



**YILDIZ TEKNİK ÜNİVERSİTESİ
ELEKTRİK-ELEKTRONİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

BİLGİSAYAR PROJESİ

**AĞ MADENCİLİĞİ YÖNTEMLERİYLE
İNTERNET KULLANICILARININ
SINIFLANDIRILMASI**

Proje Yöneticisi: Yrd. Doç. Dr. M. Elif Karslıgil

Proje Grubu
02011040 Eren Aykın

İstanbul, 2006

© Bu projenin bütün hakları Yıldız Teknik Üniversitesi Bilgisayar Mühendisliği Bölümü'ne aittir.

İÇİNDEKİLER

Kısaltma Listesi.....	ii
Şekil Listesi	iii
Tablo Listesi	iv
Önsöz.....	v
Özet.....	vi
Abstract.....	vii
1.Giriş.....	1
1.1. Ağ Madenciliğiyle İlgili Önceki Çalışmalar.....	1
2. Sistem Analizi ve Fizibilite Çalışması.....	6
2. Sistem Analizi ve Fizibilite Çalışması.....	6
2.1. Teknik Fizibilite.....	7
2.2. Gannt Diyagramı.....	9
2.3. Ekonomik Fizibilite.....	10
2.4. Alternatif Fizibilite.....	10
3. Ağ Madenciliği Yöntemleriyle İnternet Kullanıcılarının Sınıflandırılması.....	12
3.1. Veri Madenciliği Süreci.....	13
3.2. Sunucu Tarafındaki Log Dosyasının Değerlendirilmesi.....	14
3.3. Naive Bayes Yöntemi.....	15
3.4. Sözlük Oluşturulması.....	16
3.5. Öğrenme.....	19
3.6. Sınıflandırma.....	21
4. Uygulama.....	23
4.1. Ekran Görüntüleri.....	23
5. Sonuç.....	27
6. Sözlük.....	28
7. Kaynaklar.....	30
Özgeçmiş.....	32

KISALTMA LISTESİ

URL:	Uniform Resource Locator
OPS:	Open Profiling Standart
Wbext:	Web Browser Extended
IDE:	Integrated Development Environment
JSDK:	Java Source Development Kit

ŞEKİL LİSTESİ

Şekil 3-1	Temel Blok Diyagramı.....	12
Şekil 3-2	Veri Madenciliği Süreci.....	14
Şekil 3-3	Her Kullanıcıya Ait Site Bilgilerin Diziye Atanması Akış Diyagramı...15	
Şekil 3-4	Sözlük Oluşturulması Akış Diyagramı.....	18
Şekil 3-5	Öğrenme Akış Diyagramı.....	20
Şekil 3-6	Sınıflandırma Akış Diyagramı.....	22
Şekil 4-1	Eğitim Dosyalarının Seçilmesi Ekran Görüntüsü.....	24
Şekil 4-2	Eğitim Ekran Görüntüsü.....	25
Şekil 4-3	Sınıflandırma Ekran Görüntüsü.....	26
Şekil 4-4	Grafikle Gösterim Ekran Görüntüsü.....	27

TABLO LİSTESİ

Tablo 2-1 Gannt Diyagramı.....	9
--------------------------------	---

ÖNSÖZ

Projedeki katkılarından dolayı Yrd. Doç. Dr. M. Elif Karşılıgil'e teşekkürlerimi sunarım.

ÖZET

Bu uygulamada, yerel bir ağ içerisindeki kullanıcıların, internette bağlandıkları sitelerin içeriklerinin incelenmesi ve sınıflandırılması yoluyla kişinin ilgi alanlarının ve hangi tür web sitelerini gezdiğinin belirlenmesi amaçlanmıştır.

Bir kullanıcının bağlandığı web sitelerindeki yazıların sınıflandırılması için Naive Bayes yöntemi kullanılmıştır. Bunun için önce eğitim dosyaları hazırlanmıştır. Eğitim dosyaları, o dosyanın temsil edeceği sınıfla ilgili web sitelerinin yazı içeriğinden oluşturulmuştur. Eğitim sırasında eğitim dosyalarındaki her bir kelimenin önceden belirlenmiş sınıflarda ne oranda bulunduğu bakılır. Kelimelerle ilgili bu bilgi sayesinde sınıflandırma aşamasında, kullanıcıların takip ettiği her sitenin sınıflandırılması gerçekleştirilir. Uygulama sonucunda her kullanıcının ne zaman ne türdeki siteleri ziyaret ettiği bilgisi oluşturulur.

Bu uygulama sayesinde yöneticiler çalışanlarını daha yakından tanıyabileceği gibi, interneti ne amaçla kullandıklarını da öğrenebilirler. Ayrıca çeşitli alanlarda farklı ürünleri olan bir şirket, bir yerel ağ içerisindeki farklı ilgi alanlarına sahip kullanıcılara ilgili ürün ya da hizmetlerini önerebilirler.

Sistemin başarısını değerlendirme amacıyla uygulanan test işleminde sistemin 5 farklı kullanıcı tarafından ziyaret edilmiş 20 adet web sitesini %90 oranında başarıyla sınıflandırdığı gözlemlenmiştir.

ABSTRACT

In this project a system that classifies internet users in a local network, based on contents of the web sites they have visited is developed.

Naive Bayes Algorithm is used in order to classify the text content achieved from the sites the user has visited. In order to educate the system, education files are created from the contents of the websites of the relevant subjects. In the education process, the frequency of occurrence of each word in each of the classes is determined. By the help of this information about the words, it becomes possible to classify the sites users have visited. By the help of this program, the information of who has visited which sites can be attained.

This project may help managers to know and be informed about their employees more, and tell them for what purposes is the internet being used. Furthermore, a system may be developed that offers advertisements to the users based on their interests.

As the result of the test process that was formed in order to test the success of the system, the program has classified 20 websites that belong to 5 users with a 90% success.

1. GİRİŞ

Günümüzde internette yaşanan bilgi patlaması yüzünden gereksiz bilgiler arasından değerli ve istenilen bilgiyi çıkarmak amacıyla pek çok yöntem geliştirilmiştir. Bu konudaki araştırmalardan bazıları aşağıda incelenmiştir.

1.1. Ağ Madenciliğiyle İlgili Önceki Çalışmalar

Andrew McCallum, Kamal Nigam, Jason Rennie, Kristie Seymore

Makine Öğrenmesi Teknikleriyle Alan Bazlı Arama Motorları Oluşturma [1]

Hiyerarşi ve her kategori için birkaç anahtar kelime kullanılarak, Naive Bayes, hiyerarşik büzüşme (shrinkage) ve beklenti maksimizasyonu (Expectation-Maximization) yöntemleri bir arada kullanılır. Sınıflandırıcının eğitilmesi sırasında insan emeğine gerek yoktur. Bu yöntemler sayesinde, sadece bilgisayar bilimleri alanında arama yapan *Cora arama motoru* oluşturulmuştur. Ağı büyütmek için gerekli bilgiler makalelerin referans kısımlarındaki bilgilerin ayrıştırılmasıyla elde edilir.

Jose Borges

Ağ Kullanımı Madenciliği İçin Bir Ortalama Lineer Zamanlı Algoritma [2]

Kullanıcının ağ navigasyonu oturumları ile ilgili bilgiler log bilgisinden elde edilir ve Markov zinciri olarak modellenir. Zincirde olasılığı en yüksek olan izler, ağda izlenmesi muhtemel yola karşılık gelir. Önerilen algoritma Markov modelini depth-first şeklinde tarar ve karmaşıklığı izlenen web sayfası sayısı ile birlikte lineer olarak artar.

Massimiliano Albanese, Antonio Picariello, Carlo and Lucio Sansone

Ağ Kullanıcı Madenciliği Teknikleriyle Oluşturulan Bir Ağ Kişiselleştirme Sistemi [3]

Ağ kişiselleştirilmesi için bir ağ madenciliği tekniği önerilir. Bu yöntem sabit ve değişken özellikleri analiz edip sınıflandırabilir. Önerilen sistemin sonuçları bir e-ticaret sitesinin verileri aracılığıyla gözlemlenir. Web sitesinin içeriğinin kişiselleştirilmesinde kullanılan yöntem şu konulara değinir: i) tek fazlı sınıflandırma yöntemi yerine iki fazlı Makine öğrenmesi teknikleri kullanılarak alan özellikli arama makinelerinin oluşturulması ve bakımının incelenmesi yapılır. Naive Bayes sınıflandırıcısına eklenen sınıflandırma kullanılır ii) hem kullanıcının verdiği bilgiler hem de web kullanım örüntüleri dikkate alınır. iii) hem kullanıcılar hem de içerikler sınıflandırılır. Bayes Teorisine dayalı olarak geliştirilmiş bulanık (fuzzy) denetlemesiz (unsupervised) kümeleme algoritması *Autoclass C* kullanılır.

Danny POO, Brian CHNG, Jie-Mein GOH

Kullanıcı Profili Belirlemek İçin Bir Melez Yöntem [4]

Aşırı ve gereksiz bilginin önlenmesi amacıyla, kullanıcı profili oluşturma yaklaşımları ve bilgi filtreleme teknikleri birleştirilir. Statik ve dinamik içeriğin kişiselleştirilmesi ve statik ve dinamik içeriğe göre gruplar oluşturma yöntemleri açıklanır ve bir kütüphane uygulaması üzerinde gösterilir.

Mark Craven, Johan Kumlien

Tekst İçeriklerden Bilgi Okuyarak Biyolojik Bilgi Tabanları Oluşturmak [5]

Yazı kaynağından elde edilen bilgileri, bilgi tabanlarındaki gibi yapılandırılmış şekilde temsil etmek amacıyla makine öğrenmesi tekniklerinin kullanılması amaçlanmıştır. Bu amaçla kullanılan iki öğrenme metodu: i) istatistiksel yazı sınıflandırılması metodu ii) ilişkisel öğrenme metodudur.

Stratos Paulakis, Charalampos Lampos, Magdalini Eirinaki, Michalis Vazirgiannis

SEWeP: Anlamsal Kişiselleştirmeyi Destekleyen Bir Ağ Madenciliği Sistemi [6]

Sadece ağ kişiselleştirilmesinin doğurabileceği sakıncaları engellemek için bir çerçeve (framework) önerilmiştir. Ağ kullanım logları ve ilgili ağ içeriğine anlambilimsel yaklaşım ve tasnif için SEWeP isminde bir prototip sistem hazırlanır. İçerik bir sözlük (thesaurus) ile anlamlandırılır ve c-log olarak kaydedilir. Bu loglar ağ madenciliği aşamasında girdi olarak kullanılır. Tamamen Java tabanlı bir sistemdir.

Yongjian Fu Ming, Yi Shih

Kişisel Ağ Kullanımı Madenciliği İçin Bir Çerçeve [7]

Kullanıcı taraflı çalışmanın faydalarından biri de cache'deki bilgilerden faydalanma olanağıdır. Kullanıcıya sayfa tavsiye etmek amacıyla bir ajan geliştirilmektedir. Aktivite kaydedicisi bir web tarayıcısı için geliştirilen Java kavrayıcısı (wrapper) olarak tasarlanmıştır. Bu kaydedici kullanıcının kaydet, yazdır gibi aktivitelerini yakalayacaktır. Veri ambarı için IBM DB2 sistemi kullanılır. Veri madenciliği içinse IBM Intelligent Miner kullanılır.

Hongjun Lu, Qiong Luo, Kiu Shun

Bir Tarayıcının Kullanıcı Taraflı Madencilik ile Genişletilmesi [8]

Sunulan Wbext eklentisi, kullanıcı taraflı madencilik yapma kapasitesine sahiptir. Denetlemesiz öğrenim yöntemi kullanan eklenti Visual C++'ın Browser Helper Object'i kullanılarak yazılmıştır. Kullanıcı aktivitelerini Activity log'da tutar. Takip edilen aktiviteler: sayfa ziyareti, arama başlatma, bağlantı takip etme, favorilere ekleme, yazı seçme, sayfaya odaklanma ve yeni sayfa açma. Kullanıcı bir yandan internette gezinirken sistem eşzamanlı olarak bağlantı tavsiyeleri sunar. Vektör bazlı model kullanılmıştır. Her yeni işlemde önemli özellikler çıkartılarak elde edilir ve bir özellik vektörü oluşturulur. Apriori Algoritmasının değiştirilmiş bir versiyonu ile aktivite verisi üzerinde adapte edilebilir ve genel kuraların(generic rules) madenciliği yapılabilir.

Dimitrios Pierrakos, Georgios Paliouras Christos Papatheodorou, Constantine D. Spyropoulos

KOINOTITES: Kişiselleştirme İçin Bir Ağ Kullanımı Madenciliği Aracı [9]

KOINOTITES, ağ üzerinde kullanıcı toplulukları oluşturmak amacıyla veri madenciliği tekniklerini kullanan bir web kullanımı madenciliği sistemidir. Sunucu log dosyalarını işler ve web sitesindeki bilgileri kullanıcıların kullanım davranışlarına göre gruplandırarak organize eder.

P. Perner, G. Fiss

Ağ Madenciliği, Kişiselleştirme ve Kullanıcıya Uyumlu Arayüzler İle Akıllı E-Ticaret [10]

Bir e-ticaret sitesinden ne tür bilgilerin elde edilebileceği ve bu verilerin deha iyi hizmet, reklam ve satış için nasıl kullanılabilirliği araştırılır. Kullanıcıların tanımlanması için cookie loglarının kullanımı önerilir. Fakat kullanıcının cookie kullanımına izin vermeyebileceği düşünülerek sunucu loglarıyla beraber kullanılır. OPS sayesinde kullanıcı profillerine müşterinin tarayıcısından ulaşılabilir. Sınıflandırma için kullanılan eşleme fonksiyonu (mapping function) karar ağacı tümevarımı (decision tree induction) metoduyla öğrenilir.

Daniel Oberle, Bettina Berendt, Andreas Hotho, Jorge Gonzalez

Kavramsal Kullanıcı Takibi [11]

Ağ kullanımı madenciliğinde kullanılacak log dosyalarının, sitenin temelindeki ontolojiye dayanarak, anlamsal (semantic) olarak hazırlanması amacıyla bir çerçeve önerilir. *SEAL* Programı bir portalla bütünleşik çalışan bir servlet'tir. EM Kümeleme algoritmasının yanında sadece önemsiz birkaç kural çıkarabilen bağlantı kuralı algoritmaları da kullanılmıştır.

Bu alıřmada, kullanıcıların girdiđi sitelerin tmndeki yazı ieriđinin saklandıđı kullanıcı dosyalarının sınıflandırılmasında Naive Bayes Algoritması kullanılmıřtır.

Geliřtirilen program, sunucunun log bilgilerinden hangi kullanıcının hangi web sitelerine gittiđi bilgisini alır ve bu sitelerin ieriđini bir text dosyasına yazar. Naive Bayes Sınıflandırma algoritmasına gre eđitilen program, bu text dosyalarının hangi sınıfa ait olduđunu bulur ve bylece internet kullanıcıları sınıflandırılmıř olur.

2. SİSTEM ANALİZİ VE FİZİBİLİTE ÇALIŞMASI

Uygulama, kullanıcıların girdikleri web sitelerine göre kullanıcıları sınıflandırır. Bütün kullanıcıların bağlanacağı bir vekil (Proxy) sunucuya ihtiyaç vardır. Her kullanıcı önce vekil sunucuya oradan da internete ulaştığı için her kullanıcının hangi siteleri ziyaret ettiği öğrenilebilir. Program, vekil sunucudaki kimin hangi siteleri ziyaret ettiği bilgisini kullanarak, ilgili sitelerle bağlantı kurar ve bu sitelerdeki yazı içeriğini her site için ayrı ayrı oluşturduğu dosyalara aktarır. Bu dosyalar daha sonra sınıflandırılacak ve kullanıcının ne çeşit siteleri takip ettiği belirlenecektir.

Uygulamada kullanılan Naive Bayes algoritmasının eğitim aşamasında her kelimenin önceden belirlenmiş sınıflarda bulunma oranı hesaplanır. Bunun için eğitim dosyalarındaki bütün kelimeleri içeren bir sözlük dosyası oluşturulur. Eğitim dosyaları benim tarafımdan, ilgili konulardaki web sayfalarındaki yazıları bu dosyalara yazmak suretiyle oluşturulmuştur. Örneğin ekonomiyle ilgili bir siteye girilmiş ve bu sitedeki yazılar “Ekonomi.txt” isimli bir dosyaya eklenmiştir.

Sözlük oluşturma aşamasında program, önceden oluşturulan eğitim dosyalarındaki kelimeleri inceler ve her kelime yalnız bir kez kullanılacak şekilde sözlüğe ekler.

Bu uygulamada 8 adet sınıf kullanılmıştır. Bunlar: Biyoloji, Ekonomi, Elektronik, Haber, Network, Spor, Tarih ve Yazılım’dır. İnternet sitelerinin bu 8 sınıftan hangisine ait olduğu hesaplanır.

Sınıflandırma amacıyla Naive Bayes algoritması kullanılacağı için, öğrenme aşamasında, bu algoritmaya uygun şekilde sistem eğitilir. Sözlükteki kelimeler teker teker ele alınarak, önceden belirlenen sınıflara ait olma olasılıkları, o sınıfla ilgili eğitim dosyalarında bulunma frekanslarına göre belirlenir.

Sınıflandırma aşamasındaysa, program kullanıcıların girdiği sitelerin içeriğini sınıflandırır. Her kelimenin hangi sınıfta ne oranda bulunduğu, eğitim aşamasında hesaplanmış olduğundan, bütün kelimelerin bir gruba ait olma olasılıkları çarpılarak,

bütün yazının o gruba hangi oranda ait olduđu hesaplanır. Aitlik hesaplarından hangi sınıfa ait olanı en büyükse, yazı o sınıfa aittir.

2.1. Teknik Fizibilite

Yazılım geliştirme sürecinin teknik fizibilitesi aşağıda incelenmiştir:

2.1.1. Donanım

Sađlıklı bir sınıflandırma için sözlükteki kelime sayısının fazla olması gerekir. Bu amaçla geliştirilen sistemde 20000'den fazla kelime kullanılmıştır ve bu durum sistem üzerine düşen yükü arttırmıştır. Ayrıca Naive Bayes algoritmasıyla hesaplanan aitlik değerleri çok sıfırlı sayılardır ve fazla yer tutarlar. Bu sebeplerden dolayı, programın çalışacağı sunucu bilgisayarda bellek miktarı ve işlemci kapasitesi yüksek olmalıdır. Önerilen donanım özellikleri:

Sunucu Bilgisayar

İşlemci: 1100 Mhz veya üzeri

Bellek: 512 MB SD RAM veya üzeri

İstemci bilgisayarın sahip olması gereken tek özellik, sağlıklı bir şekilde internette gezinmeye izin verecek bir kapasiteye sahip olmasıdır. Bu sebeple istemci bilgisayarın sahip olması tavsiye edilen özellikler:

İstemci Bilgisayar

İşlemci: 486 veya üzeri

Bellek: 64 MB SD RAM veya üzeri

2.1.2. Yazılım

Bu uygulamanın geliştirilmesinde Java platformu tercih edilmiştir. Bu seçimin bir çok avantajları vardır: Java, platform bağımsız bir dil olduğundan, üzerinde çalıştığı işletim sisteminin belirli bir platformda olması gerekmez. Örneğin Microsoft'un .NET platformunu kullanmış olsaydım yazdığım program Linux işletim sistemine sahip bir makinede çalışmayacaktı. Ayrıca kullandığım yazılım geliştirme ortamı NetBeans ücretsiz olduğundan, bu seçimim bana maliyet avantajı da sağladı.

Uygulama Java ile geliştirildiğinden, ve proxy Sunucu olarak kullanılan Apache sunucu da hemen her türlü platformda çalışabildiğinden Sunucu veya istemci bilgisayarlarda kullanılan yazılımların bir önemi yoktur. Bu sebeple, uygulamanın açık kaynak ve ücretsiz olan Linux platformunda kullanılması tavsiye edilir.

Reel sayılarla yapılan işlemlerde, hassasiyetin yüksek olması gerekir. Gerekli hassasiyeti sağlamak için, Java'nın gerektiği zaman virgülden sonra milyonlarca sıfır kullanılmasına izin veren BigDecimal sınıfı kullanılmıştır.

Güvenlik duvarı kullanılıyorsa, bağlantılara izin vermesi için güvenlik duvarının ayarlanması gerekir. Bu ayarların, vekil sunucu kullanımını engellememesi ve kullanıcıların vekil sunucuya, vekil sunucunun da internete bağlanmasına izin vermesi sağlanmalıdır.

Kullanıcı bilgisayarlarının sunucuya bağlanabilmesi için gerekli ağ ayarlarını yapılmalıdır. Örneğin yerel ağ ortamında birden fazla bilgisayar vekil sunucuya bağlanacaksa Router kullanılmalıdır.

Proje geliştirilirken ve yazılım test edilirken kullanılan bilgisayarlardaki yazılımların özellikleri aşağıda verilmiştir:

Sunucu Bilgisayar

İşletim Sistemi: Windows XP

Sunucu: Apache2

IDE: Netbeans 4.0

JSDK: j2sdk1.4.2_06

İstemci Bilgisayar

İşletim Sistemi: Windows 2000

2.2. Gantt Diyagramı

Görevler	Hafta											
	1	2	3	4	5	6	7	8	9	10	11	12
Planlama	■	■										
Sorunların Belirlenmesi		■	■									
Donanım ve Yazılımın Hazırlanması			■	■								
Tasarım				■	■							
Kodlama				■	■	■	■	■	■	■	■	
Test											■	■
Kilometre Taşları												
1. Rapor						X						
2. Rapor									X			
Program gösterimi										X		
3. Rapor												X

Tablo 2-1 Gantt Diyagramı

2.3. Ekonomik Fizibilite

Yazılım geliştirmede kullanılan Sun Microsystems'e ait Netbeans 4.0 IDE'si ve Apache Group'a ait Apache2 Sunucusu ücretsizdir. Uygulama platform bağımsız olduğundan herhangi bir işletim sistemine bağımlı değildir.

Sistem yükü sunucu bilgisayarda olduğundan ve işlemler fazla kaynak kullandığından, sunucu bilgisayarın en az, proje geliştirilen ortamdaki kadar kapasiteye sahip olması önerilir. İstemci bilgisayarların donanım özellikleri önemli değildir.

Sun Netbeans 4.0 IDE: Ücretsiz

Apache 2 Sunucu: Ücretsiz

Donanım: 300 YTL

Kod geliştirme maliyeti: 8 Hafta * 50 YTL = 400 YTL

Toplam: 700 YTL

2.4. Alternatif Fizibilite

Tarayıcının history dizinindeki URL adresleri, RMI, JMS gibi teknolojiler kullanılarak, tek merkeze gönderilebilir ve işlemler bu merkezde yapılabilirdi.

Bir yerel ağda proxy sunucu kullanmak yerine, çeşitli sitelerin bulunduğu sunucular üzerinde çalışılabilirdi. Bu sayede yerel ağın sınırlandırmasından kurtulunmuş olurdu ancak internetteki belirli sunuculara programı yüklemek verimsiz olacağından, programı sadece takip edilmek istenen sitelerin sunucularına yüklemek gerekirdi.

Daha fazla sınıf kullanılarak sınıflandırmadan elde edilecek sonuç daha güvenilir yapılabilirdi.

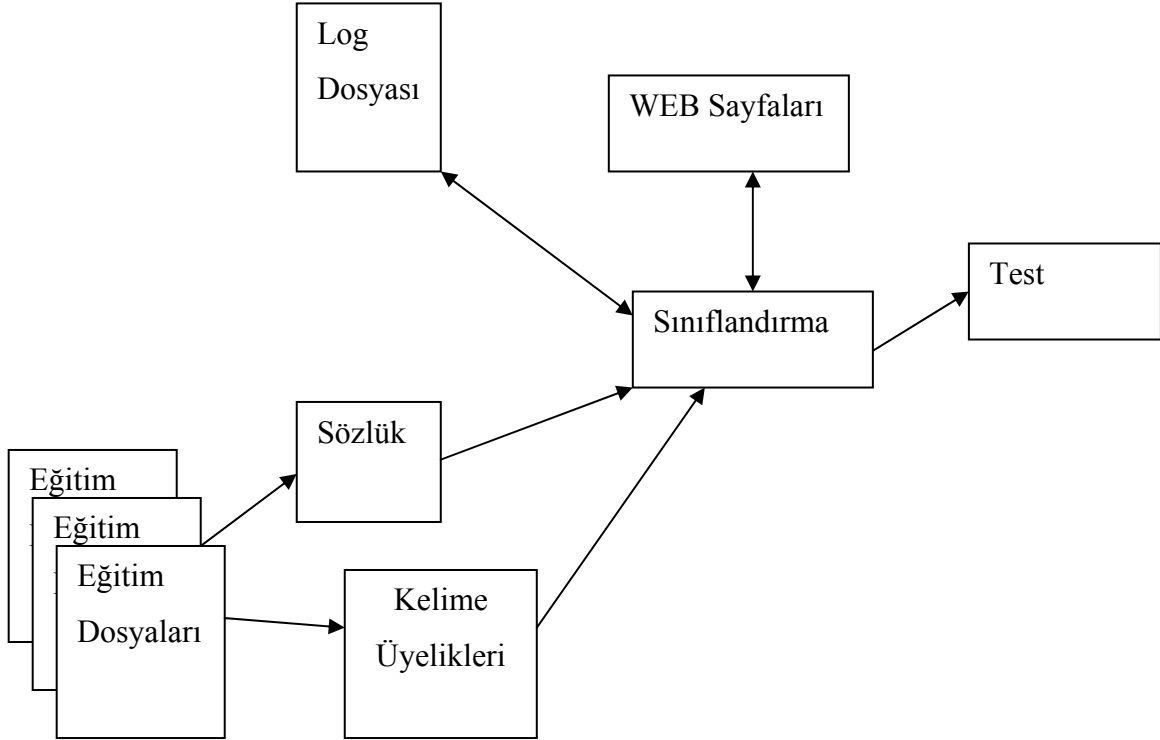
Eđitim dosyalarının statik olarak dıřarıdan verilmesi yerine sistemin dinamik olarak kendi kendine ğrenmesi sađlanabilirdi.

ekim eki alan dosyaların birden fazla kez deđerlendirilmesi nlenebilirdi. rneđin “yapıyorum” ve “yapıyorlar” kelimeleri ayrı ayrı deđerlendirilmek yerine “yap” olarak deđerlendirilebilirdi.

3. AĞ MADENCİLİĞİ YÖNTEMLERİYLE İNTERNET KULLANICILARININ SINIFLANDIRILMASI

Projede öncelikle önceden oluşturulmuş eğitim dosyalarındaki bilgiler kullanılarak eğitim dosyalarındaki her kelimenin bulunduğu sözlük dosyası ve sözlük dosyasındaki her kelimenin hangi gruba ne oranda ait olduğu bilgisini içeren Kelime Üyelikleri dosyası oluşturulur. Sunucudaki internet erişimi log dosyasından bir kullanıcının hangi siteleri gezdiği bilgisi alınır ve bu sitelere bağlanır. Bu sitelerdeki yazı içerikleri her site için ayrı bir dosyada saklanır. Sınıflandırma aşamasında bu dosyalar Naïve Bayes yöntemiyle sınıflandırılarak, hangi kullanıcının ne türdeki siteleri ziyaret ettiği bilgisi oluşturulur ve istatistikler grafik ile gösterilir.

Projenin işleyişiyle ilgili temel blok diyagramı Şekil 3-1’de verilmiştir:



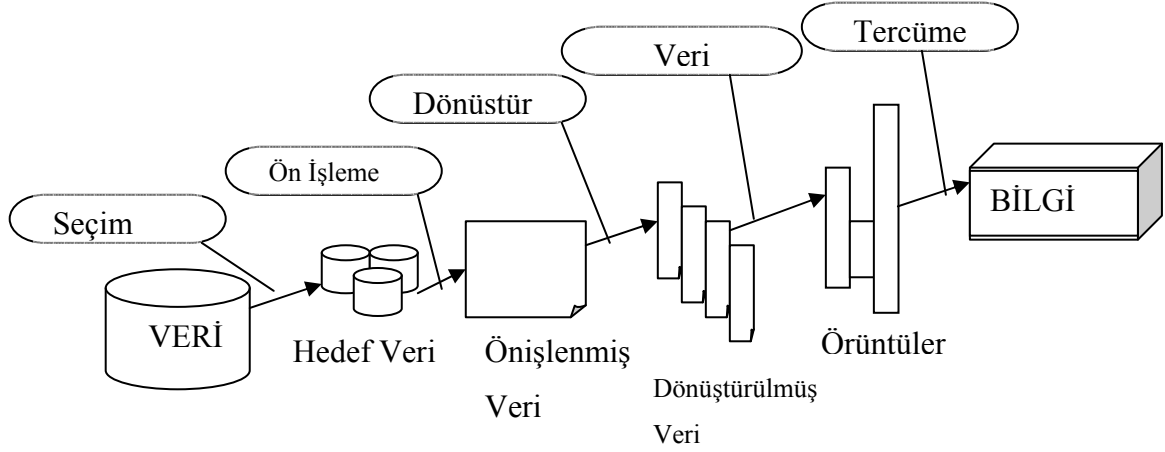
Şekil 3-1 Ağ Madenciliği Yöntemleriyle İnternet Kullanıcılarının Sınıflandırılması Temel Blok Diyagramı

3.1. Veri Madenciliği Süreci

Bir web sitesinde üç çeşit bilgi yönetilir: içerik, yapı ve log bilgisi. İçerik bilgisi sitede mevcut olan bilgiden oluşur. Yapı bilgisi, içeriğin organizasyonu ve siteler arası bağlantılarla ilgilidir. Kullanım bilgisi ya da log bilgisi sitelerin kullanım örüntüleridir (pattern). Bu bilgi kümelerine veri madenciliği tekniklerinin uygulanmasıyla ağ madenciliğinin üç temel araştırma alanı oluşmuştur.

Bu projenin konusu ağ kullanımı madenciliği ve ağ içeriği madenciliği alanlarıdır. Veri madenciliği için tanımlanan esas aşamalar (bkz. Şekil 3-2) ağ kullanımı madenciliği için de geçerlidir: Sunucu veya kullanıcı tarafından verinin toplanması. Verinin ön işlenmesi. Verinin temizlenmesini, kullanıcının tanınmasını ve oturumun tanınmasını içerir. Örüntü tanıma. Kümeleme, sınıflandırma ilgi kuralı keşfetme gibi bilgisayar öğrenmesi teknikleriyle bilgi elde edilir. Bilgi üst işleme (post processing). Elde edilen bilgiler insan anlayışına uygun şekilde temsil edilir.

Şu anda ağ kullanımı madenciliği konusuyla ilgi araştırmaların çoğu sunucu taraflıdır. Genel amaç web sitesinin servisini geliştirmek ve sunucu performansını arttırmaktır. Ağ kullanımı madenciliği için kullanılan ana kaynak sunucu loglarıdır. Ağ kullanımı madenciliği URL örüntülerinin yanında olay örüntüleriyle de ilgilenir. Ağ kullanımı ve ağ yapısı madenciliğinden farklı olarak ağ içeriği madenciliği, siteler arası bağlantılarla da (link) ilgilenir. Ağ üzerindeki veriler düzensiz yapıda olduklarından içerik madenciliği karmaşıktır.



Şekil 3-2 Veri Madenciliği Süreci [15]

3.2. Sunucu Tarafındaki Log Dosyasının Değerlendirilmesi

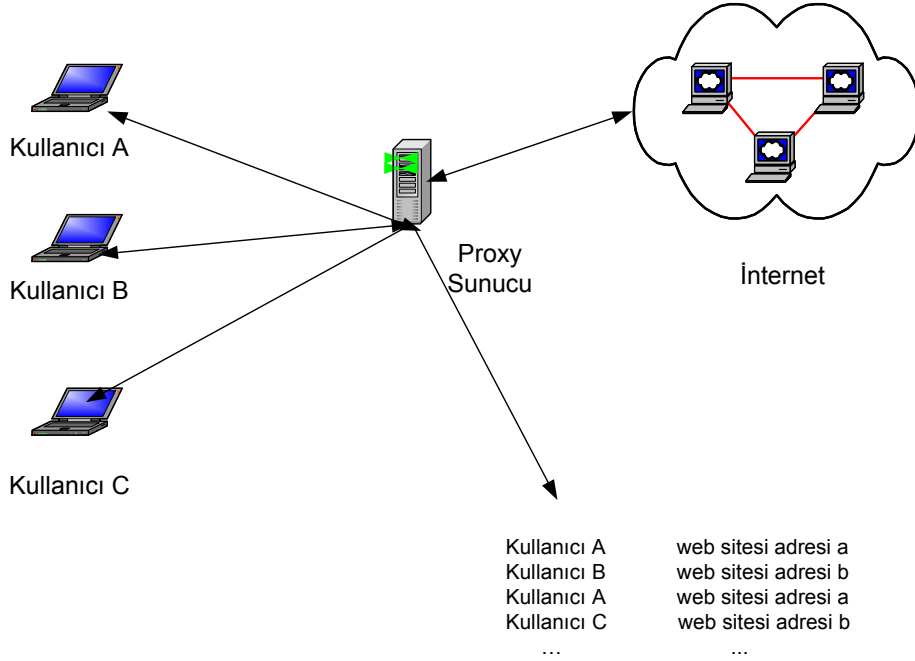
Uygulamada öncelikle sunucu tarafındaki log dosyasından ilgilendiğimiz bilgiler okunur. Bize gereken bilgiler sadece IP adresi ve URL adresidir ve . Log dosyasındaki internet erişim bilgileri şu şekilde saklanır:

```
127.0.0.1 - - [05/Apr/2005:11:52:28 +0300] "GET http://www.milliyet.com.tr/" 200
1494
```

```
127.0.0.2 - - [05/Apr/2005:11:52:28 +0300] "GET http://www.fazlamesai.net/" 200
2326
```

```
127.0.0.1 - - [05/Apr/2005:11:52:29 +0300] "GET http://ce.yildiz.edu.tr/" 404 283
```

Uygulama sırasında, örneğin 127.0.0.1 ile http://ce.yildiz.edu.tr bilgileri iki boyutlu bir dizide saklanır. Program daha sonra http://ce.yildiz.edu.tr adresine bağlanır ve bu adres için bir dosya oluşturur. Bu sitedeki yazı içeriği bu siteyle ilgili dosyaya aktarılır. Sınıflandırma aşamasında bu site sınıflandırılarak, 127.0.0.1 IP'li kullanıcının ne türde bir web sitesini ziyaret etmiş olduğu belirlenir.



Şekil 3-3 Her kullanıcıya ait site bilgilerin diziye alınması

3.3. Naive Bayes Yöntemi

Naive Bayes kolay uygulanabilir olduğu kadar üstün performansıylada metin sınıflandırma çalışmalarında en çok kullanılan metotlardan biri haline gelmiştir. Metodda önce tüm eğitim verisindeki metinlerde kullanılan kelimelerden bir sözlük oluşturulur. Daha sonra her bir kelimenin her bir sınıftaki tekrar sayıları(frekansı) bulunur. Sınıflandırılması istenen yeni bir metin önceden geldiğinde oluşturulan sözlükte var olan kelimelerin her bir sınıftaki frekansları bulunur. Bir metnin C sınıfına dahil olma olasılığı C sınıfının eğitim setindeki oranıyla, metnin içindeki her bir kelimenin C sınıfına ait olma olasılıkları çarpılarak bulunur.[16]

3.3.1. Öğrenme

Örnekler, hangi gruba ait oldukları belli olan eğitim dosyalarıdır. V_j , ait olunması muhtemel gruplardır. $P(w_k | v_j)$: w_k kelimesinin v_j sınıfına ait olma ihtimalidir.

1. Örneklerdeki bütün kelimeleri, yalnızca bir kez bulunacak şekilde sözlüğe aktar.
2. Her grup için

Aynı gruba ait eğitim dosyalarını birleştir

n = birleştirilen dosyalardaki kelime sayısı

sözlükteki her kelime için

n_k = kelimenin eğitim dosyasında kaç kez geçtiği

voc_cnt = sözlükteki kelime sayısı

$P(w_k | v_j) = (n_k + 1) / (n + voc_cnt)$

3.3.2 Sınıflandırma

D , Kullanıcının gezdiği sitelerdeki içeriğe sahip kullanıcı dosyasıdır. w_i : Bu dosyadaki i . kelimedir. $P(w_i | C)$: i . kelimenin C sınıfına ait olma olasılığıdır. D dokümanının C sınıfına ait olma olasılığı olan $p(D | C)$ şöyle bulunur:

$$p(D|C) = \prod_i p(w_i|C)$$

3.4. Sözlük Oluşturulması

Sözlük, seçilen eğitim dosyalarındaki kelimelerin sadece bir kez kullanılmasıyla oluşturulur. Bu sebeple, eğitim dosyalarında karşılaşılan her kelime için bu kelimenin sözlükte olup olmadığı kontrol edilir. Eğer yoksa sözlüğe eklenir.

3.4.1. Pseudocode

1'den sınıf sayısına kadar

1'den bu sınıfa ait dosya sayısına sayısına kadar

eđitim dosyasından satır oku

satırını token'lerine ayır

hala token varken

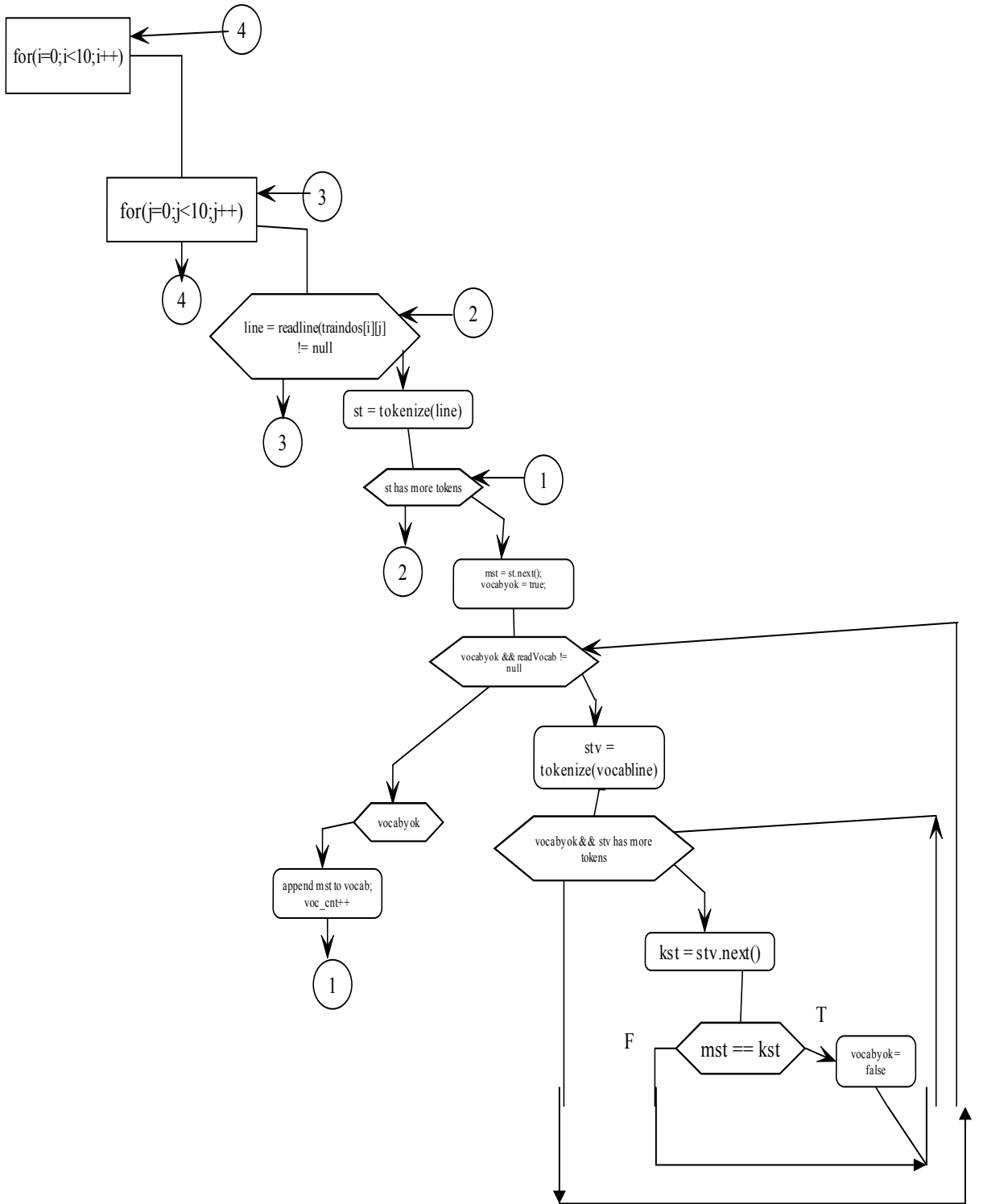
token sözlükte yokken ve sözlük tamamen okunmamışken

sözlük token'i ile eğitim dosyası token'i eşitse kelime sözlükte var

kelime sözlükte yoksa sözlüğe ekle

3.4.2. Akış Diyagramı

i: sınıf sayısı, j: her sınıfa ait eğitim dosyası adedidir. Eğitim dosyalarındaki kelimeler teker teker alınır ve bu kelimenin sözlükte olup olmadığı kontrol edilir. Bulunamazsa, kelime sözlüğe eklenir ve sözlükteki kelime sayısı arttırılır. (bkz. Şekil 3-4)



Şekil 3-4 Sözlük Oluşturma Akış Diyagramı

3.5. Öğrenme

Öğrenme aşamasında sözlükteki her kelimenin, önceden belirlenmiş sınıflarda hangi oranda bulunduğu belirlenir. Her kelimenin her gruba ne oranda ait olduğu hesaplanır.

3.5.1. Pseudocode

1'den sınıf sayısına kadar

1'den bu sınıfa ait dosya sayısına kadar

eğitim dosyalarını birbirine ekle

birleşmiş dosyanın ilk satırını oku

sözlükten okunan satır null değilken

sözlük satırını token'lerine ayır

hala token varken

eğitim dosyasından okunan satırı token'lere ayır.

hala token varken

iki token birbirine eşitse eğitim dosyasındaki kelimenin sözlükte bulunma

sayısını arttır

sözlükteki kelimenin bu gruptan olma olasılığını hesapla

3.5.2. Akış Diyagramı

i : sınıf sayısı, j : her sınıfa ait eğitim dosyası adedidir. Önce, ayrı eğitim dosyaları, her sınıfa ait sadece bir dosya olacak şekilde birleştirilir.

Sözlükteki her kelime teker teker ele alınarak, sınıflara ait eğitim dosyalarında bu kelimenin kaç kez geçtiği hesaplanır.

Oluşturulan `vocamArray` dizisi, her kelime için, o kelimenin sınıflarda bulunma katsayısını tutar. Sözlükteki kelime sayısı kadar satıra ve sınıf sayısı kadar sütuna sahiptir.

Naive Bayes algoritmasına uygun şekilde, katsayılar hesaplanır. (bkz. Şekil 3-5)

3.6. Sınıflandırma

Sınıflandırma aşamasında, her kullanıcıya ait, o kullanıcının bağlandığı sitelerden elde edilmiş yazı içeriğinden oluşan dosyalardaki kelimeler teker teker ele alınır. İncelenen kelimenin sözlükte bulunup bulunmadığına bakılır. Naive Bayes algoritmasına uygun şekilde üyelikler hesaplanır. İşlemler sonucunda, kullanıcının aidiyet katsayısının en büyük olduğu sınıf, kullanıcının sınıfıdır. (bkz. Şekil 3-6)

3.6.1. Pseudocode

```

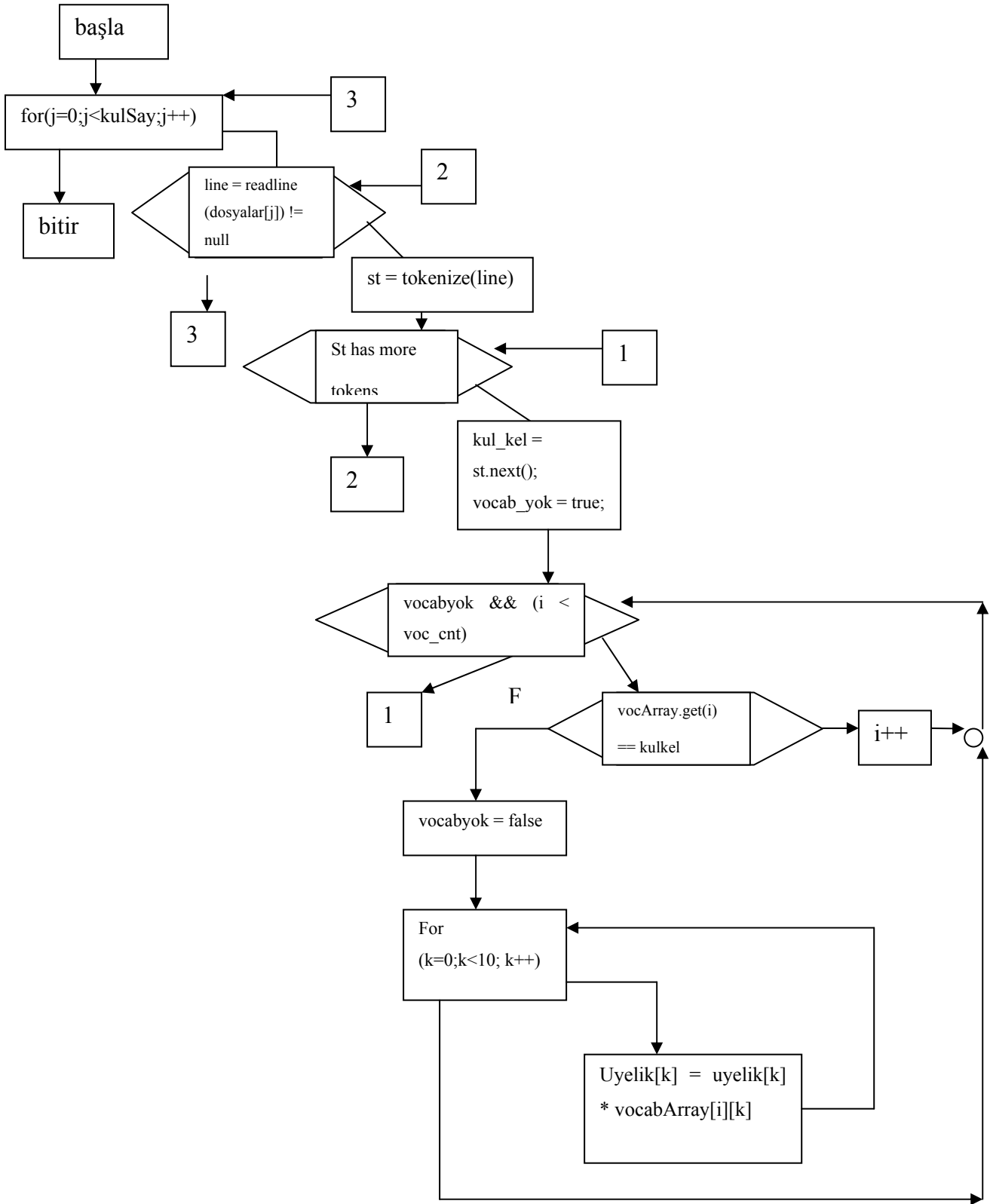
1'den kullanıcı sayısına kadar
  sınıflandırılacak dosyadan bir satır oku
  satırı token'lerine ayır
  hala token varken
    kelime sözlükte varsa
      1'den grup sayısına kadar
        sınıflandırılacak dosyanın grup üyeliklerini hesapla

```

3.6.2. Akış Diyagramı

Sınıflandırma işleminin akış diyagramı Şekil 3-6'da verilmiştir.

vocArray (vocabArray'den farklı olarak) sözlük oluşturma aşamasında oluşturulan bir vektördür. Sözlükteki bütün kelimeleri içerir kulGrup dizisi, kullanıcı sayısı kadar satırdan ve grup sayısı kadar sütundan oluşmuştur ve kullanıcıların sınıflara aitlik katsayılarını içerir. uyelik[k]: bir dosyanın k. Sınıfa ait olma olasılığıdır. Dosyalar[j], sınıflandırılacak j. Dosyadır.



Şekil 3-6 Sınıflandırma Akış Diyagramı

4. UYGULAMA

Bu uygulamada 8 adet sınıf (Ekonomi, Haber, Spor, Yazılım, Elektronik, Network, Tarih, Biyoloji) kullanılmıştır. Eğitim dosyaları WEB üzerinden ilgili sayfalar bulunarak buradaki içeriğin yaklaşık 15-20 KB büyüklüğündeki 4-5 adet ayrı metin dosyasına aktarılmasıyla oluşturulmuştur. Bu eğitim dosyalarından oluşturulan sözlükte 20000'den fazla Türkçe kelime mevcuttur. Ve, neden, ile, ki, dahi, için, çünkü, fakat kelimeleri sözlüğe dahil edilmemiştir.

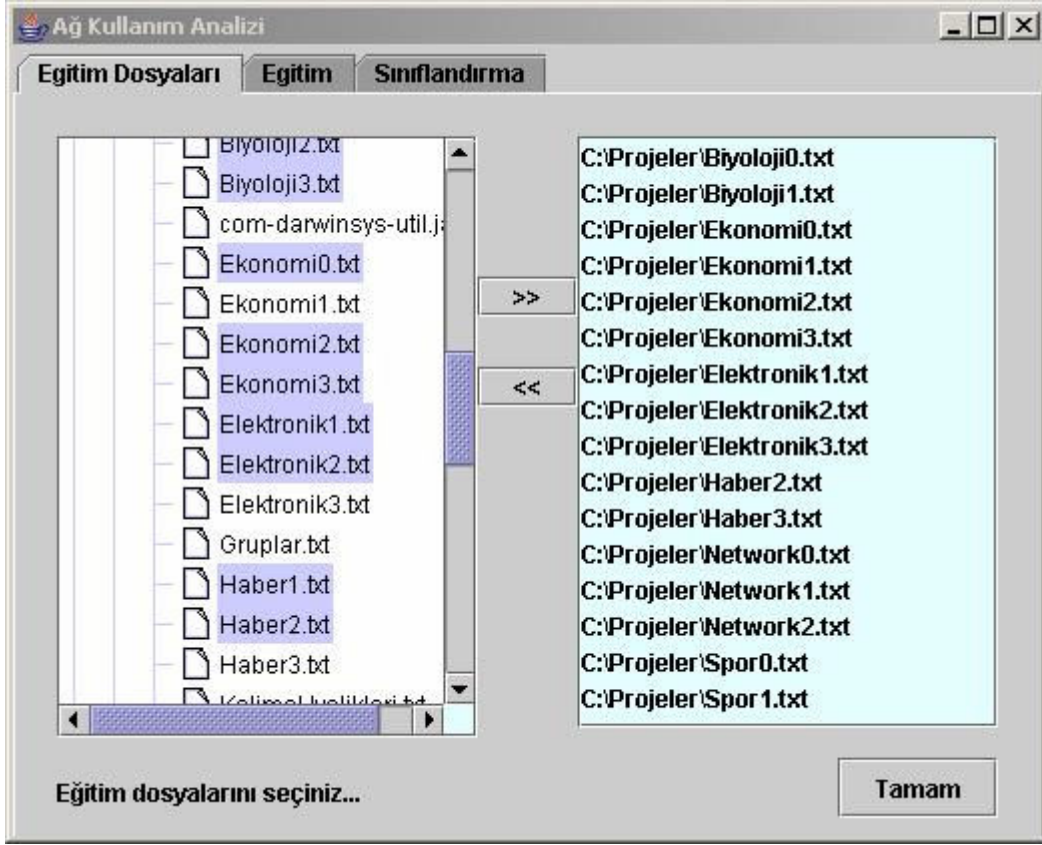
Yazılımı test etmek amacıyla Apache erişim logu formatında, ilgi alanları birbirinden farklı dört kişiye ait log dosyası oluşturulmuştur.

4.1. Ekran Görüntüleri

Bu bölümde programın ekran çıktıları yorumlanacaktır.

4.1.1. Eğitim Dosyalarının Seçilmesi

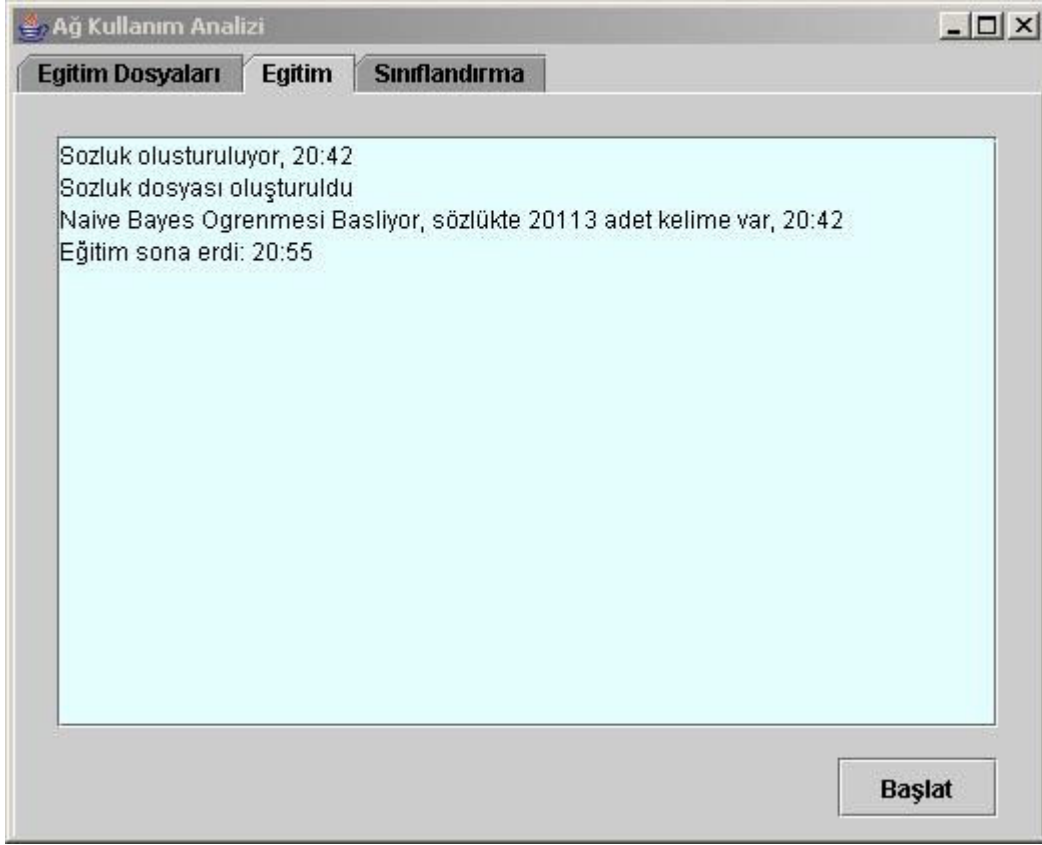
Şekil 4-1'de Soldaki bölüm, kullanıcının bilgisayarından istediği eğitim dosyalarını seçmesine izin verir. Bu bölümde seçilen dosyalar belirli bir formata sahip olmalıdır. Eğer seçilen dosya büyük harfle başlayan bir kelimedenden sonra gelen bir sayıdan oluşmuş bir isme sahip değilse hata mesajı çıkar. Dosyalar böyle bir formata sahip olmalıdır çünkü program, sistemdeki sınıfların isimlerini bu dosya isimlerinden öğrenir. Örneğin Elektronik2.txt dosyası Elektronik sınıfına aittir.



Şekil 4-1 Eğitim Dosyalarının Seçilmesi Ekran Görüntüsü

4.1.2. Eğitim

Eğitim için Ekonomi, Haber, Spor, Yazılım, Elektronik, Network, Tarih ve Biyoloji sınıflarına ait dörder adet yaklaşık 17KB'lık text dosyaları kullanılmıştır. Bu dosyalardan oluşturulan sözlükte 20113 adet kelime vardır. Bu kelimeler arasında Ve, neden, ile, ki, dahi, için, çünkü, fakat yoktur. Sözlükteki bütün harfler küçük harflidir ve sınıflandırılacak dosyalardaki kelimeler de sözlüğe bakılmadan önce küçük harfli hale getirilir. Eğitim 13 dakika sürmüştür.



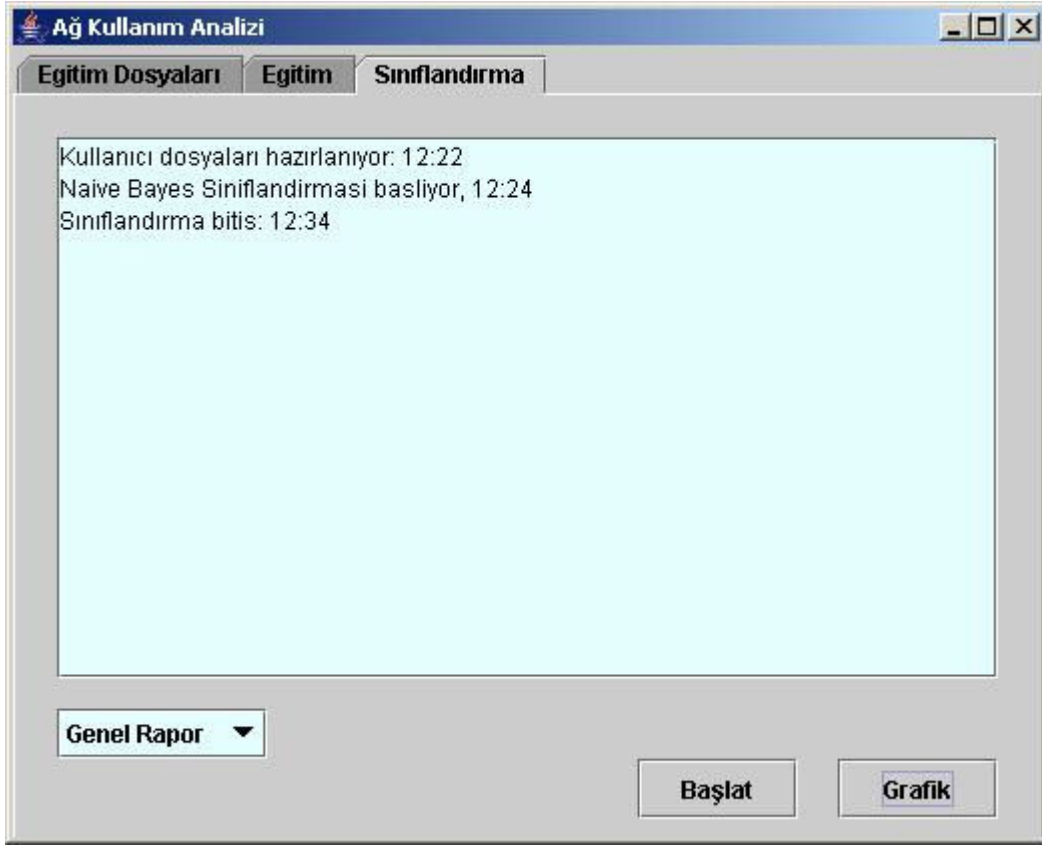
Şekil 4-2 Eğitim Ekran Görüntüsü

4.1.3. Sınıflandırma

Sınıflandırma aşamasında 5 adet kullanıcı tarafından ziyaret edilmiş 20 farklı web sitesi sınıflandırılmıştır. Sınıflandırma sonucunda 2 sitede yanlış sınıflandırma oluşmuştur. Bunlardan ilkinde haber sınıfında değerlendirilmesi gereken bir sitenin içeriği spor olarak belirlenmiştir. Öteki sitenin içeriği ise 5 kelimelik bir reklam mesajıdır ve yine spor sınıfında değerlendirilmiştir. Sonuç olarak %90 başarı ile kullanıcıların ziyaret ettiği siteler sınıflandırılmıştır.

Arayüzün alt tarafındaki seçeneklerden aylık rapor veya genel rapor seçeneği seçilebilir. Genel rapor seçildiğinde, internet erişimi log dosyasının tamamını değerlendirirken, aylık rapor seçildiğinde sadece o aya ait erişim bilgileri dikkate alınır.

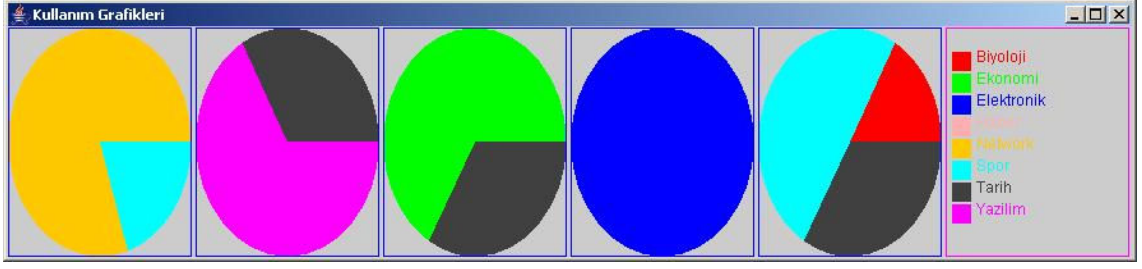
Grafik düğmesine basılırsa kullanıcılar hakkında istatistiksel bilgi veren bir pencere açılır.



Şekil 4-3 Sınıflandırma Ekran Görüntüsü

4.1.4. Grafik Gösterimi

Bu aşamada, hangi kullanıcının ne tür siteleri ziyaret ettiği bilgisi grafik yardımıyla gösterilmektedir. Grafiklerin üzerine fare ile gelindiğinde hangi IP adresli kullanıcıya ait olduğu bilgisi görülebilir. Her Kullanıcı için ayrı grafik oluşturulmuştur. En sağdaki karede ise hangi rengin hangi sınıfı temsil ettiği bilgisi yer almaktadır.



Şekil 4-4 Grafik Gösterimi Ekran Görüntüsü

5. SONUÇ

Bu uygulamada bir yerel ađ ortamındaki internet kullanıcılarının takip ettikleri web siteleri sınıflandırılarak, kullanıcıların internette ne tür web sitelerini takip ettikleri bilgisine ulaşılabildiği sağlandı. Sınıflandırma işlemi için Naïve Bayes algoritması kullanıldı.

Gelecek çalışmalarda bu programın eğitim dosyalarını kendi kendine oluşturmasını sağlayacak bir yöntem geliştirilebilir. Web tarayıcılarındaki cache bellek kullanımının da dikkate alınması sağlanabilir. Arayüz daha iyi hale getirilebilir.

Sistem yerel ađlardaki internet kullanımı hakkında fikir edinilebilmesini sağladığı için kontrol ve verim artımı açısından oldukça faydalıdır. Naïve Bayes yönteminin güvenilirliği ve uygulamada elde edilen %90 başarı sayesinde yeterli güvenilirliğe sahip olduğunu göstermiştir.

6. SÖZLÜK

Ağ Madenciliği (Web Mining): Ağdaki veriler arasından istenilen bilgiyi elde etmek için veri madenciliği tekniklerinin kullanımı

Ağ Kullanımı Madenciliği (Web Usage Mining): Web sitelerinin içeriklerinin ve yapılarının, kullanıcıların ilgisini çekecek bilgileri onlara sağlamak amacıyla, kullanım bilgilerinden yola çıkılarak düzenlenmesi ve işlenmesi. Sunucu kullanıcı arası işlemlerden oluşan veriden anlamlı örüntüler çıkartma

Ağ Kişiselleştirilmesi (Web Personalization): Bir web sitesinin her değişik kullanıcının veya kullanıcı gruplarının ihtiyaçlarına göre düzenlenmesi süreci

Oturum (Session): Bir kullanıcının bir ziyarette web sitesinden isteklerinin tümüdür.

Akıllı Ajan (Intelligent Agent): İnternet ajanı olarak da adlandırılan bu varlıkların kullanım amacı ağ üzerinden otomatik olarak veri elde etmektir. Genellikle cookie aracılığıyla bilgi edinirler.

Naive Bayes: Olasılık sınıflandırılmasında kullanılır. Büyük miktarda eğitim kümesine ihtiyaç duymaz.

Markov Modelleri: Genellikle web üzerindeki kullanıcı isteklerinin modellenmesi amacıyla kullanılırlar.

Pekiştirme Yoluyla Öğrenim (Reinforcement Learning): Ödül ve cezalandırma yoluyla en uygun kararın verilmesidir.

Denetleme Yoluyla Öğrenim (Supervised Learning): Öğrenen varlığa belirli bir durum için yapılması gereken hareket söylenmez. Seçilen hareketin ne kadar iyi ya da kötü olduğu söylenir.

Denetlemesiz Öğrenim (Unsupervised Learning): Kullanıcıdan ön bilgi istenmez. Sistem kullanıcıyı izler, davranışları öğrenir ve kendini duruma adapte eder

Küme veya Salkım (Cluster): Bir tek işlem kapasitesi oluşturmak için çalışan tamamlanmış sistemler grubu

Sınıflandırma (Classification): Bir veri nesnesini daha önceden belirlenmiş sınıflardan birine eşlemek için kullanılan teknik

Kümelendirme (Clustering): Benzer özelliklere sahip kullanıcıları veya veri nesnelerini aynı gruba koymak için kullanılan teknik

Örüntü Tanıma (Pattern Recognition): Nesnelerin belirli özelliklerine göre, bütünden ayrılarak tanımlanması

İlişki Kuralları (Association Rules): Önceden belirlenmiş bir eşik değerinden daha yüksek bir destek (support) değerine sahip sayfalar kümesine işaret eder

7. KAYNAKLAR

1. A. McCallum, K. Nigam, J. Rennie, K. Seymore, Building Domain-Specific Search Engines with Machine Learning Techniques
2. J. Borges, An Average Linear Time Algorithm for Web Usage Mining
3. M. Albanese, A. Picariello, C ve L Sansone, A Web Personalization System Base on Web Usage Mining Techniques
4. D. Poo, B. Chng, J. Goh, A Hybrid Approach for User Profiling 1
5. M. Craven, J. Kumlien, Constructing Biological Knowledge Bases by Extracting Information from Text Sources
6. S. Paulakis, C. Lampos, M. Eirinaki, M. Vazirgiannis, SEWeP: A Web Mining System supporting Semantic Personalization
7. Y. Fu Ming, Y. Shih, A Framework for Personal Web Usage Mining
8. H. Lu, Q. Luo, K. Shun, Extending a Web Browser with Client-Side Mining
9. D. Pierrakos, G. P. C. Papatheodorou, C. D. Spyropoulos, KOINOTITES: A Web Usage Mining Tool for Personalization
10. P. Perner, G. Fiss, Intelligent E-Marketing with Web Mining, Personalization and User-adpated Interfaces
11. D. Oberle, B. Berendt, A. Hotho, J. Gonzalez, Conceptual User Tracking
12. T. Loton, Web Content Mining with Java, John Wiley & Sons, 2002

13. I. Darwin, Java Cookbook, O'Reilly, 2001
14. T. M. Mitchell, Machine Learning, McGraw-Hill, 1997
15. J. Srivastava , Web Mining: Accomplishments & Future Directions, University of Minnesota
16. F. Amasyalı, Otomatik Haber Metinleri Sınıflandırma

ÖZGEÇMİŞ

Ad Soyad : Eren Aykın
Doğum Tarih : 21 Haziran 1983
Doğum Yeri : İstanbul
Lise : Kdz. Ereğli Anadolu Lisesi
Staj Yaptığı Yerler : Microsoft Türkiye
İSDEMİR AŞ
Mikrosay Yazılım AŞ