlerin güçlendirilmesi sağlanabilir. Ayrıca sunuculara yapılan saldırı ve ataklar ile sisteme zarar veren, dolandırıcılık ve hile ile kullanıcı şifrelerini elde etmeye çalışan kullanıcıların tespiti sağlanabilir. Çizelge 4.8'de görüldüğü gibi sistem geliştirme alanında da yapılmış çalışmalar mevcuttur.

4.4.3. Site Güncelleme

Birçok web uygulamalarının işleyişi, gerekliliği ve kullanımı açısından web sitesinin çekiciliği hem içerik hem de yapı bakımından çok önemlidir. Örneğin; şirketlerin e-ticaret için kullandıkları bir ürün katalogu, çevrimiçi satış modülleri, üniversitelerin web sitesi üzerinde aktif olarak kullandıkları öğrenci işleri otomasyonu, personel

maaş otomasyonu, akademik bilgi sistemleri veya bankaların yaygın olarak kullandıkları internet bankacılığı web modüllerinin etkili, yararlı ve son derece önemli oldukları aşikârdır. Bu web sitelerini değerlendirmek ve geliştirmek için internet kullanıcılarından detaylı geri dönüş bilgileri almak gerekmektedir. Web kullanım madenciliği, internet kullanıcı davranışlarını derin bir şekilde inceleyip, web sitelerinin güncellenmesi ya da yeniden tasarlanması konusunda web tasarımcılara, web yöneticilerine ayrıntılı olarak rapor sunmaktadır.

Uyarlamalı Web sitesi projesi (Perkowitz and Etzioni, 1998; Perkowitz and Etzioni, 1999), bir sitenin içeriği ve yapısının yeniden tasarlanması için Web erişim kayıtlarından SCML algoritması ile otomatik olarak bilgi çıkarmaya yönelmiştir. Bu projede, sayfaların kümelenmesi direk olarak bağlantılı olan sayfaları tanımlamak için kullanılmıştır.

4.4.4. İş Zekası

E-ticaretle uğraşan şirketlerin web sitesini hangi müşteri kitlesi nasıl kullanıyor sorusuna cevap veren bilgilerin tespit edilmesi için araştırmalar yaptığı bir çalışma alanıdır. Buchner et al.(1999), web kayıtlarından akıllı alışveriş işlemlerini tespit etmek için bir bilgi keşfi uygulaması geliştirmişlerdir. Geliştirdikleri bu elektronik ticaret uygulamasında çok büyük olan alışveriş verileri ile aşırı derecede büyük olan kullanım veri kümelerini birleştirmişlerdir. Uygulamalarında geliştirdikleri bilgi keşfi teknikleri müşteri ilişkileri yaşamında müşterinin ilgisi, müşterinin devamlılığı, çapraz satış ve müşterinin web sitesinden ayrılışı olmak üzere dört ayrı basamak tanımlamışlardır. Elektronik ticarette iş zekasının amacına ulaşması için internetteki web trafiklerini analiz eden SurfAid, Accrue, NetGenesis, Aria, Hitlist, WebTrends 14 gibi birçok ticari yazılımlar mevcuttur (Cooley, 2000).

4.4.5. Kullanım Karakteristiği

Web kullanım madenciliği ile web karakterizasyon araştırması arasında büyük oranda bir örtüşme vardır. Catledge and Pitkow (1995), Pitkow (1997) and Pitkow and Kehoe(1995) Georgia Teknoloji Enstitüsü'nde geliştirmiş oldukları Xmosaic adlı

Web tarayıcı yazılımı ile istemci taraflı aktivitelerin kaydetme işlemlerini gerçekleştirmişlerdir. Özellikle tarayıcılar aracılığıyla bir web sitesiyle etkileşim halinde bulunan kullanıcılardan elde edilen kayıtların sonuçları, kullanıcıların davranışları, kullanım karakteristiği hakkında detaylı bilgiler sunmaktadır (Cooley, 2000).

Web kullanım madenciliği ile ilgili yapılmış araştırma projelerinde birbirinden farklı birçok yazılım geliştirilmiştir. Bu araştırma projelerini uygulama alanlarına, kullandıkları veri kaynaklarına ve veri tiplerine göre sınıflandırmak mümkündür. Çizelge 4.9'de geliştirilen yazılımların çoğu sunucu temelli verileri kullanmaktadır. Görüldüğü üzere yazılım projelerinin tümü kullanım verilerini, birkaçı ise kullanımın yanı sıra yapı, içerik veya profil verilerini kullanarak analiz yapabilmektedir. Tek kullanıcılı projeler genellikle kişiselleştirme uygulama alanını içermektedir. Çoklu site analizini destekleyen projelerde ise birden fazla web sitesinin kullanım verilerine kolayca erişebilmek için ya istemci ya da vekil sunucu seviyesinde giriş verileri kullanılmaktadır. Çoğu web kullanım madenciliği projelerinde tek ve çok kullanıcılı siteler, web sunucu kayıtları gibi sunucu temelli kullanım verileri kullanılmaktadır.

Çizelge 4.9. Web kullanım madenciliği projeleri ve yazılımları (Cooley, 2000).

Proje Adı	Uygulama Alanları	Veri Kaynağı			Veri Tipi				Kullanıcı		Site	
		Sunucu	Vekil	İstemci	Yapı	İçerik	Kullanım	Profil	Tek	Çok	Tek	Çok
WebSIFT	Genel	X			X	X	X			X	X	
SpeedTracer	Genel	X					X			X	X	
WUM	Genel	X			X		X			X	X	
Shahabi	Genel			X	X		X			X	X	
Site Micros	Kişiselleştirme	X					X		X		X	
Letizia	Kişiselleştirme			X			X		X			X
Web Watcher	Kişiselleştirme		X				X	X		X		X
Krishnapuram	Kişiselleştirme	X					X			X	X	
Analog	Kişiselleştirme	X					X			X	X	
Web Personalizer	Kişiselleştirme	X			X		X			X	X	
Tuzhilin	İş	X					X			X	X	
SurfAid	İş	X					X			X	X	
Buchner	İş	X					X	X		X	X	
WebTrend,Accrue	İş	X					X			X	X	
WebLogMiner	İş	X					X			X	X	
WebLogMiner	Site Yenileme	X					X			X	X	
PageGather,SCML	Karakterize etme	X			X		X			X	X	
Manley	Karakterize etme	X					X			X		X
Arlitr	Karakterize etme	X					X			X		X

Çizelge 4.9. (devam ediyor).

Pitkow	Karakterize etme	X		X			X			X		X
Almeida	Site Geliştirme	X					X			X		X
Rexford	Site Geliştirme	X	X				X			X	X	
Sxhechter	Site Geliştirme	X					X			X	X	
Aggarwal	Site Geliştirme		X				X			X	X	

BÖLÜM 5

UYGULAMA

Bu çalışma ile Gaziosmanpaşa Üniversitesi kurumsal web sitesine ait 6 aylık sunucu erişim kayıtları incelenerek web sitesine ait çeşitli istatistiki bilgileri çıkaran ve apriori algoritması ile birliktelik kuralarını bulan yazılım geliştirilmiştir. Hazırlanan yazılım sunucu erişim kayıtları üzerinde web kullanım madenciliği kurallarını uygulamaktadır. Yazılım Visual Studio 2005 paketine ait Visual C# dili ile hazırlanmış ve MSSQL veritabanı kullanılmıştır. Yazılımdan elde edilen sonuçlar SPSS Clementine yazılımına ait sonuçlarla kıyaslanarak geçerliliği kontrol edilmiştir.

5.1. GİRİŞ VERİLERİ

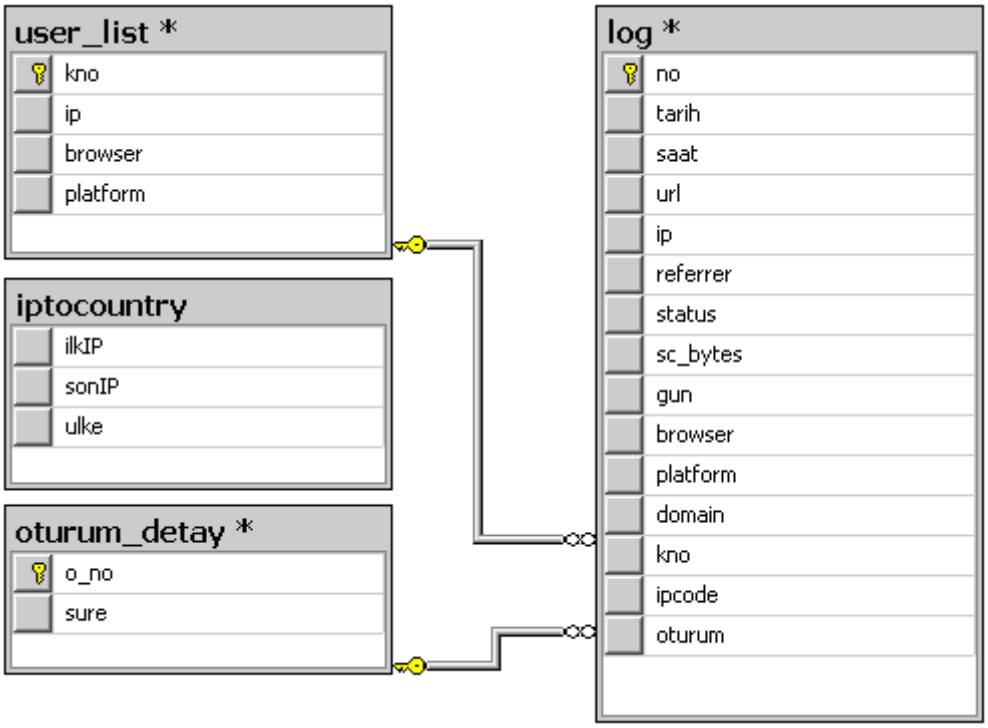
Hazırlanan çalışma Gaziosmanpaşa Üniversitesi kurumsal web sitesi ait 6 aylık sunucu erişim kayıtlarını kullanmaktadır. Kullanılacak veriye ait çeşitli değerler Çizelge 5.1'de verilmiştir.

Çizelge 5.1. Giriş verisine ait çeşitli bilgiler.

Veriye ait özellik	Değer
Kullanılacak verinin tarih aralığı	01.06.2010-30.11.2010
Erişim dosyası sayısı	3 660 adet
Toplam veri boyutu	30,5 GB
Kullanılan domain sayısı	20 adet
Erişim dosyalarının içerdiği toplam satır sayısı	123 299 164 satır
Ön işlem sonrası satır sayısı	3 858 856 satır
Veritabanı boyutu	1,76 GB

5.2. ÇIKIŞ VERİLERİ

Giriş verilerine hazırlanan yazılım ile web kullanım madenciliği uygulanarak elde edilen veriler MSSQL veritabanına aktarılmaktadır. Veritabanı üzerinde erişim kayıtlarına ait tüm bilgiler tutulduğu için hazırlanacak sorgular yardımıyla her tür istatistiki bilgi elde edilebilir. Kullanılan veritabanına ait diyagram Şekil 5.1'de verilmiştir.



Şekil 5.1. Çıkış verilerinin aktarıldığı data.mdf veritabanına ait diyagram.

Log tablosu, erişim kayıtlarında tutulan bilgilerin aktarılması için kullanılmıştır.

User_list tablosu, erişim kayıtlarına web kullanım madenciliği uygulandıktan sonra oluşturulan kullanıcıların tutulması için kullanılmıştır.

iptocountry tablosu, ülkelerin sahip olduğu IP adreslerinin aralıklarını tutmaktadır. Bu tablo içerisinde IP adreslerine ait ilkIP ve sonIP değerleri sayısal değere dönüştürülerek tutulmaktadır. Bir IP adresini sayısal değere dönüştürmek için kullanılan kod bloğu Şekil 5.2'de verilmiştir.

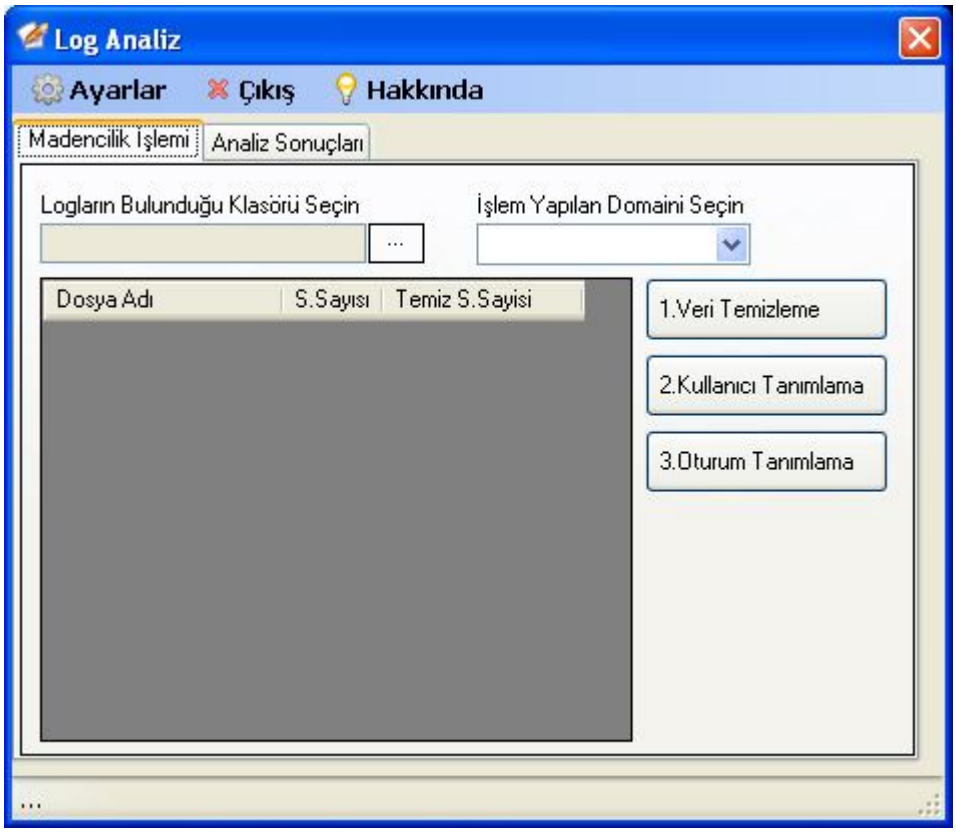
```
String ip_adresi="193.140.80.2";
String[] ip_part = ip_adresi.ToString().Split('.');
Double ipcode = 16777216 * Convert.ToDouble(ip_part[0]) + 65536 *
Convert.ToDouble(ip_part[1]) + 256 * Convert.ToDouble(ip_part[2]) +
Convert.ToDouble(ip_part[3]);
```

Şekil 5.2. IP adresini sayısal değere dönüştürmek için kullanılan C# kodu.

Oturum_detay tablosu, erişim kayıtlarından elde edilen oturumların ve sürelerinin tutulması için kullanılmıştır.

5.3. HAZIRLANAN YAZILIMIN ÖZELLİKLERİ

Hazırlanan yazılım formlar yardımıyla tasarlanarak kullanımı kolaylaştırılmıştır. Kullanıcının analiz için özel bir kod bilmesine gerek kalmamaktadır. Hazırlanan yazılıma Log Analiz ismi verilmiştir. Program çalıştırıldığında karşılaşılan pencere Şekil 5.3'de verilmiştir.



Şekil 5.3. Log Analiz programı açılış penceresi.

Program Ayarlar, Çıkış, Hakkında menülerinden ve Madencilik İşlemi, Analiz Sonuçları sekmesinden oluşmaktadır.

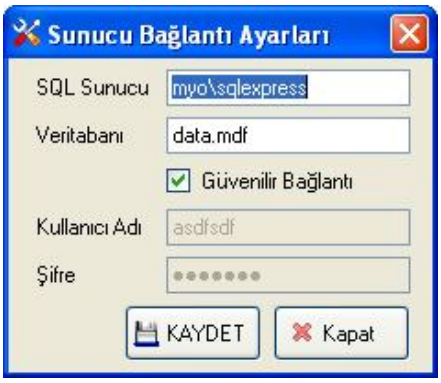
5.3.1. Ayarlar Menüsü

Ayarlar menüsü ile SQL Server sunucu ayarları, kullanılan erişim kayıtlarının formatı, analiz için kullanılacak dosya türleri ve analiz için dikkate alınmayacak örümcek yazılımlara ait anahtar kelimelerin tanımlaması yapılabilir. Şekil 5.4'de ayarlar menüsünün içeriği gösterilmiştir.

Şekil 5.4. Log Analiz programına ait Ayarlar menüsü içeriği.

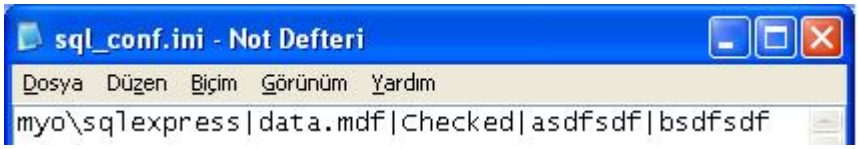
5.3.1.1. SQL Server

Madencilik sürecinde elde edilen veriler SQL Server veritabanına kaydedilmektedir. Bu bölümden veritabanı sunucusuna bağlantı için gerekli ayarlamalar yapılır. SQL Server ayarları için kullanılacak pencereye ait ekran görüntüsü Şekil 5.5'de verilmiştir.



Şekil 5.5. Sunucu bağlantı ayarları ekran görüntüsü.

Yapılan sunucu bağlantı ayarları programın kurulum klasöründe *sql_conf.ini* dosyasında tutulmaktadır. Şekil 5.6'da sql_conf.ini dosyasının içeriği verilmiştir.

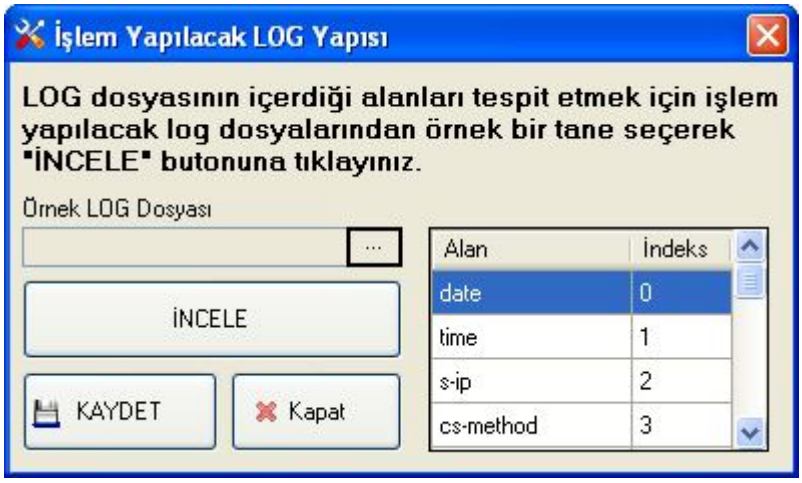


Şekil 5.6. sql_conf.ini dosyası içeriği.

5.3.1.2. LOG Yapısı

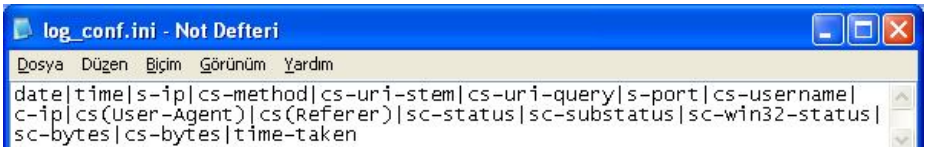
Web kullanım madenciliği için kullanılan erişim kayıtları her zaman aynı formatta olmamaktadır. Kullanılan sunucu ve yapılan sunucu konfigürasyonuna göre sunucu tarafından oluşturulan erişim kayıtlarının formatı farklı olabilir.

LOG Yapısı seçeneği ile analizi yapılacak erişim kayıt dosyalarından bir tanesi seçilerek programa erişim kayıtlarının içerdiği verinin formatı tanıtılmaktadır. Şekil 5.7'de erişim kayıtlarının formatını belirlemek için kullanılan log yapısı penceresi verilmiştir.



Şekil 5.7. İşlem yapılacak log yapısı ekran görüntüsü.

Log dosyalarına ait yapı programın kurulum klasöründe *log_conf.ini* dosyasında tutulmaktadır. Şekil 5.8'de log_conf.ini dosyasının içeriği verilmiştir.



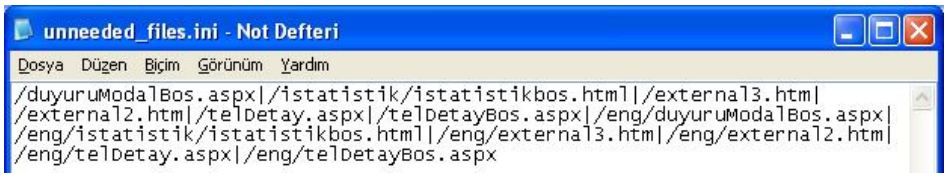
Şekil 5.8. log_conf.ini dosyası içeriği.

5.3.1.3. Dikkate Alınmayacak Sayfalar

Web sayfalarının tasarımından kaynaklı (frame, iframe v.b.) bazı sayfalara otomatik erişim sağlanmaktadır. Bu tür sayfalar bu bölümden yazılıma tanıtılarak madencilik işlemi sürecinde temizlenmektedir. Şekil 5.9'da dikkate alınmayacak sayfalara ait ekran görüntüsü verilmiştir.

Şekil 5.9. Dikkate alınmayacak sayfalar ekran görüntüsü.

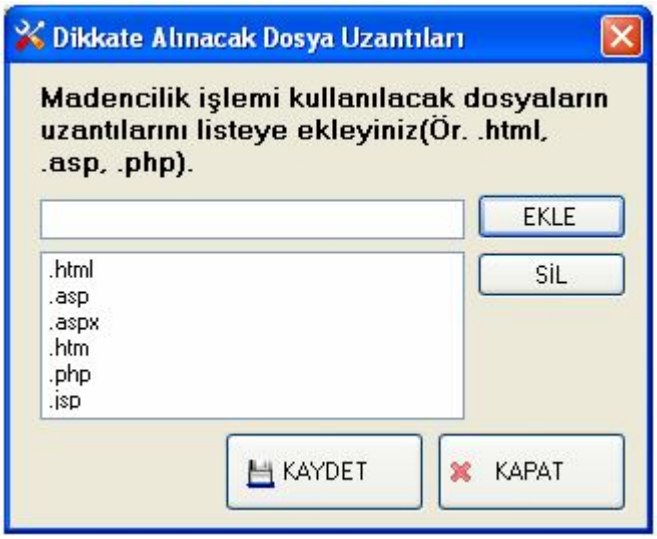
Dikkate alınmayacak sayfalar programın kurulum klasöründe *unneeded_files.ini* dosyasında tutulmaktadır. Şekil 5.10'da unneeded_files.ini dosyasının içeriği verilmiştir.



Şekil 5.10. unneeded_files_ini dosyası içeriği.

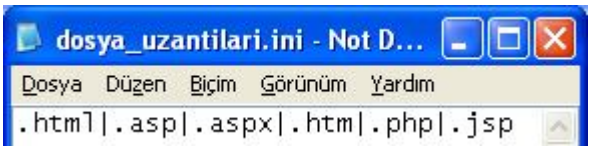
5.3.1.4. Dikkate Alınacak Dosya Uzantıları

Erişim kayıtları içerisinde bağlantılı kurulan sayfayla birlikte sayfaya ait gömülü kaynaklar da tutulmaktadır. Dikkate alınacak dosya uzantıları ayarı ile erişim kayıtları içerisinde madencilik işlemine dahil edilecek dosya uzantıları belirlenmektedir. Şekil 5.11'de dikkate alınacak dosya uzantıları penceresine ait ekran görüntüsü verilmiştir.



Şekil 5.11. Dikkate alınacak dosya uzantıları ekran görüntüsü.

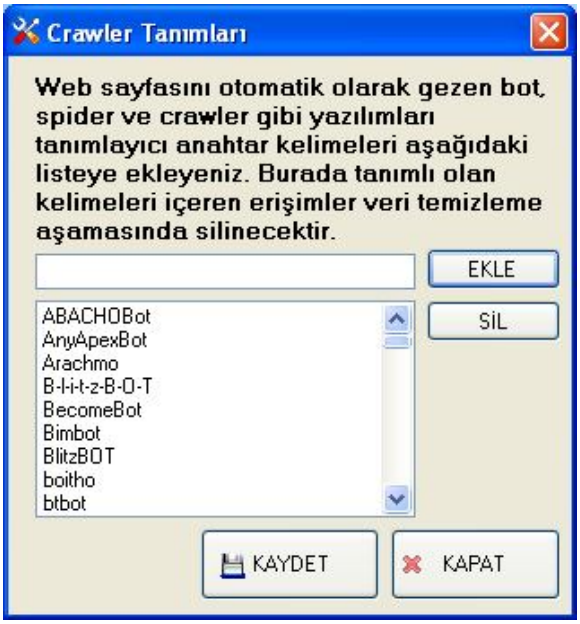
Dikkate alınacak dosya uzantıları programın kurulum klasöründe *dosya_uzantilari.ini* dosyasında tutulmaktadır. Şekil 5.12'de dosya_uzantilari.ini dosyasının içeriği verilmiştir.



Şekil 5.12. dosya_uzantilari.ini dosyası içeriği.

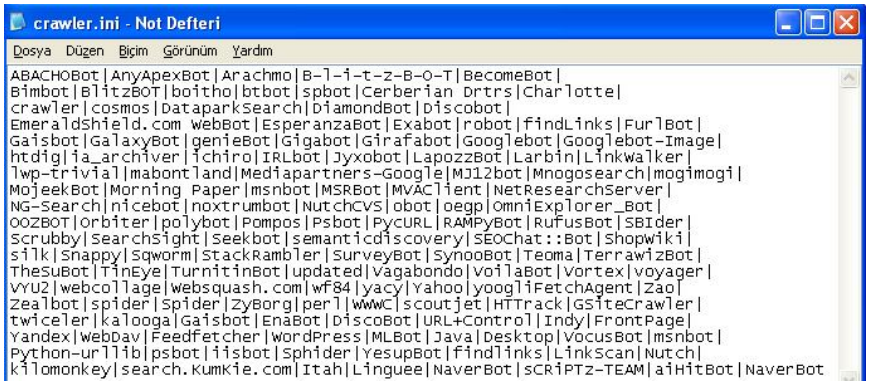
5.3.1.5. Bot, Spider, Crawler Anahtar Kelimeler

Erişim kayıtları içerisinde kullanıcı erişimlerinin yanı sıra örümcek yazılımlar tarafından yapılan sayfa ziyaretleri de bulunmaktadır. Bu tür erişimlerden madencilik sürecine dahil edilmeyecek olanları belirlemek için örümcek yazılımı tanıtan anahtar kelimeler tanımlanabilir. Şekil 5.13'de crawler tanımlamaları için kullanılacak ekran görüntüsü verilmiştir.



Şekil 5.13. Crawler tanımlamaları ekran görüntüsü.

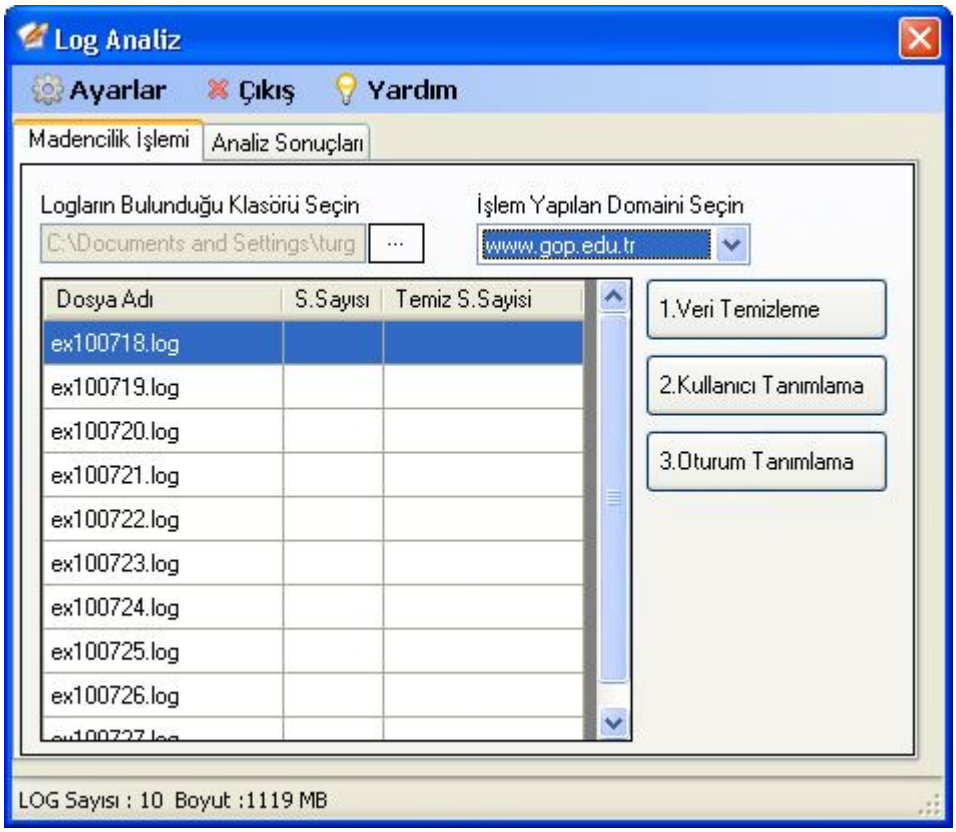
Tanımlanan anahtar kelimeler programın kurulum klasöründe *crawler.ini* dosyasında tutulmaktadır. Crawler.ini dosyasının içeriği Şekil 5.14'de verilmiştir.



Şekil 5.14. crawler.ini dosyası içeriği.

5.3.2. Madencilik İşlemleri Sekmesi

Sunucu üzerinde alt domainler için erişim kayıtları farklı klasörlerde tutulmaktadır. Hazırlanan program her bir alt domain için ayrı ayrı işlem yapmaktadır.
Erişim kayıtlarının bulunduğu klasör seçildikten sonra klasör içerisindeki dosyalar listeye eklenir. Dosyalar listeye eklendikten sonra pencerenin altında doya sayısı ve toplam boyut görüntülenecektir. Son olarak da seçilen dosyaların ait olduğu domain seçilerek madencilik işlemine başlanabilir. Örnek olarak seçilen erişim kayıtlarını içeren ekran görüntüsü Şekil 5.15'de verilmiştir.



Şekil 5.15. Log analiz programı ile örnek dosya seçimi.

Log analiz programı, web kullanım madenciliğinin ön işlem sürecini madencilik işlemi sekmesinde gerçekleştirmektedir.

Erişim kayıtları ve domain seçildikten sonra ön işlem sürecinin basamakları olan veri temizleme, kullanıcı tanımlama ve oturum tanımlama işlemi gerçekleştirmektedir.

5.3.2.1. Veri Temizleme

Erişim kayıtlarının içerdiği verilerin tamamı madencilik süreci için gerekli veriler değildir. Bu nedenle, erişim kayıtları içerisindeki geçerli ve gerekli olan veriler alınmalı diğerleri temizlenmelidir

Bir web sayfası ziyaret edildiğinde sayfayla birlikte sayfanın içerdiği gömili kaynaklar ve web sitesinin tüm dosyalarını otomatik olarak tarayan robot istekler de erişim kayıtlarına yeni satır olarak eklenmektedir. Yapılacak çalışmanın özelliğine bağlı olarak bazı durumlarda 200 durum kodu dışındaki yani başarılı erişimler dışındaki erişimler de gereksiz veri olarak düşünülebilir.

Bu çalışma da temizliğe ihtiyaç duyulan gereksiz veya alakasız iki tür veri vardır. Bunlar, web sayfasına ait dosya içerisine gömülü ek kaynaklar ve robot isteklerdir. Log analiz programının ayarlar kısmında tanımlanan dosya uzantıları dışında kalan dosyalara erişimler ve robot istekler için tanımlanan anahtar kelimeleri içeren satırlar gereksiz veri olarak belirlenmekte ve veritabanına eklenmemektedir.

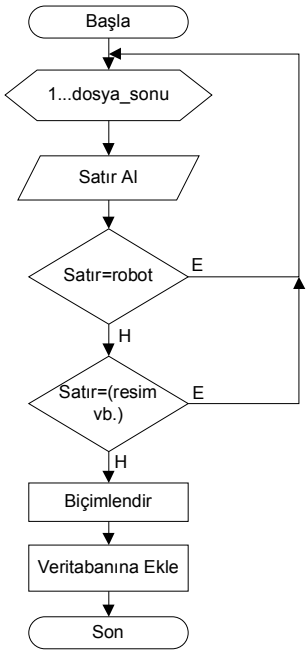
Log analiz programı veri temizleme aşamasında listede bulunan her dosya satır satır okunarak biçimlendirilmekte ve geçerli bir veri ise veritabanına eklemektedir.

Erişim kayıtları içerisindeki satırlar incelendiğinde, barındırdığı veri alanlarının boşluk karakteri ile birbirinden ayırdığı görülmektedir. Veri alanlarının içerdiği bilginin türü ve sıralaması dosyanın başında açıklama şeklinde verilmektedir. Erişim kayıtları içerisindeki veri alanları ve örnek bir satır Şekil 5.17'de verilmiştir.

```
#Software: Microsoft Internet Information Services 6.0
#Version: 1.0
#Date: 2010-07-27 00:02:23
#Fields: date time s-ip cs-method cs-uri-stem cs-uri-query s-port cs-username c-ip cs(User-Agent)
2010-07-27 00:02:23 193.140.180.4 GET /Default.aspx - 80 - 78.170.172.235 Mozilla/4.0+(compatible
2010-07-27 00:02:23 193.140.180.4 GET /ajaxtabs.js - 80 - 78.170.172.235 Mozilla/4.0+(compatible;
```

Şekil 5.17. Loglar içerisinde bulunan veri alanları ve örnek veri.

Log analiz programı veri temizleme aşamasına ait akış diyagramı Şekil 5.16'da verilmiştir.



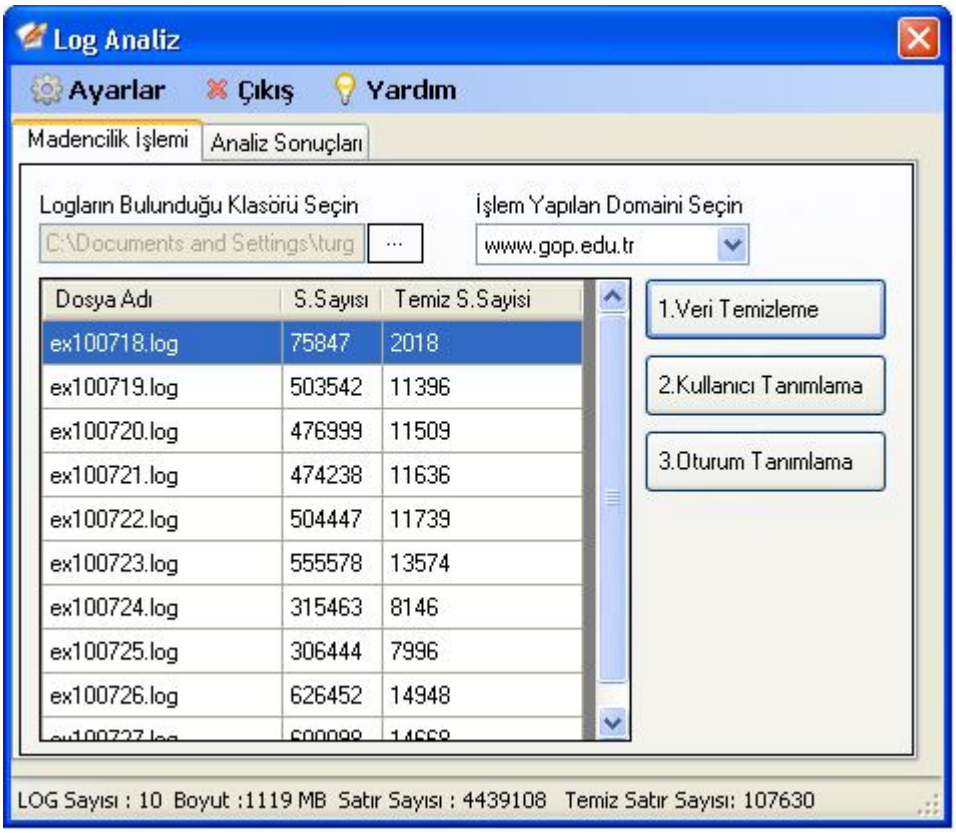
Şekil 5.16. Veri temizleme aşamasına ait akış diyagramı.

Program tarafından okunan her bir satır veri gruplarına ayrıldıktan sonra cs-uri-stem alanının içerdiği sayfa uzantısı programa tanıtılan uzantılardan birisi ise ve cs(User-Agent) alanı robot isteklere ait anahtar kelimelerden herhangi birisini içermiyorsa geçerli bir veri olarak düşünülmekte ve veritabanına aktarılmaktadır.

Geçerli olan bir veri satırının içerdiği verilerden date, time, cs-uri-stem, c-ip, cs-referer, sc-status, sc-bytes ve cs(User-Agent) verileri üzerinde aşağıda verilen biçimlendirmeler yapılmakta ve veritabanına aktarılmaktadır.

- Date verisi log tablosu üzerinde tarih sütununa ve tarihin gün karşılığı bulunarak gun sütununa eklenmektedir.
- Log tablosu üzerinde time verisi saat sütununa, cs-uri-stem verisi url sütununa, sc-status verisi status sütununa, sc-bytes verisi sc_bytes ve cs-referer verisi referrer sütununa eklenmektedir.
- cs(User-Agent) verisi üzerinden ziyaretçinin kullandığı tarayıcı ve işletim sistemi tespit edilerek log tablosu üzerine browser ve platform sütununa eklenmektedir.
- c-ip verisi üzerinden kullanıcının IP adresinin sayısal karşılığı bulunarak log tablosu üzerinde ipcode sütununa eklenmektedir.

Veri temizleme aşaması sonrasında her bir dosyanın içerdiği satır sayısı ve veri temizleme aşamasından sonra elde edilen satır sayısı liste üzerinde gösterilmektedir. Şekil 5.17'de veri temizleme aşaması sonrası ekran görüntüsü verilmiştir.



Şekil 5.17. Veri temizleme aşaması sonrası ekran görüntüsü.

Analizi yapılacak log dosyalarının tamamına ait veri temizleme süreci bitene kadar diğer aşamalar olan kullanıcı ve oturum tanımlama aşamasına geçilmemelidir. Veri temizleme işlemine tabii tutulacak dosya sayısı fazla olduğu durumda dosyalar klasörler içerisine gruplandırılarak grup grup veri temizleme işlemi yapılabilir.

5.3.2.2. Kullanıcı Tanımlama

Web kullanım madenciliği için bir kullanıcının doğrulanmasına ihtiyaç yoktur. Fakat farklı kullanıcıları ayırt etmeye ihtiyaç duyulur. Kimlik doğrulama veya kullanıcı taraflı çerezler olmaksızın kullanıcıları tanımlamak için IP adresi, tarayıcı ve işletim sistemi bilgilerini tutan user-agent bilgisi kullanılır.

Log analiz programı kullanıcı tanımlama işlemini gerçekleştirirken veri temizleme sonrası veritabanına aktarılan kayıtlar kullanılmaktadır. Şekil 5.18'de veri temizleme aşaması sonrası veritabanına aktarılan kayıtlardan bir bölüm verilmiştir.

no	tarih	saat	url	ip	referrer	status	gun	browser	platform
4...	2010-07-02	06:47:10	/Default.aspx	192.168.85.67	-	500	Cuma	Internet Explorer 6.0	Windows XP
4...	2010-07-02	06:47:10	/Default.aspx	192.168.85.67	-	500	Cuma	Internet Explorer 6.0	Windows XP
4...	2010-07-02	06:47:10	/Default.aspx	192.168.85.67	-	500	Cuma	Internet Explorer 6.0	Windows XP

Şekil 5.18. Veri temizleme aşaması sonrası veritabanına aktarılan kayıtlar.

Kullanıcı tanımlama işlemi için ziyaretçinin IP adresi, kullandığı işletim sistemi ve tarayıcı bilgileri kullanılmaktadır. Bu üç bilgisi aynı olan erişimler tek bir kullanıcı olarak tanımlanmaktadır.

Kullanıcı tanımlama programlama tarafında yapıldığında veritabanının sorgulaması ve oluşturulan kullanıcıların tekrar veritabanına aktarılması için veritabanına birçok kez bağlantı kurmak gerekir ve programın hızında ciddi yavaşlamalar meydana gelir. Bu sorunun önüne geçmek için veritabanı üzerinde user_create isminde stored procedure tanımlanarak kullanıcı oluşturma işlemi bu stored procedure yardımıyla yapılmıştır. Böylece veritabanına bir kez bağlantı yapılarak programda performans artışı sağlanmıştır.

Stored procedure, SQL Server üzerinde sorgulamalar yapmak için oluşturulan derlenmiş T-SQL ifadeleridir ve belirli bir görevi yerine getirmek için oluşturulurlar (Özseven, 2010).

user_create stored procedure log tablosu içerisindeki kayıtları ip, browser ve platform sütununa göre gruplandırılarak kullanıcıları bulmakta ve bulduğu kullanıcıları user_list tablosuna eklemektedir. Daha sonra da user_list tablosuna göre log tablosu içerisindeki kayıtların kno bilgisini güncellemektedir. Oluşturulan user_create yordamına ait T-SQL ifadesi 5.19'da verilmiştir.

```
CREATE PROCEDURE user_create
AS
INSERT INTO user_list SELECT ip,browser,platform FROM log GROUP BY
ip,browser,platform

UPDATE log SET kno = (SELECT top 1 kno FROM user_list WHERE log.ip =
ip AND log.browser = browser AND log.platform = platform)
RETURN
```

Şekil 5.19. user_create stored procedure'ne ait T-SQL kodları.

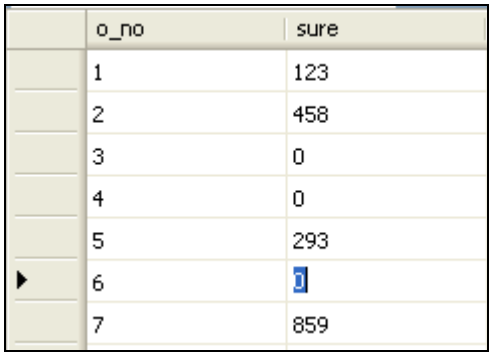
Hazırlanan çalışma için kullanılan giriş verileri üzerinde yapılan kullanıcı tanımlama işlemi sonrası 423 712 adet kullanıcı oluşturulmuştur.

5.3.2.3. Oturum Tanımlama

Bir oturum kullanıcının siteye girişi ile çıkışı arasındaki sürede gerçekleştirdiği aktiviteler grubu olarak tanımlanabilir. Oturum tanımlamadaki amaç oturumlar içerisindeki her kullanıcının sayfa erişimlerini birbirinden ayırt etmektir.

Oturum tanımlama için h-ref, h1 ve h2 yaklaşımları kullanılmaktadır. Hazırlanan çalışmada oturum tanımlama için h1 yaklaşımı kullanılmıştır. Çünkü http://www.gop.edu.tr web sitesine ait ana sayfanın içerdiği tüm bağlantılara diğer sayfalardan da erişilebilmektedir, bazı bölümlerde frame kullanılmıştır ve site ziyaretçilerinin büyük çoğunluğu Proxy sunucu üzerinden siteye bağlanmaktadır.

Log analiz programı oturum tanımlama işlemi yaparken erişim kayıtlarını kullanıcı, tarih ve saat bilgisine göre sıralayarak aynı kullanıcılar için erişimler arasındaki süre 30 dakikaya eşit veya küçükse bu erişimler aynı oturum olarak düşünülmüştür. Oturum numarası 1'den başlayıp her bir oturum için artarak oturumlar tanımlanmış ve log tablosu içerisinde ki her bir erişim için oturum numaraları eklenmiştir. Ayrıca tanımlanan oturumların numarası ve oturumun süresi oturum_detay tablosuna eklenmiştir. Bir oturum içerisinde tek bir sayfa ziyareti gerçekleştirilmişse oturum süresi 0 olarak tanımlanmıştır. Şekil 5.20'de oturum_detay tablosundan bir bölüm verilmiştir.



	o_no	sure
	1	123
	2	458
	3	0
	4	0
	5	293
▶	6	0
	7	859

Şekil 5.20. oturum_detay tablosundan bir bölüm.

Hazırlanan çalışma için kullanılan giriş verileri üzerinde yapılan oturum tanımlama işlemi sonrası 1 234 532 adet oturum oluşturulmuştur.

5.3.3. Analiz Sonuçları Sekmesi

Analiz sonuçları sekmesinde ön işlem sürecinden sonra elde edilen ve veritabanına aktarılan verilerden çeşitli istatistiki bilgiler elde edilmekte ve VM tekniklerinden apriori algoritması ile birlikte ziyaret edilme olasılığı yüksek sayfalar tespit edilmektedir.

Veritabanından elde edilen verilerin grafiksel olarak gösterimi için Visual studio 2005 ile birlikte Microsoft Chart Controls for Microsoft .NET Framework 3.5 kullanılmıştır (http://www.microsoft.com, 2010).

Log analiz programı ile elde edilebilecek bilgiler Şekil 5.21'de analiz sonuçları sekmesinde gösterilmiştir.

Şekil 5.21. Log analiz programı analiz sonuçları sekmesi.

5.3.3.1. Genel Bakış

Log analiz programı genel bakış seçeneği ile erişim kayıtlarına ait ilk erişim zamanı, son erişim zamanı, toplam ziyaret, günlük ortalama ziyaret gibi çeşitli bilgiler elde edilmiştir.

Genel bakış ile elde edilen veriler için veritabanı içerisindeki log, user_list ve oturum_detay tabloları kullanılmıştır. Genel bakış ile elde edilen bilgiler için kullanılan SQL ifadeleri Çizelge 5.2'de verilmiştir.

Çizelge 5.2. Genel bakış ile elde edilecek veriler için kullanılan SQL ifadeleri.

Elde Edilen Bilgi	SQL İfadesi
İlk Erişim Zamanı	SELECT TOP 1 tarih, saat FROM log ORDER BY tarih,saat
Son Erişim Zamanı	SELECT TOP 1 tarih, saat FROM log ORDER BY tarih DESC,saat DESC
Toplam Ziyaret	SELECT COUNT(*) as say FROM log

Çizelge 5.2. (devam ediyor).

Günlük Ortalama Ziyaret	SELECT COUNT(*) as gec2 FROM (SELECT COUNT(tarih) as aa FROM log GROUP BY tarih) gec
Ziyaret Sayısı	SELECT MAX(oturum) as o_no,MAX(kno) as k_no, COUNT(*) as s_say FROM log
Farklı Kişi Ziyaret Sayısı	
Sayfa Gösterim Sayısı	
Sitede Geçirilen Ortalama Süre	SELECT avg(sure) as o_sure FROM oturum_detay
Hemen Çıkma Oranı	SELECT count(*) as abc FROM (SELECT COUNT(oturum) as o_say FROM log GROUP BY oturum HAVING COUNT(oturum)<2) gec
Başarısız Erişimler	SELECT count(*) gec FROM log WHERE status<>200
Görüntülenen Sayfa Sayısı	SELECT count(*) gec FROM (SELECT url FROM log GROUP BY url) aa
Günlük Ortalama Görüntülenen Sayfa Sayısı	SELECT count(*) as gec2 FROM (SELECT COUNT(tarih) as aa FROM log GROUP BY tarih) gec
Hafta İçi Ziyaret Sayısı	SELECT COUNT(gun) gun_say,gun FROM log GROUP BY gun
Hafta Sonu Ziyaret Sayısı	
Haftanın En Aktif Günü	SELECT TOP 1 COUNT(gun) gun_say,gun FROM log GROUP BY gun ORDER BY gun_say DESC
Haftanın En Pasif Günü	SELECT TOP 1 COUNT(gun) gun_say,gun FROM log GROUP BY gun ORDER BY gun_say
Günü En Aktif saati	SELECT TOP 1 SUBSTRING(saat,1,2) as aa FROM log GROUP BY SUBSTRING(saat,1,2) ORDER BY COUNT(substring(saat,1,2)) DESC
Günün En Pasif Saati	SELECT TOP 1 SUBSTRING(saat,1,2) as aa FROM log GROUP BY SUBSTRING(saat,1,2) ORDER BY COUNT(SUBSTRING(saat,1,2))

Log analiz programı ile giriş verileri sonucu elde edilen genel bakış sonuçları Şekil 5.22'de verilmiştir.

Şekil 5.22. Log analiz programı genel bakış sonuçları.

Sitede Geçirilen Ortalama Süre(Saniye)	204
Hemen Çıkma Oranı	13,44%
Başarısız Erişimler	144767
Görüntülenen Sayfa Sayısı	5702
Günlük Ortalama Görüntülenen Sayfa Sayısı	31
Hafta İçi Ziyaret Sayısı	3171060
Hafta Sonu Ziyaret Sayısı	687796
Haftanın En Aktif Günü	Salı
Haftanın En Pasif Günü	Cumartesi
Günün En Aktif Saati	11:00 - 11:59
Günün En Pasif Saati	03:00 - 03:59

Şekil 5.22. (devam ediyor).

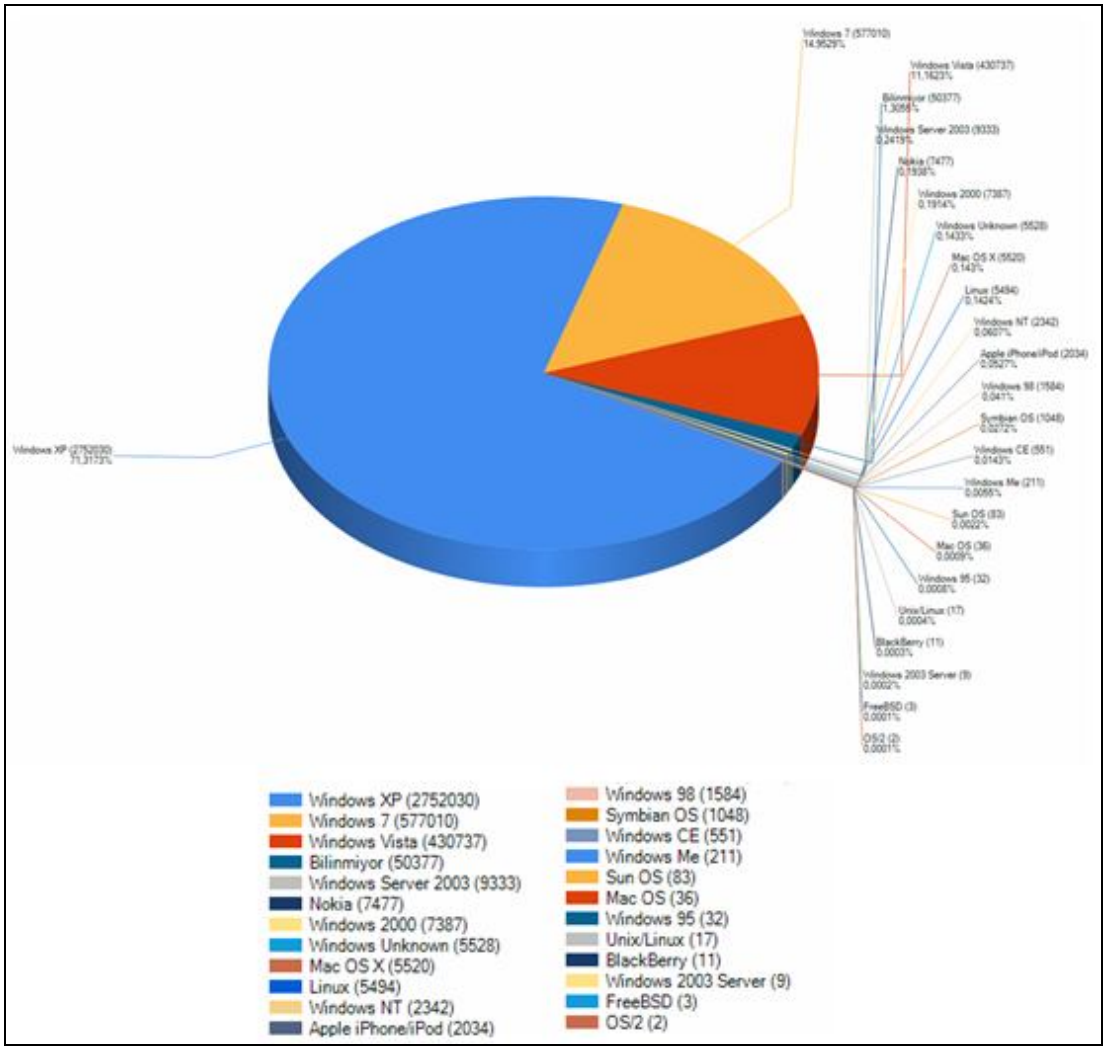
5.3.3.2. OS Dağılımı

OS dağılımı ile ziyaretçilerin kullanmış olduğu işletim sistemlerinin dağılımı grafiksel olarak elde edilmektedir.

Ziyaretçilerin kullanmış olduğu işletim sistemleri veri temizleme aşamasında user-agent bilgisinden elde edilerek veritabanı üzerinde log tablosuna kaydedilmektedir. Ziyaretçilerin OS dağılımları bulunurken log tablosu üzerinde Şekil 5.23'de verilen SQL ifadesinin çalışması sonucu elde edilen veriler Şekil 5.24'deki grafik üzerinde gösterilmektedir.

```
SELECT platform, COUNT(platform) as os_say,
ROUND(100*CAST(COUNT(platform) as float)/CAST((SELECT
COUNT(platform) as os_say FROM log) as float),4) as yuzde FROM
log GROUP BY platform ORDER BY os_say DESC
```

Şekil 5.23. OS dağılımını bulmak için kullanılan SQL ifadesi.



Şekil 5.24. OS dağılımlarının grafiksel gösterimi.

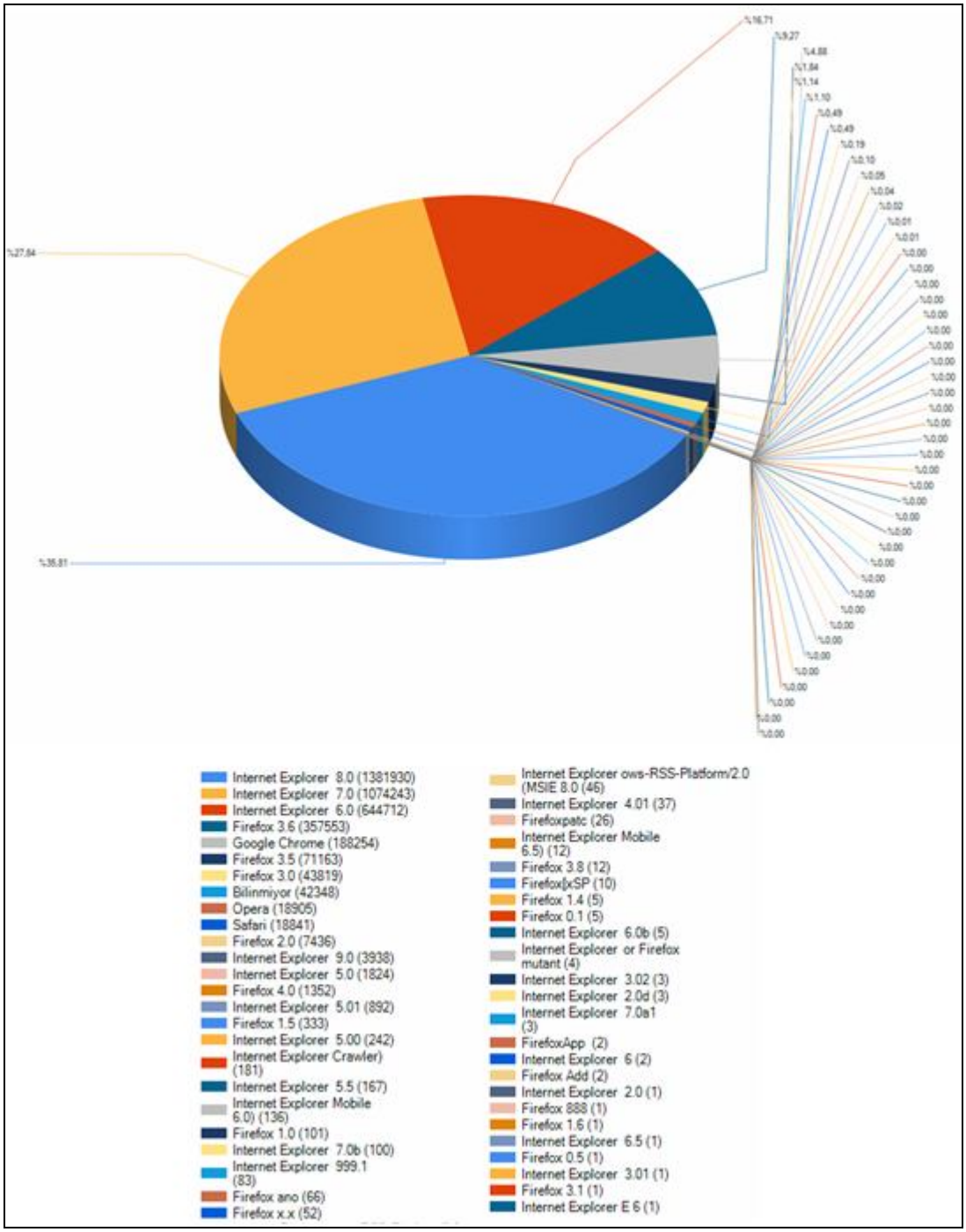
5.3.3.3. Tarayıcı Dağılımı

Tarayıcı dağılımı ile ziyaretçilerin kullanmış olduğu internet tarayıcıların dağılımı grafiksel olarak elde edilmektedir.

Ziyaretçilerin kullanmış olduğu tarayıcı veri temizleme aşamasında user-agent bilgisinden elde edilerek veritabanı üzerinde log tablosuna kaydedilmektedir. Ziyaretçilerin tarayıcı dağılımları bulunurken log tablosu üzerinde Şekil 5.25'de verilen SQL ifadesinin çalışması sonucu elde edilen veriler Şekil 5.26'daki grafik üzerinde gösterilmektedir.

```
SELECT browser, COUNT(browser) as b_say,
ROUND(100*CAST(COUNT(browser) as float)/CAST((SELECT
COUNT(browser) as b_say FROM log) as float),4) as yuzde FROM log
GROUP BY browser ORDER BY b_say DESC
```

Şekil 5.25. Tarayıcı dağılımını bulmak için kullanılan SQL ifadesi.



Şekil 5.26. Tarayıcı dağılımlarının grafiksel gösterimi.

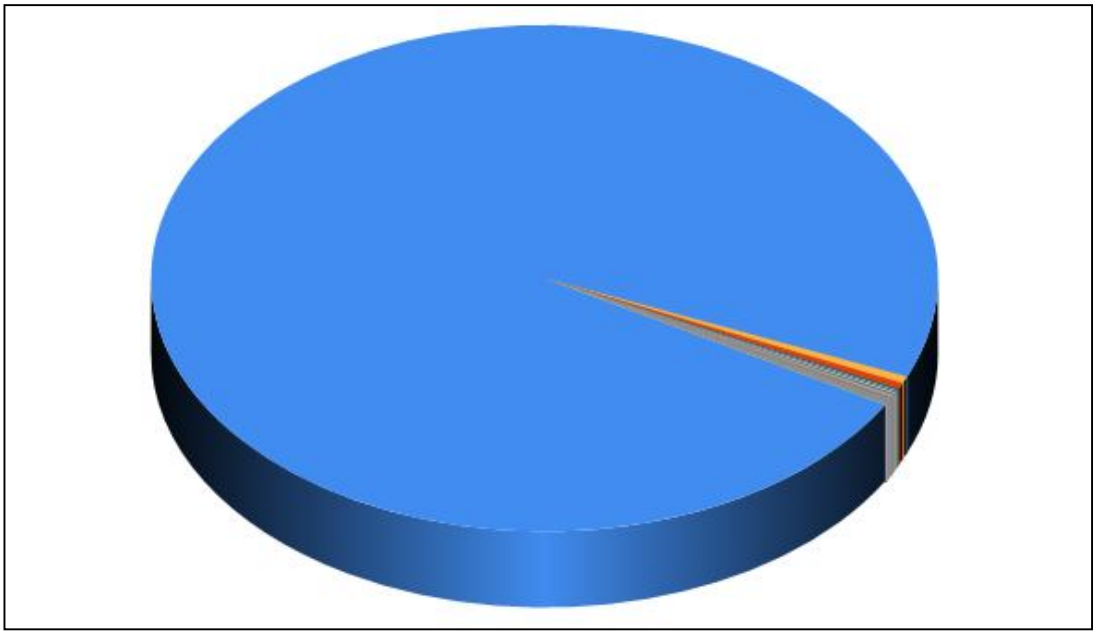
5.3.3.4. Ülke Dağılımı

Ülke dağılımı ile web sitesini ziyaret eden kullanıcıların ülke dağılımı grafiksel olarak elde edilmektedir.

Ziyaretçilerin ülkeleri IP adreslerinden elde edilmektedir. Veri temizleme aşamasında ziyaretçinin IP adresi sayısal bir değere dönüştürülerek veritabanı üzerinde log tablosuna kaydedilmektedir. iptocountry tablosu da ülkelerin sahip olduğu IP aralıklarını içermektedir. Log ve iptocountry tablosu birleştirilerek ziyaretçilerin ülkeleri tespit edilmektedir. .Ziyaretçilerin ülke dağılımları bulunurken kullanılan SQL ifadesi Şekil 5.27'de ve sorgu sonucunun grafiksel gösterimi Şekil 5.28'de verilmiştir.

```
SELECT ulke, COUNT(ulke) ulkesay FROM (SELECT (SELECT ulke FROM
iptocountry WHERE ipcode BETWEEN ilkIP AND sonIP) as ulke FROM
LOG GROUP BY ipcode) aa GROUP BY ulke
```

Şekil 5.27. Ülke dağılımını bulmak için kullanılan SQL ifadesi.



Şekil 5.28. Ülke dağılımlarının grafiksel gösterimi.

Şekil 5.28. (devam ediyor).

5.3.3.5. Günlük Dağılım

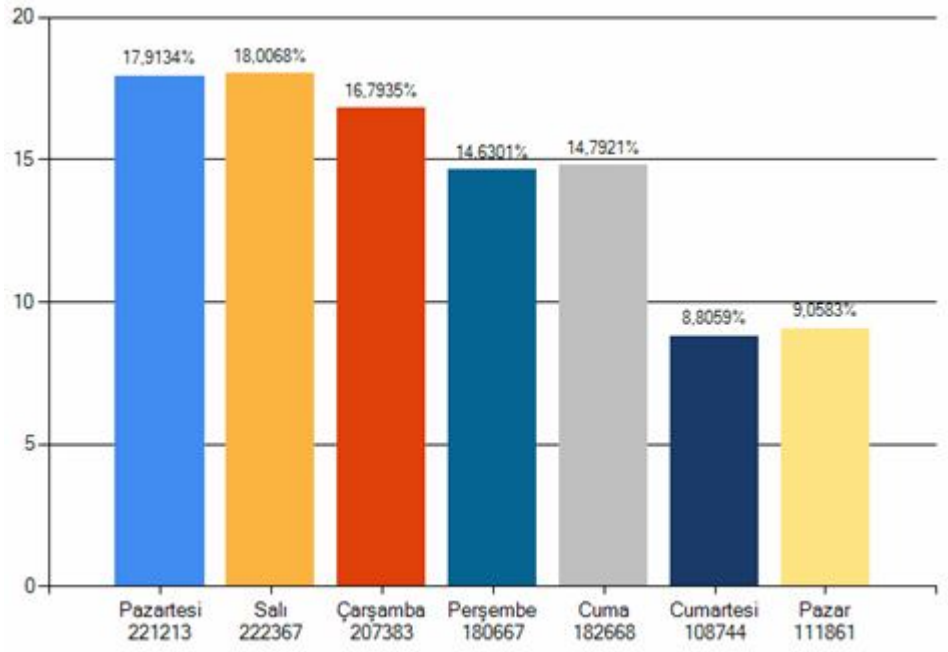
Günlük dağılım ile haftanın günlerinin ziyaret oranı bulunmakta ve grafiksel olarak gösterilmektedir.

Erişim kayıtları veri temizleme aşamasında veritabanına aktarılırken ilgili ziyaret tarihine ait haftanın günü bilgisi de bulunarak veritabanına eklenmiştir.

Ziyaretlerin gün dağılımları bulunurken log tablosu içerisinde ziyaret gününü tutan gun sütunu kullanılmaktadır. Günlük dağılımlar için kullanılan SQL ifadesi Şekil 5.29'da ve sorgu sonucunun grafiksel gösterimi Şekil 5.30'da verilmiştir.

```
SELECT COUNT(gun) as gun_say,gun,ROUND(100*CAST(COUNT(gun) as
float)/CAST((SELECT COUNT(gun) FROM (SELECT gun FROM log GROUP BY
oturum, gun) gec) as float),4) as yuzde FROM (SELECT gun FROM log
GROUP BY oturum, gun) gec GROUP BY gun
```

Şekil 5.29. Günlük dağılımı bulmak için kullanılan SQL ifadesi.



Şekil 5.30. Günlük dağılımların grafiksel gösterimi.

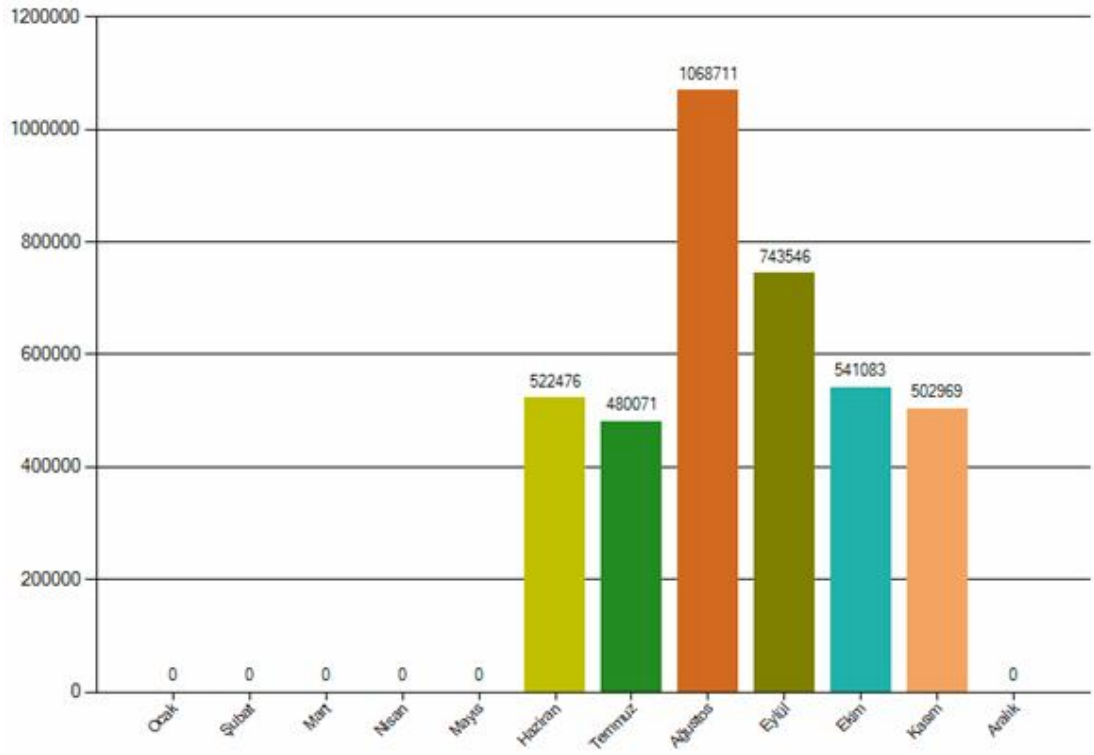
5.3.3.6. Aylık Dağılım

Aylık dağılım ile ayların ziyaret oranı bulunmakta ve grafiksel olarak gösterilmektedir.

Ziyaretlerin aylık dağılımı bulunurken log tablosu içerisinde ziyaret tarihini tutan tarih sütunu kullanılmaktadır. Aylık dağılımlar için kullanılan SQL ifadesi Şekil 5.31'de ve sorgu sonucunun grafiksel gösterimi Şekil 5.32'de verilmiştir.

```
SELECT SUBSTRİNG(tarih,6,2) ay, COUNT(SUBSTRİNG(tarih,6,2)) aa
FROM log GROUP BY SUBSTRİNG(tarih,6,2) ORDER BY ay
```

Şekil 5.31. Aylık dağılımı bulmak için kullanılan SQL ifadesi.



Şekil 5.32. Aylık dağılımların grafiksel gösterimi.

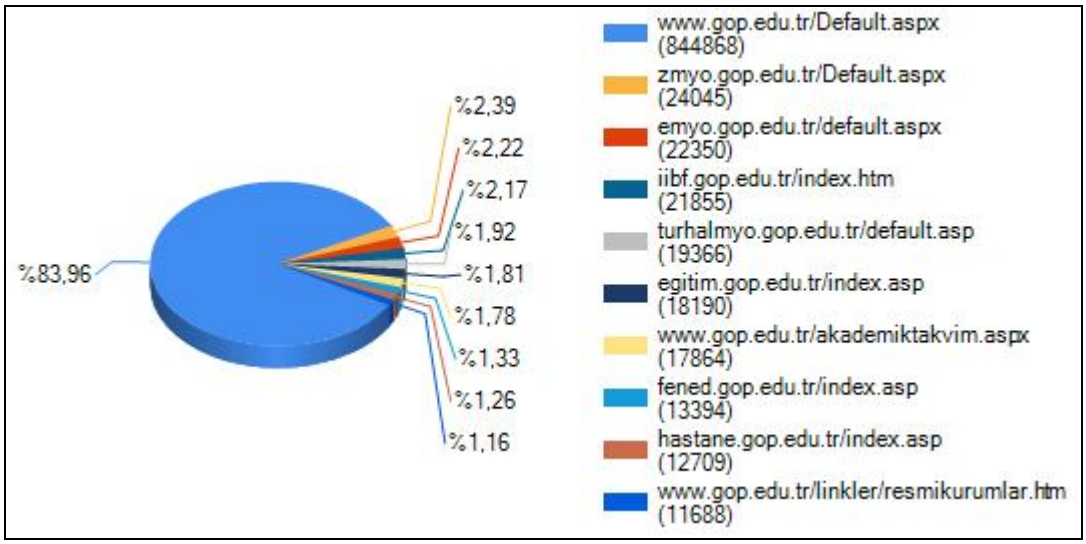
5.3.3.7. En İyi Giriş Sayfaları

En iyi giriş sayfaları ziyaretçilerin site üzerinde gezintiye başladığı ilk sayfayı ifade etmektedir. Kullanıcıların hangi sayfadan ziyarete başladığını tespit etmek için oluşturulan oturumlar ve oturumlar içerisinde gerçekleşen ziyaretler kullanılmıştır. Bir oturum içerisinde gerçekleşen ziyaretlerden tarihi ve saati en erken olan sayfa o oturum için giriş sayfası olarak belirlenmiştir.

Web sitesine ait sayfalardan en yoğun olarak kullanılan ilk 10 giriş sayfası belirlemek için kullanılan SQL ifadesi Şekil 5.33'de ve sorgu sonucu elde edilen verilerin grafiksel sonucu Şekil 5.34'de verilmiştir.

```
SELECT TOP 10 count(url) as ss,url FROM (SELECT (SELECT TOP 1 url
FROM log WHERE oturum=aa.oturum) as url FROM log aa GROUP BY
oturum) gec GROUP BY url ORDER BY COUNT(url) DESC
```

Şekil 5.33. En iyi giriş sayfalarını bulmak için kullanılan SQL ifadesi.



Şekil 5.34. En iyi giriş sayfalarının grafiksel gösterimi.

5.3.3.8. Ziyaret Süreleri

Ziyaret süreleri, web sitesini ziyaret eden kullanıcıların site üzerinde geçirdikleri süreleri grafiksel olarak göstermektedir.

Oturumlar tanımlanırken bir oturumun maksimum süresi 30 dakika olarak belirlendiği için 30 dakikaya kadar 6 farklı aralık belirlenmiştir.

Veritabanı içerisinde bulunan oturum_detay tablosu erişimler için oluşturulan oturumları ve oturumların süresini tutmaktadır. Bu tablo üzerindeki kayıtlar Şekil 5.35'de verilen SQL ifadesi ile listelenmiş ve sorgudan dönen sonuçlar Şekil 5.36'da

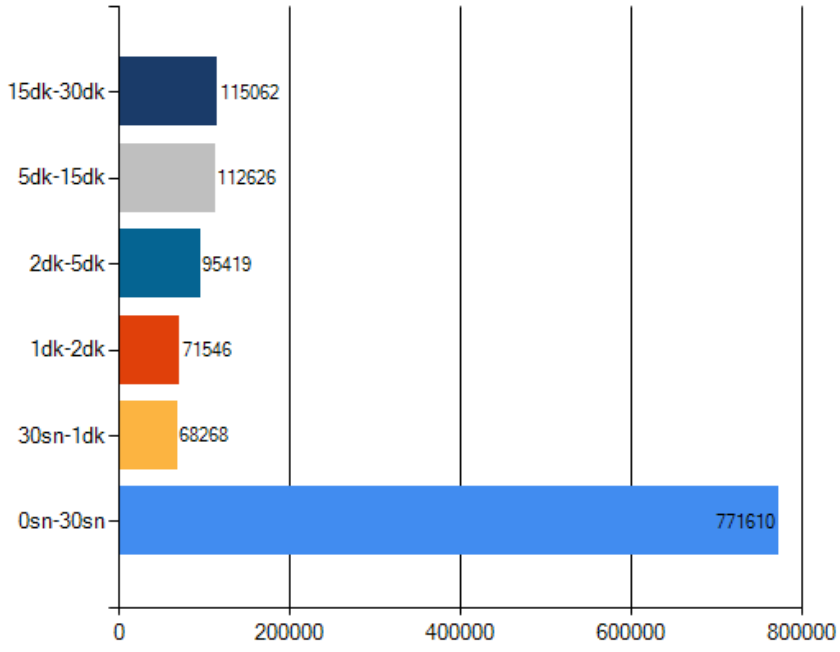
verilen C# kodları yardımı ile ilgili zaman dilimine yerleştirilmiştir. Zaman dilimlerinin içerdiği oturum sayısı *sureler* isimli *int* türündeki dizi değişkende tutulmuş ve değişkenin içeriği Şekil 5.37'de verilen grafik yardımıyla gösterilmiştir.

```
SELECT sure FROM oturum_detay
```

Şekil 5.35. Oturumları ve sürelerini bulmak için kullanılan SQL ifadesi.

```
While (db_oku.Read())
{
      int sr=Convert.ToInt32(db_oku["sure"].ToString());
      if (sr>=0 && sr<=30)
            sureler[0]=sureler[0]+1;
      else if (sr>=31 && sr<=60)
            sureler[1]=sureler[1]+1;
      else if (sr>=61 && sr<=120)
            sureler[2]=sureler[2]+1;
      else if (sr>=121 && sr<=300)
            sureler[3]=sureler[3]+1;
      else if (sr>=301 && sr<=900)
            sureler[4]=sureler[4]+1;
      else
            sureler[5]=sureler[5]+1;
}
```

Şekil 5.36. Sorgu sonucunu dizi değişken yerleştirmek için kullanılan C# kodları.



Şekil 5.37. Ziyaret sürelerinin grafiksel gösterimi.

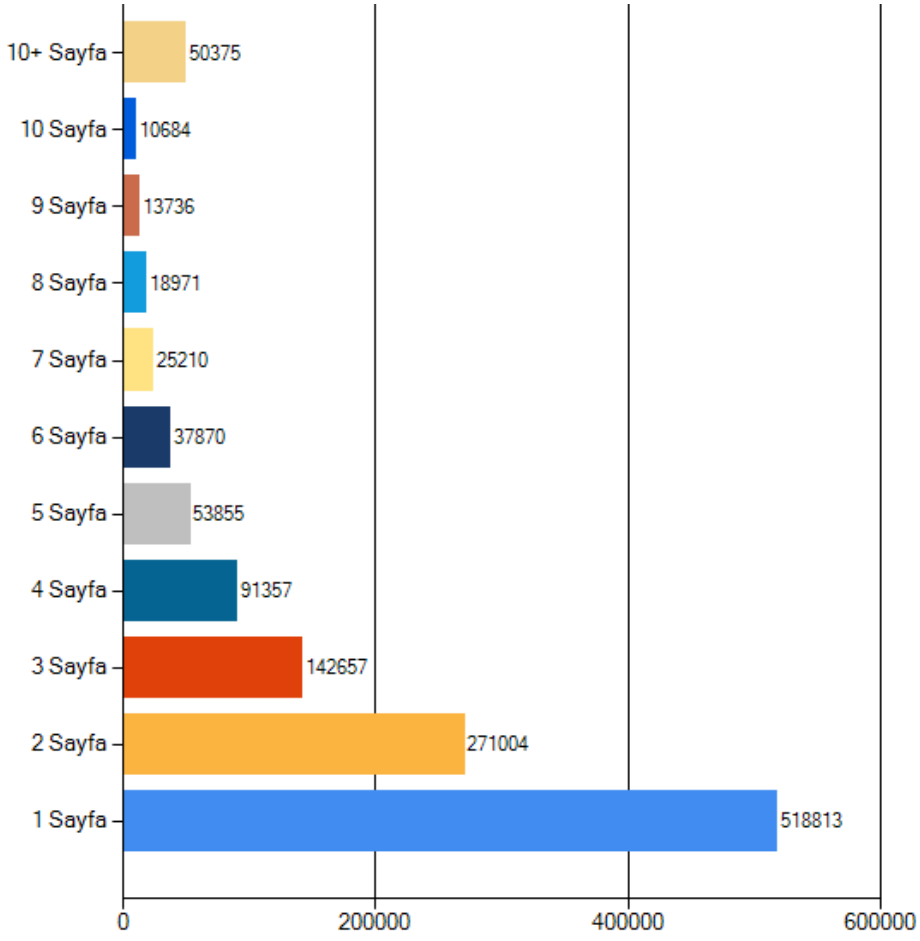
5.3.3.9. Ziyaret Derinliği

Ziyaret derinliği, ziyaretçilerin her bir oturumda ziyaret ettiği sayfa sayısının grafiksel olarak gösterilmesini sağlar. 10 sayfadan fazla olan oturumlar tek bir sütun üzerinde gösterilmiştir.

Oturumlar üzerinde ziyaret edilen sayfa sayısını bulmak için log tablosu içerisindeki kayıtlar Şekil 5.38'de verilen SQL ifadesi ile oturum sütununa göre gruplandırılmış ve count fonksiyonu ile de her bir oturumun içerdiği ziyaret sayısı bulunmuştur. Sorgu sonucu elde edilen verilerin grafiksel gösterimi Şekil 5.39'da verilmiştir.

```
SELECT COUNT(url) as say FROM log GROUP BY oturum
```

Şekil 5.38. Oturumların içerdiği ziyaret sayısını bulan SQL ifadesi.



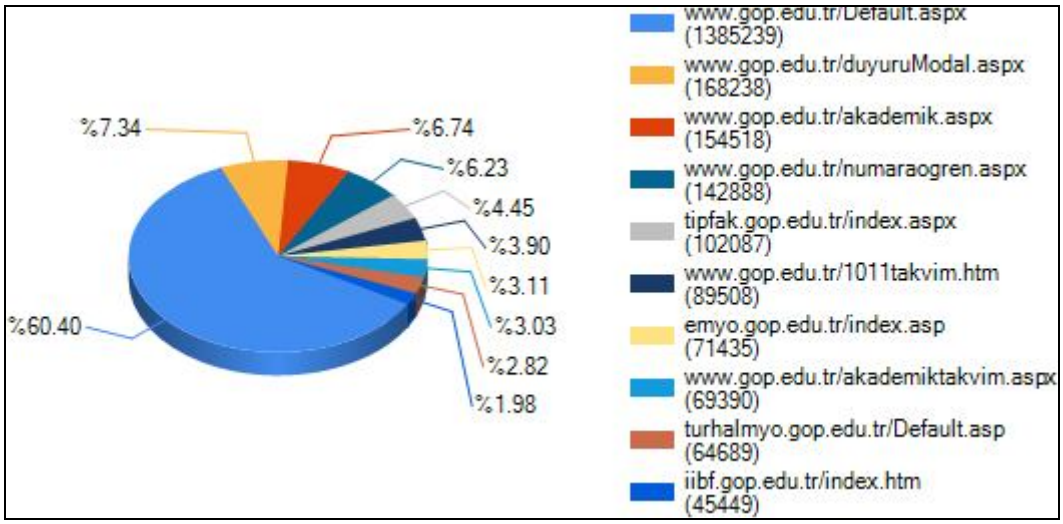
Şekil 5.39. Ziyaret derinliğinin grafiksel gösterimi.

5.3.3.10. Top 10

Top 10, web sitesi içerisinde en yoğun olarak kullanılan ilk 10 sayfayı grafiksel olarak göstermektedir. En yoğun olarak kullanılan 10 sayfayı bulmak için kullanılan SQL ifadesi Şekil 5.40'da ve sorgu sonucunun grafiksel gösterimi Şekil 5.41'de verilmiştir.

```
SELECT TOP 10 url,COUNT(url) as ss FROM log GROUP BY url ORDER
BY ss DESC
```

Şekil 5.40. En yoğun kullanılan sayfaları bulmak için kullanılan SQL ifadesi.



Şekil 5.41. Top 10 dağılımının grafiksel gösterimi.

5.3.3.11. Trafik Dağılım

Trafik dağılımı, web sitesi ziyaret eden kullanıcıların siteye hangi kaynaktan ulaştığını grafiksel olarak göstermektedir. Trafik dağılımı doğrudan erişim, arama motorları ve yönlendirme olmak üzere 3 gruba ayrılmıştır.

Erişim kayıtları içerisinde bulunan cs(Referer) verisi ziyaret edilen sayfaya hangi adresten gelindiğini göstermektedir. Eğer bu bilgi boş ise siteye doğrudan erişildiğini göstermektedir. Yani kullanıcının tarayıcı adres çubuğuna adresi yazarak yaptığı erişimlerdir. Boş olan referans verisi haricindeki veriler arama motorlarının anahtar

kelimelerine (google, bing, search v.b.) göre kontrol edilmekte bunlardan birisi var ise arama motorları grubu içerisine alınmaktadır. Bu iki durum dışındaki erişimler ise yönlendirme olarak düşünülmekte ve yönlendirme grubuna eklenmektedir.

Gelinen kaynağı belirlerken erişimlerin tamamını kullanmak gereksiz bir iş yükü ve yanlış sonuçlara sebep olacağı için oturumlara ait ilk erişimler kullanılmaktadır.

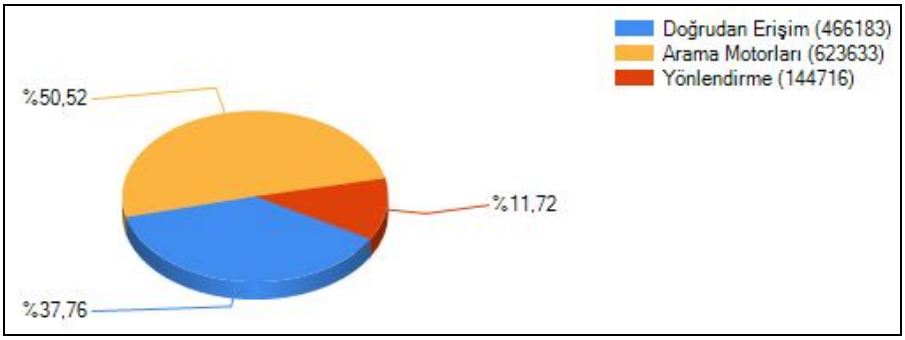
Şekil 5.42'de verilen SQL ifadesi oturumlar içerisindeki ilk erişimlere ait referans verilerini bulmaktadır. Sorgudan elde veriler Şekil 5.43'de verilen C# kodları ile ilgili gruba yerleştirilmekte ve Şekil 5.44'de verilen grafik ile sonuçlar gösterilmektedir.

```
SELECT (SELECT TOP 1 referrer FROM log WHERE oturum=aa.oturum) as
rf,oturum FROM log aa GROUP BY oturum ORDER BY oturum
```

Şekil 5.42. Oturumlara ait ilk erişimlerin referans verisini bulan SQL ifadesi.

```
While (db_oku.Read())
{
      if (db_oku["rf"].ToString() == "-")
            t_say[0] = t_say[0] + 1;
      else if (db_oku["rf"].ToString().Contains("google") ||
            db_oku["rf"].ToString().Contains("babylon") ||
            db_oku["rf"].ToString().Contains("bing") ||
            db_oku["rf"].ToString().Contains("yahoo") ||
            db_oku["rf"].ToString().Contains("search"))
            t_say[1] = t_say[1] + 1;
      else
            t_say[2] = t_say[2] + 1;
}
```

Şekil 5.43. Sorgudan elde edilen verileri ilgili gruba yerleştiren C# kodları.



Şekil 5.44. Trafik dağılımının grafiksel gösterimi.

5.3.3.12. Durum Kodu Dağılımı

Web sitesine yapılan erişimlerin tamamı başarılı olarak gerçekleşmez. Sunucudan veya kullanıcıdan kaynaklı hatalardan dolayı erişim gerçekleşmemiş olabilir. Bu tür durumlarda erişim kayıtları içerisinde kayıt altına alınmaktadır.

Erişim kayıtları içerisindeki sc-status verisi erişim durumunu göstermektedir. 200 değerini içeren erişimler başarıyla gerçekleşmiş demektir.

Veri temizleme aşamasında erişim kayıtları içerisindeki durum bilgileri de log tablosu içerisine status verisi olarak kaydedilmektedir.

Durum kodu dağılımı, yapılan site erişimleri sonucunda oluşan durum kodlarının dağılımını grafiksel olarak göstermektedir. Şekil 5.45'de verilen SQL ifadesi durum kodlarını gruplandırarak gerçekleşme sayısını vermektedir.

```
SELECT LEFT(status,1) as ii, COUNT(status) as say FROM log GROUP
BY LEFT(status,1)
```

Şekil 5.45. Durum kodlarının dağılımını bulmak için kullanılan SQL ifadesi.

Şekil 5.45'de verilen SQL ifadesi sonucu elde edilen verilerin grafiksel dağılımı Şekil 5.46'da verilmiştir.

Şekil 5.46. Durum kodu dağılımının grafiksel gösterimi.

5.3.3.13. Alt Domain Analizi

Bu çalışma da giriş verisi olarak kullanılan web sitesi kendisiyle birlikte alt domainler de içermektedir.

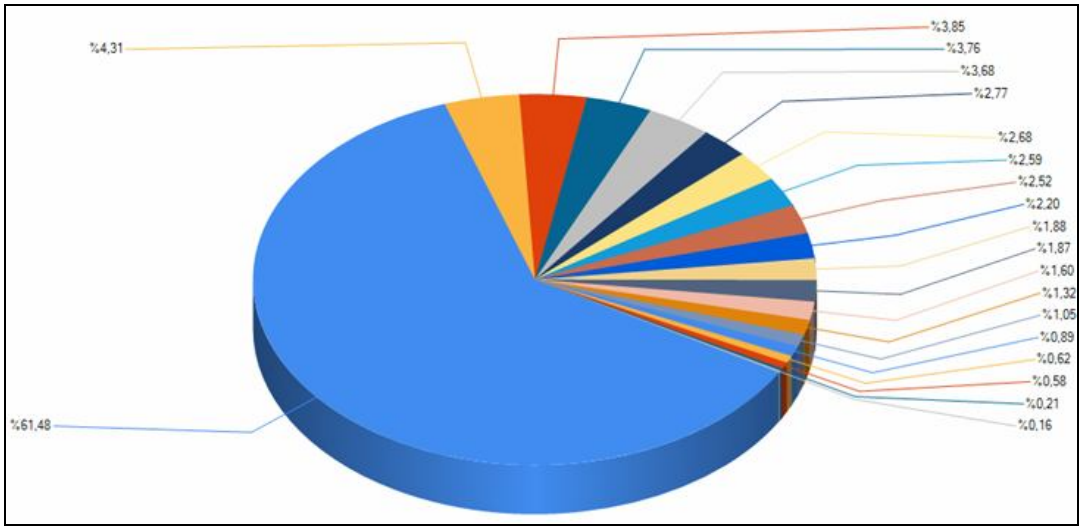
Alt domain analizi ile kuruma ait yoğun olarak kullanılan domainlerin erişim dağılımları grafiksel olarak gösterilmektedir.

Veri temizleme aşamasında erişim kayıtları ile birlikte ait olduğu domain bilgisi de seçilmektedir. Veri temizleme işlemi sonrasında temiz veriler veritabanına aktarılırken ait olduğu domain bilgisi de log tablosu içerisindeki domain sütununa eklenmektedir.

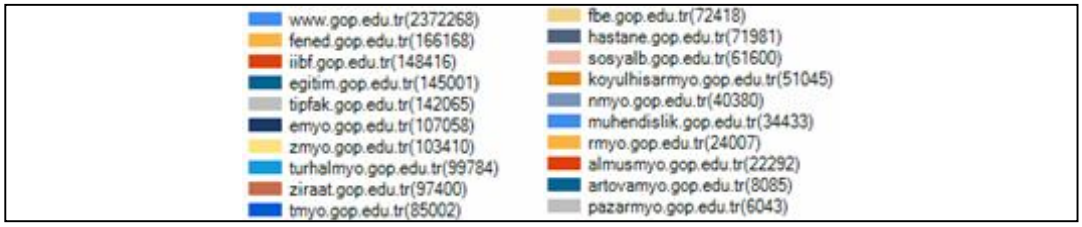
Alt domain analizi için kullanılan SQL ifadesi Şekil 5.47'de ve sorgu sonucu elde edilen verilerin grafiksel gösterimi Şekil 5.48'de verilmiştir.

```
SELECT LEFT(status,1) as ii, COUNT(status) as say FROM log GROUP
BY LEFT(status,1)
```

Şekil 5.47. Alt domain analizi için kullanılan SQL ifadesi.



Şekil 5.48. Alt domain analizinin grafiksel gösterimi.



Şekil 5.48. (devam ediyor).

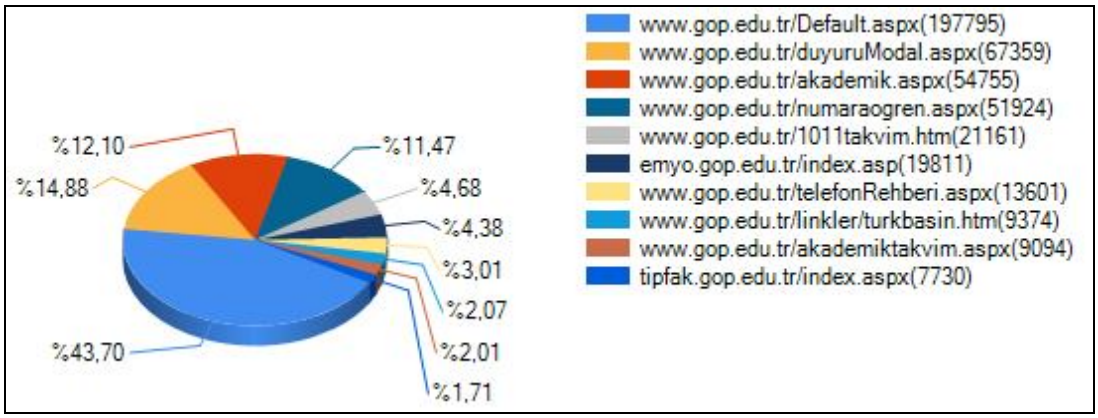
5.3.3.14. Çıkış Sayfaları

Çıkış sayfaları, ziyaretçilerin ziyaretleri sonrası siteden ayrıldıkları ilk 10 sayfayı göstermektedir.

Ziyaretçilerin, bir oturum içerisinde ziyaret ettiği en son sayfa çıkış sayfası olarak düşünülebilir. Bu sayfaları tespit etmek için kullanılan SQL ifadesi Şekil 5.49'da ve sorgu sonucu elde edilen verilerin grafiksel gösterimi Şekil 5.50'de verilmiştir.

```
SELECT TOP 10 COUNT(url) as ss,url FROM (SELECT (SELECT TOP 1 url
FROM log WHERE oturum=aa.oturum ORDER BY saat DESC) as url FROM
log aa GROUP BY oturum HAVİNG COUNT(oturum)>1) gec GROUP BY url
ORDER BY COUNT(url) desc
```

Şekil 5.49. Oturumlarda en son ziyaret edilen sayfa için kullanılan SQL ifadesi.

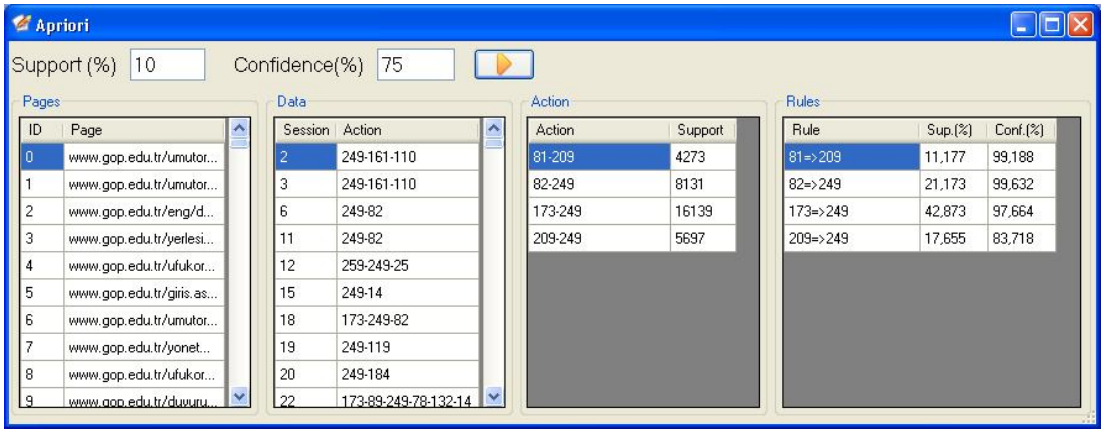


Şekil 5.50. Çıkış sayfalarının grafiksel gösterimi.

5.3.3.15. Apriori

Log analiz programı apriori seçeneği Gaziosmanpaşa Üniversitesi web sitesine ait sayfalardan birlikte kullanılanları apriori algoritması ile tespit etmek için kullanılmıştır.

Şekil 5.51'de verilen ekran görüntüsü Log Analiz programının apriori işlemini gerçekleştirmektedir. Apriori algoritmasının kullanacağı destek değeri Support(%) kısmından ve birliktelikler için kullanılacak olan güven değeri de Confidence(%) kısmından girilmektedir.



Şekil 5.51. Log analiz programı apriori penceresi.

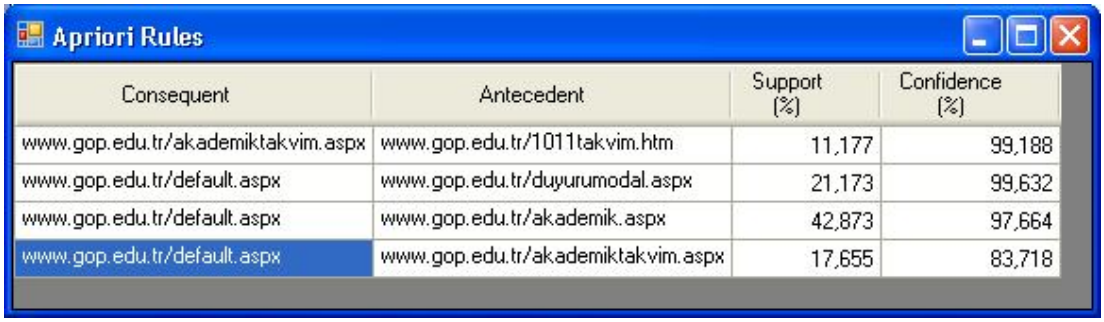
Pencere üzerinde bulunan datagrid nesnelerinin içerdiği veriler aşağıda açıklanmıştır.

- Veritabanına aktarılan erişim kayıtlarında kullanıcılar tarafından erişilen sayfalar ID numarası eklenerek pages bölümünde bulunan grid içerisine eklenir. ID değeri veritabanı içerisinde bulunan kullanıcı hareketlerini tespit ederken sayfa ismi yerine kullanılır. Ayrıca grid üzerinde gizli olarak bulunan ve her sayfanın destek değerini tutan bir sütunu bulunmaktadır. Bu sütun apriori algoritmasına ait 1-elemankümeyi oluşturmak için kullanılır.
- Veritabanı taranarak her bir oturumda kullanıcıların ziyaret etmiş olduğu sayfalar bulunmakta ve oturum numarası ile birlikte hareket olarak data bölümündeki grid içerisine eklenir. Burada yer alan veriler apriori algoritması için giriş verilerini oluşturmaktadır. Veritabanı içerisinde yer alan tüm

hareketler grid içerisine eklendiği için apriori her defasında veritabanına bağlantı kurup destek değerlerini bulmak yerine buradaki verileri kullanır.

- Algoritmanın çalışması esnasında oluşturulan L öğe kümeleri ve destek değerleri apriori'nin her basamağı için action bölümünde listelenmektedir.
- Apriori algoritmasının son bulduğu L öğe kümelerine ait çıkarılan kurallar, destek ve güven değerleri rules bölümünde yer alır.

Apriori algoritması çalışmasını bitirdiğinde çıkarılan kurallar Rules içerisinde bulunan sayfa kodları yerine sayfa isimleri ile birlikte Şekil 5.52'de verilen apriori rules penceresinde listelenmektedir.



Şekil 5.52. Apriori sonucu çıkartılan kurallar.

Log analiz programında apriori algoritmasının uygulama adımları aşağıda verilmiştir.

- Log tablosu içerisinde bulunan adresler gruplandırılarak belirlenir. Belirlenen adresler pages datagrid nesnesi ve mevcut_url isimli arraylist içerisine eklenir.
- Log ve oturum_detay tablosu kullanılarak her bir oturumda ziyaret edilen sayfalar belirlenerek oturum numarası ile birlikte datagrid nesnesine eklenir. Elde edilen bu hareketler apriori için veritabanı olarak kullanılacaktır.
- Apriori_start() isimli yordam ile Ck aday kümeleri ve Lk sık geçen öğe kümeleri belirlenir.
- Ck aday küme içerisindeki her bir öğe kümenin destek değerlerini bulmak için destek_bul(String transaction) isimli fonksiyon tanımlanmıştır.

- destek_bul fonksiyonu kendisine gelen transaction için data içerisinden destek değerini bularak, bulunan destek değeri minimum destek eşik değerini sağlıyorsa destek değerini, sağlamıyorsa -1 değerini döndürmektedir.
- Apriori algoritması sık geçen öğe kümesi bulunmayana kadar devam eder. Son bulunan sık geçen öğe kümesi sonrasında güven değerlerini belirlemek için sık geçen öğe kümelerin alt kümeleri confidence_list() isimli yordam ile gerçekleştirilmektedir.
- Confidence_list ile bulunan her bir alt kümenin güven değerlerini hesaplamak için conf_hes isminde bir fonksiyon hazırlanmıştır.
- Hesaplanan güven değerleri sonrasında belirlenen güven eşik değerini sağlayan kurallar listelenip diğerleri silinmektedir.

Örnek veri kümesi üzerinde %5 destek ve %75 güven değeri ile log analiz programından elde edilen kurallar Şekil 5.53'de ve SPSS Clementine ile elde edilen kurallar Şekil 5.54'de verilmiştir.

Consequent	Antecedent	Support (%)	Confidence (%)
www.gop.edu.tr/akademiktakvim.aspx	www.gop.edu.tr/0910takvim.htm www.gop.edu.tr/akademik.aspx	6,741	100,000
www.gop.edu.tr/akademiktakvim.aspx	www.gop.edu.tr/0910takvim.htm www.gop.edu.tr/akademik.aspx www.gop.edu.tr/default.aspx	5,907	100,000
www.gop.edu.tr/akademiktakvim.aspx	www.gop.edu.tr/0910takvim.htm www.gop.edu.tr/default.aspx	18,574	99,900
www.gop.edu.tr/akademiktakvim.aspx	www.gop.edu.tr/0910takvim.htm	21,889	99,831
www.gop.edu.tr/0910takvim.htm	www.gop.edu.tr/akademik.aspx www.gop.edu.tr/akademiktakvim.aspx www.gop.edu.tr/default.aspx	5,926	99,688
www.gop.edu.tr/0910takvim.htm	www.gop.edu.tr/akademik.aspx www.gop.edu.tr/akademiktakvim.aspx	6,778	99,454
www.gop.edu.tr/0910takvim.htm	www.gop.edu.tr/akademiktakvim.aspx www.gop.edu.tr/default.aspx	18,741	99,012
www.gop.edu.tr/0910takvim.htm	www.gop.edu.tr/akademiktakvim.aspx	22,093	98,910
www.gop.edu.tr/default.aspx	www.gop.edu.tr/akademik.aspx	47,667	97,514
www.gop.edu.tr/default.aspx	www.gop.edu.tr/0910takvim.htm www.gop.edu.tr/akademik.aspx	6,741	87,637
www.gop.edu.tr/default.aspx	www.gop.edu.tr/0910takvim.htm www.gop.edu.tr/akademik.aspx www.gop.edu.tr/akademiktakvim.aspx	6,741	87,637
www.gop.edu.tr/default.aspx	www.gop.edu.tr/akademik.aspx www.gop.edu.tr/akademiktakvim.aspx	6,778	87,432
www.gop.edu.tr/default.aspx	www.gop.edu.tr/0910takvim.htm www.gop.edu.tr/akademiktakvim.aspx	21,852	84,915
www.gop.edu.tr/default.aspx	www.gop.edu.tr/0910takvim.htm	21,889	84,856
www.gop.edu.tr/default.aspx	www.gop.edu.tr/akademiktakvim.aspx	22,093	84,828

Şekil 5.53. Log analiz programı ile elde edilen birliktelik kuralları.

Consequent	Antecedent	Suppo...	Confi...
www.gop.edu.tr/akademiktakvim.aspx = T	www.gop.edu.tr/0910takvim.htm = T www.gop.edu.tr/akademik.aspx = T	6,741	100,0
www.gop.edu.tr/akademiktakvim.aspx = T	www.gop.edu.tr/0910takvim.htm = T www.gop.edu.tr/Default.aspx = T	16,519	99,888
www.gop.edu.tr/akademiktakvim.aspx = T	www.gop.edu.tr/0910takvim.htm = T	21,889	99,831
www.gop.edu.tr/0910takvim.htm = T	www.gop.edu.tr/akademiktakvim.aspx = T www.gop.edu.tr/akademik.aspx = T	6,778	99,454
www.gop.edu.tr/0910takvim.htm = T	www.gop.edu.tr/akademiktakvim.aspx = T	22,093	98,91
www.gop.edu.tr/0910takvim.htm = T	www.gop.edu.tr/akademiktakvim.aspx = T www.gop.edu.tr/Default.aspx = T	16,685	98,89
www.gop.edu.tr/Default.aspx = T	www.gop.edu.tr/linkler.aspx = T	5,13	96,39
www.gop.edu.tr/Default.aspx = T	www.gop.edu.tr/duyuruModal.aspx = T	17,056	94,463
www.gop.edu.tr/Default.aspx = T	www.gop.edu.tr/telefonRehberi.aspx = T	6,889	93,28
www.gop.edu.tr/Default.aspx = T	www.gop.edu.tr/akademik.aspx = T	47,667	91,88
www.gop.edu.tr/Default.aspx = T	www.gop.edu.tr/duyuruAlt.aspx = T	6,667	90,278
www.gop.edu.tr/linkler/turkbasin.htm = T	www.gop.edu.tr/linkler.aspx = T	5,13	80,144
www.gop.edu.tr/Default.aspx = T	www.gop.edu.tr/akademiktakvim.aspx = T	22,093	75,524
www.gop.edu.tr/Default.aspx = T	www.gop.edu.tr/0910takvim.htm = T www.gop.edu.tr/akademiktakvim.aspx = T	21,852	75,508
www.gop.edu.tr/Default.aspx = T	www.gop.edu.tr/0910takvim.htm = T	21,889	75,465

Şekil 5.54. SPSS Clementine ile elde edilen birliktelik kuralları.

BÖLÜM 6

SONUÇLAR VE ÖNERİLER

İnternet kullanımının her geçen gün artması sunucular üzerinde tutulan verilerin hızlı bir şekilde artmasına neden olmaktadır. Metin dosyaları içerisinde tutulan bu veriler erişim kayıtları olarak geçmekte ve analiz edilerek anlamlandırılması web kullanım madenciliği ile gerçekleştirilmektedir.

Bu çalışma da, öncelikle veri madenciliği, web madenciliği, web kullanım madenciliği ve aşamaları bölümler halinde verilmiştir. Uygulama bölümünde ise web kullanım madenciliğinin tüm süreçlerini içeren ve ilgili erişim kayıtlarından çeşitli istatistiki bilgiler çıkartan bir yazılım tasarlanmıştır. Hazırlanan yazılım ile Gaziosmanpaşa Üniversitesi kurumsal web sitesine ait altı aylık erişim kayıtları analiz edilmiş elde edilen sonuçlar uygulama bölümünde verilmiştir.

Gaziosmanpaşa Üniversitesi web sitesi analizi ile

- Erişim kayıtlarına ait genel bilgiler,
- Ziyaretçilerin kullanmış olduğu işletim sistemleri dağılımı,
- Ziyaretçilerin kullanmış olduğu tarayıcı dağılımı,
- Ziyaret gerçekleştirilen ülkelerin dağılımı,
- Aylık erişim dağılımı,
- En iyi giriş sayfaları,
- Ziyaret süreleri,
- Ziyaret derinliği,
- En çok ziyaret edilen ilk 10 sayfa,
- Trafik dağılımı,
- Durum kodu dağılımı,
- Alt domain analizi,

- Çıkış sayfaları,
- Birlikte ziyaret edilen sayfalar gibi çeşitli sonuçlar elde edilmiştir.

Hazırlanan yazılım ile ön işlem sürecinden geçirilen veriler SQL veritabanına aktarılarak sonraki süreçlerin daha hızlı bir şekilde gerçekleştirilmesi sağlanmıştır.

Bu ve benzeri çalışmalar ile web site yöneticilerine web sitesinin geliştirilmesi veya yeniden tasarlanması için önemli bilgiler sunulmaktadır. Yapılan analizler sonucunda;

- Web sitesinin yoğun olarak ana sayfasının kullanıldığı iç kısımlarda kalan sayfaların çoğunlukla kullanılmadığı tespit edilmiştir.
- Ziyaret edilen sayfa sayıları incelendiğinde çoğunlukla duyuru, haber ve akademik takvimle ilgili sayfaların ziyaret edildiği görülmektedir ve bu durum ziyaretçilerin büyük çoğunluğunun mevcut öğrenciler ve personel olduğunu göstermektedir.
- Çeşitli destek ve güven değerleri ile yapılan birliktelik analizi sonuçlarının tamamında ana sayfa, akademik takvim ve aktif eğitim öğretim yılına ait akademik takvim birliktelikleri çıkmaktadır. Ziyaretçiler aktif akademik takvime ana sayfa üzerinde bulunan akademik takvim linki ile ulaşmaktadır. Aktif döneme ait akademik takvim ana sayfaya yerleştirilerek erişim zamanı kısaltılabilir.
- İsteklerin oluşturduğu http durum kodu incelendiğinde sunucudan kaynaklı hataların oluşturduğu 5XX değerinin düşük olması genel olarak sunucunun sorunsuz çalıştığını göstermektedir.
- Domain analizi sonuçlarına göre, ana domain erişimi %61.48 ile ilk ve pazarmyo alt domaini %0.16 ile son sırada yer almaktadır. pazarmyo, yeni açılan bir yüksekokul olduğu için elde edilen sonuçlar mantıklıdır. Tüm domainler incelendiğinde kuruma ait alt domainlerin etkin bir şekilde kullanılmadığı görülmektedir. Alt domainlerin etkinliğini artırmak için sitelerin sadece duyuru ve haber dışında ek bilgi ve belgeleri içermesi sağlanabilir.

KAYNAKLAR

Adriaans, P. And Zantinge, D., "Data mining", ***Addison Wesley Longman***, England, 5-8, 69-70 (1996).

Akpınar, H., "Veri tabanlarında bilgi keşfi ve veri madenciliği", ***İstanbul Üniversitesi İşletme Fakültesi Dergisi,*** 29(1): 1-22 (2000).

Alpaydın, E., "Zeki veri madenciliği: Ham veriden altın bilgiye ulaşma yöntemleri", ***Bilişim 2000 Eğitim Semineri***, (2000).

Aynekin, G., "İnternet içerik madenciliğinde yapay sinir ağları ve bir uygulama", Yüksek Lisans Tezi, ***Uludağ Üniversitesi Fen Bilimleri Enstitüsü***, Bursa, 21-26 (2006).

Berendt, B., Mobasher, B., Nakagawa, M. And Spiliopoulou, M., "The impact of site structure and user environment on session reconstruction in web usage analysis", ***Proceedings of the WebKDD 2002 Workshop***, Canada, (2002).

Berendt, B., Mobasher, B., Spiliopoulou, M. And Wiltshire, J., "Measuring the accuracy of sessionizers for web usage analysis", ***Proceedings of the Workshop on Web Mining at the First SIAM International Conference on Data Mining***, Chicago, (2001).

Bing, L., "Web data mining : Exploring hyperlinks, contents, and usage data", ***Springer-Verlag Berlin Heidelberg***, New York, 4-6 (2007).

Belen, E., Özgür, Ç. Ve Özakar, B., "WALA : Web erişim kütük araştırmacısı", ***9.Türkiye'de İnternet Konferansı***, İstanbul(2008).

Buchner, A.G., Baumgarten, M., Anand, S.S., Mulvenna, M.D. And Hughes, J.G., "Navigation pattern discovery from internet data", ***In WEBKDD***, San Diego, CA, (1999).

Catledge, L. And Pitkow, J., "Characterizing browsing behaviors on the world wide web", ***Computer Networks and ISDN Systems***, 27(6): 1065-1073 (1995).

Cerny, P.A., "Data mining and neural networks from a commercial perspective", ***Conference Twenty Naught One of the Operational Research Society of New Zealand***, Australia, (2001).

Chaofeng, L., "Research and development of data preprocessing in web usage mining", ***International Conference on Management Science and Engineering***, South-Central University for Nationalities, China (2006).

Cooley, R., "Web usage mining: Discovery and application of interesting patterns from web data", PhD thesis, ***University of Minnesota***, USA, 170-180 (2000).

Cooley, R., Mobasher, B. And Srivastava, J., "Data preparation for mining world wide web browsing patterns", ***Knowledge and Information Systems***, 1: 1–27 (1999).

Cooley, R., Mobasher, B. And Srivastava, J., "Web mining: Information and pattern discovery on the world wide web", ***In Proceedings of the 9th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'97)***, USA, 558 – 567 (1997).

Çetiner, M.H., Karagöz, N.A. Ve Erten, Y.M., "Uzaktan eğitim web sunucularının erişim kütükleri analizi", ***Akademik Bilişim 2000***, Süleyman Demirel Üniversitesi, Isparta (2000).

Daş, R., "Web kullanıcı erişim kütüklerinden bilgi çıkarımı", Doktora Tezi, ***Fırat Üniversitesi Fen Bilimleri Enstitüsü***, Elazığ, 32-33 (2008).

Daş, R., Türkoğlu, İ. Ve Poyraz, M., "Web kayıt dosyalarından ilginç örüntülerin keşfedilmesi", ***Fırat Üniversitesi Fen ve Mühendislik Bilimleri Dergisi***, Elazığ, 19(4): 493-503 (2007).

Daş, R., Türkoğlu, İ. Ve Poyraz, M., "Analyzing of the user access logs of a website using web usage mining method: Example of Firat University", ***e-Journal of New World Sciences Academy (NWSA), Natural and Applied Sciences***, 3(2):310-320 (2008).

Daş, R. Ve Türkoğlu, İ., "Web tabanlı öğretim materyallerinin web kullanım madenciliği ile analiz edilmesi", ***Fırat Üniversitesi Fen ve Mühendislik Bilimleri Dergisi***, Elazığ, 22(1): 111-122 (2010).

Elmas, Ç., "Yapay zeka uygulamaları", ***Seçkin Yayıncılık***, Ankara, 22-26 (2007).

Enke, D. And Thawornwong, S., "The use of data mining and neural networks for forecasting stock market returns", ***Expert Systems with Applications,*** 29(4): 927-940 (2005).

Etzioni, O., "The world wide web: Quagmire or gold mine", ***Communications of the ACM***, 39(11): 65-68, (1996).

Fayyad, U., Shapiro G. And Smyth, P., "From data mining to knowledge discovery in databases", ***AI Magazine***, 37-54 (1996).

Gezer, M., Erol, Ç. Ve Gülseçen, S., "Bir web sayfasının web madenciliği ile analizi", ***Akademik Bilişim 2007***, Dumlupınar Üniversitesi, Kütahya (2007).

Gündüz, Ş. Ve Adalı, E., "Web kullanıcılarının davranışları için örüntü bulma ve modelleme", ***itüdergisi/d***, İstanbul, 3(6): 15-24 (2004).

Gürcan, F. Ve Köse, C., ”Web içerik madenciliği ve konu sınıflandırması”, ***Akademik Bilişim 2008***, Çanakkale 18 Mart Üniversitesi, Çanakkale (2008).

Gürsoy, U. T., “Veri madenciliği ve bilgi keşfi”, ***Pegem Akademi***, Ankara, 3-21, 25-28, 31-45 (2009).

Güvenç, E., “Student performance assesment in higher education using data mining”, Yüksek Lisans Tezi, ***Boğaziçi Üniversitesi Fen Bilimleri Enstitüsü***, İstanbul, 5-14 (2001).

Han J. And Kamber M., ”Data mining: Concepts and techniques”, ***Morgan Kaufmann Publishers***, San Francisco, 234-240 (2001).

Hornick, M. F., Marcade E. And Venkayala S., “Java data mining: Strategy, standard, and practice”, ***Morgan Kaufmann Publishers***, United States of America, 52-60 (2007).

Ian, W. And Eibe, F., “Data mining practical machine learning tools and techniques 2nd ed.”, ***Morgan Kaufmann Publishers***, 8-10 (2005).

Iocchi, L., “The Web OEM approach to web information extraction”, ***Journal of Network and Computer Applications***, 22: 259-269 (1999).

İnternet: Analog “Kişiselleştirme amaçlı yazılım projesi” **http://www.analog.cx** (2010).

İnternet: Cross Industry Standard Process for Data Mining, “CRISP-DM 1.0 Step-by-step data mining guide” **http://www.crisp-dm.org/CRISPWP-0800.pdf** (2010).

İnternet: Data Mining Community’s, “KDNuggets polls-industries / fields where you applied data mining in 2009” **http://www.kdnuggets.com/ polls/2009 /industries-data-mining-applications.htm** (2010).

İnternet: Microsoft Corporation “Microsoft Chart Controls for Microsoft .NET Framework 3.5” **http://www.microsoft.com** (2010).

İnternet: NetIQ “NetIQ web trends log analyzer” **http://www.netiq.com** (2010).

İnternet: Nihuo “Nihuo Web Log Analyzer” **http://www.nihuo.com** (2010).

İnternet: W3C Working Draft “Web characterization terminology & defininiation sheet” **http://www.w3.org/1999/05/WCA-terms** (2010).

İnternet: WebTrends Marketing Web Analytics and Web Statistics “Web Madenciliği Uygulama Yazılımı” **http://www.webtrends.com** (2010).

Joachhims, T., Freitag, D. And Mitchell, T., “Webwatcher: A tour guide for the world wide web”, ***In The 15th International Conference on Artificial Intelligence***, Nagoya, Japan (1997).

Kantardzic M., "Data mining: Concepts, models, methods and algorithms", ***John Wiley&Sons***, (2003).

Kaya, H. Ve Köymen, K., "Veri madenciliği kavramı ve uygulama alanları", ***Fırat Üniversitesi Doğu Anadolu Bölgesi Araştırmaları Dergisi***, Elazığ, 6(2): 159-164 (2008).

Kosala, R. And Blockeel, H., "Web mining research: a survey", ***SIGKDD: SIGKDD explorations: newsletter of the special interest group (SIG) on knowledge discovery & data mining, ACM***, 2(1): 1–15 (2000).

Larose, D.T., "Discovering knowledge in data", ***John Wiley & Sons, Inc.***, 16-17, 128-129 (2005).

Leieberman, H., "Letizia: An agent that assists web browsing", ***In Proc. Of the 1995 International Joint Conference on Artificial Intelligence***, Canada (1995).

Liao, S. And Wen, C., "Artificial neural networks classification and clustering of methodologies and applications – literature analysis from 1995 to 2005", ***Expert Systems with Applications***, 32: 1-11 (2005).

Liu, B., "Web data mining: Exploring hyperlinks, contents and usage data", ***Springer***, 452-455 (2006).

Liu, H. And Keselj, V., "Combined mining of web server logs and web contents for classifying user navigation patterns and predicting users' future requests", ***Data & Knowledge Engineering***, 61: 304-330 (2007).

Mobasher, B., Cooley, R. And Srivasta, J., "Creating adaptive web sites through usage-based clustering of URLs", ***In Knowledge and Data Engineering Workshop***, (1999).

Moore, J., Han, E.H., Boley, D., Gini, M., Gross, R., Hastings, K., Karypis, G., Kumar, V. And Mobasher, B., "Web page categorization and feature selection using association rule and principal component clustering", ***In 7th Workshop on Information Technologies and Systems***, (1997).

Nadjarbashi, M. And Ghorbani, A., "Improving the referrer-based web log session reconstruction", ***Second Annual Conference on Communication Networks and Services Research***, Canada, (2004).

Ngu, D.S.W. And Wu, X., "Sitehelper: A localized agent that helps incremental exploration of the world wide web", ***In 6th International World Wide Web Conference***, Santa Clara, (1997).

Nisbet, R., Elder, J. And Miner, G., "Handbook of statistical analysis and data mining applications", ***Elsevier Inc.***, Canada, 23-24, 35-46 (2009).

Özekes, S., "Veri madenciliği modelleri ve uygulama alanları", ***İstanbul Ticaret Üniversitesi Dergisi***, 3: 65-82 (2003).

Özkan. Y., "Veri madenciliği yöntemleri", ***Papatya Yayıncılık Eğitim***, İstanbul, 45-46, 53-54 (2008).

Özmen, Ş., "İş hayatı veri madenciliği ile istatistik uygulamalarını yeniden keşfediyor", ***V.Ulusal Ekonometri ve İstatistik Sempozyumu***, Çukurova Üniversitesi, (2001).

Özseven, T., "Veritabanı yönetim sistemleri II", ***Murathan Yayınevi***, Trabzon, 57-58 (2010).

Perkowitz, M. And Etzioni, O., "Adaptive web sites: Automatically synthesizing web pages", ***In Fifteenth National Conference on Artifical Intelligence,*** Madison, WI, (1998).

Perkowitz, M. And Etzioni, O., "Adaptive web sites: Conceptual cluster mining", ***In Sixteenth International Joint Conference on Artifical Intelligence***, Stockholm, Sweden, (1999).

Pitkow, J., "In search of reliable usage data on the www", ***In Sixth International World Wide Web Conference***, Santa Clara, CA., 451-463 (1997).

Pitkow, J. And Kehoe, C.M., "Results from the thrid www user survey", ***The World Wide Web Journal***, 1(1) (1995).

Ramkumar, G.D. And Swami, A., "Clustering data without distance functions", ***IEEE Bulletin ofthe Technical Committee on Data Engineering***, 21(1): 9-14 (1998).

Sever, H. Ve Oğuz, B., "Veri tabanlarında bilgi keşfine formel bir yaklaşım: Kısım I - eşleştirme sorguları ve algoritmalar", ***Bilgi Dünyası***, 3(2): 173-204 (2002).

Silahtaroğlu, G., "Kavram ve algoritmalarıyla temel veri madenciliği", ***Papatya Yayıncılık Eğitim***, İstanbul, 11-18 (2008).

Singh, Y. And Chauhan, A. S., "Neural networks in data mininig", ***Journal of Theoretical and Applied Information Technology***, 5(1): 37-42 (2009).

Soares, C., Peng, Y., Meng, J., Washio, T. And Zhou, Z., "Applications of data mining in e-business and finance", ***IOS Press***, Amsterdam, 2-4 (2008).

Spiliopoulou, M., Mobasher, B., Berendt, B. And Nakagawa, M., "A framework for the evaluation of session reconstruction heuristics in web-usage analysis", ***INFORMS Journal on Computing***, 15 (2): 171–190 (2003).

Srivastava, J., Cooley, R., Deshpande, M. And Tan, P., "Web usage mining: discovery and applications of usage patterns from web data", ***SIGKDD Explorations***, 1(2): 12-23 (2000).

Srivastava, J., Desikan, P. And Kumar, V., "Web mining: Concepts, applications and research directions", ***Studies in Fuzziness and Soft Computing***, 180: 275-307 (2005).

Susan, P., "Effective use of the KDD process and data mining for computer performance professionals", ***CMG(Computer Measurement Group) Conference,*** California, (2001)

Tantuğ, A.C., "Veri madenciliği ve demetleme", Yüksek Lisans Tezi***, İstanbul Teknik Üniversitesi Fen Bilimleri Enstitüsü***, İstanbul, 17-18 (2002).

ÖZGECMİŞ

Turgut ÖZSEVEN 1981 yılında Tokat'ta doğdu; ilk ve orta öğrenimini aynı şehirde tamamladı. Lise eğitimini Tokat Teknik Lise Bilgisayar bölümünde tamamladı. 1999 yılında Kocaeli Üniversitesi Teknik Eğitim Fakültesi Elektronik-Bilgisayar Eğitimi Bilgisayar Öğretmenliği Bölümü'nde öğrenime başlayıp 2003 yılında mezun oldu. 1997-1999 yılları arasında yerel bir televizyon kanalın reklam departmanında, 2001-2003 yılları arasında özel bir şirkette teknik servis, program desteği ve network departmanlarında, 2003-2008 yılları arasında Milli Eğitim Bakanlığı bünyesinde öğretmen olarak görev yaptı. 2008 yılı sonunda Gaziosmanpaşa Üniversitesi Turhal Meslek Yüksekokulu'nda öğretim görevlisi olarak göreve başladı ve halen aynı yerde çalışmaya devam etmektedir.

ADRES BİLGİLERİ

Adres : Gaziosmanpaşa Üniversitesi
Turhal Meslek Yüksekokulu
Turhal/TOKAT
Tel : 0(533) 224 5325
E-Posta : turgutozseven@gmail.com