

سیستم جامع اطلاع رسانی سیمرغ

روشهای جستجوی پیشرفته در سیمرغ نوسا



نوسا 

شرکت نرم افزار و سخت افزار ایران
www.nosa.com

تهران، خیابان سپهبد قرنی، خیابان شاداب شرقی، پلاک ۶
تلفن/نمابر: ۸۸۸۲ ۲۹ ۷۹ - ۸۸۸۳ ۵۳ ۶۰
پست الکترونیک: info@nosa.com



مروری بر مفاهیم جستجو

در گذشته کتابخانه ها و مراکز اطلاع رسانی برای ارائه مجموعه اسناد خود از روش فهرست برگه های مرتب شده در برگه دانها استفاده می کردند. به این صورت که اطلاعات کتابشناختی مهم این اسناد طبق یک قانون و مدل خاص (ISBD) در قالب یک فهرست (کاتالوگ) روی یک برگه با ابعاد خاص نوشته و در برگه دان قرار می گرفت. این برگه ها بر اساس اولین فیلد (شناسه) برگه آرابی می شد تا محقق بتواند سند دلخواه خود را جستجو کند. واضح است که در این سیستم محقق باید شکل کامل و صحیح عبارت مورد نظر خود را می دانست و سپس به جستجو می پرداخت. بخصوص واژه ابتدایی بسیار اهمیت داشت و اگر محقق واژه ابتدایی را فراموش کرده بود یا بطور صحیح نمی دانست به نتیجه نمی رسید. البته ابداع شناسه های افزوده و اضافه کردن آن به بخش اطلاعات تحلیلی اسناد تا حد کمی این مشکل را حل کرده بود اما بهر حال نقایص عمده و اساسی داشت که ادامه آن بخصوص پس از حجیم شدن اطلاعات و مجموعه ها امکان پذیر نبود. این نقایص را می توان در چند مورد خلاصه کرد:

(۱) وابسته بودن آماده سازی، تولید و برگه آرابی به اشخاص و امکان بروز اشتباه در این مراحل و نیز وابسته بودن فرایند جستجو به شخص که در این مرحله نیز ممکن است به علت عدم آشنایی محقق با قواعد برگه آرابی و نیز اشتباه چشمی او عمل جستجو موفقیت آمیز نشود.

(۲) **شناسه های محدود برای دسترسی به اسناد**: در این سیستم برای شناسه های محدودی برگه تولید و در برگه دان قرار داده می شود. آنچه که معمول بوده فیلدهای نویسنده، عنوان، و حداکثر موضوع دارای برگه های مستقل بوده و مدخل جستجو قرار می گیرند. بدیهی است که فیلدهایی نظیر ناشر، محل نشر، سال نشر، مندرجات، یادداشتهای، ... قابل جستجو نیستند.

(۳) **عدم امکان جستجوی واژه های میانی و انتهایی شناسه ها**: همانطور که قبلاً نیز توضیح داده شد، روش دستی به واژه ابتدایی و آغازین شناسه حساس است. به عبارت دیگر واژه های میانی شناسه ها قابل جستجو نیستند و نمی توان مستقل آنها را بازیابی کرد. برای مثال در برگه دان عنوان، کتابی با عنوان "آیین نگارش و دستور زبان فارسی" را نمی توان از طریق واژه های "نگارش"، "دستور"، "زبان" و "فارسی" پیدا کرد.

(۴) **عدم امکان ترکیب واژه ها و شناسه ها در جستجو**: در جستجوی دستی امکان ترکیب دو شناسه وجود ندارد. به این معنی که در هر مرحله فقط می توان در یک برگه دان (شناسه) جستجو کرد. برای مثال با می توان کتابهای یک نویسنده خاص (عبدالکریم سروش) را پیدا کرد یا اینکه در موضوع خاصی (فلسفه) جستجو کرد. در این مثال نمی توان از سیستم دستی انتظار بازیابی کتابهای فلسفی دکترا سروش را (در یک مرحله) داشت.

بنا به دلایل نامبرده و نیز به علت حجم عظیم اطلاعات موجود در کتابخانه ها و مراکز اطلاع رسانی، و نیاز به دستیابی هرچه سریعتر به این اطلاعات، روشهای سنتی و دستی دیگر جوابگو نبوده و استفاده از نرم افزارهای کامپیوتری برای ذخیره، بازیابی و پردازش اطلاعات پدیده رایجی شده است. نرم افزارهایی که این وظیفه را انجام می دهند اصطلاحاً سیستم مدیریت پایگاههای اطلاعاتی (Database Management System) نامیده می شود. این نرم افزارها اطلاعات کتابشناختی اسناد را بصورت رکورد به رکورد نگهداری می کنند که هر رکورد از تعدادی فیلد تشکیل می شود و هر فیلد نیز از تعدادی واژه تشکیل می شود. هر رکورد در بانک اطلاعاتی مشابه و متناظر با یک فهرست برگه در سیستم دستی است. بدیهی است که در بانکهای اطلاعاتی نقایص سیستم دستی رفع شده و ویژگیهای زیر موجود است:

(۱) **مکانیزه بودن مراحل پردازش اطلاعات**: اگرچه در مرحله ورود اطلاعات سیستم به فرد وابسته است و ممکن است خطاهای انسانی در این مرحله بانکها را تحت تاثیر قرار دهد اما در مراحل دیگر پردازش مانند مرتب سازی، گزارشهای چاپی، انتقال اطلاعات و غیره سیستم کاملاً مکانیزه است و امکان اشتباه و خطا در آن وجود ندارد.

(۲) **امکان ایجاد شناسه و مدخل جستجو برای تمامی فیلدها**: مداخل جستجو دیگر محدود به نویسنده، عنوان و موضوع نیست و تمام اطلاعات موجود در یک رکورد امکان جستجو دارند.

(۳) **امکان جستجو برای تمام واژه ها**: تک تک واژه های فیلدهای موجود، صرف نظر از موقعیت مکانی آنها در عبارت، قابل جستجو هستند. تمام واژه های ابتدایی، میانی و انتهایی یک فیلد (مثلاً عنوان) قابل بازیابی اند.

(۴) **امکان ترکیب کلیدواژه ها در جستجو**: یکی از اصلی ترین مزایای سیستم های کامپیوتری که تاکید اصلی این مقاله نیز روی آن است، امکان ترکیب کلیدواژه ها در جستجوست. به این صورت که نرم افزار به تعداد کلیدواژه ها عمل جستجو در واژه نامه را انجام می دهد و در انتها با ترکیب مجموعه جوابها، نتیجه نهایی را بیرون می دهد. ابزاری که بانک اطلاعاتی برای ترکیب کلیدواژه ها استفاده می کند عملگرهای جستجو نام دارد. در واقع یک جستجوی کامپیوتری، فرایند ظریف و پیچیده ای است که با یک یا چند کلیدواژه در ذهن محقق آغاز می شود و معمولاً در چند مرحله با ترکیب های متفاوت این کلیدواژه ها توسط عملگرها ادامه می یابد تا محقق به نتیجه دلخواه خود برسد.

همانطور که اشاره شد هدف اصلی این مقاله تشریح فرایندها و امکانات ترکیب جستجوها در سیمرغ است که توسط عملگرها انجام می شود. بطور کلی این عملگرها دو کار عمده را انجام می دهند:



- **حذف نتایج اضافی و محدود کردن جستجو:** بدیهی است که جستجو باید به نحوی انجام گیرد که تا حد امکان نتایج غیرمربوط و نامناسب حذف شود. در واقع این عملگرهای جستجو هستند که به مانع بودن جستجو کمک می کنند.
 - **افزایش نتایج جستجو و گسترش آن:** یعنی اینکه تا حد امکان همه موارد مناسب و مربوط به جستجوی مورد نظر آورده شود و چیزی از دست نرود. این خاصیت را می توان جامع بودن جستجو نامید.
- با این وجود، و با توجه به حجم عظیم اطلاعات روزافزون در بانکهای اطلاعاتی، بنظر می رسد که از میان دو کارکرد اصلی عملگرهای جستجو، خاصیت مانع بودن و محدود کردن نتایج جستجو و هدایت آن به سمتی که با حذف نتایج اضافی به هدف مورد نظر محقق برسد اصلی ترین کارکرد عملگرهای جستجو است. بدون این عملگرها عملاً امکان بازیابی اطلاعات وجود نخواهد داشت و محقق در دریایی از اطلاعات اضافی غرق خواهد شد.

انواع عملگرها

بسته به طراحی ساختار بانک اطلاعاتی، و ترتیب اهمیت فیلدها، معمولاً چند مدخل برای جستجو در نظر گرفته می شود. مدخل جستجو همان جایی است که محقق عبارت مورد نظر خود را برای بازیابی وارد می کند و به جستجو می پردازد. برای مثال چند مدخل جستجو از پایگاه اطلاعاتی «کتابهای فارسی» به این شرح است: شماره راهنما، عنوان، پدیدآور، تالگان، تاریخ انتشار، موضوع و ... با توجه به تعدد و گوناگونی این مداخل، عملگرهای جستجو برحسب این که درون یک مدخل باشند یا بین چند مدخل عمل کنند به دو دسته اصلی عملگرهای بین فیلدی و عملگرهای درون فیلدی تقسیم می شود:

عملگرهای بین فیلدی

همانطور که از نام آنها پیداست، این گونه عملگرها میان دو یا چند مدخل جستجو عمل می کنند. برای مثال فرض کنید شخصی به دنبال کتابهای با موضوع سیاست است و از آن میان بیشتر به دنبال کتابهایی است که توسط انتشارات طرح نو منتشر شده است. لذا این شخص باید دو مدخل «موضوع» و «ناشر» را به شکل زیر ترکیب کند.

موضوع: سیاست

و

ناشر: طرح نو

در این مثال، این دو مدخل با عملگر «و» با هم ترکیب شده اند. لازم به توضیح است که با توجه به تفاوت محتوایی مداخل، منطقاً فقط عملگرهای بول (Boolean Operators) («و»، «یا»، «بجز») و معادل لاتین آنها (AND, OR, EXCEPT) می تواند بین مداخل بکار رود و عملگرهای پیشرفته نمی توانند بین مدخل ها قرار بگیرند. به مثالهای مربوط به عملگرهای «یا» و «بجز» دقت کنید:

فرض کنید شخصی به دنبال کتابهایی است که شرکت شیلات ایران منتشر کرده است. در این جستجو مشخص نیست که شرکت شیلات ناشر یا پدیدآورنده (تالگان) کتابهاست. لذا در این جستجو باید از عملگر «یا» برای ترکیب دو مدخل پدیدآورنده و ناشر استفاده شود:

پدیدآورنده: شیلات

یا

ناشر: شیلات

برای آخرین مثال فرض کنید شخصی به دنبال کتابهای با موضوع زبانشناسی است که نعام چامسکی نوشته است. از آنجایی که چامسکی آثاری در موضوع علم سیاست نیز دارد لذا محقق می تواند برای بازیابی کتابهای غیرسیاسی این نویسنده، از روش زیر استفاده کند:

پدیدآور: چامسکی

بجز

موضوع: سیاست

عملگرهای درون فیلدی

این عملگرها در درون یک مدخل بکار می روند و دو یا چند کلمه یا عبارت را با هم ترکیب می کند. در نرم افزار سیمرغ علاوه بر عملگرهای بول (و، یا، بجز) Boolean Operators) و عملگر پسوندهای عمومی (\$)، که بطور معمول در نرم افزارهای بانک اطلاعاتی پشتیبانی می شود، طیف وسیعی از عملگرهای جستجوی پیشرفته پشتیبانی می شود که به این ترتیب می باشند:

همتکرار (Soccr)، نزدیک (Near)، همجوار (Adjacent)، همه (All)، هیچ (None)، تا (To)، .. ، = ، « > ، < ، > ، < ، = ، > »

این عملگرها را برحسب کارکردی که دارند می توان به چهار دسته کلی عملگرهای همسایگی، عملگرهای وجودی، عملگرهای تعیین محدوده و عملگرهای مقایسه ای تقسیم



بندی کرد که در ذیل پس از معرفی عملگر پسوند عمومی (\$) به تفصیل توضیح داده شده است.

عملگر پسوند عمومی «\$» (Truncation)

همانطور که می دانید واژه ها در زبان فارسی بر دو نوع اند: ساده و مرکب. به عنوان مثال به واژه کتاب توجه کنید. این واژه یک واژه ساده است که به افزودن پسوندهای مختلف می توانیم واژه های مرکب گوناگون از این کلمه بسازیم. کلماتی از قبیل:

کتاب، کتابها، کتابی، کتابداری، کتاب فروشی، کتاب سوزی، کتاب سازی، کتاب خوانی

بطور کلی وقتی کاربری به دنبال یک واژه جستجو می کند به ندرت پیش می آید که فقط به دنبال کلمات ساده باشد. زیرا که کلمات مشتق از آن واژه نیز از نظر موضوعی ارتباط تنگاتنگی با آن دارد و به احتمال قریب به یقین منظور او را برآورده خواهد کرد. لذا در گام اول بهتر است که برای جامعیت جستجو، واژه مورد نظر با تمام پسوندهای ممکن ترکیب شده و مجموعه همه واژه ها با هم جستجو شود. در نرم افزار سیمرغ برای راحتی کار عملگری پیش بینی شده که عملگر پسوند عمومی نام دارد و به شکل (\$) می باشد. قرار گرفتن این کاراکتر در انتهای هر واژه به این معناست که واژه موردنظر با تمام پسوندهای ممکن باهم جستجو شود و در انتها نتایج جستجوها با عملگر «یا» با هم ترکیب شود. یعنی اینکه جستجوی:

عنوان : کتاب\$

هم ارز است با جستجوی

عنوان : کتاب یا کتابها یا کتابی یا کتابداری یا کتاب فروشی یا کتاب سوزی یا کتاب سازی یا کتاب خوانی یا (مشتقات دیگر)

توصیه می شود کاربر در ابتدای جستجوی خود از این عملگر برای جامعیت جستجوی خود استفاده کند و در صورتی که نتایج بسیار زیاد شد در مراحل بعد با استفاده از عملگرهای دیگر جستجو، نتیجه جستجوی خود را دقیق تر و جزئی تر کند.

کاربرد دیگر این عملگر، زمانی است که کاربر املائی دقیق کلمه ای را نمی داند. این مساله بیشتر زمانی رخ می دهد که کاربر می خواهد منابع به زبان غیر از زبان مادری خود را جستجو کند. برای مثال کاربری به دنبال معادل انگلیسی واژه هیدرولیک است. اما از آنجا که در مورد املائی این کلمه بین واژه Hydrolic و Hydraulic تردید دارد لذا بجای آنها از عبارت «Hydr\$» استفاده می کند که دو واژه یاد شده و واژه های دیگری مانند Hydrometry، Hydrotherapy و ... را نیز شامل می گردد. برای مثالی در زبان فارسی فرض کنید می خواهیم نویسنده ای را با نام «حاج زاده» جستجو کنیم و از طرفی احتمال می دهیم که نام خانوادگی او «حاجی زاده» باشد. در این حالت بهتر است عبارت «حاج\$» را جستجو کنیم. البته در این مورد نیز علاوه بر دو نام یادشده، نامهایی نظیر «حاجیان»، «حاجی مرادی»، «حاجی نیا» و ... نیز بازیابی می شود که زائد می باشد. همانطور که این مثالها نیز نشان می دهد در این موارد، احتمال بازیابی جوابهای غیرمربوط بسیار بالا می رود و لذا در کاربرد آن باید دقت کرد.

عملگرهای همسایگی

همانطور که گفته شد در عمل جستجو معمولاً دو یا چند کلیدواژه در یک مدخل جستجو وارد می شود. بنا بر این اولین موردی که اهمیت پیدا می کند وجود داشتن کلیدواژه ها در مدخل مورد نظر است. اما تنها وجود آنها مشکل را حل نمی کند بلکه تقدم و تاخر کلیدواژه ها، فاصله آنها از هم، تعلق آنها به فیلدهای مختلف تشکیل دهنده واژه نامه یک مدخل و موقعیت قرار گرفتن آنها در تکرارهای مختلف فیلدهای تکرار پذیر نیز در میزان نزدیک شدن رکورد به خواسته ما بسیار موثر و با اهمیت است. لازم به توضیح است که بعضی از مداخل جستجو محتویات چند فیلد را در بر می گیرد. بعنوان مثال مدخل پدیدآور شامل فیلدهای «پدیدآور سرشناسه»، شرح پدیدآور و پدیدآورهای افزوده است یا مدخل عنوان که شامل فیلدهای عنوان اصلی، عنوان قراردادی، مندرجات و عناوین افزوده است. در این موارد تعلق دو کلیدواژه مورد نظر ما به یک فیلد اهمیت پیدا می کند. در مورد فیلدهای تکرارپذیر نیز تعلق کلیدواژه ها به یک تکرار جستجو را دقیق تر و صحیح تر خواهد کرد.

و اما از آنجایی که عملگر «و» تنها به وجود کلیدواژه ها حساس بوده و نسبت به موارد یادشده بی تفاوت است برای رسیدن به نتایج جستجوی مناسب کافی نیست. لذا برای رفع این مشکل در سیمرغ عملگرهایی طراحی شده که به موقعیت مکانی، تعلق به یک فیلد خاص و وضعیت تکرار کلیدواژه ها حساس اند و عملگرهای همسایگی نامیده می شوند. این عملگرها عبارتند از:

«همتکرار»، «نزدیک» و «همجوار». عملگر «همتکرار» وجود کلیدواژه ها را در یک فیلد و در یک تکرار تضمین می کند و عملگر «نزدیک» همان خاصیت عملگر همتکرار را دارد به اضافه اینکه به فاصله کلیدواژه ها نیز حساس است و در انتها عملگر «همجوار» است که ضمن داشتن خواص عملگرهای «همتکرار» و «نزدیک»، به تقدم و تاخر کلید واژه ها نیز حساس می باشد. برای روشن تر شدن مطلب، هر کدام از این عملگرها با مثالهای مربوط به خود در زیر شرح داده می شود:

همتکرار (SOCC)

یکی از امکانات سیمرغ در زمینه جستجو این است که می تواند محتوای چند فیلد جدا از هم را در یک مدخل مشترک جستجو کند. برای مثال در پایگاه کتابهای فارسی و در مدخل جستجوی پدیدآور می توان پدیدآور سرشناسه، شرح پدیدآور و پدیدآوران افزوده را جستجو کرد. یا مثلاً مدخل جستجوی عنوان که در آن می توان عنوان اصلی، عناوین فرعی، مندرجات و عناوین افزوده را جستجو کرد. یکی دیگر از امکانات سیمرغ وجود فیلدهای تکرارپذیر در ساختار پایگاههاست (مانند فیلدهای موضوع، فروست، شناسه های افزوده در پایگاه کتابهای فارسی). در این مورد هم اگرچه تکرارهای مختلف از جهتی با هم مرتبط اند اما هر تکرار یک موجودیت مستقل است و ماهیتی جدا از تکرارهای دیگر



دارد. در جستجوی عادی که کلیدواژه‌ها با عملگر «و» با هم ترکیب می‌شوند، این دو مورد در نظر گرفته نمی‌شود و لذا شمار زیادی از نتایج نامربوط برای کاربر بازیابی می‌شود که او را دچار مشکل کرده و از مسیر مناسب منحرف می‌کند. مثالهای زیر به روشن تر شدن مفاهیم بالا کمک می‌کند:

• مثال در مورد تعلق کلید واژه‌ها به فیلدهای مختلف:

فرض کنید کتابی دارای دو پدیدآور به شکل زیر باشد:

پدیدآور سرشناسه: صالحی، علی

پدیدآور همکار: معتمدی، حسین

حال اگر کاربری به دنبال کتابهای نویسنده ای بنام «علی معتمدی» باشد، و بصورت ساده واژه‌های «علی» و «معتمدی» را با عملگر «و» با هم ترکیب کند:

پدیدآور: علی و معتمدی

با توجه به اینکه کلید واژه «علی» در یک فیلد و کلید واژه «معتمدی» در فیلد دیگر آمده و مدخل پدیدآور شامل محتوای هر دو فیلد است، لذا نرم افزار کتاب فوق را نیز به عنوان یکی از نتایج جستجو بازیابی خواهد کرد. کاملاً واضح است که این کتاب با خواسته او مرتبط نیست و یک نتیجه زائد می‌باشد. برای رفع این مشکل، جستجو باید با استفاده از عملگر «همتکرار» و به شکل زیر انجام شود:

پدیدآور: علی همتکرار معتمدی

در این حالت تنها و تنها کتابهایی بازیابی خواهد شد که واژه‌های علی و معتمدی همزمان در یک نام یک نویسنده موجود باشند. وجود این قید کافی است تا کلیدواژه‌ها در هر فیلد به شکل مستقل جستجو شود و در نتیجه آن، رکوردهای مناسب برای محقق بازیابی شود.

• مثال در مورد تعلق کلید واژه‌ها به تکرارهای مختلف یک فیلد تکرار پذیر

فرض کنید شخصی به دنبال کتابهای با موضوع جنگهای ایران و روس می‌گردد. اگر این شخص کلمات جنگ و ایران و روس را به شکل ساده با عملگر «و» با هم ترکیب کند ممکن است کتابی با موضوعات زیر را نیز بازیابی کند:

موضوع اول: جنگهای روس و عثمانی

موضوع دوم: جنگهای عثمانی و ایران

واضح است که این کتاب در مورد جنگهای ایران و روس نیست اما از آنجا که کلیدواژه روس در تکرار اول (موضوع اول) آمده و کلیدواژه ایران در تکرار دوم (موضوع دوم)، لذا بازیابی آن با روش فوق کاملاً طبیعی است. لذا برای پیشگیری از چنین نتایجی این جستجو باید به شکل زیر انجام شود:

موضوع: جنگ \$ همتکرار روس همتکرار ایران

در این حالت نیز وجود قید همتکرار باعث می‌شود تا کتابهایی بازیابی شود که هر سه کلید واژه را همزمان در یک موضوع (تکرار) داشته باشند. مثالهای فوق اهمیت عملگر همتکرار را بخوبی نشان می‌دهد. و اما اگر بخواهیم علاوه بر همتکرار بودن فاصله کلیدواژه را نیز در نظر بگیریم باید از عملگرهای دیگری که در بخشهای بعدی توضیح داده شده استفاده کنیم.

نزدیک (NEAR)

این عملگر تمام خواص عملگر همتکرار را دارد به علاوه این که به فاصله کلید واژه‌ها نیز حساس است. بر اساس قاعده ای که در اکثر مواقع صادق است، در یک عبارت هر چه کلیدواژه‌ها به هم نزدیک تر باشد به موضوع مورد نظر نزدیک تر می‌گردد. به عبارت دیگر قرار گرفتن واژه‌های دیگر میان کلیدواژه‌های مورد نظر ما، رکورد مربوطه را از خواسته ما دور می‌کند. برای مثال سه عنوان زیر را که متعلق به سه کتاب مختلف است در نظر بگیرید:

عنوان ۱: تاریخ اسلام

عنوان ۲: تاریخ معماری در اسلام

عنوان ۳: آذربایجان در سیر تاریخ ایران: از آغاز تا اسلام

واضح است که عنوان ۲ اگرچه هنوز به نوعی به موضوع تاریخ اسلام مربوط می‌شود اما بطور مستقیم به موضوع تاریخ اسلام که به تحولات صدر اسلام و نحوه شکل گیری و



توسعه اسلام در آغاز آن می پردازد مربوط نیست و عنوان ۳ هم که کاملاً از موضوع جداست و دقیقاً عکس آن چیزی است که جستجوکننده به دنبال آن بوده است. لذا در مواقعی که نتایج جستجوی با عملگر **همتکرار** نیز زیادتز از حد مناسب است می توان بجای آن از عملگر **نزدیک n** استفاده کرد. توضیح اینکه عدد n در این عملگر تعیین کننده حداکثر فاصله ای است که می تواند میان دو کلیدواژه باشد. به این صورت که عدد (n-1) نشان دهنده حداکثر کلماتی است که می تواند میان دو کلیدواژه قرار بگیرد. بدیهی است در صورتی که بخواهیم هیچ کلمه ای میان این دو کلیدواژه قرار نگیرد باید از عملگر **نزدیک** استفاده شود. به مثالی در این زمینه توجه کنید:

فرض کنید می خواهیم کتابهای در مورد **تاریخ تمدن** را بازبایی کنیم. لذا برای این کار از عملگر **نزدیک ۴** به شکل زیر استفاده می کنیم:

عنوان: تاریخ **نزدیک ۴** تمدن

این جستجو عناوینی را که حاوی واژه های **تاریخ و تمدن** است بازبایی می کند به شرطی که در این عنوانی میان این دو واژه حد اکثر ۳ کلمه فاصله وجود داشته باشد. دقت کنید که در این مورد تقدم و تاخر این واژه ها مهم نیست (در موارد شماره ۱ و ۳ واژه تمدن قبل از تاریخ آمده است.)

(۱) **افول یک تمدن**: تاریخ روم باستان(بدون فاصله)

(۲) **تاریخ تمدن**(بدون فاصله)

(۳) گفتگوی **تمدنها** در **تاریخ**(با یک کلمه فاصله)

(۴) **تاریخ** علم در **تمدن** اسلامی(با دو کلمه فاصله)

(۵) **تاریخ** ارتباطات میان دو **تمدن** ایران و مصر(با سه کلمه فاصله)

با نگاهی به مثالهای فوق می توان به این نتیجه رسید که عملگر نزدیک اگرچه تا حد زیادی در حذف نتایج اضافی جستجو موثر است اما چون به تقدم و تاخر کلیدواژه ها حساس نیست باعث بازبایی درصدی از پاسخهای زائد می گردد. لذا برای دقت بیشتر و محدود کردن نتایج، به عملگر دیگری نیاز است که در بخش بعد توضیح داده شده است.

همجوار (ADJACENT)

تجربه نشان می دهد که کاربران و محققان بطور معمول دو کلمه (کلیدواژه) را در کنار هم جستجو می کنند. برای مثال: آبیاری قطره ای، تاریخ تمدن، گرافیک کامپیوتری و ... یکی از اساسی ترین موارد در موتورهای جستجو، نحوه و استراتژی برخورد با این نوع جستجوهاست. در اکثر موتورهای جستجو و پایگاههای داده ها، عملگر «و» بطور اتوماتیک بین این دو یا چند کلمه قرار داده می شود و فرض می شود که محقق ترکیب عطفی میان این دو را در نظر داشته است. همانطور که قبلاً بطور کامل توضیح داده شده این روش مناسب نیست. عملگرهای «همتکرار» و «نزدیک» نیز با وجود اینکه از عملگر «و» مناسب تر می باشند اما باز بخاطر عدم توجه به تقدم و تاخر کلیدواژه ها درصدی از خطا را از خود نشان می دهند.

بهرحال تجربه عملی روی این قضیه، طراحان نرم افزار در شرکت نوسا را به این نتیجه رهنمون کرد که در این نوع موارد بهتر است که عملگر دیگری بنام همجوار n تعریف شود و مثالهای یاد شده مورد خاصی از این عملگر باشد. عملگر همجوار همانطور که از نام آن پیداست، همجواری و در کنار هم بودن کلیدواژه ها را تضمین می کند و درواقع تمام خواص عملگر نزدیک را دارد به علاوه این که ترتیب کلمات را نیز در نظر می گیرد. لازم به توضیح است که در حالت های خاصی که دو کلیدواژه بلافاصله بعد از هم می باشد نیازی به وارد کردن این عملگر نیست (نبودن عملگر، هم ارز است با عملگر همجوار ۱ که در این مورد خاص $n=1$ نیازی به وارد کردن عدد ۱ هم نیست). برای مثال به جستجوی زیر دقت کنید:

عنوان: گرافیک \$ کامپیوتر\$

این جستجو کاملاً هم ارز است با این جستجو:

عنوان: گرافیک \$ همجوار کامپیوتر\$

در مثال فوق تنها کتابهایی بازبایی خواهد شد که در عنوان آنها عبارت «**گرافیک \$ کامپیوتر\$**» وجود داشته باشد. به چند مورد از جوابهای احتمالی این جستجو دقت کنید:

(۱) مبانی **گرافیک کامپیوتری** برای کمودور ۶۴

(۲) **گرافیک کامپیوتری** با توربوپاسکال

(۳) مایا: نرم افزاری پیشرفته در **گرافیک کامپیوتری**

شکی نیست که در این روش نیز ممکن است بعضی از جوابهای بسیار نزدیک به هدف مورد نظر بازبایی نشود. به عبارت دیگر جامعیت جستجو در قبال مانع بودن آن کمتر می شود. اما با توجه به همه جوانب قضیه، در پایگاههای اطلاعاتی با حجم بالا، این روش بهتر و سریعتر به هدف می رسد. با همه اینها، برای اینکه جستجو جامع تر شود می توان از عملگر همجوار با اعدادی بیشتر از یک استفاده کرد مثلاً:



عنوان: گرافیک همجوار ۴ کامپیوتر \$

این جستجو کتابهایی را بازیابی خواهد کرد که در عناوین آنها اولاً دو کلیدواژه «گرافیک» و «کامپیوتر» وجود داشته باشند و ثانیاً میان آنها حداکثر ۳ کلمه فاصله وجود داشته باشد و نیز واژه «گرافیک» قبل از «کامپیوتر» آمده باشد. مجموعه جوابهای این جستجو علاوه بر جوابهای قبلی می تواند شامل موارد زیر باشد:

(۱) طراحی مدار کارتهای گرافیک برای کامپیوترهای شخصی (با یک کلمه فاصله)

(۲) محیطهای گرافیکی در برنامه نویسی کامپیوتری (با دو کلمه فاصله)

(۳) واسط گرافیکی در سیستم عامل کامپیوترهای اپل-مک اینتاش (با سه کلمه فاصله)

واضح است که این سه مورد نیز اگر نه بطور مستقیم اما بهرحال تا اندازه ای به موضوع مربوط اند. لذا جستجو کننده اگر بدون اپراتور یا عملگر جستجو اقدام به جستجو کند و نتایج اندکی بدست آورد می تواند با افزودن عدد عملگر همجوار، دامنه جستجوی خود را بیافزاید.

عملگرهای وجودی

این عملگرها، وجود و یا عدم وجود یک مدخل جستجو را صرف نظر از هر کلیدواژه خاصی بررسی می کنند. در سیمرغ، عملگرهای همه و هیچ در این دسته قرار می گیرند. به عنوان مثال یک کتابدار می خواهد بداند کدام کتابها شماره ثبت ندارند تا آنها را وارد و تکمیل کند. لذا او باید از عملگر هیچ در مدخل شماره ثبت استفاده کند: شماره ثبت: هیچ

برای مثالی در مورد عملگر «همه» فرض کنید یک کتابدار می خواهد تمام کتابهایی را که فروست (سلسله انتشارات) دارند پیدا کند تا بعنوان مثال گزارشی چاپی در مورد این کتابها تهیه کند. روش این کار استفاده از عملگر «همه» در مدخل «فروست» است: فروست: همه

از مثالهایی که زده شد می توان نتیجه گرفت که این نوع عملگرها بیشتر برای کارشناسان کتابداری و اطلاع رسانی و در کارهای جاری کتابخانه کاربرد دارد و برای محققین و کاربران نهایی (End Users) شاید کارکرد چندانی نداشته باشند.

عملگرهای تعیین محدوده

در بعضی از مداخل جستجو بویژه مداخل عددی بحث تعیین محدوده، معنا و اهمیت پیدا می کند. برای مثال یکی از معمولترین جستجوها، جستجوی کتابهایی است که در یک محدوده تاریخی خاص منتشر شده است. در این گونه موارد باید از عملگرهای تعیین محدوده استفاده کرد. دو عملگر تا و .. برای چنین مواردی بکار می رود. برای مثال کتابهای با شماره ثبت ۲۰۰۰۰ تا ۳۰۰۰۰ به شکل زیر بازیابی می شود:

شماره ثبت: ۲۰۰۰۰ تا ۳۰۰۰۰

یا فرض کنید هدف محقق، بازیابی کتابهایی باشد که در محدوده سالهای ۱۳۷۰ تا ۱۳۸۲ منتشر شده باشد. در این صورت نیز باید در مدخل جستجوی سال نشر به شکل زیر جستجو کند:

سال نشر: ۱۳۷۰ تا ۱۳۸۲

عملگر .. نیز درست مثل تا می باشد بجز اینکه حد پایین و حد بالای محدوده را در نظر نمی گیرد. برای مثال اگر در نمونه بالا بجای تا از .. استفاده کنیم، سالهای ۱۳۷۰ و ۱۳۸۲ که حد پایین و بالای محدوده می باشند از پاسخها خارج خواهد شد و محدوده ۱۳۸۱ تا ۱۳۷۱ بازیابی خواهد شد.

هرچند استفاده از عملگرهای تعیین محدوده در مداخل جستجوی حرفی خیلی معمول نیست اما بهرحال قابل استفاده است برای مثال فرض کنید کتابخانه ای بخواهد برای اصلاح موضوعی، مجموعه کتابهای خود را به سه قسمت تقسیم کرده و اصلاح هر قسمت را به یک کارشناس بسپارد. برای این کار می توان جستجوهای زیر را انجام داد:

موضوع: آ تا خ

موضوع: د تا ض

موضوع ط تا ی

با این جستجو ها کتابها به ترتیب موضوعی به سه قسمت تقسیم می شود .

• نکته: در عملگرهای تعیین محدوده این امکان وجود دارد که ابتدا و انتهای محدوده باز باشد. یعنی اینکه از ابتدا تا یک کلیدواژه خاص (یا از یک کلیدواژه خاص تا انتها) جستجو شود. برای مثال اگر بخواهیم تمام کتابهای موجود را که از اول تا سال ۱۳۸۲ منتشر شده اند بازیابی کنیم کافی است به شکل زیر جستجو کنیم:

سال نشر: تا ۱۳۸۲ (باز بودن ابتدای محدوده)

یا اگر بخواهیم تمام کتابهای منتشر شده از سال ۱۳۷۰ تا حال حاضر را پیدا کنیم باید به شکل زیر جستجو کنیم:



سال نشر: ۱۳۷۰ تا (باز بودن انتهای محدوده)

عملگرهای مقایسه ای

عملگرهای مقایسه ای در واقع حالت خاصی از عملگرهای تعیین محدوده اند و همانند آنها بیشتر در مداخل عددی معنی پیدا می کنند. تنها تفاوتی که با آنها دارند در این است که برخلاف عملگرهای تعیین محدوده که میان دو کلیدواژه قرار می گیرند، این عملگرها تک کلیدواژه ای (عملوندی) اند و معیار آنها تنها یک کلید واژه است. عملگرهای مقایسه ای در سیمرغ به این ترتیب می باشند: <، >، =، <=

برای مثال اگر کاربری لیست کتابهای از سال ۱۳۷۰ به بعد را بخواهد باید از عملگر <= به شکل زیر استفاده کند:
سال نشر: ۱۳۷۰ >=

اگر بجای عملگر <= از عملگر > استفاده شود، سال ۱۳۷۰ از جوابها حذف می شود و کتابهای منتشر شده در سالهای ۱۳۷۱ به بعد بازیابی خواهد شد. از این عملگرها برای مقایسه حروف نیز می توان استفاده کرد. به مثالی در این مورد دقت کنید:

فرض کنید در یک نمایشگاه کتاب که غرفه ها به ترتیب ناشران است، می خواهید لیست خرید کتابها را به دو قسمت کرده و هر کدام را به یک نفر بسپارید. برای این کار جستجوی زیر جوابگو خواهد بود:

ناشر : <= شباویز

این جستجو، تمام کتابهایی را که ناشر آنها در زنجیره الفبایی از شباویز به بعد باشد بازیابی می کند. چند نمونه از این ناشران از این قرار اند:

شباویز

شرکت سهامی انتشار

طرح نو

مرکز نشر دانشگاهی

نشر نی

یساولی

مثالهای فوق نشان می دهد که جهت این عملگرها با توجه به جهت ورود عبارتها تعیین می شود. به این معنی که در مداخل جستجوی چپ به راست مانند سال نشر، شماره ثبت و ... علامت بزرگتر یا مساوی از چپ به راست است و در مداخل حرفی از راست به چپ.

برای مثالی دیگر در این زمینه فرض کنید هدف ما بازیابی کتابهای با شماره ثبت کمتر از ۱۰۰۰۰ باشد. برای این کار از عملگر < به شکل زیر باید استفاده کرد:

شماره ثبت: < ۱۰۰۰۰

عملگر <> زمانی بکار می رود که بخواهیم یک مورد خاص را از مجموعه جوابها حذف کنیم. مثلاً فرض کنید به دلایلی می خواهیم کتابهای سال ۱۳۸۲ بازیابی نشوند. لذا از عبارت زیر استفاده می کنیم:

سال نشر: ۱۳۸۲ <>

عملگرهای جستجوی تمام عبارت

گاهی اوقات جستجوکننده ممکن است که به دنبال یک مدرک (رکورد) خاص باشد. به عنوان مثال یک کتابدار می خواهد بداند یک کتاب خاص آیا در مجموعه کتابخانه موجود است یا خیر. در این حالت، جستجوی کلیدواژه ای قطعاً علاوه بر رکورد مورد نظر و مطلوب، نتایج اضافی را بازیابی خواهد کرد که احتمالاً نزدیک و مرتبط با رکورد مطلوب است اما دقیقاً آن رکورد نیست و لذا جستجوکننده باید بار دیگر در میان انبوه جوابهای نامربوط و نامناسب، به دنبال جواب مطلوب خود بگردد. لذا در این موارد، برای صرفه جویی در وقت و انرژی جستجوکننده، بهتر است که جستجوی دقیق (تمام عبارت) انجام شود. از طرفی عبارتها هرچه طولانی تر می شود احتمال وجود واژه های ربط در آنها بالا می رود. از انجایی که از تعدادی از این حروف ربط (و، یا، تا) در سیمرغ بعنوان عملگر جستجو استفاده شده لذا سیستم این حروف را عملگر فرض کرده و جستجو را از مسیر مناسب دور خواهد کرد. علاوه بر اینها وجود بعضی از علائم خاص مانند \$ و % نیز که بازم نقش عملگر (پسوندهای عمومی) را دارند و یا وجود عناوین مداخل جستجو (مثل عنوان، موضوع، ...) در عبارت جستجو نیز باعث اختلال در عملیات خواهد شد. لذا برای پیشگیری از این موارد در سیمرغ از عملگرهای «جستجوی تمام عبارت» استفاده می شود. این عملگرها عبارتند از:

(=) و (" ")

عملگر =

این عملگر زمانی کاربرد دارد که محقق بخواهد عبارتی را عیناً و در کلیت آن جستجو کند. در این حالت نرم افزار، هیچ دخل و تصرف و پردازشی روی عبارت مورد نظر انجام نمی دهد و کل عبارت را جستجو می کند. فرض کنید شخصی می خواهد کتاب با عنوان «تاریخ جنگهای ایران و روس» را پیدا کند. واضح است که در اینجا واژه «و» بعنوان یک واژه عادی و جزئی از عنوان است و با عملگر «و» تفاوت دارد. لذا در این موارد برای اینکه نرم افزار دچار خطا نشود، کل عبارت را باید پس از یک علامت = قرار داد:



عنوان: = تاریخ جنگهای ایران و روس
به چند مثال دیگر در این زمینه توجه کنید:
عنوان: = از صبا تا نیما («تا» بعنوان حرف ربط)
عنوان: = طراح باغ و فضای سبز («و» بعنوان حرف ربط)
عنوان: = خصوصی سازی یا ترکیب مطلوب دولت و بازار در توسعه («یا» و «و» بعنوان حروف ربط)

عملگر " "

کارکرد این عملگر تا حد زیادی شبیه به عملگر مساوی (=) است و تنها تفاوت آنها در این است که عملگر " " فقط شناسه‌ها را جستجو می‌کند. لذا قبل از هر مثالی بهتر است مفهوم شناسه توضیح داده شود. شناسه‌ها آن دسته از فیلدهایی هستند که ارزش دسته بندی بیشتری دارند و معمولاً بهتر است جستجوکنندگان از طریق آنها به مدرک مورد نظر خود برسند. به عنوان مثال در پایگاه کتابهای فارسی معمولاً فیلدهای عنوان، موضوع، نویسنده و ... شناسه قرار می‌گیرند. اما فیلدهایی نظیر یادداشت، مشخصات ظاهری، ویرایش و ... اگر چه توضیحاتی در مورد رکورد می‌دهند، اما ارزش اطلاعات چندانی به شکل مستقل ندارند و نیز امکان اینکه محقق بخواهد از طریق آنها کتابها را جستجو کند نیست لذا چنین فیلدهایی شناسه قرار نمی‌گیرند و در نتیجه با عملگر " " قابل جستجو نیستند.
یکی از کاربردهای بسیار مفید عملگرهای جستجوی دقیق، حذف نتایج اضافی است. برای مثال فرض کنید شخصی به دنبال کتاب فیزیک نوشته هالیدی باشد. لذا او تصمیم می‌گیرد در مدخل عنوان جستجوی زیر را انجام دهد:

عنوان: فیزیک

بدیهی است که چنین جستجویی تمام کتابهایی را که در عنوان آنها کلمه فیزیک وجود داشته باشد بازبایی می‌کند. چند نمونه از جوابهای احتمالی را در زیر می‌بینید:

(۱) فیزیک

(۲) فیزیک الکتربیسته

(۳) اصول شیمی فیزیک

(۴) مبانی فیزیک هسته ای

(۵) تائوی فیزیک

(۶) آشنایی با فیزیک اتمی

واضح است که تنها مورد (۱) جواب صحیح می‌باشد و جوابهای دیگر اگرچه در رابطه با فیزیک است اما مورد نظر جستجوکننده نیست. لذا برای حل این مشکل باید جستجو را با استفاده از عملگر جستجوی دقیق و بصورت زیر انجام داد:

عنوان: «فیزیک»

با این کار کتابهایی بازبایی می‌شود که عنوان آنها تنها و تنها کلمه «فیزیک» باشد یا به عبارت دیگر واژه «فیزیک» بعنوان یک شناسه جستجو می‌شود و نه بعنوان یک واژه عادی.

در یک مثال دیگر محقق می‌خواهد کتاب با عنوان دقیق «تاریخ ایران» را جستجو کند. برای این کار باید عنوان را بین این دو علامت (" ") قرار دهد:

عنوان: «تاریخ ایران»

در این حالت تنها کتابهایی پیدا می‌شوند که عنوان آنها عیناً و بی‌کم و کاست عبارت «تاریخ ایران» باشد. بدیهی است که بدون استفاده از این عملگر، کتابهایی با عناوین زیر نیز بازبایی خواهد شد که مورد نظر نیستند:

تاریخ ایران باستان

روزشمار تاریخ ایران از مشروطه تا انقلاب

نگرشی بر تاریخ ایران نوین



مطالعی که در بالا آمد، بخشی از امکانات جستجوی پیشرفته در سیمرغ نوساست که تنها به عملگرهای جستجو مربوط می باشد. امکانات دیگری نظیر جستجو در مداخل دلخواه، جستجوی ترکیبی و جستجوی همزمان در چند مرکز اطلاع رسانی امکانات دیگری است که در نوشته های بعدی به آن پرداخته خواهد شد.