ساختار نمایه‌سازی در موتورهای کاوش وب

نوشته: احمد کمیجانی*

چکیده

حجم وسیع اطلاعات بر روی شبکه وب باعث می‌گردد تا پاسخ‌دهی به کاوش‌های ارسالی از سوی کاربرن، بدون دسترسی به تمام متون و فقط با استفاده از فایل‌های نمایه صورت گیرد. بدین منظور، در سطح شبکه از روش‌های مختلف نمایه‌سازی استفاده می‌گردد. روش نمایهٔ انتهای کتاب، استفاده از ابرداده‌ها، شاخه‌های موضوعی و ساختار متمرکز و پراکنده در فن‌آوری موتورهای کاوش از روش‌های دیگر می‌باشد.

بسیاری از موتورهای کاوش از یک ساختار متمرکز خزنده-نمایه‌ساز، سود می‌جونید. خزنده‌ها برنامه‌های نرم‌افزاری هستند که عمل پیمایش وب را انجام داده و صفحات جدید و یا به روز در آمده را به سرویس‌دهنده‌ای که قرار است این صفحات در آنجا نمایه شوند، می‌فرستند.

در ساختار پراکنده که بسیار موثرتر از نوع متمرکز است، مشکلات ناشی از استفاده از ساختار متمرکز چون:

۱) دریافت درخواست صفحات، توسط خزنده‌های متفاوت موتورهای کاوش، از سرویس‌دهندگان وب؛

۲) افزایش ترافیک در وب به علت استخراج تمام اشیا و اجزا صفحات وب و نادیده گرفتن اکثر آن‌ها هنگام نمایه‌سازی؛

۳) جمع‌آوری اطلاعات بدون همکاری و آگاهی سایر خزنده‌ها و موتورهای کاوش، مورد توجه قرار گرفته و مرتفع شده است.

مقدمه

از زمان پایه‌گذاری وب جهانی در اواخردههٔ ۱۹۸۰، هیچ کس نمی‌توانست پیش‌بینی وضعیت و تأثیرات فعلی آن را بکند. رونق وب و رشد فزاینده آن بر کسی پوشیده نیست، به نحوی که فقط میزان اطلاعات متنی قابل دسترس آن در حدود ۱ ترابایت تخمین زده می‌شود (Baeza-Yates, 1999).

اندازه‌گیری حجم اطلاعات بر روی شبکه اینترنت به ویژه وب کار بسیار دشواری است. بر طبق آمارهایی که گروه Cyveillance عرضه کرده است، بیش از ۲/۱ میلیارد صفحه اطلاعات، بدون تکرار و قابل دسترس تا نیمه دوم سال ۲۰۰۰ بر روی وب موجود بوده و برساس همین مطالعات نرخ رشد انفجاری صفحات وب ۷ میلیون در روز بوده است. (Pasore, 2000)

این بدان معناست که در حال حاضر تعداد صفحات وب، به میزان سه برابر آن افزایش یافته است و این

* دانشجوی کارشناسی ارشد اطلاع‌رسانی

اطلاعات، براساس آمار NetCraft (2002) به وسیله بیش از ۲۷ میلیون سرویس‌دهنده وب در اختیار مشتریان قرار می‌گیرد.

حجم وسیع اطلاعات بر روی شبکه وب باعث می‌گردد تا پاسخ‌دهی به کاوش‌های ارسالی را بدون دسترسی به تمام متون و فقط با استفاده از فایل‌های نمایه صورت دهیم زیرا در غیر این صورت یا بایستی نسخه‌ای از اطلاعات درخواستی به صورت محلی ذخیره گردد و یا تمام صفحات از راه دور و از طریق شبکه، در هنگام جستجو دسترس‌پذیر باشد که این روش‌ها بسیار گران و کند است. تمام این‌ها تأثیر و اهمیت، تلاش برای بهبود روش‌های نمایه‌سازی و الگوریتم‌های جستجو را مشخص می‌سازد.

براساس استاندارد نمایه‌سازی بریتانیا نمایه، ترتیب اصولی از مداخل است که به منظور قادر ساختن استفاده‌کنندگان برای یافتن اطلاعات خود در یک مدرک ایجاد می‌شود. نمایه‌سازی وب کار ساده‌ای نیست و لذا برای کمک به درخواست‌کنندگان اطلاعات در رسیدن به آن در سطح شبکه از روش‌های مختلف نمایه‌سازی استفاده می‌گردد.

روش نمایه انتهای کتاب در وب

بسیاری از سایت‌های وب، برنامه‌ای برای جستجوی سایت خود طراحی کرده‌اند. این برنامه‌ها همچون جستجو در فایل‌های تمام متن می‌تواند در نتایج جستجوی خود دارای مدارک نامرتبط و در اصطلاح همراه با ریزش کاذب باشد.

اگر در چنین سایت‌هایی نمایه‌ای شبیه آنچه در انتهای کتاب‌ها دیده می‌شود به وجود آید، مراجعه‌کننده می‌تواند به سرعت و با استفاده از لیست به مدخل مورد نظر خود وارد شود و با سرعت بالا و از دست دادن زمان کمتری به مدارک مورد نیاز خود وحتی به مدارک مرتبط با آن نیز دسترسی یابد.

سایت وب شرکت نرم‌افزاری Adobe با داشتن نمایه‌ای از نوع کواک[۱] و نیز موتور کاوش ویژه سایت، سعی در بالا بردن دسترس‌پذیری موٴثر اطلاعات سایت خود نموده است.

سایت دانشگاهی جورج تاون نیز نمایه موضوعی A-Z را در قسمت ابزارهای دسترسی به اطلاعات در صفحه خانگی خود قرار داده است.

ابرداده[۲] و وب

ابرداده به طور مکرر، داده‌ای برای داده تعریف شده است. این تعریف در عین ضروری بودن ناکافی است. ابرداده، داده‌ای است درباره‌ٔ داده، که برای شرح منابع یا شیء اطلاعاتی پایه‌ریزی شده است و داده‌های منابع و روابط بین آن‌ها را تشریح می‌کند. پدیدآورندگان منابع، ناشران، کتابداران و سایر متخصصان اطلاع‌رسانی می‌توانند ابرداده را تولید کنند. ابرداده می‌تواند در درون منابع اطلاعاتی جاسازی[۳] و یا در کنار منبع اطلاعاتی و به طور مجزا حفظ شود. (Cleveland, 2001,P.223)

قالب ابرداده‌ای دوبلین کور نمونه‌ای پیشنهادی از ابرداده است که دستاورد نشست متخصصان اطلاع‌رسانی در دوبلین اوهایو به منظور حل مشکلات موجود در توصیف منابع اطلاعاتی موجود بر روی شبکه‌های کامپیوتری است. این نمونه‌ای از مفهوم پیوند بین ابرداده و وب است.

شاخص‌های عنوان، پدیدآور، موضوع، ناشر، توصیف (همچون چکیده)، تاریخ ارائه، نوع مدرک، قالب[۴] (نیازهای سخت‌افزاری و نرم‌افزاری جهت ارائه مدرک)، برچسب منحصر به فرد شناسایی[۵]، محل تولید مدرک، زبان اصلی مدرک، چگونگی و محل ارتباط مدرک با سایر منابع، پوشش (بیانگر دامنه، محدوده و عمق مدرک) و مدیری حق موٴلف، در قالب دوبلین کور پیشنهاد گردیده است. (Cleveland, 2001,P.224)

شاخه‌های موضوعی

بعضی از ابزارهای جستجوی وب سعی در مرور سایت‌ها توسط افراد متخصص کرده و پس از تحلیل محتوی سایت، کلیدواژه مناسب را انتخاب و آن را در محل موضوعی، براساس لیست موضوعی ویژه خود قرار می‌دهند

و در واقع یک راهنمای موضوعی را برای استفاده‌کننده فراهم می‌آورند. این در واقع به آن معناست است که در زمانی که موتور کاوش به طور معمول برای هدایت فرد به سایت، تمام صفحات آن سایت را نمایه کرده است، ولی از سوی دیگر یک راهنمای موضوعی بسیار شبیه یک پیوند به صفحه خانگی آن سایت تلقی می‌گردد. (Tyner, 2001)

سایت Open Directory و Looksmart دو نمونه بارز از این نوع سایت‌ها می‌باشند.

فن‌آوری موتورهای کاوش

ساختار متمرکز

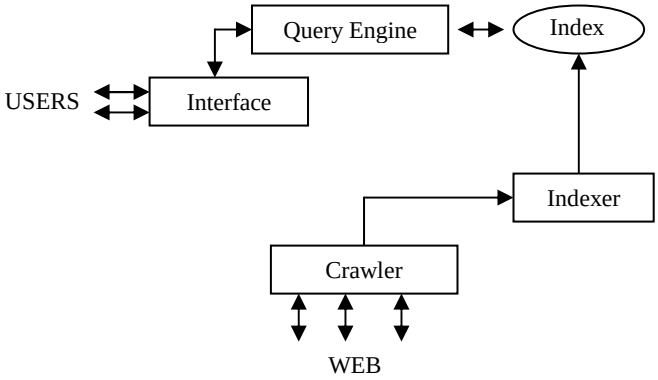
بسیاری از موتورهای کاوش از یک ساختار متمرکز خزنده-نمایه‌ساز[۶] سود می‌جویند. خزنده‌ها برنامه‌های نرم‌افزاری هستند که عمل پیمایش وب را انجام داده و صفحات جدید و یا به روز درآمده را به سرویس‌دهنده‌ای که قرار است این صفحات در آنجا نمایه شوند، می‌فرستند.

یک خزنده به عنوان نقطه شروع یک URL را دریافت کرده و انتقال صفحات وب را-همچون ایستگاه کاری که صفحات را مرور می‌کند-به سرویس‌دهنده آغاز می‌نماید. پس از انتقال یک مدرک، سازه‌یاب[۷] شروع به استخراج واژه‌های مرتبط با متن کرده و آن‌ها را به پایگاه داده می‌افزاید. هر رکورد اطلاعاتی در این پایگاه شامل واژه استخراج شده و URL مربوط به آن می‌باشد. قابل ذکر است، تعدادی از خزنده‌ها واژه‌های موجود در بین برچسب‌هایی خاص نظیر، <TITLE>,<H1>... و یا واژه‌های با بسامد بالا می‌نمایند.

JumpstationII از این نوع است که علاوه بر آن واژه‌های موجود در عناصر <HEADER> (<Hi>,1<=I<=6) و واژه‌های با بسامد بالا در برچسب <BODY> را استخراج می‌کند.

خزنده WWWW واژه‌های موجود در عنصر <TITLE>, URL و <A> (ابر پیوندهای موجود در مدرک) را نمایه می‌کند.

در حقیقت خزنده به درون رایانه نفوذ نمی‌کند، بلکه بر روی یک رایانه محلی اجرا شده و درخواست‌های خود را به رایانه‌های سرویس‌دهنده در نقاط مختلف ارسال می‌کند. عمل نمایه‌سازی در این روش به طور متمرکز صورت می‌گیرد. شکل زیر ساختار نرم‌افزاری متمرکز موتور کاوش AltaVista را نشان می‌دهد.



اصلی‌ترین مشکل در این نوع ساختار، به دلیل طبیعت پویای وب، جمع‌آوری اطلاعات، پیوندهای ارتباطی اشباع شده به سرویس‌دهندگان وب و سربار شدن[۸] آن‌هاست. مشکل دیگر حجم اطلاعاتی است که در حقیقت ساختار متمرکز نمایه‌سازی توان مقابله با آن را ندارد. (Baeza-Yates, 1999, P.374)

در واقع به دلیل درخواست‌های سریع ارسالی خزنده‌ها، و اشغال حجم قابل توجه‌ای از پهنای باند ارتباطی-و حتی تمام پهنای باند در حوزه‌های کوچک[۹]-و برای رفع این مشکل در ۳۰ ژوئن سال ۱۹۹۴ استانداردی را پدیدآوردگان برنامه‌های خزنده، برای محدودیت عمل برنامه‌های خودکار خزنده در سراسر وب به وجود آوردند. براساس این استاندارد، اگر ما می‌خواهیم خزنده‌ها از سرویس‌دهنده ما بازدید نکنند بایستی فایلی متنی با نام robots.txt بر روی ریشه[۱۰] سرویس‌دهنده‌ٔ وب خود قرار دهیم و براساس توافقات پدیدآوردندگان خزندگان یا روبوت‌ها، این برنامه بایستی در اولین مرحله به دنبال این فایل در روی سرویس‌دهنده‌ٔ وب جستجو کند و در صورت وجود به فرامین قرار داده شده در آن پاسخ داده و سپس به اعمال دیگر بپردازد.

البته با استفاده از ابربرچسب[11] ROBOTS نیز می‌توان از نمایه شدن صفحهٔ وب توسط روبوت جلوگیری کرد. (Sullivan, 2000)

```
<HEAD>
<TITILE>Page I Don`t Want To Search Engines</TITLE>
<META NAME=”ROBOTS” CONTENT =”NOINDEX”>
</HEAD>
```

ساختار پراکنده

این نوع ساختار بسیار موٴثرتر از نوع متمرکز است و مشکلات ناشی از استفاده از ساختار متمرکز چون:

۱) دریافت درخواست صفحات، توسط خزنده‌های گوناگون موتوهای کاوش، از سرویس‌دهندگان وب،

۲) افزایش ترافیک در وب به علت استخراج تمام اشیا و اجزا صفحات وب و نادیده گرفتن اکثر آن‌ها هنگام نمایه‌سازی؛

۳) جمع‌آوری اطلاعات بدون همکاری و آگاهی سایر خزنده‌ها و موتورهای کاوش، در این ساختار مورد توجه قرار گرفته و مرتفع شده است.

برای رفع مشکلات ذکر شده، Harvest یکی از مهمترین مدل‌ها در این نوع دو عنصر اصلی را معرفی می‌کند: گردآورنده[12] و واسط[13]. (Baeza-Yates, 2000, P.375)

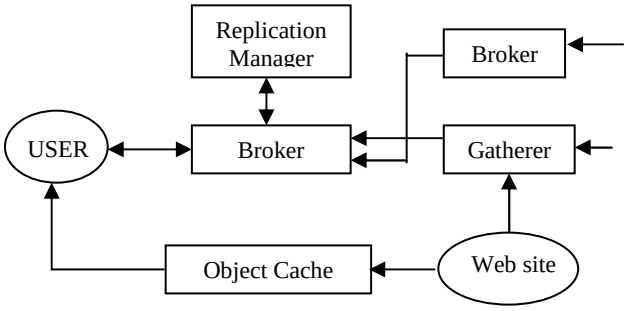
گردآورنده یک نرم‌افزار خودکار است که بر روی سرویس‌دهندهٔ وب اجرا می‌شود و عمل جمع‌آوری و استخراج اطلاعات لازم برای فایل نمایه را انجام می‌دهد. البته این نرم‌افزار بر روی سایر سرویس‌دهندگان وب نیز می‌توان اجرا شود که این خود با ساختار Haverst در تناقض است.

واسط به استخراج اطلاعات از گردآورنده‌ها پرداخته و ضمن ایجاد یک نمایه قابل جستجو، واسط کاربری آن را نیز فراهم می‌کند. همان گونه که دیده می‌شود، یک نرم‌افزار گردآورنده بر روی سرویس‌دهندهٔ وب، بدون هیچ ترافیک خارجی اجرا شده و اطلاعات جمع‌آوری شده به چندین واسط قابل ارسال می‌باشد. هر واسط می‌تواند بعد از فیلتر کردن اطلاعات آن را به سایر واسط‌ها ارسال کند.

یکی از اهداف Harvest ایجاد واسط‌های ویژه موضوعی[14] و اجتناب از حوزهٔ وسیع لغات و مشکلات نمایه‌های عمومی است.

ساختار Harvest، تکرارکننده‌ها و حافظه‌های نهانی اشیا[15] را نیز جهت افزایش سرعت دسترسی به پایگاه داده‌ها فراهم می‌کند. امروزه برنامه‌های کاربردی این شیوه در مراکزی چون ناسا[16] و آکادمی ملی علوم آمریکا مورد استفاده قرار می‌گیرد و نمونه مورد استفاده در بخش تجاری در وب، سرویس‌دهنده فهرست، شرکت Netscape است.

نمونه‌ای از ساختار Harvest در شکل زیر دیده می‌شود:



نتیجه‌گیری

افزونی افسار گسیخته اطلاعات بر روی شبکه وب، متخصصان رایانه و اطلاع‌رسانی را در جهت بهبودی کارائی نظام‌های نمایه‌سازی و به تبع آن بازیابی اطلاعات سوق می‌دهد.

حرکت از سوی نمایه‌های دست‌ساز تا نمایه‌های خودکار و ارائهٔ شیوه‌های گوناگون آن، به جهت سرعت بخشیدن برای ارائهٔ اطلاعات و رفع مشکلات فنی موجود بوده است. با وجود این علیرغم خوش‌بینی در مورد ابزارهای نمایه‌ساز و جستجوگر، هنوز هم بازیابی موضوعی در هر پایگاهی، بر اساس نمایه‌سازی کلیدواژه‌ای صورت می‌گیرد و جستجو براساس منطق بولی بوده و ریزش کاذب در آن قابل توجه است.

اهداف آتی در این حوزه، مطالعه هر چه بیشتر پردازش اطلاعات توسط انسان و چگونگی فهم انسان از اطلاعات و رسیدن به نظام‌های هوشمند نمایه‌ساز و بازیاب، با استفاده از تجارب و نتایج امیدوارکننده‌ٔ حوزه‌هایی نظیر هوش مصنوعی و نظام‌های خبره می‌باشد.

پی‌نوشت‌ها

1. KeyWord In Context
2. Metadata
3. Embedded
4. Format
5. Identifier
6. Crawler-indexer
7. Parser
8. High load
9. Domain
10. Root
11. Meta Tag
12. Gathere
13. Broker
14. Topic-specific
15. Object Cache
16. NASA

منابع

1.*American Society of Indexer*. Main Page: http://www.asindexing.org
2. Baeza-Yates, Ricardo;Ribeiro-Note, Berthier (1999). *Modern Information Rerieval*. NewYork: ACM Press
3. Cleveland, Donald B.; Cleveland, Ana D. (2001). *Introduction to Indexing and Abstracting*. Englewood: Libraries Unlimited
4. *Distributed Indexing Systems for Organizing the Web:*
http://eubdl.ugr.es/temp/serbydor/g9/know4_i.htm
5. George Town University website: http://www.gerogertown.edu
6. *Netcraft*. Main Page: http://www.netcraft.com
7. *Open Directory* Homepage: http://www.dmoz.org
8. Pastore, Michael (2000). *The Web: More Than 2 Billion Pages Strong*
http://cyberatlas.internet.com/big_picture/traffic_patterns/article/0,,5931_413691,00.htm
9. Sullivan, Danny (2000). *How to use HTML Meta Tags*.
http://searchenginewatch.com/webmasters/meta.html
10. *Toward the automation of a routine task: Using Spidrs to index the Web:*
http://eubdl.ugr.es/temp/serbydor/g9/know2_i.htm
11. Tyner, Ross(2001). *Sink or Swim: Internet Search Tools & Techniques*.
http://www.ouc.bc.ca/libr/connect96/search.html