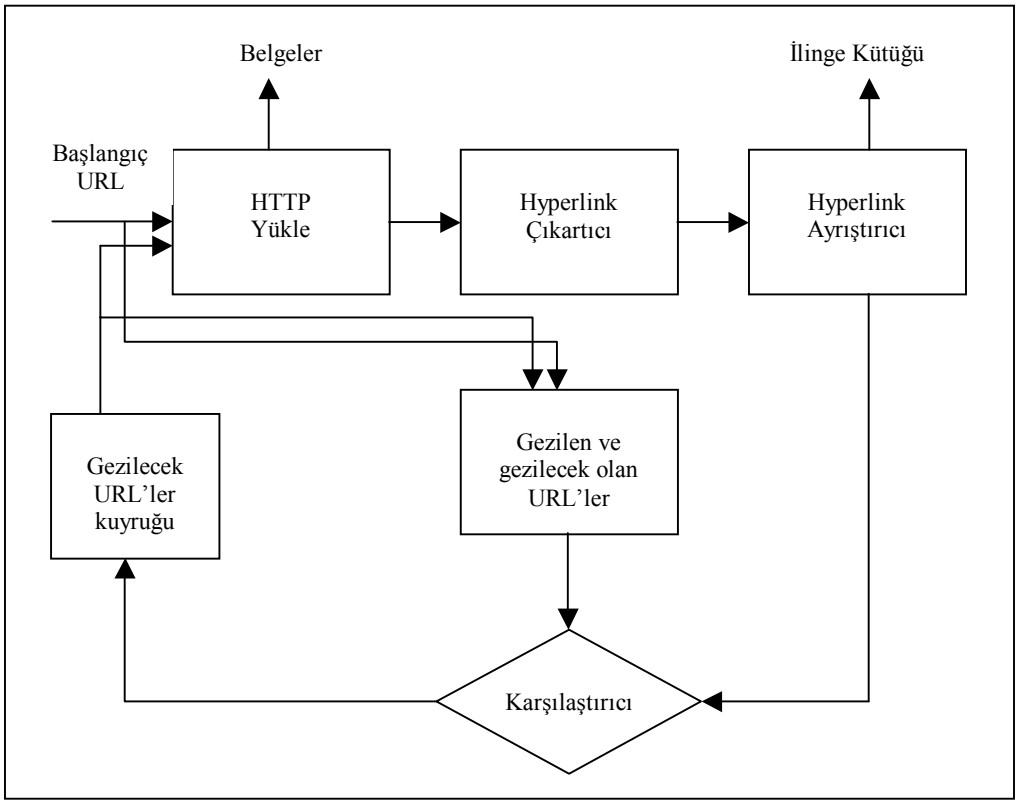
3 ARAMA MOTORLARI

Arama motorları, son elli yıldır geliştirilmekte olan bilgi erişim sistemlerini temel almaktadırlar. Bununla birlikte, arama motorları gerek mimari açıdan gerekse işlevsel özellikleri açısından bilgi erişim sistemlerinden farklılıklar gösterir. Bu bölümde arama motorları hakkında hem mimari hem de işlevsel açıdan tanımlayıcı bilgiler verilmekte ve geleneksel bilgi erişim sistemleriyle arama motorları arasındaki farklı yönlere dikkat çekilerek, konuyla ilgili literatür kısaca incelenmektedir.

3.1 Mimari Yapı

Arama motorlarının esas bileşenlerden birisi, Web üzerindeki herhangi bir sitenin yerel diske indirilmesini sağlayan ağ sörfçüsü (*network surfer*) işlevini gören bir robottur (web crawler, spider). Tipik bir robotun genel görünümü Şekil 2'de verilmiştir.



Şekil 2. Robotun işlevsel görünümü

Robot, hafif bağlı üç alt modülden oluşmaktadır: *yükleyici* (*downloader*), *çıkartıcı* (*extractor*) ve *ayrıştırıcı* (*parser*). Yükleyici, bir başlangıç adresi (*seed node, root node*) ile çalıştırılır.[1] Yükleyici gezilecek URL'ler *(*Uniform Resource Locator*)* kuyruğundaki adresleri önce-enlemesine (breadth-first) dolaşmaya başlar. Yükleyici dolaştığı belgeleri HTTP protokolünü kullanarak getirir ve getirdiği belgeleri hiper-bağlantı çıkartıcıya iletir. Yanlış ya da geçersiz bağlantıların tespit edilmesi, bu bağlantıların ziyaret edilmek istenmesi durumunda, Web sunucusundan gelen hata iletileri yardımıyla yapılır. Çıkartıcı, gelen belge içerisinde <*a href*="..."> ya da <*frame src="..."*> biçimindeki HTML (Graham, 1997) takılarını araştırır. Ayrıştırıcı, göreli URL'leri tam URL adreslerine çevirir. Belge içindeki tam URL adresleri için herhangi bir işlem yapılmaz. Ayrıştırıcıdan gelen URL adresi

[1] Web alanı düzeyinde yeni çekirdek (seed) adreslerin nasıl elde edildiği bilgisi literatür taraması esnasında elde edilememiştir. Fakat bu konudaki güvenilir çözüm, herhangi bir sitedeki Web sayfaları işlenirken elde edilen site dışı hiper-bağlantıların bir veri tabanında sonradan işlenmek üzere ayrık (distinct) olarak saklanmasıdır. Arama motorları Internet'teki yeni sayfaları robotların sörfü esnasında tespit edebildiği gibi, bu sayfaların yaratıcıları arama motoruyla bağlantı kurup, arama motorunu sayfadan haberdar edebilmektedirler (Gordon ve Pathak, 1999). Arama motorları genelde eldeki siteleri düzenli aralıklarla ziyaret edip her site ile ilgili değişiklikleri tespit eder (Sullivan, 2000).

gezilecek URL'ler kuyruğuna atılmadan önce bir döngüye girilip kilitlenme durumu oluşmasın diye o ana kadar gezilen URL'ler ile karşılaştırılır. Bu aşamada ayrıca gezilecek URL değerlerinin başlangıç olarak verilen URL ile aynı alan adını (domain name) taşıyıp taşımadığı da kontrol edilir. Aynı alan adını taşımayan URL'ler ziyaret edilmez. Böylece robotun sadece istenen bir sitedeki belgeleri getirmesi sağlanmış olur. İşletimin sonunda, getirilen belgeler yerel olarak depolanır. Ayıca, verilen başlangıç URL değeri için dolaşılan URL'lerin listesi ve onların ilinge (topology) bilgisi çıktı olarak verilmektedir. İlinge bilgisi, getirilen belgede referans verilen tam URL adreslerinin listesini içermektedir. Bu tür ilingesel bilgiler bir elektronik katalogda dizinlenecek sayfaların tespitinde doğrudan kullanılmaktadır (Deogun, Sever ve Raghavan, 1998).

Robotun bulduğu her şey, arama motorlarının ikinci bileşeni olan "veri tabanı"na kaydedilir. Arama motorunun diğer bir bileşeni ise "ajan" olarak adlandırılan arama motoru yazılımıdır. Bu yazılım, dizinde kayıtlı olan milyonlarca sayfa içinden en ilgili olduğunu "düşündüğü" siteleri eleyerek bunları (genelde) ilgililik derecelerine göre sıralar (Sullivan, 2001).

Web robotları basit programlar olmasına rağmen Web üzerinde bulunan milyonlarca dokümanı kullanıcıların hizmetine sunmak ve aranan bilgiye kolay ve doğru bir şekilde erişilmesini sağlamak amacıyla çalışmaktadırlar. Hatta zaman zaman site sahibinin saklı tuttuğu materyalleri de otomatik ve hızlı bir şekilde keşfedebilmektedirler. Bu yüzden birçok robot gayri resmi "robotları dışlama protokolü"ne (robots exclusion protocol) göre belirlenmiş kurallar kümesi dahilinde hareket etmek zorunda kalmaktadır.

İsimleri "AbachoBOT"'dan "ZyBorg"'a kadar değişen bu robotlar tüm popüler arama motorları tarafından kullanılmaktadırlar. Örneğin; Inktomi "Slurp", AltaVista "Scooter", Google ise "Googlebot" robotlarını kullanmaktadır. Bazı arama motorları değişik amaçlar için birden fazla robot da kullanmaktadır (örneğin, yeni sayfaları bulmak için bir robot, sayfa bağlantılarını kontrol etmek için başka bir robot şeklinde). Ama bu robotların tümü arama motorları için çalışmamaktadır. Kimi robotlar sayfa bağlantılarının canlı olup olmadığını kontrol etmekte (link checker), kimisi sayfa değişimini denetlemekte (page change monitors), kimisi sayfanın HTML kodunun doğruluğunu ve standartlara uyumluluğunu kontrol etmekte (validators), kimisi FTP istemcisi (FTP client) olarak indirilecek olan dosyaların yönetiminde, kimisi de sayfa ziyaretlerinde (web browser) kullanılmaktadır.

3.2 Dizinleme

Dizinlemede ve ilgili belgeleri saklamada arama motorlarının karşılaştıkları tipik sorunlar ile bilgi erişim sistemlerinin çözmesi gereken sorunlar birbirinden farklıdır. Bu tür sorunlar, ki bu alt bölümün gerisinde işlenecektir, çoğunlukla değişik ve kendine özgü çözümleri gerekli kılar.

Bilgi erişim sistemlerinde dizinlenecek belgeler durağandır (statik). Başka bir deyişle, bir belge bir defa dizinlendikten sonra bir daha dizinleme işlemine tabi tutulmaz. Halbuki Web kaynakları tahmini olarak ortalama 75 gün değişmeden kalmaktadırlar (Brake, 2001). Kahle, yapılan tahminlere göre Web kaynaklarının %40'ının her ay değiştiğini (Kahle, 1996), Internet ortamındaki bir bağlantının (link) ortalama ömrünün 44 gün olduğunu belirtmektedir (Kahle, 1997).[2] Web'deki belgelerden örneklem seçilerek yapılan ve 120 hafta süren "uzunlamasına" bir araştırmada bir Web sayfasının ya da bir Web sitesinin "yarı ömrü"nün (half-file) iki yıl civarında olduğu bulunmuş ve Web sayfası/sitesi içeriğinin bir yıllık bir sürede değiştiği saptanmıştır (Koehler, 1999). Bu tür bilgiler, arama motorlarının mimarisinin bilgi erişim sistemlerininkine göre elbette farklı olmasını gerektirmektedir. Örneğin, daha önce dizinlenmiş kaynaklarda belirli aralıklarla günleme yapılabilir (HTTP protokolü bu tür kararları destekleyen sorgulara olanak vermektedir). Günleme yapılacaksa ilgili kaynağı yeniden dizinleyen ve eskisinin yerine yerleştiren bir robot modülün belirli aralıklarla, mevcut veri tabanının belirli bir kısmını rastgele denetleyerek işletilebilir. Aslında, Internet'in üssel olarak büyümesi (her sekiz ayda bir bilgi hacminin ikiye katlanma eğilimi göstermesi) ve Internet kaynaklarının sık sık değiştirilmesi mimariyi daha karmaşık hale getirdiği gibi bilinen arama motorlarının toplam Web kaynaklarının ne kadarını dizinleyebildiklerini ve bunların kesişim oranlarını tahmin etmeyi de güçleştirmektedir. Bu konuda göstergeler ümit verici olmaktan uzaktır: Internet'in küçük bir yüzdesi dizinlenebilmekte ve bu yarış her geçen gün arama motorları aleyhine işlemektedir (Lawrence ve Giles, 1998; Bergman, 2001; Kobayashi ve Takeda, 2000).

Dizinlenebilen Web sayfalarının azlığı problemi, dizinlenecek sayfaların *kalite*sinin göz önünde bulundurulmasını gündeme getirmiştir. Kimi çalışmalarda kaliteli olma hiper-metnin cebrik çizge özelliklerinden yola çıkılarak gündeme getirilmiştir (Deogun et al., 1998; Furner, Ellis ve Willet, 1996; Doorenbos, Etzioni ve Weld, 1996; Etzioni ve Weld, 1994). Örneğin, Deogun ve diğerlerinin (1998) çalışmasında, yazarlar çevrimiçi bir katalogda detaylı ürün

[2] Sadece bu istatistiki bilgi bile Türkiye'de bir süre önce kanunlaştırılmaya çalışılan (ancak veto edilen) Internet'i "zapt-u rapt" altına almaya yönelik girişimlerin ne kadar "naïf" olduğunu göstermeye yeterlidir kanısındayız.

bilgisinin bulunduğu sayfaları (hiper-metinde düğüm olarak da adlandırılır) referans sınıfında, benzer ürünlere ait bilgilerin tablolar ve/veya listeler kullanılarak topluca verildiği sayfaları da 'özel' sınıfında toplamışlardır. İlgili sınıflar gerek cebrik özellikleri (giriş veya çıkış bağlantıları istatistiği) gerekse kullanılan HTML yapılarının türlerine bakılarak tanınmıştır. Büyük hacimli dört Web kataloğunda (toplam 122 MB) SMART sistemi kullanılarak yapılan çalışmada, referans sayfalarında ve 'özel' sayfalardaki liste ve tablo yapılarının içeriklerini dizinlemekle tüm katalogları dizinlemenin performans açısından birbirlerinden bir farkı olmadığı rapor edilmiştir. Bu da, sonuç olarak, gerçekleştirilen deneydeki toplam 122 MB'lık dizinleme uzayını 17 MB'a indirgemiştir. HTML yapılarının Web üzerinden bilgi keşfedilmesinde kullanılması aslında yeni bir olay değildir. Doorenbos ve diğerleri, bilgi keşfetme şemsiyesi altında bir alış veriş aracı (shopping agent) geliştirmişlerdir (Doorenbos et al., 1996). Geliştirilen bu araçta, verilen bir ürün ismi için en ucuz fiyatları veren alış veriş siteleri (veya katalogları) taranırken, HTML yapılarının ve gösterim biçimlerinin (kalın, italik, boşluk, vb) uyumlu ve sürekli kullanıldığı varsayılmıştır.

Geleneksel yaklaşımda, örneğin, belgelerin yazım kalitesi (ya da metin değeri) oldukça yüksektir. Halbuki, Web sayfalarında yapılan yazım hataları bir istisna olmanın çok ötesindedir. Örneğin, yapılan bir doktora çalışmasında her sitede sık olarak kullanılan kelimelerden ortalama 200 tanesinin ve her üç yabancı soyadından birisinin yanlış hecelenmiş olduğu görülmüştür (Badino, 2001). Bunun arama motorlarına getirdiği ek yük sadece gövdelemeyle sınırlı değildir; arama motorlarının aynı zamanda düzeltme yapabilme yeteneklerinin de olması gerekmektedir.[3] Bir Internet sayfasının kaliteli olması, kimi zaman da, ne kadar sayfanın kendisine referans verdiği (authoritative) veya ne kadar çok authoritative sayfalara referans verdiği (hub) ile ölçülebilir olmuştur (Kleinberg, 1998; Lynch, 1997).

Başka bir sorun ise ikilenen (duplicate) sayfaların yüzdesinin giderek artmasıdır. Bir araştırmaya göre Web sayfalarının %30'u tekrarlardan oluşmaktadır (Kirsch, 1998). Tekrarlı sayfaların tanınması ve yalnızca bir kez dizinlenmesi birçok araştırmaya konu olmuştur (Kobayashi ve Takeda, 2000; Kirsch, 1998). Tipik olarak herhangi bir arama motorunun değerlendirilmesinde de tekrarlı Internet kaynaklarının erişim çıktısında yer alıp almadığı ölü bağlantılar ile birlikte sık sık anılan bir kriter olmuştur.

---

[3] Bilindiği üzere, Türkçe gövdeleme (Duran, 1999) bir dil üyesini *tanıma* işlemi; halbuki düzeltme ise dil üyelerini *üretme* problemidir. Verilen bir kelimeye ortalama 1.65 gövde karşılık gelmektedir. Bu da, kelime düzeltmenin Türkçede sezgisel yöntemlerle çözümlenebileceğini göstermektedir.

3.3 Belgelerin Gösterimi

Arama motorları dizinlemeyi azaltmak için, geleneksel bir bilgi erişim sisteminin aksine, verilen bir belgeyi olduğu gibi dizinlemez (Kobayashi ve Takeda, 2000; Laursen, 1998). Tipik olarak, bir Web sayfasının[4] başlık kısmı, üst veri belirteçlerinin (metadata tags) içerikleri, tam metnin ilk bir-iki paragrafı dizinlenir. Web sayfalarının insan gözüne hitap eden bir şekilde hazırlanması, ama öte yandan bu sayfaların arama motorları tarafından kolayca bulunmasının beklenmesi arama etkinliğini (örneğin duyarlık) olumsuz etkilemektedir (Olgun ve Sever, 2000; Küçük, Olgun ve Sever, 2000).

Web sayfalarının arama motorlarına hitap eden kısmıyla ilgili ilk adım, HTML 3.2 standardında belirteçlerinin tanımlanmasıyla atılmıştır.[5] HTML kodunun basında bulunan ve <head> … </head> alanı ile sınırlanan, üst veri belirteçleri görüntülenebilir olmayıp tamamen robotlara hitap etmektedir. Arama motorları açısından ilginç olabilecek iki belirteç ismi "tanım" (description) ve "anahtar sözcük"tür (keyword). Aşağıda Türk Kütüphaneciler Derneği'nin (TKD) Web sitesinden (http://www.kutuphaneci.org.tr/turk/) alınan bir örnekte "tanım" ve "anahtar sözcük" üst veri belirteçleri görülmektedir (Şekil 3).

Şekil 3. Türk Kütüphaneciler Derneği Web sitesi üst veri alanları

[4] Burada dolaylı olarak ilgili Web sayfasının kalite açısından robot trafından indirilip dizinlemeye değer bulunduğunu varsayıyoruz.

[5] HTML 4.1 için bakınız: http://www.w3.org/TR/REC-html40/struct/global.html#h-7.4.4

Bir Web sitesinde yer alan üst veri belirteçlerinin listesi (author, description, keyword, vs.) <head> etiketi içinde yer alan bir profil niteliğindeki biricik URI adresi ile kontrol edilebilir. Ancak bu, zorunlu değildir. Üst veri içeriklerini belirli bir sözcük haznesi ve kodlama kuralları ile kontrol etmek mümkün değildir. Bu durum arama motorları açısından ciddi bir sorun yaratmazken, duyarlık ve anma değerlerinin yüksek olması gereken veri tabanı uygulamaları için yeterli olmaktan çok uzaktır. Örneğin, yazar adı alanında isim ve soyad olarak mı yoksa soyad ve isim olarak mı kodlama yapılmıştır? Ya da birbirinden farklı iki ayrı tanım alanı içinde yer alan "bilgisayar ürünlerinin fiyat listesi" ile "bilgi teknolojisi malları ve ücretleri" anlamsal olarak sayfaları birbirlerine ne derece yaklaştırmaktadır? Bu sorunun cevabı veri tabanı sistemlerinde, bilgi erişim sistemlerindekinden farklı olarak, kesin olmak zorundadır. Bu amaçla yönlü çizge tabanlı bir veri tabanı modeli olan RDF (Resource Description Framework) (W3C, 1999) ve RDF'nin serileştirilmesi[6] için kullanılan XML (W3C, 1997) dili tanımlanmıştır. Internet kaynakları arasında ilişki kurabilen ve genişletilebilir olan RDF üstüne kütüphanecilik uygulamaları için kullanılmak üzere 15 elemandan oluşan Dublin Core (DC) standardı tanımlanmıştır (Dublin Core, 1998).[7] Başka bir deyişle, üst veri, Web kaynağının içeriğini makinenin anlayabileceği dilde tanımlamak amacı ile kullanılmaktadır.

Üst verinin bir Web kaynağına yerleştirilmesi kolay olmasına karşın mevcut Web sayfalarında kullanımı düşüktür. 1998'de yapılan bir araştırmada polimer kimya konulu Web sayfalarının yaklaşık %25'inde HTML üst veri belirteçleri kullanıldığı ortaya çıkmıştır (Qin ve Wesley, 1998). 1999'da yapılan bir başka araştırmada ise bu oran %34 olarak bulunmuştur (Lawrence ve Giles, 1999). Ancak Web sayfalarında Dublin Core üst veri belirteçlerinin kullanımı ise çok daha düşüktür. 1998'de yapılan bir araştırmada örnek olarak seçilen 1024 ev sayfasının sadece yedisinin Dublin Core üst veri belirteçleri içerdiği görülmüştür (O'Neill, Lavoie ve McClain, 1998). Bir başka çalışmada bu oran binde üç olarak bulunmuştur (Lawrence ve Giles, 1999). 2001 yılında yapılan bir çalışmada Web sayfalarında üst veri kullanmayanların %50'sinin üst veri hakkında herhangi bir bilgileri olmadığı ortaya çıkmıştır (Klarin, Pavelić ve Pigac, 2001). Dolayısıyla üst veri hakkında Web editörlerinin yeterince bilgi sahibi olmadıkları görülmektedir.

---

[6] Bir kodlama dili aracılığı ile metin türü bilgilerin bilgisayarın işleyebileceği hale çevrilmesi işlemine "serileştirme" adı verilmektedir.

[7] Türkçe RDF/DC editörü ve bu konuda genel bir tartışma için bkz. (Olgun ve Sever, 2000; Küçük et al., 2000).

Üst verilerle ilgili bir başka nokta "spam"dır.[8] Web sayfalarının dizinlenmesine çözüm olarak düşünülen üst veri sistemi kısa bir süre sonra kötüye kullanılmaya başlanmış, Web sitelerinin arama motorlarında üst sıralarda yer almasını sağlayabilecek "spam" teknikleri geliştirilmiştir (Henshaw, 2001). Böylece Web kaynağının üst verisine, kaynak ile ilgili olmayan ve arama motorlarında arama için kullanılan en güncel, en genel ve en popüler sözcükleri yerleştirerek erişilen sonuç listelerindeki sıralamalarda üst sıralara çıkmak amaçlanmaktadır. Kuşkusuz erişim açısından önemli dizinleme bilgileri içermesi gereken üst veri belirteçlerinin "spam" ile kirletilmesi erişim etkinliğini azaltmaktadır. Arama motoru servisleri "spam"ı tanıyabilecek ve önlem alabilecek algoritmalar geliştirmeye çalışmaktadırlar (Notess, 2001). Ancak kişilerin bilgiye erişimi engelleme pahasına da olsa kendi popülarite veya ticari kazançlarını ön planda tutmaları bu çalışmaların henüz tam anlamıyla başarı kazanmasını engellemektedir. Bundan dolayı, AltaVista, HotBot, Infoseek ve WebCrawler gibi arama motorları HTML üst veri belirteçlerini belgelerin gösteriminde sınırlı olarak kullanmalarına karşılık, Excite ve Lycos gibi bazı arama motorları üst veri etiketlerinden yararlanmamaktadır (Laursen, 1998). Onüç arama motoru üzerinde yapılan bir başka araştırmada ise tüm motorların "başlık" belirtecini (title tag), AltaVista, HotBot ve Infoseek'in anahtar sözcük ve tanım belirteçlerini, HotBot'ın "yazar" belirtecini (author tag), AltaVista ve Lycos'un şekil, resim ve görüntülerle ilgili başlık ya da resim altı (caption) gibi alternatif metin bilgisi veren "alt" belirtecini (alternative tag)[9] dizinledikleri gözlenmiştir (Mettrop ve Nieuwenhuysen, 2001).

3.4 Erişim Fonksiyonu

[8] "Spam" kelime olarak, 'genellikle öğleleri sade veya sandviç içinde tüketilen pembe renginde bir konserve et' anlamına gelmektedir. Spam, Amerika Birleşik Devletleri'nde göreceli olarak popüler olan ama birçok kimse tarafından da hiç bir estetik ve beslenme değeri olmayan yiyecek türü olarak değerlendirilmektedir. Kelimenin bilişim jargonuna, "aksi takdirde istenmeyecek veya sorulmayacak olan aynı mesajın/e-postanın birçok e-posta hesabına ve/veya Usenet haber grubuna gönderilmesi" anlamıyla girmiştir. (Spam karşıtı bir portal adresi için bkz.: http://spam.abuse.net/). Bu mesaj bombardımanı çoğunlukla bir ticari avantaj sağlamak için kullanılmaktadır. Bu çalışmada söz ettiğimiz 'spam' ise 'SEP' (Search Engine Persuasion) veya 'Web Spam' olarak adlandırılmaktadır. Burada söz konusu olan, bir arama motorunun erişim fonksiyonun nasıl çalıştığını ve bir belgenin nasıl dizinlendiğini doğruya yakın kestirebilmek ve bu bilgiyi bir avantaj (veya kişisel tatmin için) sağlamak üzere kullanmaktır. Bir başka deyişle spam, arama motorlarına bir belgeyi o belgenin HTML koduyla oynayarak gerçek içeriğinin ötesinde başka birşeyle ilgiliymiş gibi "yutturmak"tır (örnekler için bkz: (Laursen, 1998)).

[9] Alt belirteci şekil, resim ve görüntülerin yüklenmediği ya da kullanıcının bu özelliği kullanmak istemediği durumlarda sayfayla ilgili alternatif metin bilgisi sunması açısından yararlıdır. Bu belirtecin Web madenleme araçları (Web mining tools) tarafından sayfalar arasındaki ilişkilerin ortaya çıkarılmasında ya da bağlantıların anlamsal olarak sınıflandırılmasında da kullanıldığı görülmektedir.

İkinci bölümde genel bilgi erişim sistemleri için verilen erişim fonksiyonları arama motorları için de geçerlidir. AltaVista, Yahoo! gibi nispeten büyük arama motorları hem ticari sır olması açısından hem de "spam"a yol açmamak için başvurdukları erişim fonksiyonlarını ve dizinleme tekniklerini açıklamamaktadır. Bununla birlikte, söz konusu arama motorlarının çoğunun daha önce akademik ortamda geliştirildikleri bilindiği için, kullandıkları erişim fonksiyonları şu veya bu şekilde tahmin edilebilmektedir. Örneğin, Infoseek arama makinesi Massachusetts Üniversitesi tarafından geliştirilen INQUERY[10] bilgi erişim sisteminin ticari sürümüdür ve ilgililik (relevance) hesaplamasının belge istatistiği (*tf*idf*), kısmen sayfanın başka sayfalar tarafından ne kadar sıklıkla referans verildiğine (popülaritesine) ve bu sayfadan bağlantı verilen sayfaların popülaritesine dayanmaktadır (Kirsch, 1998). Google arama motoru yalnızca belge istatistiğini değil, sayfanın 'hub' ve 'authoritative' bağlantılarını da dikkate almaktadır (Kleinberg, 1998; Kobayashi ve Takeda, 2000). AltaVista ise belge sıklığına dayalı ağırlıklı Boole araması (weighted Boolean search) yapmaktadır (Silverstein, Henziger, Marais ve Moricz, 1999). Excite kavram tabanlı arama yapan, Boole sorgu dilini kullanan ve gövdeleme tekniğinden yararlanmayan bir arama motorudur (Jansen, Spink, Bateman ve Saracevic, 1998).[11] Kavramlar, terimlerin kümelendirilmesine (çevrimiçi eşanlamlı sözlük) dayanır. Excite aramada ise 'latent semantic' analiz metodunun (Deerwester et al., 1990; Foltz, 1996) hesaplama-zaman etkinliği açısından basitleştirilmiş şeklini kullanmaktadır.[12]

Erişim fonksiyonunda bir sorgu ile belge arasındaki benzerlik hesaplamasında basit olarak her ikisinde de geçen ortak terimler temel alınabileceği gibi, bir belgeyi kendisini oluşturan yapısal bileşenlerin (başlık, anahtar sözcükler, özet, tam metin, vb. gibi) bir bütünü gibi görüp, belgenin çeşitli bileşenlerinde geçen arama terimlerine farklı ağırlıklar verilebilir. Örneğin, erişim fonksiyonu çeşitli belge bileşenlerinin sorgu ile benzerliklerinin toplamı olan bir polinom şeklinde düşünüldüğünde, başlık bileşeninin sorgu ile benzerliği belgenin tam

---

[10] INQUERY çıkarsama-ağı tabanlı (inference network-based) bir bilgi erişim sistemidir (Turtle ve Croft, 1991).

[11] Kanımızca Excite arama motorunun ilginç yanlarından birisini oluşturan 'More Like This' (Buna benzer diğer sayfaları bul) özelliği bu çalışmanın kaleme alındığı sırada doğrulanamadı. Büyük bir olasılıkla kaldırılmış olan bu özellik 'ilgililik geribildirimi" (relevance feedback) tekniğine' başvuruyordu ve bu yönüyle arama motorları arasında biricik (unique) bir perspektif sağlıyordu. Klasik bilgi erişim sistemlerinde kullanılan tekniğe bağlı olarak, bilinen küçük veri derlemlerinde %28-%46 arasında (Salton ve Buckley, 1990), büyük veri derlemlerinde (TREC D1 ve D2 gibi) %14-%21 arasında (Lee, 1995) performans artırımı sağlayan ilgililik geribildirimi tekniğinin arama motorları arasında aynı öneme sahip olmaması, arama motorlarında üzerinde araştırma yapılan sorunların klasik bilgi erişim sistemlerinin sorunlarından farklı olduğunun önemli bir göstergesidir.

[12] "Intelligent Concept Extraction" adı altında Excite tarafından patenti alınmıştır (bkz. http://www.excite.com/ice/tech.html).

metniyle benzerliği ile aynı kefeye konmayabilir. Bir başka deyişle, örneğin, belge başlığında geçen bir terim, belgenin konusunu belirlemede daha ağırlıklı olarak değerlendirilebilir. Deneysel olarak bilinen bu gerçek, bir anlamda eldeki belgenin ilgililik derecesini tayin etmede farklı kaynaklardan gelen kanıtların birleştirilmesi şeklinde düşünülebilir. Nitekim '90'ların ortalarında ortak bir veri tabanı (ya da belge derlemi) üzerinde farklı erişim modelleri çalıştırılarak eldeki sorgular değerlendirildiğinde, farklı erişim fonksiyonlarına göre erişilen sonuçların birleştirilmesinin erişim performansını büyük ölçüde (tek bir işlemeye, sorguya ya da alt modele göre göreceli olarak) artırdığı gözlenmiştir (Lee, 1997, 1995).[13]

Erişim fonksiyonunun belgenin çeşitli bileşenlerini eldeki sorguyla eşleştirirken farklı ağırlıklar kullanabileceğini daha önce belirtmiştik. Bu özellik aşağıda verilen örnekle daha ayrıntılı olarak açıklanmaktadır (Yuwono ve Lee, 1996).

Boole modelindeki erişim fonksiyonunun ikil (binary) mantıkla çalıştığı bilinen bir gerçektir. Zaten bu yüzden Boole modelinde erişim çıktısındaki belgelerde sıralama yoktur (Salton, 1989). Bir başka deyişle, erişim çıktısının en başında yer alan bir belge ile en sonunda yer alan belge aynı erişim değerlerine sahiptir. Fakat ufak bir trük ile -ki çoğu arama motorlarında bu yapılmaktadır- Boole erişim fonksiyonu kullanılarak sıralama yapmak mümkün olabilmektedir:

$$\text{İç Çarpımı } (D_r, Q_s) = \sum^{t} a_{ri} * q_{si.} \qquad (10)$$

Burada D_r'yi r ile gösterilen URL adresine sahip bir Web belgesi ve Q_s'yi s ile gösterilen bir numaraya sahip bir sorgu ifadesi olarak düşünebiliriz. Daha da ileri giderek D_r ve Q_s'yi sırasıyla belge terimlerinden, a_{ri}, ve arama terimlerinden, q_{si}, oluşan listeler olarak yorumlayalım ($1 <= i <= t$). Bu iç çarpım bize ortak terimler için eşit bir tamsayı değeri döndürecektir. Yukardaki iç çarpım kolayca görülebileceği gibi Boole 'VE' işlecine karşılık

[13] Bu tür aramaya iç üst arama (internal metasearch) adı verilir. Başka bir deyişle, kendi basına işletimde olmayan fakat bir ana makineye bağlı olarak çalışan alt bileşenlerin erişim çıktıları seçilen bir birleştirme (combination/fusion) algoritması çerçevesinde tek bir erişim çıktısı haline getirilir. İç üst arama tekniklerini daha popüler olan dış arama motorları, örneğin, profusion (http://www.profusion.com) ya da metacrawler (http://www.metacrawler.com) ile karıştırmamak gerekir. Burada söz konusu olan, bu çalışmanın ana temasını oluşturan ve kendi başına işletimde olan arama motorlarına bir yazılım aracı (meta search engine agent) tarafından ilgili sorgu ifadesinin yönlendirilip sonuçların tek bir erişim çıktısı altında birleştirilmesidir. Dış üst arama motorunun, ilgili bağımsız çalışan arama motorlarına müdahale imkanı olmayıp, nasıl bir erişim fonksiyonu kullanıldığı da bilinmeyebilir. Hatta kullanılan veri tabanları (dizinlenen Web sayfaları) ortak olmak zorunda değildir. Burada kritik nokta döndürülen sonuçların normalize edilmesi (belgelerin sıralama değerlerinin modellenmesi) (Montague ve Aslam, 2001) ve birleştirilmesidir (Belkin et al., 1995). Birleştirmedeki espri farklı erişim stratejilerinin (Boole, bulanık mantık, vektör, olasılık, vb. gibi) benzer ilgili belgeler ve farklı ilgisiz belgeler döndürmeleridir.

gelir. Doğal olarak 'VEYA' ve 'DEĞİL' işleçleri nasıl yorumlanacak diye sorulabilir. Her bir Boole ifadesi anlamı değişmeksizin DNF (Disjunctive Normal Form) formuna çevrilebilir. DNF formuna çevrilen bir sorgu, birbirlerine 'VEYA' ile bağlanmış bağımsız cümleciklerden oluşur -ki her bir cümlecikteki terimler de birbirlerine 'VE' ile bağlanmıştır. Bu bağımsız cümlecikler kendi başına sorgu olarak düşünülüp yukardaki iç çarpım işlemi gerçekleştirilir. Sonuç listeler aşağıdaki gibi birleştirilebilir: Bir belgenin toplam erişim değeri ilgili sonuç listelerindeki erişim değerlerinin toplamıdır. 'DEĞİL' işleci DNF çevriminin sonucunda terimlere tümleyen olarak yansıtılır (belgenin ilgili terimi içermemesi anlamına gelmektedir). Bu da erişim fonksiyonuna sonradan budama işlemi (post-pruning technique) yapma fırsatını verir: birleştirilen sıralı erişim çıktısı üzerinden bir geçiş yapılarak ilgili terimi içeren belgeler sonuç listeden çıkarılır.

Şimdi de Boole modelinde referansların nasıl işleneceğini tartışalım.

Internet yapısal açıdan yorumlanacaksa yönlü çizge (ya da hiper-metin veri tabanı) olarak düşünülebilir. Bu bağlamda bir Web belgesinin (daha genel olarak Internet kaynağının) bir uzaklıktaki komşuluk kümesini ilgili belgeye bağlantı veren ya da ilgili belgenin bağlantı verdiği belgelerin kümesi olarak tanımlayalım. Bu kavram *yakın komşu* (D_r) ile gösterilsin. Referans ilişkisinin Internet ortamında yakın komşuluk ilişkisi ile özdeş olduğunu düşünelim.[14] Bu durumda, yukarda verilen iç çarpımdaki belge terimi ağırlığı aşağıdaki gibi düşünülebilir:

Sorgu terimi belge terimleri içinde ise $a_{ri} = c_1$; sorgu terimi yakın komşuluk içindeki belgelerin herhangi birisinde geçiyorsa c_2; aksi tadirde *0*. c_1 ve c_2 sabitleri tasarımcının ilgili yapısal benzerlikleri nasıl ağırlıklandıracağına bağlı olarak değişir. Örneğin c_2 değeri tayin edilirken referans edilen/eden referans sayısı (Google motoru tarafından tutulmaktadır) veya referans eden/edilen belgenin kalitesi hesaba katılabilir.

3.5 Arama Motorlarında Performans Değerlendirmeyle İlgili Çalışmalar

[14] Dikkatli okuyucu referans ile yakın komşuluk ilişkilerinin özdeşliğinin gerçek hayattaki durumu yansıtmaktan uzak olduğunu düşünebilir. Kaba bir sınıflama ile her bir bağlantı ya organizasyon türü (bir sonraki, bir önceki, üstteki, ev, vb.) ya da çeşitli anlamsal ilişkileri içine alan referans türü (genelleştirme/özelleştirme veya alt/üst bileşen içinde düşünülebilir (Frei ve Stieger, 1995). Bu açıdan düşünüldüğünde, her bir bağlantının referans etme/edilme anlamına gelmeyeceği bir gerçektir; fakat her bir soyutlamanın kendi içinde yanlışlık içerebileceği düşünülerek basitlik uğruna yukarıdaki özdeşliğin geçerli olduğu varsayılabilir.

Bundan önceki alt bölümlerde arama motorlarının çeşitli yönleriyle ilgili araştırmalara yeri geldikçe değinildi. Bu alt bölümde arama motorlarında bilgi erişim performansının değerlendirilmesiyle doğrudan ilgili çalışmalar kısaca özetlenmektedir.

Geleneksel bilgi erişim sistemlerinin performans değerlendirmesinde kullanılan anma ve duyarlık gibi ölçümler arama motorlarının performans değerlendirmesinde de genellikle kullanılmaktadır. Fakat, aşağıda da açıklandığı gibi, arama motorlarının kendine özgü özelliklerinden dolayı anma ve duyarlık ölçümlerinde bazı değişiklikler yapılması gerekmektedir. Bunun yanı sıra, yapılan araştırmalarda arama motorlarının kapsam, güncellik ya da kırık bağlantılar (broken links), yanıt süresi, insan faktörleri ve kullanıcı arayüzü gibi ölçütler yönünden de incelendiği görülmektedir (Oppenheim, Morris ve McKnight, 2000).

Anma, bilindiği gibi, erişilen ilgili belgelerin derlemdeki toplam ilgili belgelere oranını Arama motorları tipik bilgi erişim sistemleriyle karşılaştırılamayacak kadar büyük hacimli belge derlemleri üzerinde aramalar gerçekleştirdiklerinden, belirli bir soru için derlemdeki toplam ilgili belge sayısını bulmak hemen hemen olanaksızdır. Buna benzer bir sorunla daha önce yüz yüze gelen TREC (Text REtrieval Conference) konferansları (http://trec.nist.gov/), sorunu "havuzlama" yöntemi ile çözmeye çalışmışlardır.[15] Bu yönteme göre, bir bilgi ihtiyacı ile ilişkili her bir işlemenin[16] (run) sonucunda dönen 1000 belgeden oluşan erişim çıktısının

[15] Bilgi erişim sistemlerinin değerlendirilmesinde yöntemler ve kalite testleri (benchmark collections) yönünden geçmişten gelen oldukça zengin bir birikim vardır (Sparck Jones, 1971; Salton, 1971). Bilinen test derlemleri CACM, CISI, Cranfield ve NPL olup, tam bilgi verirler; yani, sorgular ve belgeler terim vektörleri cinsinden tanımlı olup, her bir sorgu için ilgili belgeler liste halinde tutulur (bkz: ftp://ftp.cs.cornell.edu/pub/smart/). Bu testler bilgi erişim alanında karşılaşılan meydan okuyucu sorunların çözümü doğrultusunda oluşturulan yeni modellerin test edilmesinde ve, daha önemlisi, ortak bazda karşılaştırılmasında zamanla yetersiz kalmışlardır. Bü nedenle, 1990'da Amerikan İleri Savunma Araştırma Projeleri Ajansı'nın (DARPA) TIPSTER metin projesi (http://www.nist.gov/itl/div894/894.02/related_projects/tipster/) çerçevesinde, Ulusal Standartlar ve Teknoloji Enstitüsü'nün (NIST: National Institute of Standards and Technology) bilgi erişim teknolojilerini değerlendirmede kullanılmak üzere çok geniş bir metin (ya da genel olarak belge) derlemi oluşturması istendi (Voorhees ve Harman, 1999). İlk TREC konferansı 1992 yılında ticari kuruluşların ve çoğu DARPA veya NIST tarafından desteklenen akademik çevrelerin katılımıyla gerçekleştiğinde, eldeki derlem 2GB büyüklüğündeki yaklaşık bir milyon belgeden oluşuyordu (1998'e kadar süren TIPSTER programı 4 ciltlik Tipster CD'leri ile anılmaktadır). Ticari ve akademik bilgi erişim sistemlerinin test yatağı (test bed) olarak hizmet veren TREC, ulusal kimlikten sıyrılarak zamanla uluslararası bir yarış arenası haline dönüşmüştür. (2000 yılının Kasım ayında yapılan 9. TREC konferansına 17 ülkeden 69 akademik veya ticari grup katılmıştır). Yeni modellerin ya da tekniklerin denendiği bu konferanslar birkaç ana görev (task) ve kimisi sonradan ana görev olan bir çok izlerden (tracks) oluşmaktadır. İşte bu görevlerden birisi olan 'ad hoc' (bilgi ihtiyaçlarından oluşturulan sorgular aracılığı ile belgeler derlemini araştıran ve ilgili olduğuna inanılan belgelerin bir belge erişim çıktısı içersinde düzenlenerek geri getirilmesi sürecini yöneten sistemlerin başarılarının incelenmesi) TREC-8'den sonra yerini Web erişim izine bıraktığında Web için oluşturulan derlemin büyüklüğü 100 GB büyüklüğünde 18.5 milyon sayfadan oluşuyordu.

[16] Bir bilgi erişim sistemi (ya da arama motoru) bir göreve ya da ize birden fazla katılabilir. Örneğin, 'ad hoc' görevinde bir bilgi ihtiyacı (TREC terminolojisinde "konu" olarak adlandırılır) başlık, tanım, açıklama (narrative) ve kavramlar (TREC-2'den sonra "kavramlar"dan vazgeçildi) yapılarından oluşan bir mizanpajla ifade ediliyordu. Bir sistem yalnızca başlığı ya da tüm kısımları otomatik olarak ya da elle (orijinal ya da genişletilmiş Boole ya da geribildirim teknikleri ile sorguların genişletilmesi yolu) işleyerek sorguları oluşturabilir. Herhangi bir kombinasyon bir "işleme" olarak anılır.

ilk 100 belgesi bir havuzda toplanır. Bir değerlendirici, ki çoğunlukla bilgi ihtiyacını oluşturan kişidir, havuzda toplanan tekil (unique) belgelerin (ki konu başına ortalama 1500-2000 civarındadır) üzerinden geçerek ilgili belgeleri saptar. Eldeki derlemde bunlar dışında ilgili belge olmadığı kabul edilir ve bununla birlikte 1000'lik erişim çıktısı kullanılarak her bir işlemenin ilgili konuya göre anma ve duyarlık değerleri hesaplanır. Buradaki espri iki temel varsayıma dayanmaktadır: (1) İlgili belgeler büyük bir olasılıkla üst sıralara (örneğin, erişim çıktısının %10'luk kesimi) yerleşecektir (Voorhees ve Harman, 2000); ve (2) Kullanılan birbirinden oldukça farklı arama stratejileri sonucu farklı belgelere erişim sağlanacaktır (Lee, 1997; 1995; Belkin et al., 1995). Bu iki varsayım zaman içinde çeşitli deneylerle doğrulanmıştır.

Havuzlama yöntemine benzer bir başka yöntem de gerçek hayatta işletimde olan arama motorlarının ortalama anma değerlerinin hesaplanmasında kullanılmak üzere Clarke ve Willet (1997) tarafından önerilen "göreli anma" (relative recall) değeridir. Bu yöntem bir arama motoru tarafından bulunan ilgili belgelerin diğer arama motorlarının bulduğu ilk belgeler arasında yer alıp almadığının kontrol edilmesine dayanmaktadır.

Arama motorlarında duyarlık değerlerinin ölçülmesi geleneksel bilgi erişim sistemlerinden biraz farklılık göstermektedir. Geleneksel sistemlerde çoğu zaman erişilen tüm belgelere bakarak duyarlık değeri hesaplanırken, arama motorlarında ise erişilen belge sayısının çok yüksek olması ve bu belgelerin hepsinin tek tek değerlendirilememesi nedeniyle belirli kesme (cut-off) noktalarında duyarlık değerlerinin hesaplanması yoluna gidilmektedir. Bir başka deyişle, belirli bir soru için erişim çıktısında yer alan tüm belgeler üzerinden duyarlık değerini hesaplamak yerine, belirli sayıda (5, 10, 15, 20... gibi) belge görüldükten sonra her aşamada duyarlık değerlerinin nasıl değiştiği hesaplanmaktadır. Buradaki varsayım, çoğu arama motoru kullanıcılarının erişim çıktısında yer alan belgelerin çok azını (bir ya da iki ekran dolusu) görmek istemeleridir. Nitekim, yapılan araştırmalarda bu varsayımın geçerliliği kanıtlanmış, kullanıcıların gözden geçirdikleri ekran sayısı ortalama 1,39 (standart sapma 3,74) olarak bulunmuştur (Silverstein et al. 1999). Konuyla ilgili bir başka çalışmada (Jansen et al., 1998) ise kullanıcıların ilk ve ikinci ekranları görme oranı sırasıyla %58 ve %19 olarak bulunmuştur.

Geleneksel bilgi erişim sistemleriyle arama motorları arasındaki önemli farklardan birisi de sorgu cümlelerinde kullanılan ortalama sözcük sayısıdır. Tipik bir bilgi erişim sisteminde sorgu ifadelerinde ortalama 7,9 ile 14,95 sözcük yer almasına (Jansen et al., 1998) rağmen, arama motorlarına girilen sorgularda bu rakam ortalama 2,3 civarındadır (Silverstein et al., 1999; Kirsch, 1998; Jansen et al., 1998). Bu durumu Infoseek şirketinin başkanı S. Kirsch,

“Web kullanıcıları bir-iki kelimelik sorgularıyla bizden mucizeler yaratmamızı bekliyorlar” diye alaycı bir şekilde özetlemiştir (Kirsch, 1998). Gerçekten de arama motorlarının işlem kütükleri kullanılarak yapılan araştırmalarda en popüler sorguların tek sözcükten oluşan sorgular olduğu görülmektedir. Örneğin, aralarında "sex", "Playboy", "Penthouse", "chat", "nude", "porn", "erotica", "games" gibi sözcüklerin de bulunduğu toplam 15 sözcük Infoseek'te yapılan bütün aramaların %12'sini oluşturmaktadır (Kirsch, 1998). AltaVista'da yapılan yaklaşık bir milyar arama sorusunun incelenmesinden de benzer sonuçlar elde edilmiş, sırasıyla "sex", "applet", "porno", "mp3" ve "chat" gibi tek sözcükten oluşan sorular en sık aranan sözcükler olmuştur (Silverstein et al., 1999). Arama motorları, tek sözcükle arama yapma konusundaki bu meydan okumayı, Web kullanıcılarının tipik olarak anmadan çok duyarlık ile ilgilendiği ilkesini de göz önünde bulundurarak, çok referans alan sayfalara öncelik verme yolunu seçerek karşılamaya çalışmaktadır.

Arama motorlarında performans değerlendirmesi konusunda bu zamana dek yapılan araştırmalar birkaç çalışmada topluca özetlenmiştir (Oppenheim et al., 2000, s. 14, 23; Soydal, 2000).

Konuyla ilgili olarak yapılan ilk çalışmalardan birisinde Gudivada ve diğerleri (1997) iki soruyu (“latex software” ve “multiagent system architecture”) 13 farklı arama motoru üzerinde Boole işleçlerini kullanarak ve tamlama olarak ayrı ayrı aramışlar ve elde ettikleri sonuçları erişilen belge sayıları açısından karşılaştırmışlardır. Erişim çıktılarında ilgili belgelerin ilgisiz belgeler arasında dağıldığı görülmüş, bu nedenle kullanıcıların salt sıralamada başta gelen belgelere bakmalarının yeterli olmayacağı sonucuna varılmıştır. Arama motorlarının, kapsamları birbirinden farklı dizinler üzerinde arama yapmaları nedeniyle bu çalışmada performans değerlendirme ölçümleri kullanılmamıştır.

Chu ve Rosenthal’ın (1996) çalışması geleneksel performans değerlendirme ölçümlerinden duyarlığın kullanıldığı ilk araştırmalardan birisidir. Araştırmacılar AltaVista, Excite ve Lycos üzerinde gerçekleştirilen 10 arama sorgusu için duyarlık oranlarını sırasıyla %78, %55 ve %45 bulmuşlardır. Benzer bir çalışmada Leighton ve Srivastava (1999) 15 soru için erişilen ilk 20 Web sitesi üzerinden AltaVista, Excite, HotBot, Infoseek ve Lycos’un duyarlık değerlerini hesaplamışlardır. AltaVista, Excite ve Infoseek’in daha iyi performans gösterdikleri (%50’nin üzerinde), Lycos’un kısa ve yapılanmamış sorularda, HotBot’un ise yapılanmış sorularda daha başarılı olduğu görülmüştür.

AltaVista, Yahoo! gibi popüler arama motorlarının günümüzde yüz milyonlarca Web sayfasını dizinledikleri bilinmektedir. Bu tür büyük derlemlerde kesin anma (absolute recall) değerini hesaplamak için gerekli olan derlemdeki toplam ilgili belge sayısını bulmak hemen

hemen olanaksız olduğundan, yapılan ilk çalışmalarda anma değerlerinin ölçülmesi yoluna gidilmediği görülmektedir. Her arama motorunun farklı Web sayfalarını dizinlemesi, farklı arama motorları için elde edilen performans değerlerini karşılaştırmayı da güçleştirmektedir. Clarke ve Willet (1997) göreli anma (relative recall) değerini kullanarak AltaVista, Excite ve Lycos üzerinde 30 soruya dayanan bir araştırma gerçekleştirmişlerdir. Bu araştırmada söz konusu arama motorları için bulunan ortalama anma değerlerinin (yaklaşık %60), geleneksel bilgi erişim sistemlerinde genelde elde edilen sonuçların aksine, ortalama duyarlık değerlerinden (%35) daha yüksek olduğu görülmüştür. Anma değerleri açısından söz konusu arama motorları arasında istatistiksel açıdan anlamlı bir faklılık yoktur. Duyarlık açısından ise AltaVista (%46) ile Lycos (%25) arasındaki performans değerleri istatistiksel açıdan anlamlı bulunmuştur.

Göreli anma değerlerinin kullanıldığı bir başka araştırma Gordon ve Pathak (1999) tarafından gerçekleştirilmiştir. Araştırmacılar gerçek bilgi gereksinimlerinden kaynaklanan toplam 33 soruyu sekiz farklı arama motoru üzerinde deneyerek, bilgiye gereksinim duyan deneklerin yaptığı ilgililik değerlendirmelerine göre çeşitli kesme (cut-off) noktalarında anma ve duyarlık değerlerini hesaplamışlardır.[17] Buna göre çeşitli arama motorlarında erişilen ilk 10 belgede duyarlık değerleri %41 (AltaVista) ile %18 (Yahoo!), anma değerleri ise (erişilen ilk 15-25 belgede) %16 (AltaVista) ile %6 (Yahoo!) arasında değişmektedir.

Soydal (2000) AltaVista, Excite, HotBot, Infoseek ve Northern Light üzerinde gerçekleştirdiği bir araştırmada erişilen ilk 10 ve ilk 20 belge üzerinden ortalama (göreli) anma ve duyarlık değerlerini hesaplamıştır. Adı geçen arama motorları arasında ortalama duyarlık değerleri (yaklaşık %50) açısından anlamlı bir farklılık olmadığı görülmüştür. Ortalama anma değerleri ise %14 (Infoseek) ile %31 (Northern Light) arasında değişmektedir. Infoseek ile Northern Light arasındaki anma değerleri istatistiksel açıdan anlamlı bulunmuştur.

Yukarıda (3.3) Web sayfalarının hazırlanmasında yazar, anahtar sözcük, tanım vb. gibi HTML üst veri belirteçlerinin (meta tags) belgelerin içeriğini tanımlamada kullanıldığından söz etmiş ve arama motorlarının erişim amacıyla bu alanlardan yeterince yararlanmadığını vurgulamıştık. Web belgelerinin hazırlanmasında HTML üst veri belirteçleri kullanımının arama motorlarında erişim etkinliğini artırıp artırmadığı çeşitli araştırmalara konu olmuştur. Turner ve Brackbill (1998) AltaVista ve Infoseek üzerinde yaptıkları kontrollü araştırmada anahtar sözcük (keyword) üst veri belirtecinin kullanıldığı belgelerde üst veri belirteci

[17] TREC derlemiyle çalışan Web erişim grubundaki araştırmacılar da duyarlık değerlerini kesme noktası kullanarak hesaplamışlardır (bkz. Hawking, Craswell, Thislewaite ve Harman, 1999).

kullanılmayanlara oranla erişilebilirliğinin önemli ölçüde arttığını saptamışlardır. Ancak, popüler arama motorları kullanılarak yapılan bir başka kontrollü araştırmada üst veri belirteçlerinin kullanımının erişim sonuçlarını pek etkilemediği ortaya çıkmıştır. Elektronik bir dergi olan *First Monday*'de (http://www.firstmonday.dk) yayımlanan ve üst veri belirteçleri boş olan makalelere arama motorları kullanılarak erişim sağlanmışır. Daha sonra ise bu makalelere üst veri belirteçleri eklenmiş ve aramalar tekrarlanarak söz konusu makalelerin erişim çıktısında daha üst sıralarda yer alıp almadıkları test edilmiştir. Yapılan testlerde üst veri belirteçlerinin kullanımının erişim sıralamasını tek başına etkilemediği görülmüştür (Henshaw ve Valauskas, 2001). Anlaşıldığı kadarıyla, Web sayfalarının hazırlanmasında üst veri belirteçlerinin kullanımı açısından henüz bir standartlaşmaya gidilmediğinden, çoğu arama motorları üst veri belirteçlerini erişim sırasında dikkate almamaktadırlar.

Çeşitli araştırmacılar arama motorlarında çeşitli erişim ve sıralama algoritmalarının performanslarını değerlendirmişlerdir. Savoy ve Picard (2001) basit anahtar sözcüğe dayalı dizinleme stratejilerinin terim sıklığına dayanan dizinleme stratejilerinden daha başarılı olduğunu, sorgu cümlesinde daha fazla anahtar sözcük kullanmanın ortalama duyarlığı artırdığını, dur listesi kullanmanın erişim etkinliğini artırdığını, TREC 8'de kullanılan bilgi erişim modellerinin yaklaşık 2 GB'lık Web derlemi üzerinde de yüksek performans sergilediğini, Web sayfası başlığında yer alan terimleri ağırlıklandırmanın ortalama duyarlık üzerinde önemli bir etkisi olmadığını, sadece başlıkta yer alan terimlerin dizinlenmesinin erişim etkinliğini zayıflattığını, gövdeleme kullanılmadığında çoğu arama stratejilerinde ortalama duyarlığın önemli ölçüde düştüğünü bulmuşlardır. Yuwono ve Lee'nin (1996) araştırmasında ise vektör uzayı modeline dayalı erişim algoritmalarının daha başarılı sonuçlar verdiği, sadece üst veri alanlarında yer alan bilgilere dayanan algortimaların, sezgisel olmalarına rağmen, pek başarılı olmadığı ortaya çıkmıştır.

Arama motorları tarafından erişilen ilgili belgeler arasındaki çakışma oranı (overlap) çeşitli araştırmalara konu olmuştur. Yukarıda anılan Gordon ve Pathak'ın (1999) çalışmasında yedi arama motoru arasındaki çakışma oranı sadece %7 olarak bulunmuştur. Soydal'ın (2000) çalışmasında da beş arama motoru için benzer bir sonuç (%11) elde edilmiştir. Bharat ve Broder (1998) ise dört arama motoru (AltaVista, HotBot, Excite ve Infoseek) arasındaki çakışma oranının %1'den az olduğunu bulmuştur. 1997 yılında söz konusu dört arama motoru tarafından dizinlenen toplam 200 milyon civarındaki Web sayfasından sadece yaklaşık iki milyonu dört arama motoru tarafından da dizinlenmiştir. Bir başka deyişle, bu bulgular farklı arama motorlarının Web uzayında farklı ilgili belgelere erişim sağladığını ortaya çıkmaktadır.