MAKİNE ÖĞRENMESİ TEKNİKLERİ İLE RSS BESLEME YÖNETİMİ

Tuğrul YARDIMCI

YÜKSEK LİSANS TEZİ
BİLGİSAYAR EĞİTİMİ

GAZİ ÜNİVERSİTESİ
BİLİŞİM ENSTİTÜSÜ

ŞUBAT 2011

ANKARA

Tuğrul YARDIMCI tarafından hazırlanan “Makine Öğrenmesi Teknikleri ile RSS Besleme Yönetimi” adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

Doç. Dr. H. İbrahim BÜLBÜL

Tez Yöneticisi

Bu çalışma, jürimiz tarafından oy birliği ile Bilgisayar Eğitimi Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir.

Başkan : Prof. Dr. Ali Paşa AYDIN

Danışman Üye: Doç. Dr. H. İbrahim BÜLBÜL

Üye : Yrd. Doç. Dr. Tolga GÜYER

Tarih : 06/01/2011

Bu tez, Gazi Üniversitesi Bilişim Enstitüsü tez yazım kurallarına uygundur.

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

Tuğrul YARDIMCI

MAKİNE ÖĞRENMESİ TEKNİKLERİ İLE RSS BESLEME YÖNETİMİ
(Yüksek Lisans Tezi)

Tuğrul YARDIMCI

GAZİ ÜNİVERSİTESİ

BİLİŞİM ENSTİTÜSÜ

ŞUBAT 2011

ÖZET

Son yıllarda Internet' in hızlı gelişimi ve buna bağlı olarak da internette bilgi birikimi artmaktadır. Forumlar, Blog sayfaları, haber kaynakları, e-ticaret ve e-öğrenme gibi kavramlar yaygınlaşmaktadır. İster nitelikli ister niteliksiz olsun bütün bu birikimler bilgi kirliliğine yol açmaktadır. Bilgiye erişmek için birden fazla aynı türde sitenin ziyaret edilmesi ve bu sitelerden uygun bilgilerin bir araya getirilmesi gerekmektedir. Kullanıcı erişilen bilgileri sınıflandırma ve analiz etme ve gereksiz bilgileri ayırt etme gereksinimi duymaktadır. Bu çalışmada Makine Öğrenmesi teknikleri ile kullanıcıya bilgiyi filtrelemesinde veya analiz etmesinde kolaylık sağlayan bir sistem sunulması amaçlanmıştır. "Rss" teknolojisinin Web ortamına getirdiği yeniliklerle tek bir Web sayfasından bilgilere erişim sağlanabilmektedir.

Bu çalışmada, birden fazla kaynağa ait "Rss" adresleri, bir Script dili ve veri tabanı programı yardımı ile kullanıcının haber okuma alışkanlığının Web sitesine öğretilmesi amaçlanmıştır.

Bilim Kodu : **702.3.006**
Anahtar Kelimeler : **RSS, RSS yönetimi, makine öğrenmesi teknikleri, naive bayes algoritması**
Sayfa Adedi : **52**
Tez yöneticisi : **Doç. Dr. H. İbrahim BÜLBÜL**

RSS FEEDING MANAGEMENT BY MACHINE LEARNING TECHNIQUES
(Master Thesis)

Tugrul YARDIMCI

GAZI UNIVERSITY

INFORMATICS INSTITUTE

FEBRUARY 2011

ABSTRACT

The knowledge accumulation is being increased by the effects of the radip development of the internet in recent years. Terms are becoming common in use such as forums, blog pages, news sources, e-commerce and e- learning. Whether qualified or unqualified, all these accumulation causes the knowledge pollution. In order to reach the information; more than one site, which are similar with eachother, must be visited and appropriate information needs to be brought together. The user needs to classify and analyze the accessed data and distinguish these data from the unnecassary information. In this study, it has been purposed to present a system which provides convenience to the user while analizing or filtering the data by the machine learning tecniques. It's provided to access the information from only one web site, by the innovations which are brought by the Rss Technology to the web environment.

In this study, it's aimed to teach the users' news reading habits to the web site by the help of the Rss adressess which belong more than one source, a script language and a database programme.

Science Code : 702.3.006
Keywords : RSS, RSS management, machine learning techniques, naive bayesian algorithm.
Page Number: 52
Adviser : Assoc. Prof. Dr. H. Ibrahim BULBUL

TEŞEKKÜR

Çalışmam süresince benden sonsuz sabır ve hoşgörüsünü esirgemeyen, yardım ve katkılarıyla beni yönlendiren değerli danışman hocam Sayın Doç. Dr. H. İbrahim BÜLBÜL'e, yine tavsiyeleriyle çalışmama yön veren Sayın Prof. Dr. Şeref SAĞIROĞLU 'na ve Doç. Dr. Tolga GÜYER' e, desteklerinden dolayı eşime ve aileme teşekkür ederim.

İÇİNDEKİLER

Sayfa

ÖZET iii

ABSTRACT v

TEŞEKKÜR vii

İÇİNDEKİLER viii

ŞEKİLLERİN LİSTESİ x

RESİMLERİN LİSTESİ xi

ÇİZELGELERİN LİSTESİ xii

SİMGELER VE KISALTMALAR xiii

1. GİRİŞ 1

2. MEVCUT LİTERATÜRÜN İNCELENMESİ 4

3. MATERYAL VE METOD 6

4. MAKİNE ÖĞRENMESİ 7

4.1. Veri Madenciliği 7

4.1.1. Sorunun tanımlanması 9

4.1.2. Verilerin hazırlanması 9

4.1.3. Toplama ve uyumlaştırma 9

4.1.4. Birleştirme ve temizleme 10

4.1.5. Seçim 10

4.1.6. Modelin kurulması ve değerlendirilmesi 11

4.1.7. Modelin kullanılması 11

4.1.8. Modelin izlenmesi 11

4.2. Veri Madenciliğinde Kullanılan Yöntemler 12

4.2.1. İstatistiksel yöntemler 12

Sayfa

4.2.2.Bellek tabanlı yöntemler 12
4.2.3. Yapay sinir ağları 13
4.2.4. Karar ağaçları 13
4.2.5. Genetik algoritmalar 14
4.3. Rss 14
4.4. PHP Dili 15
4.5. MySQL Veri Tabanı 17
5. RSS BESLEME YÖNETİM SİSTEMİ 20
5.1.Öğrenmenin Gerçekleşmesi 21
6. SONUÇ 28
EKLER 30
ÖZGEÇMİŞ 52

ŞEKİLLERİN LİSTESİ

Sayfa

Şekil 4.1. Veri madenciliği süreci..... 8

Şekil 4.2. Bir yapay sinir ağı 13

Şekil 4.3. Veri madenciliğin birçok alanla bileşimi 14

Şekil 4.4. RSS kaynağına ait simgeler 14

Şekil 4.5. MYSQL veritabanı yönetim sistemi 18

Şekil 5.1. RSS belseme yönetimi sistemi yapısı.....20

RESİMLERİN LİSTESİ

Sayfa

Resim 5.1. Veri tabanında bulunan “Kelime” tablosunun yapısı 23

Resim 5.2. Veri tabanında bulunan “Rss_Kaynak” tablosunun yapısı.....23

Resim 5.3. Örüntü Gör sayfasında verilerin görünümü.....27

Resim 5.4. Öğrenme işlemi gerçekleştikten sonra öğrenilmiş veri sayfası.....27

ÇİZELGELERİN LİSTESİ

Sayfa

Çizelge 5.1. Kümeleme sistemi ile önem dereceleri listesi.....25

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış bazı simgeler ve kısaltmalar, açıklamaları ile birlikte - aşağıda sunulmuştur.

Kısaltmalar	**Açıklamalar**
XML	Extensible Markup Language (Genişletilebilir işaretleme dili)
PHP	Personal, Home Page,Hypertext Preprocessor (Türkçe: Üstün yazı Önişlemcisi),
WAN	Dünya Gazeteler Biriliği
RSS	Rich Site Summary (Zengin Site Özeti)

1. GİRİŞ

Bilişim teknolojilerinin baş döndürücü bir hızla ilerlediği yüzyılımızda internet ortamında bilgi paylaşımı da buna paralel bir hızda ilerlemektedir. Bilgi çağı olarak nitelendirilen günümüz, artık bilginin hüküm sürdüğü bir dönemde yaşadığımızı bize açıkça ifadece etmektedir.

Internet bize bu çağa ayak uydurmamızı sağlayan temel yapıdır. Artık Cep Telefonları, Tablet PC' ler ve hatta uydu alıcılar da internete erişim sağlayabiliyorlar. Bu teknolojiler bilgiye erişimde bizlere özgürlük sunmaktadırlar. Fakat bilgiye erişimde özgürlüğü sağladığı gibi bilgi paylaşımına da özgürlük sağlanmaktadır. Günümüzde hangi konu ile ilgili olursa olsun insanlar araştırma yapmak amacıyla bilgisayarın karşısına geçtiklerinde etkin bilgiye ulaşmak için saatlerini harcayabiliyorlar, aynı bilgi ile defalarca karşılaşabiliyorlar veya istedikleri bilgiye erişebilmek için gereksiz bilgi yığınlarını aşmaları gerekebiliyor. Arama motorları (Google, Altavista, Lycos, Yahoo... vb) bize bu yığınlar arasından kurtulmamıza ve etkin bir şekilde doğruluk oranı yüksek bir şekil bilgiye erişmemizde bize fazlaca yardımcı olmaktadırlar. Fakat yinede dolandırıcılık, Fishing, , reklam veya ticaret amaçlıda olsa günümüzde internet ortamı gereksiz bilgiler çöplüğü ile kirlenmektedir [1].

Internet ortamında bilgi kaynağı olarak gazete web siteleri gerek güncellenme sıklığı gerekse bilgi potansiyeli bakımından önde gelen kaynaklardandır. Dünya Gazeteler Birliği (WAN) tarafından yapılan araştırmaya göre gazete tirajları düşmektedir[2]. Bunun başlıca nedeni ise internet üzerinden gazete okunma oranlarının yükselmesi gösteriliyor. Mobil teknolojilerin de gelişmesi ile mekân gözetmeksizin insanlar gazetelerini elektronik ortamda okuyorlar. Buna bağlı olarak da gazeteler reklam piyasasını internete kaptırmış durumdalar.

Haber platformlarının elektronik ortama kaydığı bu dönemde haber kaynaklarının analiz edilmesi, sınıflandırılması büyük önem arz etmektedir.

Eskiden gazete okurken haber alışkanlıkları gazetenin hangi sayfasından başlanıldığı ile ifade ediliyordu. Siyaset, Spor, Ekonomi, Magazin ve Güncel gibi kategoriler bunların bazıları. Artık kategorilere ayrılmış bir şekilde haber siteleri bilgiyi kullanıcılara sunmaktadırlar ve kullanıcılarda hangi sayfadan başlayacağı değil hangi kategoriyi seçecekleri kalmaktadır.

Fakat elektronik ortamda kaynakların fazla olması da en az bilginin kirliliği kadar sorun teşkil etmektedir. İstenilen bilgiye erişmede isabet oranı buna bağlı olarak düşmektedir. İşte sorunun tespiti burada yapılmaktadır. "İnsanlar istedikleri bilgi dışındaki verilerden nasıl kurtulabilirler?" İşte tam burada devreye RSS girmektedir. RSS Web teknolojisinin bize getirdiği yeniliklerden biridir. İstemci ile kaynak arasında güvenli köprü görevi görmektedir. Bilgi kaynağı RSS yayınlarını kullanarak istediği formatta yayın yapabiliyor ve istemcilerden bu yayınlara RSS kaynak adresinden güvenli bir şekilde erişip kullanabiliyorlar. RSS yayını yapan herhangi bir Web sitesinin kaynağını kullanarak tek yönlü veri akışı sağlanabilmektedir.

RSS bize temiz ve güveli bir kanal hizmeti sunsa da sorunu tamamen ortadan kaldıramamaktadır. Bir süre sonra kaynak adres miktarı arttıkça istenilen bilgiye net bir şekilde ulaşma oranı düşmektedir. Verinin yönetiminin önemi burada ön plana çıkmaktadır.

Gazete okumak isteyen bir insan sadece kendi haber okuma alışkanlığı ile uyuşan haberleri görebilmek için birden fazla haberi veya haber kaynağını taramak zorunda kalabiliyor. Bu probleme çözüm olabilmesi için RSS kaynak verilerinin yönetimi kaçınılmaz hale gelmektedir. Bu çalışmada bir yöntem geliştirilerek sistemin haber okuyan kişinin okuma alışkanlığını öğrenmesi ve tekrar haber okunmak istendiğinde örenilen alışkanlığa göre haberlerin sunulması sağlanmıştır.

Bu çalışmanın amacı, internet üzerinden haber okuyan kişilerin okuduğu haber başlıkları analiz edilerek haber alışkanlıklarının sisteme öğretilmesi ve bu öğrenilmiş bilgi ile farklı kaynaklardan gelen haberlerin bu öğrenilmiş bilgiye göre kullanıcıya sunulması ve sistemin kullanıcıya sağladığı yararının tespit edilmesidir.

İkinci bölümde, bu konuda daha önce yapılmış olan çalışmalardan bahsedilmektedir.

Üçüncü bölümde ise çalışmanın yöntemi ve bulunan materyallerden bahsedilmektedir.

Dördüncü bölümde, bu çalışma için geliştirilen sistemde kullanılan yazılım, geliştirilen sistemin yapısı ve kullanımı, araçlar ve araştırma modeli üzerinde durulmaktadır.

Son bölümde ise alınan sonuçların yorumlanması ve elde edilen sonuçların yorumlanması ve önerilerden bahsedilmektedir.

2. MEVCUT LİTERATÜRÜN İNCELENMESİ

Bu bölümde yapılan çalışma ile ilgili literatür Gazi Üniversitesi Kütüphanesi, internet kaynakları ve YÖK Dokümantasyon Merkezi dahil olmak üzere taranmış olup önce yapılan çalışmalar incelenmiş ve bu çalışmaların ortaya koyduğu tespit ve çözümler özetlenmiştir.

Yavanoğlu (2009), tarafından yapılan bir çalışmada Web sayfaları için karar destek yazılımı geliştirmiştir. Geliştirilen sistemde Web tabanlı otomatik dil tanıma ve çevirme işlevini yerine getirmektedir. Kelimelerin sınıflandırılarak hangi dilde yazıldığı tespit edilmiştir. Bu tespit ile dokümanların hangi dilde yazıldığının tespitini kolaylaştırmıştır [3].

Konu ile ilgili yapılan diğer bir çalışmada, Sağıroğlu ve arkadaşları (2008), Yapay Sinir Ağları ile Web İçeriklerini Sınıflandırma uygulaması (WESAKA) gerçekleştirmişlerdir. Sistem Naive Bayes metodunu kullanarak haber sitelerinden alınan verilerle sistemi eğittikten sonra alınan bir haber cümlesinin hangi haber kategorisine ait olduğu tahmin edilmektedir. Bu çalışma kelimelerin sınıflandırılmasına bir çözüm getirmiştir [4].

Altan ve Orhan (2004), yaptıkları çalışmada sözcük anlamlarını açıklaştırmak amacıyla öğrenme algoritmalarını uygulamışlardır. İstatistiksel ve Örnek tabanlı yöntemlerin kelime anlam açıklaştırma için daha uygun olduğunu tespit etmişlerdir [5].

Pilavcılar (2007), çalışmasında Naive Bayes metodunu kullanarak kelime kategorilendirilmesinin daha hızlı olacağı ve net sonuçlar vereceği, sınıflandırmada kolaylık sağlayacağı tespit etmiştir [6].

Ünsal (2010), çalışmasında makine öğrenmesi algoritması ile meslek alanlarının belirlenmesi için Naive Bayes metodunu kullanarak Naive Bayes ile öğrencilerin meslek seçimi işlemin daha iyi sonuç alabilmesini sağlamıştır [7].

Yukarıda verilen çalışmalarda genellikle sınıflandırma işlemleri gerçekleştirilmiştir. Çalışmamızda sınıflandırma tekniğinde Naive Bayes metodundan esinlenilse de daha önceki çalışmalardan farklıdır.

3. MATERYAL VE METOD

Bu bölümde çalışmanın yapılabilmesi için kullanılan materyal ve metotlardan bahsedilecektir. Verilerin işlenebilmesi için gereken kaynakların nereden alınacağı, hangi programlama dili ile işleneceği ve nerede depolanacağı belirtilmiştir. Kullanılan materyallerin neden seçildiklerinden bahsedilmiştir.

Bu çalışmada literatür' e dayalı tarama modeli kullanılmıştır. Bu çalışmanın teorik kısmında Tarama modeli kullanılmıştır. Elde edilen verilerin değerlendirilmesinde ise Nicel gözlem kullanılmıştır. Tarama modelleri genel ve örnek olay olmak üzere 2 çeşittir. Tarama modellerinin tek başına uygulandığı araştırma yaklaşımları olmakla birlikte, taramanın yer almadığı bir başka araştırma modelinin tek başına var olması düşünülemez. Tarama araştırmacısı, nesnenin ya da bireyin doğrudan kendisini inceleyebileceği gibi, önceden tutulmuş çeşitli kayıtlara (yazılı belge ve istatistik kayıtları vb.) eski kalıntılar ve alandaki kaynak kişilere başvurarak, elde edeceği dağınık verileri, kendi gözlemleri ile bir sistem içinde bütünleştirerek yorumlamak durumundadır [8]. Çalışmada materyal olarak PHP Script dili, veri depolamak amacıyla MySQL veri tabanı programı, öğrenme verisi elde etmek için RSS kaynaklar kullanılmıştır.

Çalışmada;

- **a.** RSS besleme yönetimi yazılımı yerel olarak sisteme kurulmuştur.
- **b.** Çeşitli RSS kaynak adresleri girilmiştir
- **c.** Girilen kaynaklardan gelen haberlere tıklanarak eğitim işlemi yapılmıştır
- **d.** Öğrenilmiş veri ile alınan haber sayısı ve niteliği, öğrenme olmadan alınan haber sayısı ve niteliği ile karşılaştırılmıştır

Bu çalışmada, literatür taramasında incelenen çalışmalardan farklı olarak sınıflandırma için Naive Bayes algoritmasından farklı, sadece kelimelerin tekrarlanma sayıları kullanılarak sınıflandırma yapan yeni bir metot geliştirilmiştir.

4. MAKİNE ÖĞRENMESİ

Makine öğrenmesi, bilgisayarların algılayıcı verisi ya da veritabanları gibi veri türlerine dayalı öğrenimini olanaklı kılan algoritmaların tasarım ve geliştirme süreçlerini konu edinen bir bilim dalıdır. Makine öğrenimi araştırmalarının odaklandığı konu bilgisayarlara karmaşık örüntüleri algılama ve veriye dayalı akılcı kararlar verebilme becerisi kazandırmaktır. Bu, makine öğreniminin istatistik, olasılık kuramı, veri madenciliği, örüntü tanıma, yapay zeka, uyarlamalı denetim ve kuramsal bilgisayar bilimi gibi alanlarla yakından ilintili olduğunu göstermektedir [9].

Bir başka deyişle Makine Öğrenmesi programın veri akışından yaralanarak yeni örüntüler oluşturur ve kendini geliştirir. Sisteme girdi ve çıktı işlemleri öğrenmeye katkıda bulunan etmenlerdir.

Makine öğrenmesinin günlük hayatımızdaki kullanım alanları aşağıda verilmiştir;

- Parmak izi tanıma sistemleri
- Göz taraması (İris) ile tanıma sistemleri
- Yüz tanıma sistemleri
- El yazısı veya imza tanıma sistemleri
- Tıbbi verileri tanımlamada kullanılan sistemler
- Metin ve Mail analizinde kullanılan sistemler

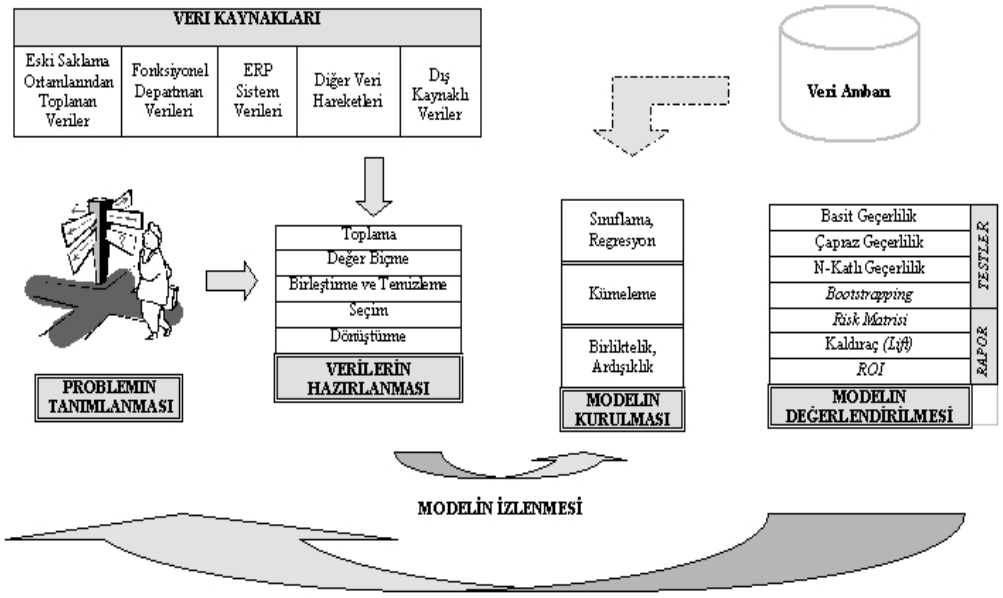
4.1. Veri Madenciliği

Veriyi genel olarak, enformasyonel veri ve operasyonel veri olarak ikiye ayırmak mümkündür. Enformasyonel veri, kişiye yönelik, bütünleşik, zaman içinde oluşan ve birleştirilmiş veriler olarak tanımlanabilir. Operasyonel veri ise, uygulamaya yönelik, dağınık, kısa zamanda oluşan ve tekrarlayabilen veriler olarak tanımlanmaktadır .

Erişilmek istenen veriler doğru anda doğru sıra ile bulunması veri madenciliğinin kullanım amaçlarından biridir. Veri madenciliği bir veri tabanında saklanabilecek bütün veriler için kullanılabilir. Veri madenciliğinin kullanım alanları;

- Bankacılık: Risk analizleri ve usulsüzlük tespiti.
- Pazarlama: Çapraz satış analizleri, müşteri segmentasyonu.
- Sigortacılık: Müşteri kaybı sebeplerinin belirlenmesi, usulsüzlüklerin önlenmesi.
- Telekomünikasyon: Hile tespiti, hatların yoğunluk tahminleri.
- Borsa: Hisse senedi fiyat tahmini, genel piyasa analizleri.
- Tıp: Tıbbi teşhis, uygun tedavi sürecinin belirlenmesi.
- Bilim ve Mühendislik: Ampirik veriler üzerinde modeller kurularak bilimsel ve teknik problemlerin çözümlenmesi.
- Endüstri: Kalite kontrol, lojistik.

Veri Madenciliği bir yöntem değildir bir süreçtir. Bu süreçte ana unsur süreci gerçekleştiren uygulamacıdır. Süreçte bulunan adımlar doğru olarak yerine getirilmediği sürece istenilen sonuca ulaşılması mümkün değildir. Veri madenciliği süreçleri aşağıdaki şekilde görülmektedir;



Şekil 4.1. Veri madenciliği süreci [10].

4.1.1. Sorunun tanımlanması

Veri madenciliği çalışmalarında başarılı olmanın ilk şartı, uygulamanın hangi kuruluş amacı için yapılacağının açık bir şekilde tanımlanmasıdır. İlgili kuruluş amacı, sorun üzerine odaklanmış ve açık bir dille ifade edilmiş olmalı, elde edilecek sonuçların başarı düzeylerinin nasıl ölçüleceği tanımlanmalıdır. Sorun ile tam örtüşmeyen bir veri madenciliği çalışması, sorunu çözmeye yetmeyeceği gibi sonuçta başka problemlerin de ortaya çıkmasına neden olabilecektir. Ayrıca yanlış kararlarda katlanılacak olan maliyetlere ve doğru kararlarda kazanılacak faydalara ilişkin öngörülere de bu aşamada yer verilmelidir.

4.1.2. Verilerin 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 karar vericinin veri keşfi sürecinin toplamı içerisindeki enerji ve zamanının % 50 - % 85' ini harcamasına neden olmaktadır.
Verilerin hazırlanması aşaması kendi içerisinde toplama ve uyumlaştırma, birleştirme ve temizleme ve seçme adımlarından meydana gelmektedir.

4.1.3. Toplama ve uyumlaştırma

Tanımlanan sorun için gerekli olduğu düşünülen verilerin ve bu verilerin toplanacağı veri kaynaklarının belirlenmesi adımıdır. Hangi veri kaynaklarından yararlanılacağı önemli bir karardır. Çünkü gereğinden az veri kaynağı veri madenciliği çalışmasını eksik bırakacağı gibi, gereğinden fazla veri kaynağı sürecin uzamasına neden olabilecek veri kirliliğine yol açabilecektir. Verilerin toplanmasında kuruluşun kendi veri kaynaklarının dışında, nüfus sayımı, hava durumu, merkez bankası kara listesi gibi çeşitli veri tabanlarından veya veri pazarlayan kuruluşların veri tabanlarından faydalanılabilir.

Veri madenciliğinde kullanılacak verilerin farklı kaynaklardan toplanması, doğal olarak veri uyumsuzluklarına neden olacaktır. Bu uyumsuzlukların başlıcaları farklı

zamanlara ait olmaları, güncelleme hataları, veri formatlarının farklı olması, kodlama farklılıkları (örneğin bir veri tabanında cinsiyet özelliğinin e/k, diğer bir veri tabanında 0/1 olarak kodlanması), farklı ölçü birimleri ve varsayım farklılıklarıdır. Ayrıca verilerin nasıl, nerede ve hangi koşullar altında toplandığı da önem taşımaktadır. Güvenilir olmayan veri kaynaklarının kullanımı tüm veri madenciliği sürecinin de güvenilirliğini etkileyecektir.

Bu nedenlerle, iyi sonuç alınacak veri madenciliği çalışmaları ancak iyi verilerin üzerine kurulabileceği için, toplanan verilerin ne ölçüde uyumlu oldukları bu adımda incelenerek değerlendirilmelidir.

4.1.4. Birleştirme ve temizleme

Bu adımda farklı kaynaklardan toplanan verilerde bulunan ve bir önceki adımda belirlenen sorun ve uyumsuzluklar 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.

4.1.5. Seçim

Bu adımda kurulacak modele bağlı olarak veri seçimi yapılır. Örneğin tahmin edici bir model için bu adım, bağımlı ve bağımsız değişkenlerin ve modelde kullanılacak veri kümesinin seçilmesi anlamını taşımaktadır.

Sıra numarası, kimlik numarası gibi anlamlı olmayan değişkenlerin modele girmemesi gerekmektedir. Çünkü bu tip değişkenler, diğer değişkenlerin modeldeki ağırlığının azalmasına ve veriye ulaşma zamanlarının uzamasına neden olabilmektedir. Bazı veri madenciliği 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 araçlar ve bunların sunduğu ilişkiler, bağımsız değişkenlerin seçilmesinde önemli yararlar sağlayabilir. Genellikle

yanlış veri girişinden veya bir kereye özgü bir olayın gerçekleşmesinden kaynaklanan verilerin, veri kümesinden atılması tercih edilir.

Veri madenciliği çalışmasında geliştirilen modelde kullanılan veri tabanının çok büyük olması durumunda, rastgeleliği bozmayacak şekilde örnekleme yapılması uygun olabilir. Ayrıca burada seçilen örneklem kümesinin tüm popülasyonu temsil edip etmediği de kontrol edilmelidir. Halen kullanılan işletim sistemleri ve paket programlar ne kadar gelişmiş olursa olsun, çok büyük veri tabanları üzerinde çok sayıda modelin denenmesi zaman kısıtı nedeni ile mümkün olamamaktadır. Bu nedenle tüm veri tabanını kullanarak bir kaç model denemek yerine, rastgele örneklenmiş bir veri tabanı parçası üzerinde bir çok modelin denenmesi ve bunlar arasından en güvenilir ve güçlü modelin seçilmesi daha uygun olacaktır. Diğer bir deyişle modellerin performansları uygun bir karar yöntemi ile sınanmalıdır.

4.1.6. Modelin kurulması ve değerlendirilmesi

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.

4.1.7. Modelin kullanılması

Oluşturulan model tanımlanan sorunun çözümüne uygun bir şekilde uygulamanın içerisine gömülür.

4.1.8. Modelin izlenmesi

Zaman içerisinde bütün sistemlerin özelliklerinde ve dolayısıyla ürettikleri verilerde ortaya çıkan değişiklikler, kurulan modellerin sürekli olarak izlenmesini ve gerekiyorsa yeniden düzenlenmesini gerektirecektir. Tahmin edilen ve gözlenen değişkenler arasındaki farklılığı gösteren grafikler model sonuçlarının izlenmesinde kullanılan yararlı bir yöntemdir.

Bir veri madenciliği sistemi, aşağıdaki temel bileşenlere sahiptir:

1. Veritabanı, veri ambarı ve diğer depolama teknikleri
2. Veritabanı ya da Veri Ambarı Sunucusu
3. Bilgi Tabanı
4. Veri Madenciliği Motoru
5. Örüntü Değerlendirme
6. Kullanıcı Ara yüzü [11].

4.2. Veri Madenciliğinde Kullanılan Yöntemler

4.2.1. İstatistiksel yöntemler

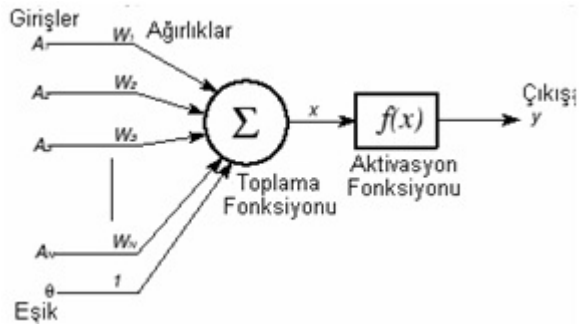
Veri madenciliği çalışması esas olarak bir istatistik uygulamasıdır. Verilen bir örnek kümesine bir kestirici oturtmayı amaçlar. İstatistik literatüründe son elli yılda bu amaç için değişik teknikler önerilmiştir. Bu teknikler istatistik literatüründe çok boyutlu analiz (multivariate analysis) başlığı altında toplanır ve genelde verinin parametrik bir modelden (çoğunlukla çokboyutlu bir Gauss dağılımından) geldiğini varsayar. Bu varsayım altında sınıflandırma (classification; discriminant analysis), regresyon, öbekleme (clustering), boyut azaltma (dimensionality reduction), hipotez testi, varyans analizi, bağıntı (association; dependency) kurma için teknikler istatistikte uzun yıllardır kullanılmaktadır.

4.2.2. Bellek tabanlı yöntemler

Bellek tabanlı veya örnek tabanlı bu yöntemler (memory-based, instance-based methods; case-based reasoning) istatistikte 1950'li yıllarda önerilmiş olmasına rağmen o yıllarda gerektirdiği hesaplama ve bellek yüzünden kullanılamamış ama günümüzde bilgisayarların ucuzlaması ve kapasitelerinin artmasıyla, özellikle de çok işlemcili sistemlerin yaygınlaşmasıyla, kullanılabilir olmuştur. Bu yönteme en iyi örnek en yakın k komşu algoritmasıdır.

4.2.3. Yapay sinir ağları

1980'lerden sonra yaygınlaşan yapay sinir ağlarında (artificial neural networks) amaç fonksiyon birbirine bağlı basit işlemci ünitelerinden oluşan bir ağ üzerine dağıtılmıştır (Bishop, 1996). Yapay sinir ağlarında kullanılan öğrenme algoritmaları veriden üniteler arasındaki bağlantı ağırlıklarını hesaplar. YSA istatistiksel yöntemler gibi veri hakkında parametrik bir model varsaymaz yani uygulama alanı daha geniştir ve bellek tabanlı yöntemler kadar yüksek işlem ve bellek gerektirmez.



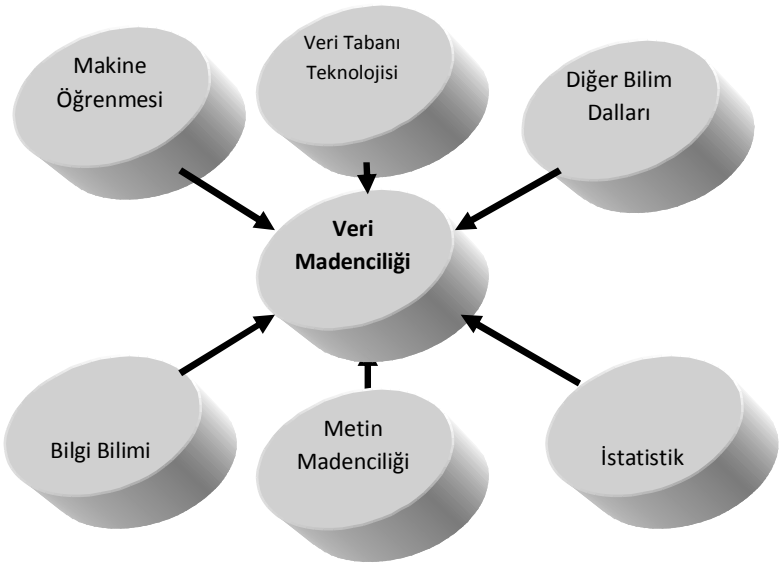
Şekil 4.2. Bir yapay sinir ağı [12].

4.2.4. Karar ağaçları

İstatistiksel yöntemlerde veya yapay sinir ağlarında veriden bir fonksiyon öğrenildikten sonra bu fonksiyonun insanlar tarafından anlaşılabilecek bir kural olarak yorumlanması zordur. Karar ağaçları ise veriden oluşturulduktan sonra yukarıdaki örnekte de olduğu gibi ağaç kökten yaprağa doğru inilerek kurallar (IF-THEN rules) yazılabilir . Bu şekilde kural çıkarma (rule extraction), veri madenciliği çalışmasının sonucunun geçerlenmesini sağlar. Bu kurallar uygulama konusunda uzman bir kişiye gösterilerek sonucun anlamlı olup olmadığı denetlenebilir. Sonradan başka bir teknik kullanılacak bile olsa karar ağacı ile önce bir kısa çalışma yapmak, önemli değişkenler ve yaklaşık kurallar konusunda bize bilgi verir ve tavsiye edilir.

4.2.5. Genetik algoritmalar

Diğer veri madenciliği algoritmalarını geliştirmek için kullanılan optimizasyon teknikleridir. Sonuç model veriye uygulanarak gizli kalmış kalıpları ortaya çıkarılmakta ve bu sayede tahminler yapılabilmektedir. Doğrudan postalama, risk analizi ve perakende analizlerinde kullanılabilir.



Şekil 4.3. Veri madenciliğin birçok alanla bileşimi

4.3. Rss

RSS, genellikle haber sağlayıcıları, bloglar ve podcastler tarafından kullanılan, yeni eklenen içeriğin kolaylıkla takip edilmesini sağlayan özel bir XML dosya formatıdır. İnternet kullanıcısı RSS teknolojisi ile düzenli olarak içerik sunan sitelere abone olabilir ve çeşitli RSS istemcileri sayesinde içeriği takip edebilir. RSS olarak sunulan içerik web sitesinde sunulan içeriğin tamamını, özetini veya sadece başlığını içerebilir [13].

RSS kaynağı sağlayan internet sitelerinde genellikle şu simgeler bulunur:

Şekil 4.4. RSS kaynağına ait simgeler

Bu tür bir haber alma yoluna gidenler için masraflar yok denecek kadar az. İnternet sayfası için hazırlanan haberler, bir uygulama tarafından tam otomatik olarak açılıp, RSS için uygun olan XML formatına dönüştürülüyor. Bu XML verisi okuyucu için gerekli bağlantıları, başlıkları ve markalamaları da içeriyor. Her yeni haber gönderimiyle beraber XML verisi de yeniden yazılıyor ve eski kayıtlar siliniyor. Bu şekilde, çoğu haber yenilemesinden sonra bile haber gönderimini hızlandırmak için veri boyutu düşük tutulmuş oluyor. Bu şekil bir haber akışına "üye" olan kişinin yapması gereken şey, XML verisinin bulunduğu sayfayı okuyucu yazılıma girmek. Ayrıca bilgileri alırken virüs vb… zararlı yazılımlardan da korunmuş olunuyor. RSS yayınları okuyabilmek için web tarayıcısının RSS okuyucusunu kullanılabileceği gibi Web tabanlı RSS okuyucular veya bilgisayara kurulabilen RSS okuyucularda mevcuttur. RSSOwl, RSS Reader, FeeD Reader bu programlara örnek verilebilir.

4.4. PHP Dili

Açılımı Personal Home Page, olan PHP, ilk kez Rasmus Lerdorf tarafından, web sayfalarını ziyaret edenleri izlemek amacıyla bir dizi Perl Script (betik) kullanılarak geliştirilmişti. İnsanlar kısa zamanda bununla ilgilenmeye ve bu konuyla ilgili sorular sormaya başladıklarında, Rasmus kararını verdi ve bir script motoru oluşturdu. Ayrıca formlara da destek verdi ve böylece PHP/F1'i biçimlendirmiş oldu. Adını duyurdukça bir gurup yazılım geliştirmecinin dikkatini çekti ve böylece bir API oluşturuldu ve PHP3 ortaya çıktı. Daha sonraları yeniden ele alınması gerekti ve Zend motoru PHP4'ü yaratmış oldu. Artık PHP önünde pek engel bulunmuyordu, PHP Hypertext Processor fetihlere çıkmaya hazırdı. PHP gibi bir script motorunun verimliliğini en yüksek düzeye çıkartan 4 temel etmen bulunuyor. Bunlar; Hız, İstikrar, Güvenlik ve Basitlik olarak sayılabilir. Uygulama hızı da önemlidir tabi ki, ancak bununla birlikte bilgisayarın diğer fonksiyonları yavaşlamamalı. Bu nedenle bir sürü sistem kaynağına gerek duymamalı. PHP, özellikle Unix tabanında çalışıyorsa, diğer yazılımlarla iyi uyum sağlamaktadır, az yer kaplar ve bir Apache modülü olarak çalıştırıldığında hemen kullanıma geçer. Bir kaç bin sayfalık bir işte, sistem çöküyorsa eğer hızın pek bir anlamı kalmayacaktır. Her uygulamanın hata

sorunu vardır. Ancak bir gurup yazılım geliştiricilerinden oluşmuş bir topluluğa sahip bir uygulama söz konusuysa, işler biraz değişir ve böcek (bug) olarak tabir edilen hatalar saklanacak pek bir delik bulamaz. Bunun yanı sıra PHP kendi işletim sistemi kaynaklarını kullanıyor ve veri transferi ve denetiminde çok başarılı ve karmaşık bir metot getiriyor. Sistemin bazı saldırgan tavırlı kullanıcılara karşı korunması zaruridir. PHP istenilen düzeyde .ini dosyaları olarak kurulabilen farklı güvenlik düzeylerine sahiptir. Programcıların uygulama üzerinde hızlı bir biçimde üretime geçmeleri gerekmektedir. PHP üzerinde, HTML kodlamacıları hiç zorlanmadan web sayfalarını yazmaya başlayabilir. C dilinde deneyim sahibi olan programcılar, hatta javascript kullananlar kısa bir sürede hızlanabilirler. Ayrıca bağlanabilirlik de PHP'nin artılarından biridir. Modül uzantılar sistemi çeşitli kütüphanelerle (veritabanları) kolayca arabirim oluşturabiliyor. Yeni uzantılar eklemek çok daha kolay.

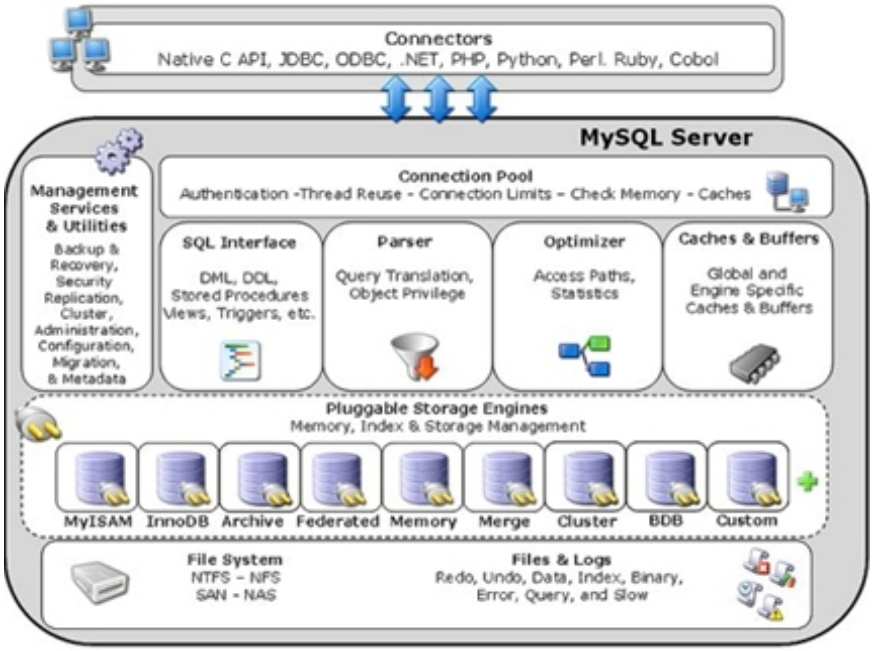
PHP ' nin diğer avantajları PHP hemen hemen her platformda çalışabiliyor olmasıdır. PHP aynı kod temelini kullandığı için, UNIX, Windows (95/98/NT/2000/XP/Server) ve Mac OS dahil olmak üzere 25 platformda derlenip kurulabilir. Kodlar aynı olduğundan script' ler platformdan bağımsız olarak çalışacaktır. PHP, uzantı alabilmektedir. Uygulamanın içerisinde yer alan çekirdek motor , bir dizi asal kod modüllerinden ve kod uzantılarından oluşmaktadır. Bu nedenle programcılara PHP uzantıları yaratarak bazı özel işlemlerini yapabilmeleri için iki seçenek sunuluyor; ya uzantı modüllerini yazarak uygulanabilen bir derleme yapmak, ya da PHP ' nin dinamik yükleme mekanizmasıyla yüklenebilecek uygulanabilir uzatmalar yaratmak.

PHP pek çok HTTP server arayüzü barındırıyor. PHP Apache'ye, AOL server'a, Roxen ve THTTPD'ye doğrudan yüklenebiliyor. Alternatif olarak CGI modülü olarak da kullanılabilir. PHP pek çok veritabanı ara yüzü bulunduruyor. PHP, MySQL, MS SQL, Oracle, Informix, PostgreSQL ve diğerleriyle doğrudan çalışabiliyor. Bunlar ikili sayı düzenindeki ara yüzlerden oluşmaktadır ve bu çözümler için veritabanının desteklenmediği yerlerde ODBC desteği sağlıyor. Bir PHP kullanıcısı herhangi bir kütüphane için ara yüz oluşturmakta zorluk çekmez. Pek çok kullanıcı bu yolu seçmiş, grafik rutinleri, PDF dosyaları, Flash Movie'leri, Cybercash cetvelleri, XML, IMAP, POP ve diğerleriyle ilgili modüller bulabilmiştir.

PEAR, PHP'nin uzantısı ve Add-on deposudur. Pear, Perl için geliştirilen CPAN'e benzemektedir. Halen başlangıç aşamasında olmasına rağmen PEAR, PHP'nin kurulumuyla birlikte gelecek bir dizi PHP script'ini kullanıma sunmaktadır. # PHP bir açık kod uygulamasıdır ve pek çok profesyonel kullanıcı için çok şey ifade etmektedir. Basitçe açıklamaya çalışırsak PHP kullanıcıyı, çalışmayan uygulamalar için üretici firmanın keyfini beklemekten, her yıl sistemini belli paralar ödeyerek güncelleme zorunluluğundan kurtarmaktadır. Eksik yönleri neler Hata denetimi Cold Fusion ya da ASP uygulamasındaki kadar etkili değil. ASP ile IIS muhtemelen PHP ile IIS'den daha iyi. Ancak tamamen teknik bir altyapıda PHP, WindowsNT üzerinde, diğer platformlardaki performansına ulaşmakta [14].

4.5. MySQL Veri Tabanı

Makine öğrenmesi ile RSS besleme yönetimi için veri tabanı dikkatle hazırlanmalıdır. . Veri tabanı sistemi tüm verileri sağlıklı bir şekilde sakladığı gibi en az seviyede sistem kaynağı kullanılarak en fazla bilgiyi derleyebilecek bir şekilde tasarlanmalıdır. Şekil 3.2’ deki MySQL veritabanı yönetim sistemi altı milyondan fazla sistemde yüklü bulunan çoklu iş parçacıklı (multi-threaded), çok kullanıcılı (multi-user), hızlı ve sağlam bir veritabanı yönetim sistemi olması, RSS besleme yönetimi sisteminde MySQL veritabanı seçilmesi sistem kaynağının daha iyi kullanılmasına olanak sağlamaktadır [15]. MySQL veritabanı ücretsiz bir veritabanı yönetim sistemidir ve web hosting hizmeti olarak ücretsiz sunulmaktadır.



Şekil 4.5. MYSQL veritabanı yönetim sistemi [15].

MySQL veritabanı çoklu iş parçacıklı özelliği olan bir program içinde, programın çeşitli bölümleri ayni anda paralel olarak çalışabilir. Çok kullanımlı (Multithreading) özelliği programın hızını ve performansını arttırır. Çoklu iş parçacıklı özelliği, MySQL veritabanına aynı anda birden fazla kullanıcının bağlanıp, sorgulama (query) yapması imkânını verir. MySQL veri yönetim sistemi 113 milyon kayıt (7.5 GB veri + 5.2 GB index) içerebilir [15]. MySQL, yüksek performanslı sorgu motoru sayesinde yüksek trafik web siteleri için, çok hızlı veri eklemek yeteneği standart, hızlı ve tam metin arama gibi özel Web işlevler için güçlü desteğine sahiptir.

MySQL Cluster, arızalı veya kritik konumda olan veritabanı kümelerini dağıtmak ve mimari olarak kümeleri oluşturmak için MySQL tarafından üretilmiştir. Şekil 3.2'de gösterildiği gibi MySQL veritabanı iki farklı tür tablo yapısını desteklemektedir. InnoDB ve Berkeley DB (BDB) işlem (Transaction) tablolarıdır. *MyISAM*, *HEAP*, *MERGE*, *ISAM* ise atomik işlem tablolarıdır. MyISAM çok hızlı olmasına rağmen Windows tabanlı sunucuda çalışıyorsa çok güvenli değildir. Bozulması kolay olmasına rağmen tamir etmesi kolaydır. Fakat tamir edildiğinde kayıt kayıpları oluşabilmektedir. Yine MySQL ile Kayıt Birleştirme (*MERGE*) işlemi

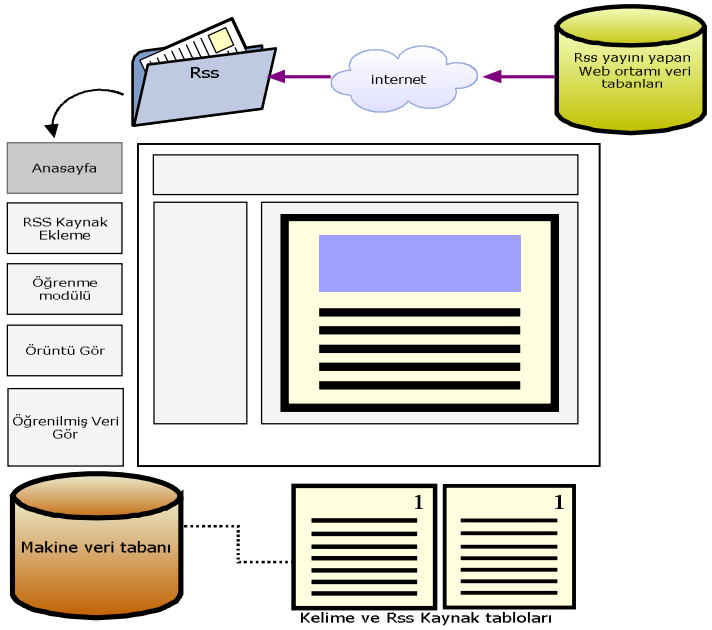
yapılabilmektedir. *MERGE* tablo türü, aynı yapıya sahip tabloların verilerini beraber tutan bir tablo türüdür. *Merge* tablo türü, aynı yapıya sahip tabloların verilerini beraber tutan bir tablo türüdür. Genellikle birbiriyle yakından alakalı veriler tutan tabloların hepsine birden erişmek için kullanılır. *MEMORY* Tablo türlerinin gerçek anlamda en hızlısı memory türüdür. Fakat bu türündeki tablolar RAM ' de tutulduğu için herhangi bir güç kesintisinde bu tablolardaki bütün veriler kaybolacaktır. MySQL tarafından geçici tablolar için kullanılan bu tablo türü belirli bir boyutu aşarsa otomatik olarak MyISAM türüne çevriliyor. Eğer sınırları düzgün belirlenmemişse sunucudaki bütün hafızanın kullanılmasına sebep olabilirler. *FEDERATED* tablo türü bir tablonun başka bir tablonun kopyası olmasını sağlar. Bu iki tabloda yapılan herhangi bir değişiklik aynı zamanda diğer tabloya da uygulanır [16].

5. RSS BESLEME YÖNETİM SİSTEMİ

Bu bölümde;

- RSS kaynak adreslerini kaydedilmesi,
- Kaydedilen kaynaklardan verilerin çekilmesi,
- Çekilen veri içerisinden başlık bilgisinin ayrılması,
- Başlık bilgisi içersinde bulunan cümleden kelimelerin çıkarılması,
- Gerekli temizleme ve denetimlerin uygulanması,
- Veri tabanının yapısı ve ara yüzle etkileşimi,
- Öğrenmenin Gerçekleşmesi için veri tabanına gerekli kriterler eşliğinde kaydın yapılması,
- Veri tabanı üzerinde gerektiğinde düzeltme ve optimizasyonların yapılması,
- Veri tabanında oluşturulan kümeleme sistemi ile verilerin öğrenilmiş örüntüye göre RSS ‘ ten çekilmesi (Diğer bir deyişle süzülmesi).

Adımları anlatılmıştır. Öncelikle sistemi bir Şekil 5.1. ile pekiştirmek yerinde olacaktır.



Şekil 5.1. RSS belseme yönetimi sistemi yapısı.

Çeşitli haber sitelerinden alınan RSS adresleri RSS kaydet Linkine Tıklanarak Veri tabanında saklanmaktadır. Bu çalışmada örnek olarak;

- http://www.milliyet.com.tr/D/rss/rss/Rss_1.xml
- http://www.ntvmsnbc.com/id/24927681/device/rss/rss.xml
- http://rss.hurriyet.com.tr/rss.aspx?sectionId=1
- http://rss.hurriyet.com.tr/rss.aspx?sectionId=2
- http://www.milliyet.com.tr/D/rss/rss/Rss_2.xml
- http://www.milliyet.com.tr/D/rss/rss/Rss_3.xml
- http://www.milliyet.com.tr/D/rss/rss/Rss_4.xml
- http://www.milliyet.com.tr/D/rss/rss/Rss_31.xml
- http://www.milliyet.com.tr/D/rss/rss/Rss_36.xml

RSS kaynakları kullanılmıştır.

<description>……</description>

<title>……</title>

<link>…….</link>

XML tag' ları arasındaki veriler ana sayfaya çekilmiştir. Çekilen bu verilerin "link" kısmına tıklandığında sayfa öğrenme modülüne yönlendirilmektedir. Öğrenme modülünde öğrenmenin gerçekleşmesi için kullanılan kodlar ekler' de bulunmaktadır. Sistemde Rss kaynakların kullanılması hem kullanıcıya kaynak tarama safhasında büyük kolaylık sağlamaktadır. Yavanoğlu' nun Wesaka sisteminde haber kategorizasyonu yapılırken sayfalara teker teker girilmek zorunda idi. Rss beslemeler ile buna gerek kalmamıştır.

5.1. Öğrenmenin Gerçekleşmesi

Öğrenme modülünde öncelikle başlıktan gelen cümle içeriğinin parçalara ayrılması gerekmektedir. Bu arada sadece başlık verisinin kullanılma sebebi ise; haberi en doğru şekilde özetleyebilecek veri kümesi olduğu içindir. Ayırma işlemi bir fonksiyon yazılarak halledilmiştir. Bu fonksiyon gelen kelimeleri aradaki boşlukları

nirengi noktası alarak kaç adet kelime var ise o kadar kelimeye bölmektedir ve aynı zamanda cümle içerisindeki kirlilik yaratabilecek boşluk, noktalama işaretleri, rakamlar gibi karakterleri de temizlemektedir. Fakat PHP ‘ de özel karakter yerine geçen “ ‘ ” karakterini temizlenmesinde başarılı olunmamıştır. Bunun için sistemde bulunan örüntü gör kısmında gerekli düzeltmeler yapılabilmektedir. Kelimelerin veri tabanına kaydedilmeden önce sadece noktalama ve ya diğer karakterlerden değil Türkçe ‘ de bulunan yapım, çekim ekleri gibi eklerinde temizlenmesi gerekmektedir. Sebebi ise sistemin herhangi bir kelimeyi kayıt yaparken kayıtları tek bir karakteri değişse bile farklı bir kelimeymiş gibi görüp veri tabanına kayıt yapmasıdır.

“belge” kelimesi ile;

“belgeye” kelimesi farklı iki kelimeymiş gibi davranılmaktadır.

Fakat eklerin temizlenmesi işleminde başarılı olunamamıştır. Yapılan araştırmada “Zemberek” adlı kelime işleme kütüphanesinin bu sorunu ortadan kaldırdığı öğrenilmiştir. Fakat JAVA platformunda yazılan bu kütüphanenin sisteme entegrasyonu yapılamadığından kullanılamamıştır [17].

Kelimelerin ekleri ile kaydedilmesi veri tabanı üzerinde fazla veri tutma gibi eksileri olduğu gözlemlenmiştir. Fakat olaya başka bir yönden bakacak olursak bazen bu farklılıkların kaydedilmesinin öğrenme sırasında çekilen verilerde seçicilik yarattığı gözlemlenmiştir. Bu bazen olumlu etki yapsa da çoğu kez olumsuz yönde etkilemiştir. Bu sorun yukarıda da bahsedildiği üzere örüntü gör modülünde manuel olarak müdahale edilerek ortadan kaldırılmaya çalışılmıştır. Ayrıca kelimelerin çıkarımında veri tabanına kelimeyi oluşturan tüm kelimeler küçük harf ile kayıt altına alınmıştır. MySql veri tabanının küçük ve büyük har ayrımı yaptığını göz önünde bulundurursak kelime veri tabanının boyutunu arttırmamak adına olumlu bir adımdır. PHP dilinde bulunan *“strtolower”* hazır fonksiyonu Türkçe karakterleri küçük harfe çevirirken karakter bozulmaları yarattığı için Türkçe karakterleri de küçüğe çevirebilecek *“strtolower_utf8”* fonksiyonu yazılmıştır. Bu fonksiyon ekler bölümünde verilmiştir.

Öğrenme işlemi gerçekleşirken, yeni gelen veriler arasında hali hazırda veri tabanında bulunan bir değer var ise var olan kaydın frekans değeri bir arttırılmıştır. Böylece RSS' den gelen bütün veriler içersinde bulunan başlık bilgilerinin kelime değerleri veri tabanına frekans değerleri ile kaydedilmiştir. İlk kez kayıt yapılan verilerin frekans değerleri "1" kabul edilmiştir. MySql veri tabanı programının Türkçe karakterleri kaydederken veriler bozuk görünmesine rağmen PHP ile kullanılmak üzere sayfaya çekildiğinde karakterlerin düzgün çıktığı tecrübe edinilmiştir.

İsim	Türü	GEÇERSİZ (NULL)	Varsayılan	Ekst...
Birincil İndeks	Id			unique
Id	int(11)	Hayır	<auto_increment>	
birim	varchar(255)	Evet	<GEÇERSİZ (NULL)>	
frekans	bigint(20)	Evet	1	
agir	varchar(9)	Evet	<GEÇERSİZ (NULL)>	
kumea	varchar(1)	Evet	<GEÇERSİZ (NULL)>	
kumeb	varchar(1)	Evet	<GEÇERSİZ (NULL)>	
kumec	varchar(1)	Evet	<GEÇERSİZ (NULL)>	
tarih	varchar(255)	Evet	<GEÇERSİZ (NULL)>	

Resim 5.1. Veri tabanında bulunan "kelime" tablosunun yapısı

İsim	Türü	GEÇERSİZ (NULL)	Varsayılan	Ekst...
Birincil İndeks	Id			unique
Id	int(11)	Hayır	<auto_increment>	
rss	text	Evet		

Resim 5.2. Veri tabanında bulunan "Rss_kaynak" tablosunun yapısı

Veri tabanına kaydedilen veriler "örüntü gör" sayfasına tıklandığında liste halinde Resim 5.3.'de de olduğu gibi

görülebilmektedir.

380 adet kelime.

Kelime:	Frekans	Sil	Ağırlık	Karar Kümesi A	Karar Kümesi B	Karar Kümesi C	Okuma Tarihi
ölüm Düzelt	5		0,027473	1	1	1	05.01.2011
hırsız Düzelt	4		0,021978	1	1	1	04.01.2011
için Düzelt	4		0,021978	1	1	0	04.01.2011
tahliye Düzelt	4		0,021978	1	1	0	04.01.2011
yeni Düzelt	4		0,021978	1	1	0	04.01.2011
sakatlanan Düzelt	7		0,019231	1	1	0	03.01.2011
şantaj Düzelt	3		0,016484	1	0	0	05.01.2011
çıplak Düzelt	3		0,016484	1	0	0	05.01.2011
beşiktaş Düzelt	3		0,016484	1	0	0	04.01.2011
fotoğraflı Düzelt	3		0,016484	0	0	0	05.01.2011
galatasaray Düzelt	3		0,016484	0	0	0	04.01.2011
işadamına Düzelt	3		0,016484	0	0	0	05.01.2011
tutuklandı Düzelt	3		0,016484	0	0	0	04.01.2011

Resim 5.3. Örüntü gör sayfasında verilerin görünümü.

Uygun olmayan veriler “sil” komutu ile veri tabanından silinebilirler. Veya veriler üzerinde değişiklik yapılarak değişiklikler yapılarak frekans değeri değişmeden düzeltmeler yapılabilir. Eğer düzeltme işlemi yapıldığında yeni veri daha önceden kayıtlı bir veri ile eşleşiyorsa frekans değerleri toplanıp tek bir veri haline dönüştürülmektedir. Bu bir şekilde aynı iki verinin tabloda bulunduğu haller içinde kullanılabilir. Yapılan sisteme de frekans değeri “1” olan veriler ağırlık hesabı ve kümeleme işlemine tabi tutulmamışlardır. Sebebi ise frekans değeri 1 olan bir değer çok nadir olarak kullanılan bir veri olabilir ver gereksiz yere ağırlık ve küme hesabının değerlerini etkilemektedir

Verilerin sağ tarafında bulunan “küme” değerleri ise verilerin önem derecelerini hesaplamak için kullanılan veri alanlarıdır. Bu alanlar hesaplanırken;

Küme A için; frekansı 1 den büyük olanlar sıralamasında yüzde değeri olarak ilk %15 ‘ e giren kelimelerin küme A değeri “1” olarak kaydedilmektedir.

Küme B için; frekansı 1 den büyük olanlar sıralamasında yüzde değeri olarak ilk %10 ‘ a giren kelimelerin küme b değeri “1” olarak kaydedilmektedir.

Küme C için; frekansı 1 den büyük olanlar sıralamasında yüzde değeri olarak ilk %5 ‘ e giren kelimelerin küme C değeri “1” olarak kaydedilmektedir.

Böylelikle karşımıza 3 – Bit’ den oluşan bir önem derecesi çıkmaktadır.

Çizelge 5.1. Kümeleme sistemi ile önem dereceleri listesi

Küme A	**Küme B**	**Küme C**	
1	**1**	**1**	**Desimal değeri 7** **Yüksek önem derecesi**
1	**1**	**0**	**Desimal değeri 6** **Orta önem derecesi**
1	**0**	**0**	**Desimal değeri 4** **Az önem derecesi**

Çizelge 5.1. de verilen örnekte önem derecelerini alabileceği değerler verilmiştir. Önem dereceleri Küme A’ dan Küme C’ ye piramit şeklinde gitmektedir. Kayıt sayısı bu şekle etken olmadığı tecrübe edilmiştir. Bu kümeleme işlemi her öğrenme gerçekleştiğinde yeniden dinamik olarak yeniden hesaplanmaktadır çünkü verilerin frekans değerleri haber başlıklarına tıklandığı sürece değişmektedir. İstenildiği takdirde küme sayıları arttırılabilmektedir. Fakat bunun için veri tabanında bulunan veri miktarının çok büyük olması gerekmektedir.

Ağırlık hesabına katılan bir diğer etmen ise kelimeye rastlanma tarihidir. Bunun önemini şöyle açıklayabiliriz. Herhangi bir kelime veri tabanına kayıt edildiğinde ve frekans değeri arttığında o kelime güncel bir olay veya dönemlik bir olaya ait olabilir. Yıl başı, Bayram, Seçim gibi kelimeler olabilir. Fakat günler geçtikçe kelimeler frekans değerleri ne olursa olsun önemini yitirmektedir. doğal olarak

dönemlik bir habere ait kelimenin frekansı yüksek değerlere ulaştığında diğer frekans değerleri yanında uzunca bir süre yüksek önemde görünebilir. İşte burada güncel olarak rastlanmayan bu verinin önemini düşürebilmek için kelimenin rastlanma tarihi ağırlık hesabına katılmıştır. Buna göre ağırlık hesabı aşağıdaki gibi hesaplanmaktadır;

$$Ağırlık = \frac{Kelime\ frekansı}{(Frekansı\ 1'den\ büyük\ kelimelerin\ frekansı \times Rastlama\ tarihi\ ile\ güncel\ tarih\ farkı)}$$

Burada ağırlık değeri en az "0" en fazla "1" değeri alabilmektedir. Fakat gerek kelime sayısı gerekse frekans sayıları alt ve üst sınır değerleri teoride bırakmaktadır. Ağırlık hesabında karşılaşılabilecek olan bir engel 0' a bölünmedir. Rastlama tarihi ile güncel tarih arasındaki fark aynı gün içerisinde rastlanan kelimelerin ağırlık hesabında paydayı "0" yapacağından 0' a bölünmeden dolayı hata ile karşılaşılacaktır. Aynı zamanda tarih farkı hesabında "1" sayısı da etkisiz eleman olacağından hesaba katmanın bir anlamı olmayacaktır.

Rastlama tarihi ve güncel tarih farkı "1" ve "0" olan kelimelerin ağırlık hesaplarında bu fark kullanılmamış ve karşılaşılabilecek hesap hataları engellenmiştir.

Bu hesapta kelimeye rastlanılan tarih ile güncel tarih arasındaki fark ağırlık değerini aşağıya çekmektedir. Buda zaman geçtikçe o kelimenin önemini yitirmesini sağlamaktadır. Bu arada tarihi eski olan bir kelimeye yeniden rastlandığında ise tarih bilgisi de güncellenerek kelimenin tekrar önem derecesi kazanması sağlanmaktadır. Burada ki amaç öğrenmenin insan zihninde gerçekleşen öğrenmeye yakın olmasını sağlanmaktır.

Resim 5.4. Öğrenme işlemi gerçekleştikten sonra haberlerin listelenmesi

Resim 5.3. ‘ de görüldüğü gibi en yüksek önceliğe sahip olan kelimelerin bulunduğu haberler öğrenilmiş veri sayfası altında görülebilmektedir. Yaklaşık 250 adet haber başlığı ana sayfa da listelenmektedir, öğrenilmiş veri ile haberler süzüldüğünde bu sayı 60 ila 90 arasında değişmektedir. Yeni çekilen veriler içersinde öğrenilmiş veriler aranmakta ve öncelik sırasına göre yeni haberler listelenmektedir. Önem derecesi eşit olan veriler olduğunda öncelik ağırlık değeri fazla olana verilmektedir. Kümeler oluşturulurken verilen yüzde değerleri değiştirilerek kümeye dahil edilebilecek kelime sayısı arttırılıp azaltılabilir. Bu oran üzerinde oynama yapmak bize daha hassas bir öğrenme veya daha genel bir öğrenme sunabilmektedir.

6. SONUÇ

Öğrenmenin gerçekleşmesi için daha önce verilen RSS linklerine 20 Gün boyunca, 500 Öğrenme işlemi ve 1036 adet kelime kaydı yapılmıştır. Bu işlem;

- Intel Centrino Duo 2 P8600 2.4 Ghz
- 3 GB Ram
- 320 Gb Hdd
- Internet Explorer 8 Web tarayıcı
- PHP 5 versiyonu
- MySQL istemci sürümü: 5.0.51a
- Apache Server
- MySql Front veri tabanı yönetim sistemi

Donanım ve yazılım bileşenleri ile gerçekleştirilmiştir. Öğrenme işlemi örüntü gör sayfasında her defasında yaklaşık 2 Saniye sürmektedir. 380 adet öğrenme gerçekleştiğine göre 2 x 380 = 760 saniye, bu da yaklaşık 13 Dakika sürmüştür. Tabi ki 13 dakika blok bir zaman dilimi değil, 2' şer saniyelik parçalar halinde sürmüştür. Öğrenme sayısı arttıkça da bu süre doğal olarak artacaktır. 250 adet haber başlığının yukarıda verilen değerler ile süzülmesinden 66 adet başlık kullanıcının haber okuma alışkanlığına göre listelenmiştir. Buda verilerin yaklaşık % 73 ' ünün süzüldüğünü göstermektedir. Bu işlem sonucunda kullanıcıya haber okurken istediği haberi bulmakta zaman tasarrufu sağlamaktadır.

Sistemin diğer benzer sistemlerden farkı; sınıflandırma yapmak amacı ile değil bu sınıflandırmayı kullanarak kullanıcıya en uygun verilerin kullanıcıya sunulmasıdır. Sistem sonuçta bir değerlendirme yapmak yerine kümelenen verileri kullanarak yeni gelen verileri süzme işlemi yapmaktadır.

Sonuç olarak RSS kaynaklardan alınan veriler ile eğitilen bir sistemin bir kullanıcıya sağlayabileceği yararlar görülmüştür. Kullanıcıyı fazla bilgi sayfalarından kurtarmakta ve zaman kazandırmaktadır. Sistem sadece haber kaynaklarının yönetiminde değil e-ticaret, e-öğrenme, reklamcılık gibi alanlarda da kullanılabilir.

Bundan sonraki çalışmalar için kelime köklerinin sağlıklı bir şekilde bulunması büyük yarar sağlayacaktır. Hatta kelimelerin kökleri bulunduktan sonra öğrenme verisine köklerin alabileceği eklerde dahil edilebilirse isabet oranının artması garantilenecektir. Dil sınıflandırması çalışmaları ile birlikte çalışılarak farklı sonuçlar elde edilmesi mümkündür.

EKLER

Ek-1. Ana sayfa'da Rss başlıklarının çekilmesi

```
<?php
$k=0;
$sorgudizim="select * from rss_kaynak";
$dizimsonuc=mysql_query($sorgudizim) or die("Rss kaynak bulunmamakta veya erişilemiyor.");
while ($oku=mysql_fetch_array($dizimsonuc))
{
$sayfa=$oku["rss"];
$rss_id=$oku[0];
$kaynak = file_get_contents($sayfa);
$desc = '#<description>(.*?)</description>#si';
$desc2 = '#<title>(.*?)</title>#si';
$desc3 = '#<link>(.*?)</link>#si';
preg_match_all($desc,$kaynak,$ddesc);
preg_match_all($desc2,$kaynak,$ddesc2);
preg_match_all($desc3,$kaynak,$ddesc3);
$ddesc = $ddesc[1];
$ddesc2 = $ddesc2[1];
$ddesc3 = $ddesc3[1];
$news_total=count($ddesc);
$k=$news_total + $k;
$i=2;
while($i <= $news_total){
//$k++;
$katar=html_entity_decode($ddesc2[$i]);
echo
"<br>_____<br>";
```

Ek-1(Devam). Ana sayfa'da Rss başlıklarının çekilmesi

```
echo "<a href='learn.php?katar=$katar&rss_id=$rss_id'>$katar</a><br>";//başlık
echo "<a href=$ddesc3[$i] target=_blank >Haber Kaynağına Git...</a><br>";
//habere git
echo html_entity_decode("".$ddesc[$i]."<br>"); // haber özeti
echo
"<br>_____
__<br>";
$i++;
}
}
echo $k." adet haber alındı..";
?>
```

Ek-2. Öğrenme işleminde verilerin optimizasyonu için kullanılan “k_sil” fonksiyonu

```
function k_sil($malzeme)
{
$o_kel=array(".",",","':",";","?","!","-
","'","*","+","<",">","$","_","/","(",")","0","1","2","3","4","5","6","7","8","9","");
  foreach ($o_kel as $kar)    {
$malzeme= str_replace($kar, '', $malzeme);
}
return($malzeme);
}
function sayi_sil($number)
{
$n_kel=array();
foreach ($n_kel as $str)
{
$number=str_replace($str, '', $number);
}
return($malzeme);
}
```

Ek-3. Strtolover_utf8 fonksiyonu

```
function strtolower_utf8($string)
{
$convert_to = array( "a", "b", "c", "ç", "d", "e", "f", "g", "ğ", "h", "i", "ı", "j", "k", "l",
"m", "n", "o", "ö", "p", “q", "r", "s", "ş","t", "u", "ü", "v", "w", "x", "y", "z", "à", "á",
"â", "ã", "ä", "å", "æ", "ç", "è", "é", "ê", "ë", "ì", "í", "î", "ï", "ð", "ñ", "ò", "ó", "ô",
"õ", "ö", "ø", "ù", "ú", "û", "ü", "ý", "a", "б", "в", "г", "д", "e", "ё", "ж", "з", "и", "й",
"к", "л", "м", "н", "o", "п", "p", "c", "т", "y", "ф", "х", "ц", "ч", "ш", "щ", "ъ", "ы",
"ь", "э", "ю", "я");
$convert_from = array("A", "B", "C", "Ç", "D", "E", "F", "G", "Ğ", "H", "İ", "I", "J",
"K", "L", "M", "N", "O", "Ö", "P", "Q", "R", "S", "Ş", "T", "U", "Ü","V", "W", "X",
"Y", "Z", "À", "Á", "Â", "Ã", "Ä", "Å", "Æ", "Ç", "È", "É", "Ê", "Ë", "Ì", "Í", "Î",
"Ï", "Ð", "Ñ", "Ò", "Ó", "Ô", "Õ", "Ö", "Ø", "Ù", "Ú", "Û", "Ü", "Ý", "A", "Б", "B",
"Г", "Д", "E", "Ё", "Ж", "З", "И", "Й", "К", "Л", "M", "H", "O", "П", "P", "C", "T",
"У", "Ф", "Х", "Ц", "Ч", "Ш", "Щ", "Ъ", "Ъ", "Ь", "Э", "Ю", "Я" );
return str_replace($convert_from, $convert_to, $string);
}
```

Ek-4. Öğrenme işlemi gerçekleştirilen kod satırları

```
<?php
$katar=$_GET["katar"];
$rss_id=$_GET["rss_id"];
$kelime=explode(" ",$katar);
$kelime_sayisi = count($kelime);
for ($d=0 ;  $d <= $kelime_sayisi ; $d++ )
{
$kelime_kontrol= $kelime[$d];
$kelime_kontrol=k_sil($kelime_kontrol);
$kelime_kontrol= strtolower_utf8($kelime_kontrol);
if ($kelime_kontrol!="")
{
$varmi=mysql_query("select * from kelime where
birim='$kelime_kontrol'",$baglanti);
$k=0;
while ($oku2=mysql_fetch_array($varmi))
{
if ($oku2["birim"]==$kelime_kontrol)
{
$kelime_kontrol=trim($kelime_kontrol);
$tarih=date("d.m.Y");
mysql_query("update kelime set frekans= frekans + 1 , tarih = '$tarih' where
birim='$kelime_kontrol'",$baglanti);
$k=1;
}
if ($k==0)
{
```

```
$kelime_kontrol= strtolower_utf8($kelime_kontrol);
```

Ek-4(Devam). Öğrenme işlemi gerçekleştirilen kod satırları

```
$kelime_kontrol=trim($kelime_kontrol);
$minkarakter=mb_strlen($kelime_kontrol);
if ($minkarakter > 2 and $minkarakter!=NULL )
{
$kelime_kontrol=stripslashes($kelime_kontrol);
$tarih=date("d.m.Y");
$tamam=mysql_query("insert into kelime (birim, rss_id, tarih) values
    ('$kelime_kontrol', '$rss_id', '$tarih')",$baglanti);
if ($tamam)
{
echo "Kaydedilen kelime: ".$kelime_kontrol."<br>";
}
}
else
{
}
}
}
}

?>
```

Ek-5. İki tarih arasındaki gün sayısını hesaplayan fonksiyon kod satırları

```
function fark_bul($tarih1,$tarih2,$ayrac) {
/*  İki tarih arasındaki gün farkını bulur          */
list($g1,$a1,$y1) = explode($ayrac,$tarih1);
list($g2,$a2,$y2) = explode($ayrac,$tarih2);
$t1_timestamp = mktime('0','0','0',$a1,$g1,$y1);
$t2_timestamp = mktime('0','0','0',$a2,$g2,$y2);
if ($t1_timestamp > $t2_timestamp)
{
$result = ($t1_timestamp - $t2_timestamp) / 86400;
}
else if ($t2_timestamp > $t1_timestamp)
{
$result = ($t2_timestamp - $t1_timestamp) / 86400;
}
return $result;
}
```

Ek-6. Örüntü Gör sayfasında verilerin ağırlıklandırma, kümeleme ve frekanslarının hesaplandığı kod satırları

```
$sayi=mysql_query("SELECT * FROM kelime");
$count=mysql_num_rows($sayi);
$sayi2=mysql_query("SELECT * FROM kelime where frekans < 2");
$count2=mysql_num_rows($sayi2);
$count3=$count - $count2;
$yuzdeon=number_format($count3*0.15,0);
$yuzdeon_on=number_format($count3*0.1,0);
$yuzdeon_on_on=number_format($count3*0.05,0);
$i=1;
$sorgu2=mysql_query("select Id from kelime order by agir desc, birim asc");
while ($sonuc2=mysql_fetch_array($sorgu2))
{
// yüzde 15 1 gerisini 0 yapma
if ($l<$yuzdeon)
{
$degistir_id=$sonuc2['Id'];
mysql_query("update kelime set kumea='1' where Id='$degistir_id' ");
}
else
{
$degistir_id=$sonuc2['Id'];
mysql_query("update kelime set kumea='0'  where Id='$degistir_id'  ");              }
// yüzde 10 1 gerisini 0 yapma
if ($l<$yuzdeon_on)
{
$degistir_idk=$sonuc2['Id'];
```

```
mysql_query("update kelime set kumeb='1' where Id='$degistir_idk' ");          }
```

Ek-6(Devam). Örüntü Gör sayfasında verilerin ağırlıklandırma, kümeleme ve frekanslarının hesaplandığı kod satırları

```
else
{
$degistir_idk=$sonuc2['Id'];
mysql_query("update kelime set kumeb='0'  where Id='$degistir_idk'  ");          }
// yüzde 5 1 gerisini 0 yapma
if ($l<$yuzdeon_on_on)
{
$degistir_idm=$sonuc2['Id'];
mysql_query("update kelime set kumec='1' where Id='$degistir_idm' ");
}
else
{
$degistir_idm=$sonuc2['Id'];
mysql_query("update kelime set kumec='0'  where Id='$degistir_idm'  ");          }
$i++;
}
echo $count." adet kelime.";
$i=0;
//ağırlıkları veri tabanına yaz
$q2 = mysql_query("SELECT SUM(frekans) AS frekanstoplam FROM kelime
where frekans > 1");
while ($f_oku2=mysql_fetch_array($q2))
{
$tfrekans=$f_oku2["frekanstoplam"];
}
$q3=mysql_query("select * from kelime");
```

```
while ($res=mysql_fetch_array($q3))
                    {
```

Ek-6(Devam). Örüntü Gör sayfasında verilerin ağırlıklandırma, kümeleme ve frekanslarının hesaplandığı kod satırları

```
$id=$res["Id"];
$frekans=$res["frekans"];
if ($tfrekans!=0) {
//Ağırlıklara tarihi' de kat basla
$bugun=date("d.m.Y");
$gun_farki = fark_bul($res["tarih"], $bugun,'.');
if ($gun_farki==0)
{
$agir=$frekans/$tfrekans;
}
if ($gun_farki==1)
{
$agir=$frekans/$tfrekans;
}
if ($gun_farki > 1)
{
$agir=$frekans/ ($tfrekans * $gun_farki);
}
//Ağırlıklara tarihi' de kat son
$agir=number_format($agir, 6, ",", ".");
}
else
{
$agir=0;
}
```

```
mysql_query("update kelime set agir='$agir' where Id='$id' and frekans > 1",
$baglanti) or die ("agirliklar güncellenemedi");
$i++;
```

Ek-6(Devam). Örüntü Gör sayfasında verilerin ağırlıklandırma, kümeleme ve frekanslarının hesaplandığı kod satırları

```
}
$sorgu="select * from kelime order by agir desc, birim asc";
$sonuc=mysql_query($sorgu) or die("Veri Yok!");
while ($oku=mysql_fetch_array($sonuc))
{
echo "<tr>";
echo "<td bgcolor='#CC9966'  valign='middle'>
<form name ='form1'enctype='multipart/form-data' action='process.php'
method='post' >
<input name='yenibirim' type='text' value='$oku[1]' size='20' maxlength='255'>
<input name='id_duzelt' type='hidden' value='$oku[0]'>
<input type='submit' name='button' id='button' value='Düzelt'> </form> </td>";
echo "<td bgcolor='#CC9966' align='center' valign='middle'>$oku[2]</td>";
echo "<td bgcolor='#CC9966' align='center'><a
href='process.php?id=$oku[0]&islem=sil'><img src='images/file_delete.png'
width='32' height='32'></a></td>";
echo "<td bgcolor='#CC9966' align='center' valign='middle'>";
$q = mysql_query("SELECT agir FROM kelime where id='$oku[0]' ");
while ($f_oku=mysql_fetch_array($q))
{
if ($f_oku["agir"]!=NULL)
{
echo $f_oku["agir"];
}
else
```

```
{
echo 0;
}
```

Ek-6(Devam). Örüntü Gör sayfasında verilerin ağırlıklandırma, kümeleme ve frekanslarının hesaplandığı kod satırları

```
}
echo "</td>";
echo "<td bgcolor='#CC9966' align='center' valign='middle'>";
echo $oku["kumea"];
echo "</td>";
echo "<td bgcolor='#CC9966' align='center' valign='middle'>";
echo $oku["kumeb"];
echo "</td>";
echo "<td bgcolor='#CC9966' align='center' valign='middle'>";
echo $oku["kumec"];
echo "</td>";
echo "<td bgcolor='#CC9966' align='center' valign='middle'>";
echo $oku["tarih"];
echo "</td>";
echo "</tr>";
}
```

Ek-7. Öğrenilmiş veri üzerinde silme ve düzeltme işlemi yapan process.php dosyası kod satırları

```
<?
if ( $_GET["islem"]=='sil' )
{
$id_sil=$_GET["id"];
mysql_query("delete from kelime where Id=$id_sil") or die ("Kayıt
Silinemedi<br>");
?><br>Lütfen Bekleyin...<br>
<img src="images/loading.gif" width="16" height="16" border="0"><?
}
if (isset($_POST["yenibirim"]) )
{
$id_degistir=$_POST["id_duzelt"];
//echo $id_degistir;
$yeni_birim=strtolower_utf8($_POST["yenibirim"]);
$varmi=mysql_query("select * from kelime where birim='$yeni_birim'",$baglanti);
$k=0;
while ($oku2=mysql_fetch_array($varmi))
{
if ($oku2["birim"]==$yeni_birim)
{
mysql_query("update kelime set frekans= frekans + 1 where
birim='$yeni_birim'",$baglanti);
mysql_query("delete from kelime where Id=$id_degistir") or die ("Kayıt
Silinemedi<br>");
$k=1;
}
}
```

if ($k==0)

Ek-7(Devam). Öğrenilmiş veri üzerinde silme ve düzeltme işlemi yapan process.php dosyası kod satırları

```
{
$yeni_birim=trim($yeni_birim);
mysql_query("UPDATE kelime SET birim='$yeni_birim' WHERE Id='$id_degistir'
") or die ("Kayıt düzeltilemedi<br>");
}
?><br>Lütfen Bekleyin...<br>
<img src="images/loading.gif" width="16" height="16" border="0">
<? } ?>
```

Ek-8. Öğrenilmiş veriyi kullanarak kullanıcıya sunan sayfa (kume.php) kod satırları

```
<?
$k=0;
$t=0;
$sorgudizim="select * from rss_kaynak";
$dizimsonuc=mysql_query($sorgudizim) or die("Rss kaynak bulunmamakta veya erişilemiyor.");
while ($oku=mysql_fetch_array($dizimsonuc))
{
$sayfa=$oku["rss"];
$rss_id=$oku[0];
$kaynak = file_get_contents($sayfa) or die ("kaynağa erişilemedi");
$desc = '#<description>(.*?)</description>#si';
$desc2 = '#<title>(.*?)</title>#si';
$desc3 = '#<link>(.*?)</link>#si';
preg_match_all($desc,$kaynak,$ddesc);
preg_match_all($desc2,$kaynak,$ddesc2);
preg_match_all($desc3,$kaynak,$ddesc3);
$ddesc = $ddesc[1];
$ddesc2 = $ddesc2[1];
$ddesc3 = $ddesc3[1];
$news_total=count($ddesc);
$t=$news_total + $t;
$i=2;
while($i <= $news_total)
{
$gor=0;
$katar=html_entity_decode($ddesc2[$i]);
```

// kelimeler aranıyor

Ek-8(Devam). Öğrenilmiş veri ile sunulan sayfa (kume.php) kod satırları

```
$sorgu=mysql_query("select * from kelime order by agir desc",$baglanti);
while ($sonuc=mysql_fetch_array($sorgu))
{
if ($sonuc["agir"] != NULL )
{
$onem=$sonuc["birim"];
if (strstr($katar,$onem))
{ $gor++;  }
//kelimeler aranıyor
}
}
if ($gor != 0)
{
$k++;
echo
"<br>_____<br>";
echo $katar."<br>";
echo "<a href=$ddesc3[$i] target=_blank >Haber Kaynağına Git...</a><br>";
//habere git
echo html_entity_decode("".$ddesc[$i]."<br>"); // haber özeti
echo
"<br>_____<br>";
}
$i++;
}
}
echo $t." adet haberden ".$k." tanesi alındı";                    ?>
```

Ek-9. Rss kaynakların veri tabanına kaydedilmesi için kullanılan kod satıları

```
$adres=$_POST["rss"];
$adres=trim($adres);
if ($adres=="")
{
header("location:rss_ekle.php");
}
$sorgu="insert into rss_kaynak (rss) values ('$adres')";
mysql_query($sorgu,$baglanti) or die("<BR>Can't add record...");
```

Ek-10. Ana sayfa' da alınan haber sayısı

17:48 Fenerbahçe Kazım'ın sözleşmesini feshetti
Haber Kaynağına Git...

Fenerbahçe Kulübü, futbolcusu Colin Kazım'ın sözleşmesini karşılıklı olarak feshetti. Sarı-lacivertli kulüp, hücum oyuncusu Kazım Kazım'ın sözleşm...

E-mail this BOOKMARK

17:47 İngiltere'de rekor vergi artışı
Haber Kaynağına Git...
İngiltere'de katma değer vergisinde (KDV) şimdiye kadarki en yüksek artış yaşandı. Ülkede yüzde 17,5 olan KDV yüzde 20'ye çıkarıldı. Gıda, çocuk...

E-mail this BOOKMARK

Haber Kaynağına Git...

237 adet haber alındı..

Ek-11. Öğrenilmiş veri ile alınan haber sayısı

18:12 İşadamına çıplak fotoğraflı şantaj
Haber Kaynağına Git...
SAKARYA'da karavan imal eden firmayı silahlarla basan 7 kişi, firma sahibi Ergün B.yi etkisiz hale getirdikten sonra zorla senet imzalatıp kasada b...
E-mail this BOOKMARK

18:09 Yasak aşkta ikinci ölüm
Haber Kaynağına Git...
KIRIKKALE'de amcasının oğlunun açtığı ateşle yaralanan evli ve 1 çocuk babası Beytullah Ağırbaş da tedavi gördüğü hastanede öldü. Olayda, Tarkan A...
E-mail this BOOKMARK

17:49 Beşiktaş'a vergi cezası
Haber Kaynağına Git...
Beşiktaş Futbol Yatırımları Sanayi ve Ticaret A.Ş, şirkete tebliğ edilen rapora dayanılarak, 2005-2007 hesap dönemleri için KDV yönünden yapılan in...
E-mail this BOOKMARK

257 adet haberden 57 tanesi alındı

KAYNAKLAR

1. Internet: Türkiye bilim sitesi, "Yapay zeka mı? Makine öğrenmesi mi?", http://www.genbilim.com/content/view/7609/211/ (2010).

2. Internet : Dünya gazetesi,"İnternetten gazete okumak", http://www.dunyagazetesi.com.tr/haberArsiv.asp?id=148087 (2010).

3. Yavanoğlu, U., "Web tabanlı otomatik dil tanıma ve çevirme sisteminin geliştirilmesi", ***Gazi Üniv. Müh. Mim. Fak. Der.***, Cilt 25, No 3, 483-494, (2009)

4. Sağıroğlu, Ş., Güven, E.N., Onur, H., ***Yapay Sinir Ağları ile Web İçeriklerini Sınıflandırma*** ,Bilgi Dünyası, 9(1):158-178, (2008)

5. Orhan, Z., Altan, Z., "Makine Öğrenme Algoritmalarıyla Türkçe Sözcük Anlamı Açıklaştırma" Bilgisayar ***Mühendisliği Bölümü Mühendislik Fakültesi İstanbul Üniversitesi*** :1-4, (2004)

6. Pilavcılar, İ.F., "Metin Madenciliği ile Metin Sınıflandırma", Yüksek Lisans Tezi, ***FBE Matemetik Muhendisliği Anabilim Dalı Matematik Muhendisliği Programı Yıldız Teknik Üniversitesi,*** İstanbul, 3-50 (2007)

7. Ünsal, Ö., "Determination Of Vocational Fields With Machine Learning Algorithm" **,** ICMLA**:** 1 – 6 , (2010)

8. Karasar, N., "Bilimsel Araştırma Yöntemi, ISDN:978-975-591-046-8," ***Nobel Yayınevi 14.Baskı*****,** Ankara, 20-65 (2005)

9. İnternet:Wikipedia, "Makine Öğrenimi", http://tr.wikipedia.org/wiki/Makine_%C3%B6%C4%9Frenimi (2010).

10. Akpınar, H., ***İ.Ü. İşletme Fakültesi Dergisi,*** C:29, S: 1-22 (2000)

11. İnternet: Wikipedia, "Veri madenciliği", http://tr.wikipedia.org/wiki/Veri_madencili%C4%9Fi (2010).

12. Amasyalı, M. F., Makine Öğrenmesine Giriş, ***Yıldız Teknik Üniversitesi Bilgisayar Mühendisliği Bölümü*** :2-20 (2007)

13. İnternet: Wikipedia, "RSS nedir", http://tr.wikipedia.org/wiki/RSS (2010).

14. İnternet: MyBB Türkçe destek forumu, "PHP nedir", http://www.mybbturkiye.com/Thread-PHP-Nedir-ve-Neden-PHP (2010).

15. İnternet: National Taipei University of Technology, "MySQL Pluggable Storage Engine Architecture", http://ftp.ntut.edu.tw (2009).

16. Welling, L., "Php ve MySQL" ,***Alfa Yayıncılık***, ISBN: 9752971172 , İstanbul, 8-15 (2005)

17. İnternet: Wikipedia, "Zemberek", http://tr.wikipedia.org/wiki/Zemberek_(yaz%C4%B1l%C4%B1m) (2010).

18. Internet: Coss90 blog hizmetleri, "Bayes Teoremi" http://www.cos90.com/istatistik/bayesi-anlamak-bayes-teoremi/ (2010).

19. Karasar, N., "Bilimsel Araştırma Yöntemi, ISDN:978-975-591-046-8," ***Nobel Yayınevi 14.Baskı*****,** Ankara, 20-65, (2005)

20. Acartürk, C., Kürşat, Ç. K., "İnsan Bilgisayar Etkileşimi ve ODTÜ'de Yürütülen Çalışmalar" , ***Bilgi Teknolojileri Kongresi IV /Akademik Bilişim 2006*** **,** Ankara, 59, (2006)

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, adı	:YARDIMCI, Tuğrul
Uyruğu	:T.C.
Doğum tarihi ve yeri	:20.03.1982 Afyon
Medeni Hali	:Evli
Telefon	:0 505 677 32 63
Faks	: -
E-mail	:tugrulyardimci@yahoo.com

Eğitim

Derece	**Eğitim Birimi**	**Mezuniyet Tarihi**
Lisans	Selçuk Üniversitesi Bilg. Sist. Eğit. Bölümü	2005
Lise	Eskişehir Hoca Ahmet Yesevi Lisesi	1999

İş Deneyimi

Yıl	**Yer**	**Görev**
2006-2009	Ürgüp Anadolu Meslek ve Meslek Lisesi	Öğretmen
2009-…..	Gazi Teknik ve Endüstri Meslek Lisesi	Öğretmen

Yabancı Dil

İngilizce

Hobiler

Gitar Çalmak, Sinema, Kitap Okumak, Müzik