دانشکده مهندسی

گروه کامپیوتر

پایان‌نامه کارشناسی ارشد

ارائه یک روش جدید رتبه‌بندی مجموعه داده‌ها در موتورهای جستجوی معنایی برای مقابله با اسپم

سهیلا دهقانزاده

استاد راهنما: دکتر محسن کاهانی

شهریور ماه ۱۳۹۰

بسم الله الرحمن الرحيم

تقدیر و سپاس

بدینوسیله بر خود لازم می‌دانم که از زحمات بی‌دریغ استاد گرامي، جناب آقاي دکتر کاهانی که راهنمایی‌های ارزشمند ایشان در تمام مراحل انجام این این پایان‌نامه راه‌گشای من بوده است تشکر نمایم.

همچنین از تلاش‌های بی‌وقفه و پشتیبانی‌های پدر و مادر عزیزم کمال سپاس‌گذاری را دارم.

در انجام این پایان‌نامه از همکاری اعضای مؤسسه‌ی DERI بهره‌ی زیادی برده‌ام که در اینجا لازم است از اعضای این مؤسسه تقدیر نمایم.

---

http://*www.**deri**.org/*[۱]

چکیده

با ظهور وب معنایی و همه‌گیر شدن آن، ضرورت درک اطلاعات وب توسط ماشین بر هیچ کس پوشیده نیست. انتظار انسان از چگونگی نتایج یک موتور جستجو با انتظار عامل نرم‌افزاری از چگونگی نتایج فرق می‌کند. آنچه مسلم است اینست که تاکنون وب برای ماشین قابل فهم نبوده است و صفحات وب فقط توسط انسان‌ها قابل پردازش بوده است.

پروژه عظیم داده‌های باز پیوندی، حجم زیادی از داده‌های RDF ،که توسط ماشین و انسان قابل فهم است، را روی وب در دسترس قرار داده است. برای استفاده از این حجم انبوه داده‌ها باید بتوان آن‌ها را جستجو کرد. بنابراین، نسل دوم برنامه‌های وب معنایی، نیاز به نقطه دسترسی کارا به وب معنایی دارند که ماهیت معنایی این دانش را نیز لحاظ کند. از آنجا که موتورهای جستجو دروازه ورود به وب هستند و انسان و ماشین هم باید بتوانند روی این مدل داده جدید (RDF) جستجو انجام دهند، ضرورت یک موتور جستجوی معنایی برای انسان و یک موتور پرسش معنایی برای ماشین کاملاً احساس می‌شود.

با توجه به ظهور موفقیت آمیز "وب داده‌ها"، سوء استفاده‌های شخصی برای کسب سود و منفعت بیشتر در قالب اسپم، در وب داده‌ها رو به ظهور است. از آنجا که الگوریتم رتبه‌بندی یک موتور جستجو، تا حد زیادی وظیفه مقابله با این نوع تهدیدها را بر عهده دارد، این پایان نامه با بررسی الگوریتم‌های رتبه‌بندی "وب اسناد" و تطبیق آن برای وب داده‌ها، بدنبال پیشگیری از ظهور اسپم در نتایج موتورهای جستجوی معنایی است .

با ایجاد انواع مختلف اسپم و مشاهده نتایج رتبه‌بندی الگوریتم مشهور DING که در موتور جستجوی معنایی sindice بکار رفته است و الگوریتم رتبه‌بندی بر اساس ماتریس صلاحیت نام‌گذاری که در موتور جستجوی معنایی SWSE بکار رفته است، نقاط ضعف این الگوریتم‌ها در مقابل ارتباطات گروهی نشان داده‌شده است. الگوریتم پیشنهادی برای رتبه‌بندی مجموعه داده‌ها، با کشف ارتباطات گروهی و تنبیه این نوع اسپم، با روش جدیدی ارتباطات را وزن‌دار می‌کند و با اعمال رتبه‌بندی وزن‌دار، اعضای ارتباط گروهی را در قعر نتایج رتبه‌بندی قرار می‌دهد. از آنجا که نویسنده معتقد است برای بکارگیری تمام مفاهیم پنهان یک چهارگانه برای رتبه‌بندی دامنه‌ها، باید هر دو روش بکار رفته در صلاحیت نام‌گذاری و DING توأما بکار گرفته شوند روش ارائه شده در این پایان‌نامه ترکیبی از دو روش موجود، همراه با تکنیک‌های کشف ارتباط‌های گروهی است. نوآوری اصلی این پایان نامه، ارائه یک روش رتبه‌بندی جدید است که توسط ارتباطات گروهی گمراه نشود و تمام مفاهیم ضمنی چهارگانه را به‌کارگیرد.

برای ارزیابی روش پیشنهادی مجموعه‌داده داروها از ابر داده‌های پیوندی جمع‌آوری شده است. چهار نوع اسپم ایجاد شده در چهار تست به مجموعه‌داده تزریق شده‌اند و نتایج هر مرحله نشان دهنده اینست که روش پیشنهادی در کشف انواع اسپم موفقیت آمیز بوده است.

کلید واژه- موتورجستجوی معنایی، وب معنایی، اسپم محتوا، ارتباطات گروهی، RDF ، رتبه‌بندی، تحلیل لینک، متریک اعتماد، چهارگانه، اصالت داده.

فهرست مطالب

فهرست جداول

فهرست شکل‌ها

فهرست علائم و اختصارات

Any23: Any thing to triple.
API: Application Programming Interface.
AVL: Adelson-Velskii-Landis.
CLI: Command Line Interface.
DBMS: Data Base Management System.
DING: Dataset RankING.
HITS: Hyperlink-Induced Topic Search.
HTML: Hyper Text Markup Language.
ISBN: International Standard Book Number.
LOD: Linked Open Data.
RDF: Resource Description Framework.
RDFa: Resource Description Framework -in- attributes.
RDFS: Resource Description Framework Schema.
SIREn: Semantic Information Retrieval Engine.
SPARQL: SPARQL Protocol and RDF Query Language.
SWD: Semantic Web Document.
SWSE: Semantic Web Search Engine.
TFIDF : Term Frequency Inverse Document Frequency.
TKC: Tightly Knit Community.
URI: Uniform Resource Identifier.
URL: Uniform Resource Locator.
XML: eXtensible Markup Language.

فصل 1- مقدمه

ضرورت موتورهای جستجو یک واقعیت انکارناپذیر است زیرا موتورهای جستجو دروازه‌های ورود به وب هستند و برای قابل استفاده کردن اطلاعات انبوه روی وب ضروری هستند. از دیدگاه کاربر، یک موتور جستجوی ایده‌آل برای وب، باید قادر باشد تا جواب مستقیم یک پرسش را بیابد. موتور جستجوی گوگل با ارائه یک واسط خیلی ساده و مدل تراکنش ساده بر اساس کلمه کلیدی، زمان پاسخ بسیار کوتاه و مرتب‌سازی ماهرانه نتایج، معیار سنجش روش‌های جستجوی وب است و ۶۴/۶% از پرسش‌های وب را روی میلیاردها سند وب سرویس‌دهی می‌کند. اما در نهایت نتایج گوگل یک لیست مرتب از صفحات توصیه شده وب است و کاربر انسانی با مشاهده صفحات وب و پیمایش آن‌ها جواب مورد انتظار خود را بازیابی می‌کند. اما این نتایج برای عامل نرم‌افزاری قابل فهم نیست.

1-1-تعریف مسئله

امروزه پروژه "LOD"[1] حجم وسیعی از داده‌های RDF را (که محتوا صادر شده از پایگاه داده‌های wikiPedia و BBC و New York Times و Flicker و LastFM و... است) قابل دسترس ساخته است.

با ظهور "وب داده‌ها" و ایجاد قابلیت تعامل وب و ماشین (عامل‌های نرم‌افزاری و وب‌سرویس‌ها)، عدم قابلیت پردازش نتایج جستجو در موتورهای جستجوی فعلی توسط ماشین، ضرورت موتور جستجوی **معنایی** را آشکارتر می‌سازد. مثلاً نتایج بازیابی شده توسط گوگل تعدادی صفحه مرتبط است که به کاربر توصیه می‌شود. کاربر باید این نتایج را پردازش کرده و خودش داده‌های مرتبط را از صفحه استخراج کند. در این میان

[1] Linked Open Data

تبلیغات صفحه و سایر اطلاعات نامربوط صفحه توسط انسان فیلتر می‌شود.

مسلماً یک عامل نرم‌افزاری یا یک وب سرویس هوش کافی برای شناسایی قطعه اطلاعاتی مورد نظر را از میان اطلاعات گوناگون صفحات ارائه شده در گوگل ندارد. محدودیت‌های گوگل از آنجا ناشی می‌شود که اسناد HTML[1] ساختار قابل پردازش توسط ماشین را ندارند. بنابراین اطلاعات بازیابی شده برای ماشین باید حاوی اطلاعات قابل پردازش توسط ماشین باشد.

مشکل بعدی یکپارچه‌سازی و امکان استفاده مجدد اطلاعات، در وب است. از آنجا که اطلاعات سازمان‌ها بصورت داده‌های ساخت یافته در پایگاه داده‌های محلی نگهداری می‌شوند و این پایگاه داده‌ها بندرت به هم متصل هستند، امکان استفاده مجدد اطلاعات بین سایت‌های مختلف خیلی محدود است.

هر چند استانداردهای وب معنایی سطح بالایی از تعامل را ممکن ساخته‌اند ولی تنوع هنوز مهمترین مانع برای استفاده از این اطلاعات است. مثلاً منتشرکنندگان داده اطلاعات یکسان را با واژگان متنوع منتشر می‌کنند و یا شناسه‌های مختلف را برای یک منبع انتساب می‌دهند. حل مشکل تنوع در [Hog11] به تفصیل بررسی شده است و راه حل‌هایی برای آن ارائه شده است.

مشکل اصلی که این پایان نامه به آن پرداخته است جلوگیری از گمراه سازی الگوریتم رتبه‌بندی موتورهای جستجوی معنایی توسط تولیدکنندگان اسپم می‌باشد. بدین منظور الگوریتم رتبه‌بندی طوری طراحی شده است که با شناسایی اعضای ارتباط گروهی، آن‌ها را در قعر نتایج رتبه‌بندی قرار می‌دهد.

---

[1] Hyper Text Markup Language

1-1-1- موتورجستجوی معنایی

یک موتور جستجوی معنایی همانند سایر موتورهای جستجو باید فازهای پیمایش[1]، یکپارچه‌سازی[2]، رتبه‌بندی مستقل از پرسش[3]، استنتاج[4]، شاخص‌گذاری[5]، پردازش پرسش[6]، رتبه‌بندی وابسته به پرسش، نمایش نتایج به[7] کاربر در قالب اجزای واسط و ارائه ارزیابی[8]های مناسب، را داشته باشد[9]. تنها تفاوت روی نوع داده‌های پردازش شده و چگونگی پیاده‌سازی هر فاز است. در موتورهای جستجوی وب اسناد، واحد اطلاعاتی یک صفحه وب است که صفحات با لینک‌های بدون نوع و اصالت به هم متصل شده‌اند. در حالیکه در وب داده‌ها، واحد اطلاعاتی قابل جستجو موجودیت‌ها هستند که اعضای تشکیل دهنده سه‌تایی‌ها (RDF) می‌باشند و هر عبارت RDF در واقع با لینک‌هایی از انواع و اصالت‌های مختلف، این موجودیت‌ها را به هم مرتبط می‌سازد. الگوریتم‌های فازهای مختلف موتور جستجو باید بر روی نوع داده RDF قابلیت عملکرد بهینه را داشته باشند. در بعضی فازها اعمال اینگونه تغییرات نیاز به بازبینی کلی الگوریتم دارد چون مفهوم ابرلینک[10] وب اسناد در وب داده‌ها، کاملاً تغییر کرده و واحد جستجو نیز به جای صفحه، موجودیت است [**Error! Reference source not found.Error! Reference source not found.Error! Reference source not found.**Hog11].

---

[1] crawling
[2] consolidation
[3] query independent ranking
[4] reasoning
[5] indexing
[6] query processing
[7] query dependant ranking
[8] display
[9] offering pertinent evaluation
[10] hyperlink

2-1-1- مشکلات اسپم در موتور جستجو

برای اکثر پرسش‌ها تنها ۱۰ صفحه بالاتر نتایج جستجو توسط کاربر مشاهده می‌شوند. چون ترافیک بالاتر صفحه به معنای سود تجاری بیشتری است، تولیدکنندگان محتوا مایلند صفحات آن‌ها در بهترین رتبه‌ها ظاهر شوند. در این میان بعضی با بازی کردن با ویژگی‌های صفحات و بدون تلاش برای ایجاد صفحات با کیفیت بالا به دنبال فریب دادن الگوریتم رتبه‌بندی موتور جستجو و بالابردن مصنوعی رتبه صفحات بی‌کیفیتی هستند که اسپم نامیده می‌شوند. صفحات اسپم در وب از تکنیک‌های مختلفی برای رسیدن به رتبه‌های بالا در نتایج جستجوی موتورهای جستجو و گمراه کردن آن‌ها استفاده می‌کنند. انسان‌ها برای شناسایی صفحات اسپم و با کیفیت پایین -که ممکن است ادعا کنند هویتی هستند، اما در واقع جعل هویت هستند- مشکلی ندارند. اما استفاده از نیروی انسانی در وب امروز برای شناسایی اسپم‌ها و هویت‌های جعلی خیلی وقت‌گیر و پرهزینه و غیرمعقول است. موتورهای جستجو باید ویژگی‌های دوگانه **کیفیت نتایج** و **مرتبط بودن** را با هم برای رتبه‌بندی و نمایش نتایج به کاربر لحاظ کنند تا کاربر بتواند از حجم زیاد اطلاعات روی وب استفاده کرده و در عین حال گمراه نگردد. در تکنیک‌های بهینه‌سازی موتور جستجو[۱] و بازیابی رقابتی اطلاعات، هدف یافتن تابع نمره‌دهی موتور جستجو و بالابردن مصنوعی رتبه‌ی یک صفحه در نتایج بازیابی شده است، تا بتوان از منافع تجاری صفحاتی که در رتبه‌های بالا ظاهر می‌شوند استفاده کرد. با توجه به غیر ممکن بودن استفاده از نیروی انسانی برای کشف صفحات اسپم و هویت‌های جعلی، باید این فرآیند را خودکار کرد. تولیدکنندگان اسپم و جعل

---

[۱] search Engine Optimization

هویت‌کنندگان، متناوباً تکنیک‌های خود را تغییر می‌دهند تا موتورهای جستجو را گمراه کنند، بنابراین مقابله‌ی اتوماتیک با آن‌ها خیلی دشوار است.

خیلی از تکنیک‌های تولیدکنندگان اسپم شناسایی شده‌اند از جمله اینکه سابقاً الگوریتم‌های بازیابی اطلاعات متنی مثل TFIDF نقش مهمی در الگوریتم‌های رتبه‌بندی موتورهای جستجوی متنی داشتند. بنابراین تلاش‌های اولیه تولیدکنندگان اسپم روی محتوا صفحه متمرکز بود (مثلاً تکرار کلمات کلیدی در صفحه یا الحاق یک واژه‌نامه در محتوا صفحه). با اختراع الگوریتم‌های رتبه‌بندی بر اساس لینک مثل PageRank و HITS و موفقیت چشمگیر آن‌ها در موتورهای جستجو، تکنیک‌های جدید تولید اسپم برای هدف قرار دادن لینک‌ها دارای اهمیت شد که ارتباط گروهی یکی از آن‌هاست. ارتباط گروهی شبکه‌ای از وب سایت‌هاست که کاملاً به هم متصل هستند و مثالی از پدیده جامعه کاملاً پیوسته[1] است. چون TKC روی نتایج رتبه‌بندی خیلی مؤثر است ضرورت تشخیص و اصلاح تأثیر آن روی فرآیند رتبه‌بندی کاملاً مشخص است[Cas10].

هدف اصلی در اسپم محتوا، گمراه کردن الگوریتم TFIDF است که با بالا بردن فرکانس واژگان غیرمرتبط و مرتبط در قسمت‌های پنهان صفحه که توسط کاربر قابل مشاهده نیست، به این هدف دست می‌یابد. زیرا موتور جستجو این واژگان را شاخص‌گذاری می‌کند و این صفحات را به عنوان پاسخ‌های مرتبط به پرسش کاربر، بازمی‌گرداند. این در حالیست که صفحه، این واژگان غیر مرتبط را فقط به صورت پنهان ذکر کرده است و این باعث سردرگمی و عدم رضایت کاربر می‌شود. کشف اسپم

[1] tightly knit community effect (TKC)

محتوا ساده‌تر از سایر انواع اسپم است و در صورتیکه با نوع دوم اسپم ترکیب نشود، تأثیری روی الگوریتم pageRank ندارد.

در یافتن رتبه‌ی جستجوی طبیعی وب سایت، قسمت اصلی توسط عمومیت لینک[1] (تعداد و کیفیت صفحات سایر وب سایت‌های ارجاع دهنده به وبسایت) تعیین می‌شود. یک بخش حیاتی در معماری موتورهای جستجو الگوریتم‌های رتبه‌بندی بر اساس لینک است که اکثر آن‌ها در وب اسناد بکارگیری شده‌اند. از طرف دیگر، الگوریتم‌های رتبه‌بندی موجود برای وب داده‌ها، نیاز به دخالت عامل خبره انسانی، برای وزن دهی انواع لینک دارند مثلاً objectRank [Hwa06] . بنابراین این الگوریتم‌ها در مواردی که داده‌ها از تعداد زیادی منبع با واژگان متعدد یکپارچه می‌شوند، مثل وب داده‌ها، کارا نیستند. در چنین محیط‌هایی اصالت منبع سیگنال مهمی است که الگوریتم رتبه‌بندی باید آن را در نظر بگیرد.

فرآیند ایجاد لینک[2]، تلاش برای ساخت لینک‌های ورودی به یک وب سایت است. ایجاد لینک از منابع با کیفیت، مشکل و وقت‌گیر است چون اکثر لینک‌ها بر اساس دستور مافوق انسانی ایجاد می‌شوند. ایجاد لینک به وب سایت از روش‌های زیر مقدور است:

1. ایجاد وب سایت‌های صرفاً تقویت‌کننده محتوا[3]
2. گرفتن لینک‌های دوطرفه از وب سایت‌های غیررقیب ولی مرتبط
3. انتشار در newsletterها و e-zines
4. ارسال مقالات به پایگاه‌های مقالات
5. شرکت در انجمن‌های بحث و تبادل نظر مرتبط با سایت

---

[1] Link popularity
[2] Link building
[3] content-only feeder websites

6. وبلاگ نویسی روی سایت‌های مرتبط با سایت
7. جمع‌آوری لینک دوستان در سایت‌های اجتماعی
8. و....

ارزش نوشته‌های یکتا و شاخص قابل چشم پوشی نیست. کیفیت مطلب، لینک‌های بسیاری را ایجاد می‌کند و رتبه را بالا نگه می‌دارد. ارتباطات گروهی ساختگی درصددند تا تلاش لازم برای ایجاد لینک باکیفیت را دور بزنند و بیشتر روی تعداد لینک‌ها، بدون توجه به کیفیت لینک‌ها متمرکزند. چون موتورهای جستجو با تحلیل لینک رو به عقب بسیار پیشرفته شده‌اند، بنابراین خطر جریمه شدن برای داشتن لینک به یک ارتباط گروهی بالا است.

معمولاً الگوریتم‌های تحلیل لینک پایه از روی تعداد لینک‌های ورودی، رتبه‌ی صفحه را محاسبه می‌کنند. بنابراین توسط ارتباط گروهی که روی تعداد لینک‌های ورودی و بدون افزایش کیفیت کار می‌کنند، گمراه می‌شوند. اکثر وب سایت‌های ارتباط گروهی، محتوا خودساخته ندارند و استانداردی برای واگذاری[1] لینک هم ندارند.

تنها هدف ارتباطات گروهی افزایش مصنوعی تعداد لینک‌های ورودی به سایت برای افزایش رتبه شاخص سایت در موتور جستجو است. ارتباطات گروهی ساختگی یک نوع اسپم برای موتور جستجو تلقی می‌شوند و هدفشان گمراه کردن موتور جستجو برای گرفتن رتبه بالاتر (به دلیل گرفتن لینک‌های ورودی از یک ارتباط گروهی) نسبت به آنچه که واقعاً لایقش هستند، می‌باشد. چنین سایت‌هایی از طریق موتور جستجو باید جریمه شوند و هیچ رتبه‌یا وزنی نباید به آن‌ها تعلق گیرد. همچنین اگر سایتی به یکی از اعضای ارتباطات گروهی اشاره کند باید توسط

[1] Submission

موتور جستجو جریمه شود. برخی وب سایت‌ها برای بالا بردن رتبه سایت خود به شرکت‌های SEO مراجعه می‌کنند. بعضی از شرکت‌ها SEO علاوه بر روش‌های مجاز برای بالابردن رتبه که پیشتر ذکر شد، از روش‌های غیرمجاز مخصوصاً ارتباطات گروهی استفاده می‌کنند. خیلی از شرکت‌های SEO، که از ارتباطات گروهی استفاده می‌کنند آن را انکار می‌نمایند.

2-1-راه حل

برای حل مشکلات وب اسناد که قبلا ذکر شد، وب معنایی پشته ای از تکنولوژی‌های لازم برای انتشار داده‌های قابل پردازش توسط ماشین ارائه کرده است، که قابلیت استفاده مجدد داشته باشد و قابل فهم توسط ماشین باشد. هسته مرکزی این پشته RDF[1] است.

راه‌حل وب معنایی مبتنی بر مدل داده RDF و ساختارهای آن برای انتشار و ارتباط بین داده‌های ساخت یافته در وب، تأثیر شگرفی بر امکان استفاده مجدد و قابلیت پردازش اطلاعات توسط ماشین داشته است. RDF از URI[2] برای نامگذاری هر چیز استفاده می‌کند. بدین ترتیب یک چارچوب استاندارد و انعطاف‌پذیر برای انتشار داده‌های ساخت‌یافته در وب با قابلیت ارتباط، همکاری، گسترش و استفاده مجدد سایر RDFها مهیا می‌کند، به طوریکه داده‌های متنوع از منابع مستقل بتوانند بصورت خودکار توسط عامل‌های نرم‌افزاری پردازش شوند. همچنین معنای داده‌ها با استفاده از آنتولوژی‌ها با ساختار RDF و با استانداردهای RDFS و OWL بیان می‌شود.

برای حل مشکل قابل پردازش نبودن نتایج جستجو توسط ماشین، موتورهای جستجویی مثل Sindice ارائه شده است تا نتایج

---

[1] Resource Description Framework
[2] Uniform Resource Identifier

بازیابی شده آن مستقیماً قابل پردازش توسط عامل نرم‌افزاری باشد. Sindice برای اینکه این نتایج توسط کاربران انسانی نیز قابل ملاحظه باشد، یک واسط ساده‌ی انسانی نیز ارائه کرده است. ولی کاربران عمده آن برنامه‌های نرم‌افزاری هستند.

نوآوری اصلی این پایان‌نامه ارائه یک الگوریتم رتبه‌بندی است که در برابر ارتباطات گروهی مقاوم باشد و از همه مفاهیم ضمنی و صریح یک چهارگانه برای انتقال اعتماد استفاده کند. بدین ترتیب تولید کنندگان اسپم قادر به گمراه سازی موتور جستجو نخواهند بود و موتور جستجو نیز در وقت مشتریان خود صرفه‌جویی می‌نماید و نتایج قابل اعتمادتری به مشتریان ارائه می‌کند.

3-1-نوآوری

با توجه به اینکه یک موتور جستجوی معنایی با منبع باز در حال حاضر در دسترس نیست، بایستی فازهای مختلف پیاده‌سازی شوند تا بتوان نتایج الگوریتم رتبه‌بندی پیشنهادی را مشاهده کرد و سایر الگوریتم‌های رتبه‌بندی سایر موتورهای جستجو را نیز پیاده‌سازی کرد تا بتوان نتایج را مقایسه نمود. پیاده‌سازی فاز پیمایش با استفاده از ابزار با کد باز LDSpider انجام شده است. فاز رتبه‌بندی مستقل از پرسش، با هدف رتبه‌بندی مجموعه‌ی داده‌ها و کاهش تأثیر ارتباط گروهی، در معادلات (۱۲)، (۱۳) و (۱۴) پیاده‌سازی شده است. مفهوم چهارگانه به سه روش می‌تواند اعتماد را منتقل کند، اما دو موتور جستجوی قبلی هر یک تنها یکی از فاکتورهای انتقال اعتماد را در نظر گرفته‌اند، به همین علت نویسنده معتقد است برای تکمیل انتقال اعتماد باید دو

الگوریتم DING و صلاحیت نام‌گذاری[1] تواماً بکار روند. فاز یکپارچه‌سازی با هدف شناسایی موجودیت‌های یکسان و تعیین رتبه هر موجودیت در گراف موجودیت‌های مشابه، پیاده‌سازی شده است. این رتبه‌بندی چون در یک زمینه خاص (موجودیت خاص) انجام می‌شود منجر به حذف اثر TKC می‌شود. بدین ترتیب مجموعه‌ی داده‌های عمومی برای یک موجودیت خاص، ممکن است بالاترین رتبه را نداشته باشند.

نتایج رتبه‌بندی مجموعه‌ی داده‌ها و موجودیت‌ها با شناسه‌های کانونی هر موجودیت در شاخص Lucene با ساختار پیشنهادی ذخیره شده است. پرسش با استفاده از Lucene نتایج را بازیابی می‌کند و نتایج بر اساس رتبه محاسبه شده که تابعی از رتبه محلی موجودیت و رتبه سراسری اصالت آن است، به کاربر نمایش داده می‌شود. در این مرتب‌سازی رتبه سراسری هر اصالت و وزن محلی موجودیت، از نتایج رتبه‌بندی مستقل از پرسش که در شاخص ذخیره شده است، اخذ می‌شود.

4-1-ساختار پایان‌نامه

در فصل ۲- مروری بر کارهای مشابه انجام شده است. موتور جستجوی با کد باز Lucene که پایه روش پیشنهادی است معرفی شده است. روش‌های مختلف در الگوریتم‌های رتبه‌بندی مشهور و روش‌های شاخص‌گذاری بهینه نیز معرفی شده‌اند. در فصل ۳- فازهای مختلف موتور جستجوی معنایی در روش پیشنهادی معرفی شده و فرآیند و الگوریتم هر فاز دقیقاً ذکر شده است. در فصل ۴- بررسی‌های انجام شده در طی چهار تست مختلف انجام گردیده و نتایج حاصله ذکر شده است. در تست اول تنها اثر اسپم محتوا، و در تست دوم ارتباط گروهی با چهار مجموعه

---

[1] Naming Authority

داده جعلی بررسی شده است. در تست سوم که با هدف گمراه سازی DING طراحی شده است، یک نوع ارتباط جعلی به همراه چهار مجموعه داده‌ی جعلی در یک اسپم محتوا با اندازه‌ی ۵۰ به مجموعه داده اولیه تزریق شده و در مرحله‌ی بعد از چندین ارتباط جعلی استفاده شده است. در تست چهارم از چندین ارتباط جعلی و ۱۰ مجموعه داده جعلی در یک ارتباط گروهی با اندازه‌ی ۲۰۰ استفاده شده است؛ و در مرحله‌ی بعد نیز شبه اسپم‌ها را حذف نموده‌ایم. این تست‌ها به تدریج الگوریتم‌های رتبه‌بندی را به چالش می‌کشند تا مرحله به مرحله قابلیت‌های الگوریتم‌ها ظهور کند. در فصل ۵- نتیجه‌گیری و کارهای آینده آمده است.

فصل 2- کارهای مشابه

در این فصل ساختار موتورهای جستجوی مفهومی مطرح در وب‌داده‌ها ارائه شده است و الگوریتم رتبه‌بندی هریک بیان شده است. کارهای مشابه برای فازهای شاخص‌گذاری و رتبه‌بندی نیز توضیح داده شده است. با توجه به اینکه شاخص Lucene روش مناسبی برای شاخص‌گذاری وب‌داده‌هاست، در موتور جستجوی پیشنهادی از قابلیت‌های این شاخص استفاده کرده‌ایم، ولی روش رتبه‌بندی پیشنهادی ترکیبی از روش‌های موتورهای جستجوی مشهور است که در فصل بعدی به تفضیل در مورد آن بحث می‌کنیم.

2-1-کارهای انجام شده در زمینه‌ی موتورهای جستجوی معنایی

2-1-1- Sindice

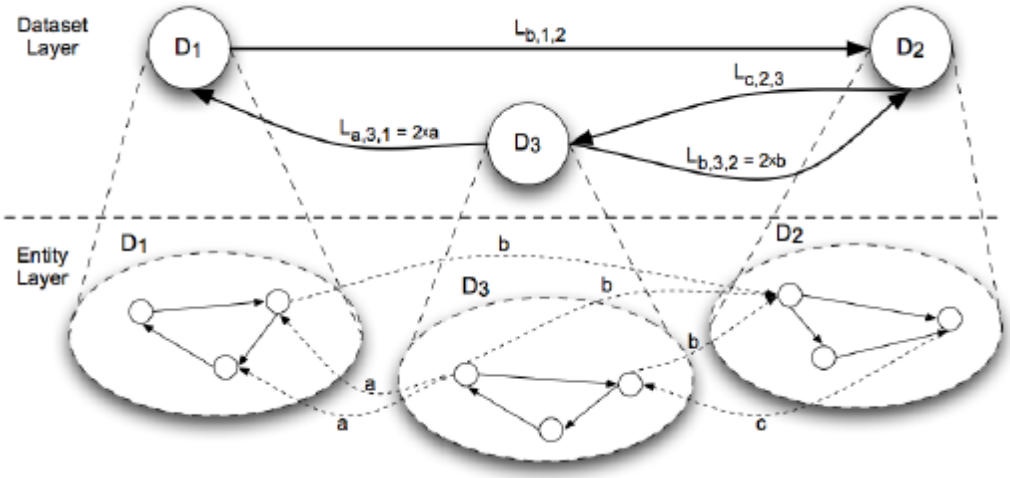
Sindice یک سرویس برخط[1] و مقیاس‌پذیر است که برای بازیابی عبارات بیان شده روی وب داده‌ها، راجع به یک منبع خاص، ایجاد شده است. بدین منظور ابتدا داده‌های وب معنایی را پیمایش و آن‌ها را شاخص‌گذاری می‌کند. با استفاده از یک API ساده، برنامه‌های کاربردی وب معنایی قادرند از Sindice استفاده نموده، منابع داده مرتبط را پیدا کرده و آن‌ها را بکار گیرند. Sindice می‌تواند یک منبع را از روی URI آن یا کلمات کلیدی یا ویژگی‌های معکوس تابعی[2] آن بازیابی کند. بنابراین هدف اصلی Sindice سرویس‌دهی به عامل‌های نرم‌افزاری و برنامه‌های کاربردی است تا منابع مرتبط برای پرسش آن‌ها بازیابی شود [Ore08].

---

[1] online

[2] inverse functional property

Sindice تنها به عنوان یابنده منابع RDF است و اشاره‌گرهایی به آن منابع را بازمی‌گرداند و یک موتور پرسش[1] نیست. بنابراین از لحاظ مفهومی به موتورهای جستجوی وب[2] (با مفاهیم وب معنایی) نزدیکتر است تا موتورهای جستجوی وب معنایی[3] مثل SWSE یا Swoogle . هدف اصلی این موتورهای جستجوی وب معنایی ارائه قابلیت‌های عمومی پرسش روی مجموعه‌ای از عبارات وب معنایی است. چون Sindice تنها اشاره‌گرهایی به منابع فراهم می‌کند، نیازی نیست که خیلی از چالش‌هایی که موتورهای جستجوی معنایی به آن‌ها برخورد می‌کنند را مدیریت کنند. چالش‌هایی مثل اعتماد و سیاست‌های یکپارچه‌سازی عمومی موجودیت و رد اختیاری یا غیرعمدی سرویس‌های پرسش‌های فوق پیچیده و یا روی داده‌های افزونه و .... مدیریت این مسائل در مقیاس عمومی سخت‌تر از مدیریت آن در سطح برنامه کاربردی است. در سطح برنامه کاربردی با استفاده از تعامل مستقیم کاربر یا سیاست‌های آن دامنه خاص و روش‌های دیگر می‌توان مسئله را مدیریت کرد [Tum08].



شکل ۱: مدل دولایه ای برای وب داده‌ها

[1] query engine
[2] Standard web search Engine
[3] semantic web search engine

Sindice از الگوریتم تحلیل لینک DING برای رتبه‌بندی موجودیت‌ها استفاده می‌کند. این الگوریتم وب داده‌ها را با یک مدل ۲ لایه‌ای مدل‌سازی می‌کند که در شکل ۱ نشان داده شده است. برای لایه مجموعه‌ی داده در وب داده، ابتدا گراف مجموعه‌ی داده را بر اساس معادله (۱) وزن‌دار می‌کند. در شکل۱ احتمال اینکه کاربر از D3 به D1 برود، متفاوت از احتمالی است که به D2 برود چون نوع و تعداد لینک‌های مرتبط با $L_{a,3,1}$ مشابه با نوع و تعداد لینک‌های مرتبط با $L_{b,3,2}$ نیست. هدف در اینجا پیاده‌سازی یک تابع وزن‌دهی مجموعه لینک است. وزن لینک هم می‌تواند بر اساس تعداد لینک‌های یک مجموعه لینک باشد و هم بر اساس اهمیت عمومی نوع لینک باشد. روش وزن‌دهی DING با استفاده از معیار TF-IDF برای اندازه‌گیری اهمیت یک نوع لینک با داشتن فرکانس آن در یک مجموعه داده است. بنابراین تابع وزن‌دهی با استفاده از معیار LF-IDF (Link Frequency - Inverse Document Frequency) بصورت معادله (۱) تعریف شده است.

$$w_{\sigma,i,j} = LF\left(L_{\sigma,i,j}\right) \times IDF(\sigma) = \frac{\left|L_{\sigma,i,j}\right|}{\sum_{L_{\tau,i,k}}\left|L_{\tau,i,k}\right|} \times \log\frac{\mathrm{N}}{1+freq(\sigma)} \qquad (۱)$$

که در آن، N تعداد مجموعه داده‌ها است و $freq(\sigma)$ فرکانس رخداد نوع σ در همه مجموعه داده‌هاست. روش LF-IDF درجه اهمیت بیشتر را به لینک‌هایی می‌دهد که فرکانس بیشتری در مجموعه داده مفروض و فرکانس کمتری در کل مجموعه داده‌ها دارد. بنابراین این روش وزن‌دهی مثلاً به foaf:knows وزن بیشتری می‌دهد تا rdfs:seeAlso.

سپس رتبه هر مجموعه‌ی داده با اعمال pageRank وزن‌دار و بصورت معادله (۲) محاسبه می‌شود.

$$r^k(D_j) = \alpha \sum_{L_{\sigma,i,j}} r^{k-1}(D_i)\, w_{\sigma,i,j} + (1-\alpha)\frac{\left|E_{D_j}\right|}{\sum_{D\epsilon\mathcal{G}}|E_D|} \qquad (۲)$$

در انتها رتبه هر موجودیت بر اساس رتبه مجموعه‌داده آن و رتبه موجودیت در آن مجموعه‌داده طبق فرمول زیر تعیین می‌شود [Del10].

$$r_g(\mathrm{e}) = \mathrm{r(D)} * \mathrm{r(e)} * \frac{|\mathrm{E_D}|}{\sum_{\mathrm{D'}\epsilon\mathrm{G}}|\mathrm{E'_D}|} \qquad (۳)$$

یکی از مشکلات اصلی الگوریتم DING اعمال بار زیادی روی سرور در فرآیند محاسبه و بهنگام سازی ماتریس سه بعدی وزن‌ها است. مخصوصاً برای محیط بزرگی مثل وب که تعداد دامنه‌ها و تعداد انواع مسندها خیلی زیاد است و پیچیدگی زمانی و مکانی الگوریتم محاسبه وزن برابر (تعداد دامنه‌ها) * (تعداد دامنه‌ها) * (تعداد مسندها) است.

SWSE -2-1-2

موتور جستجوی وب معنایی[۱] قابلیت جستجو را روی حجم رو به افزایش داده‌های RDF فراهم می‌کند. بدین منظور باید الگوریتم فازهای مختلف موتورهای جستجوی وب اسناد، برای داده‌های RDF بازنویسی شوند. طراحان SWSE برای الگوریتم هر فاز، ساختار توزیع شده‌ای را ایجاد کرده‌اند. این فازها عبارتند از پیمایش، بهبود کارایی داده‌ها[۲]، یکپارچه‌سازی، رتبه‌بندی[۳]، استنتاج، شاخص‌گذاری[۴]، رابط کاربر برای جستجو[۵]، جستجو و بازیابی اطلاعات[۶] روی داده‌های RDF است. مدل داده

---

[۱] Semantic Web Search Engine (SWSE)
[۲] data enhancing
[۳] Ranking
[۴] indexing
[۵] user interface for search
[۶] browsing and retrieval of information

RDF مشکلات خاصی در رابطه با عدم‌پایداری و ناسازگاری و نویز ایجاد می‌کند که باید همگی رفع شوند.

دو موضوع مهمی که SWSE برروی رفع آن‌ها متمرکز است عبارتند از:

- مقیاس پذیری روی حجم زیاد داده در وب داده‌ها، که برای حل این موضوع تمام فازهای مختلف بصورت توزیع شده پیاده‌سازی شده‌اند. بدین منظور هر مرحله شامل سه فاز توزیع[1]، اجرا[2] و جمع‌آوری[3] است.
- مقاومت در برابر داده‌های گوناگون و خراب و احتمالاً داده‌های متناقض که از منابع زیادی جمع‌آوری شده‌اند، که استانداردهای وب معنایی در زمینه این مشکلات راه‌گشا نیستند و حل این مشکلات در ساختار SWSE صورت گرفته است.

نویسندگان SWSE ادعا کرده‌اند که موتور جستجوی SWSE قابلیت جستجوی موجودیت‌گرا دارد در حالی‌که اعلام کرده‌اند سایر کارهای مشابه، مثل Swoogle و Sindice قابلیت جستجوی کلمات کلیدی روی اسناد وب معنایی-جستجوی سندگرا- ارائه کرده‌اند. watson و falcons روش جستجوی موجودیت‌گرا شبیه SWSE ارائه کرده‌اند. هر چند فازهای این موتورهای جستجو به اندازه فازهای SWSE کامل نیست. مثلاً Watson فازهای یکپارچه‌سازی و استنتاج را ندارد [Hog-Har11].

SWSE از مفهوم صلاحیت نام‌گذاری[4] که یک تناظر بین شناسه‌ها و منبع (مجموعه‌داده اصالت) آن‌ها است، برای رتبه‌بندی استفاده می‌کند. در اینجا pageRank روی گراف منابع که از روی

---

[1] scatter
[2] run
[3] gather
[4] Naming authority

ماتریس صلاحیت نام‌گذاری (که فقط در بر دارنده مفهوم use است) ایجاد شده است، اعمال می‌شود. درانتها شناسه‌های آن منابع مقادیر صلاحیت را از مجموعه‌داده منبع به ارث می‌برند.

این روش مستقل از طرح[1] است و نیازی به عامل خبره انسانی برای تغذیه اطلاعات ندارد و در زمینه‌های جستجو، پردازش پرسش، استنتاج و رابط کاربری روی مجموعه داده‌های یکپارچه کاربرد دارد. همچنین از لحاظ کیفیت و کارایی روی مجموعه داده‌های بزرگ و غیر آلوده وب بررسی شده است.

مدل داده روش پیشنهادی برای رتبه‌دهی شامل:

1. یک مجموعه شناسه I، شامل شناسه‌های سراسری U و شناسه‌های محلی B، و یک مجموعه رشته است.
2. یک مجموعه از منابع داده S زیر مجموعه‌ای از مجموعه شناسه‌های سراسری U است.
3. یک تابع ids که شناسه‌های سراسری و محلی و رشته‌ها را به منبعی از S که در آن ذکر شده، انتساب می‌دهد.

یک فرآیند رتبه‌بندی روی مجموعه داده‌هایی که توسط هر کسی می‌تواند ایجاد شود (مثل وب داده‌ها) باید ویژگی‌های زیر را داشته باشد:

1. استفاده از منابع A در B نشان‌دهنده تأیید صلاحیت A توسط B است و باید رتبه A را بالا ببرد.
2. استفاده مجدد نباید باعث کاهش رتبه شود.
3. صرف استفاده از یک شناسه مهم، نباید باعث بالا رفتن رتبه بشود. صرف استفاده از گراف node-link بدون توجه به اصالت (منبع) منجر به سوءاستفاده تولیدکنندگان اسپم می‌شود. زیرا با اضافه کردن یک سه‌گانه که از یک URI

---

[1] Schema

مشهور به یک URI اسپم اشاره می‌کند، URI اسپم از URI محبوب رتبه اخذ می‌کند .

الگوریتم تعیین رتبه‌ی قلم داده‌ها بدین صورت است:

1. بر اساس رخداد شناسه‌ها گراف "صلاحیت نام‌گذاری" را روی S*S ایجاد می‌کنیم.
2. این گراف رتبه‌ی pageRank هر عضو S را مشخص می‌کند.
3. برای شناسایی رتبه‌ی شناسه‌های سراسری و محلی از رتبه‌ی منبع استفاده می‌کنیم.

فرآیند تشکیل المان‌های گراف صلاحیت نام‌گذاری به صورت معادله‌ی (٤) است.

$$a_{i,j} = \begin{cases} 1 \text{ if } S_i \text{uses identifiers for which } S_j \text{has } \textit{naming authority} \\ 0 \text{ otherwise} \end{cases} \qquad (٤)$$

لینک‌های داخل یک "حوزه صلاحیت" درنظر گرفته نمی‌شوند چون اکثر آن‌ها لینک‌هایی با هدف جهت‌دهی هستند. بعد از اینکه pageRank روی ماتریس فوق اعمال شد، رتبه‌ی هر "حوزه‌ی صلاحیت" مشخص می‌شود، رتبه‌ی هر حوزه‌ی صلاحیت را به شناسه‌های سراسری و محلی و رشته‌ها، منتشر می‌کنیم.

محاسبه رتبه منبع[1] با اعمال PageRank روی گراف صلاحیت نام‌گذاری صورت می‌گیرد. رتبه‌ی شناسه‌ها[2] از روی رتبه‌ی منبع‌داده آن‌ها محاسبه می‌شود. رتبه شناسه u بسته به رتبه منابع‌داده‌ای دارد که در آن‌ها ظاهر شده است. طبق معادله (۵) وقتی یک شناسه سراسری است رتبه آن، مجموع رتبه همه منابعی است که در آن ظاهر شده است [Hog-Har11] .

$$identifierRank(u) = \sum_{s \in \{s | u \in s; s \in S\}} sourceRank(s) \qquad (۵)$$

---

[1] SOURCE RANK

[2] Identifier Rank

3-1-2- Lucene

Lucene کتابخانه موتور جستجوی متنی با کارایی بالا است که تماماً با جاوا نوشته شده است. Lucene توابع قوی‌ای برای شاخص‌گذاری و جستجو فراهم می‌کند و یک کتابخانه نرم‌افزاری است تا در برنامه‌های کاربردی به کار برده شود. Lucene این امکان را فراهم می‌کند تا جستجو را با توجه به نیازهایمان انجام دهیم در حالی‌که پیچیدگی‌های شاخص‌گذاری و جستجو را با استفاده از APIها حذف کنیم. بنابراین Lucene یک موتور جستجو با همه ویژگی‌ها نیست. به عنوان مثال فاز یکپارچه‌سازی برای داده‌های شاخص شده را انجام نمی‌دهد که امری بدیهی است چون انواع مختلف سند قابل شاخص‌گذاری هستند و یکپارچه‌سازی وابسته به سند است.

با Lucene می‌توان قابلیت شاخص‌گذاری و جستجو را به هر برنامه‌ای اضافه کرد. Lucene شاخص‌گذاری و جستجوی هر محتوایی که قابل تبدیل به متن باشد را فراهم می‌کند. این متن می تواند به هر فرمتی باشد و در هر مکانی ذخیره شده باشد. شاخص در فایل سیستم ذخیره می‌شود [McC10].

با توجه به قابلیت‌های بسیار خوب Lucene برای جستجوی شبه‌ساخت‌یافته بر آن شدیم تا این موتور جستجو پایه کار ما باشد.

مفاهیم اصلی در Lucene عبارتند از: عبارت[1]، فیلد، مستند[2] و شاخص.

- شاخص رشته‌ای از مستندهاست.
- مستند رشته‌ای از فیلدهاست.

---

[1] term
[2] document

- فیلد رشته‌ای نامگذاری شده از عبارت‌هاست.
- عبارت یک رشته کاراکتری است.

یک رشته کاراکتری در دو فیلد متفاوت، عبارت‌های متفاوتی هستند. بنابراین عبارت به صورت یک جفت رشته کاراکتری است که اولی نام فیلد و دومی متن فیلد است.

در Lucene شاخص‌گذاری به صورت معکوس[1] انجام می‌شود. شاخص برای کارا کردن جستجوی براساس عبارت[2]، آماری راجع به عبارت‌ها ذخیره می‌کند. شاخص Lucene به این دلیل جز شاخص‌های معکوس است که برای یک عبارت، تمامی مستندهایی که شامل آن عبارت هستند را بازمی‌گرداند. شاخص معکوس نیازی طبیعی است چون به صورت طبیعی هر مستند همه‌ی عبارت‌های خودش را لیست می‌کند.

در Lucene فیلدها (در حالتی‌که متن آن‌ها در شاخص رشته‌ای ذخیره می‌شوند) به صورت غیر معکوس ذخیره می‌شوند. اکثر فیلدها نشان‌گذاری[3]می‌شوند اما بعضی اوقات بهتر است برای فیلدهای شناسه‌های خاص، آن‌ها را رشته‌ای ذخیره کنیم و این فیلدها بدون هرگونه تحلیل و نشان‌گذاری ذخیره شوند.

شاخص Lucene ممکن است شامل چندین زیرشاخص باشد. هر بخش یک شاخص کاملاً مستقل است که جداگانه می‌تواند جستجو شود. رشد شاخص به دو صورت است:

۱) ایجاد بخش‌های جدید برای مستندهایی که به تازگی اضافه شده اند

۲) ترکیب بخشهای موجود

---

[1] inverted indexing
[2] term-based
[3] tokenized

هر جستجو ممکن است شامل چندین بخش یا چندین شاخص باشد. هر شاخص نیز ممکن است شامل چندین بخش باشد. این روش برای جستجو روی شاخص‌های توزیع شده بستر بسیار مناسبی است.

Lucene به مستند با شماره صحیح مستند ارجاع می‌کند. اولین مستندی که به شاخص اضافه می‌شود شماره اش صفر است و هر مستند جدید شماره‌ای بزرگتر از قبلی می‌گیرد. توجه کنید شماره مستند ممکن است تغییر کند بنابراین وقتی این اعداد را خارج از Lucene ذخیره می‌کنید باید به این نکته توجه داشته باشید.

اعداد ذخیره شده در هر بخش، تنها در همان بخش یکتا هستند. و هنگامی‌که قرار است در زمینه بزرگتری استفاده شوند باید تبدیل شوند. در تکنیک استاندارد به هر بخش بر اساس بازه اعدادی که در آن بخش استفاده می‌شوند، یک بازه از مقادیر را انتساب می‌دهیم. برای تبدیل شماره محلی مستند یک واحد به مقدار خارجی آن، شماره مستند پایه‌ی آن بخش به شماره‌ی محلی مستند اضافه می‌شود. برای تبدیل یک مقدار خارجی به یک شماره‌ی مستند محلی بخش، بخش را از روی بازه‌ی مقدار خارجی شناسایی می‌کنیم. برای مثال دو مستند ۵ بخشی اگر ترکیب شوند، بخش اول مقدار پایه صفر و بخش دوم مقدار پایه ۵ دارد و بنابراین مستند سوم از بخش دوم مقدار خارجی ۸ را دارد.

با حذف اسناد در شماره‌گذاری‌ها جای خالی ایجاد می‌شود. این جاهای خالی نهایتاً با رشد شاخص از طریق ترکیب، حذف می‌شوند. یک بخش به تازگی ترکیب شده هیچ جای خالی در شماره‌گذاری ندارد.

شاخص هر بخش شامل موارد زیر است:

- نام فیلدها: مجموعه نام فیلدهایی که در شاخص استفاده شده اند.
- مقادیر ذخیره شده فیلدها: برای هر مستند لیستی از جفت‌های مقدار-ویژگی (ویژگی نام فیلدهاست) است که برای ذخیره اطلاعات کمکی راجع به مستند مثل url, عنوان و یا شناسه دسترسی به پایگاه داده می‌باشد. مجموعه فیلدهای ذخیره شده همان چیزی است که در هر اصابت[1] هنگام جستجو برگردانده می‌شود. که با شماره مستند کلیدگذاری می‌شود.
- واژه‌نامه عبارت: یک واژه‌نامه شامل همه عبارت‌هایی که در فایل شاخص همه اسناد استفاده شده اند. واژه‌نامه همچنین شامل تعداد اسنادی است که آن عبارت را دارند و شامل اشاره‌گرهایی به داده‌های فرکانسی و تخمینی داده است.
- داده‌های فرکانسی عبارت: برای هر عبارت در واژه‌نامه، شامل همه اسنادی که شامل آن عبارت هستند و همچنین فرکانس آن عبارت در آن سند می‌باشد، اگر omitTf غلط باشد.
- داده‌های تخمینی عبارت: برای هر عبارت در واژه‌نامه مکانی است که عبارت در هر مستند ظاهر شده است.
- فاکتورهای نرمال سازی: برای هر فیلد در هر مستند، مقداری ذخیره شده است که در نمره یافته‌های آن فیلد ضرب می‌شود.
- بردارهای عبارت: برای هر فیلد در هر مستند، بردار عبارت که گاهی اوقات بردار مستند نامیده می‌شود،

---

[1] hit

ذخیره می‌شود. بردار عبارت شامل متن عبارت و فرکانس عبارت است.

- مستندهای حذف شده: یک فایل دلخواه که نشان می‌دهد کدام مستندها حذف شده اند.

4-1-2- Swoogle

Swoogle اطلاعات وب معنایی موجود در وب را کشف و تحلیل و شاخص‌گذاری می‌کند و روی این داده‌ها استنتاج انجام می‌دهد و ابرداده‌های مفید راجع به آن‌ها را ذخیره می‌کند. این سیستم، سرویس دسترسی به داده‌های معنایی روی وب را فراهم می‌کند که به کاربران انسانی و سیستم‌های نرم‌افزاری برای یافتن اسناد (واژگان و سه‌گانه‌ها) مرتبط و قابل اعتماد کمک می‌کند. همچنین Swoogle یک الگوریتم انعطاف‌پذیر الهام گرفته از PageRank ارائه می‌کند (OntoRank) که برای داده‌های معنایی تطبیق داده شده است و از الگوهای موجود در اسناد وب معنایی استفاده می‌کند. در OntoRank که بر اساس مدل پیمایش منطقی است، عامل پیمایشگر از یک سندمعنایی[1] به سندمعنایی دیگر با احتمال ثابتی حرکت می‌کند یا به یک سندمعنایی تصادفی می‌رود. عامل پیمایشگر با توجه به وزن لینک، حرکت غیرتصادفی دارد. علاوه بر این پیمایشگر با شناسایی سند معنایی، تمامی آنتولوژی‌هایی که برای توصیف واژگان آن بکار رفته است را ذخیره می‌کند تا اطلاعات سند معنایی برای او قابل فهم باشد. الگوریتم‌های رتبه‌بندی Swoogle بر طبق معادلات (۶)، (۷) و (۸) است.

$$OntoRank(a) = wPR(a) + \sum_{x \in OTC(a)} wPR(x) \qquad (۶)$$

[1] semantic Web Document(SWD)

$$wPR(a) = (1-d) + d \sum_{link(x,-,a)} \frac{wPR(x) \times f(x,a)}{\sum_{link(x,-,y)} f(x,y)} \quad (۷)$$

$$f(a,b) = \sum_{link(a,l,b)} weight(l) \quad (۸)$$

که در آنها wPR(a) نسخه‌ای از pageRank وزن‌دار می‌باشد،

f(a,b) مجموع وزن همه انواع لینک‌ها از a به b است،

d یک عدد ثابت بین ۰ و ۱ می‌باشد،

link(l,a,b) یک لینک با برچسب l از a به b است،

weight(l) ترجیح کاربر برای انتخاب لینک l می‌باشد،

و OTC(a) مجموعه سندمعنایی‌هایی است که از a به عنوان یک آنتولوژی استفاده می‌کنند.

TermRank تمامی SWTهای موجود در وب معنایی را مطابق معادله (۹) رتبه‌بندی می‌کند و رتبه یک سند معنایی میان SWTهایی که از آن استفاده می‌کنند تقسیم می‌شود. وزن عبارت t در سند معنایی d با TWeight(t,d) طبق معادله (۱۰) محاسبه می‌شود [Din04].

$$TermRank(t) = \sum_{uses(d,t)} \frac{ontoRank(d) \times TWeight(d,t)}{\sum_{uses(d,x)} TWeight(d,x)} \quad (۹)$$

$$TWeight(d,t) = cnt - uses(d,t) \times |\{d|uses(d,t)\}| \quad (۱۰)$$

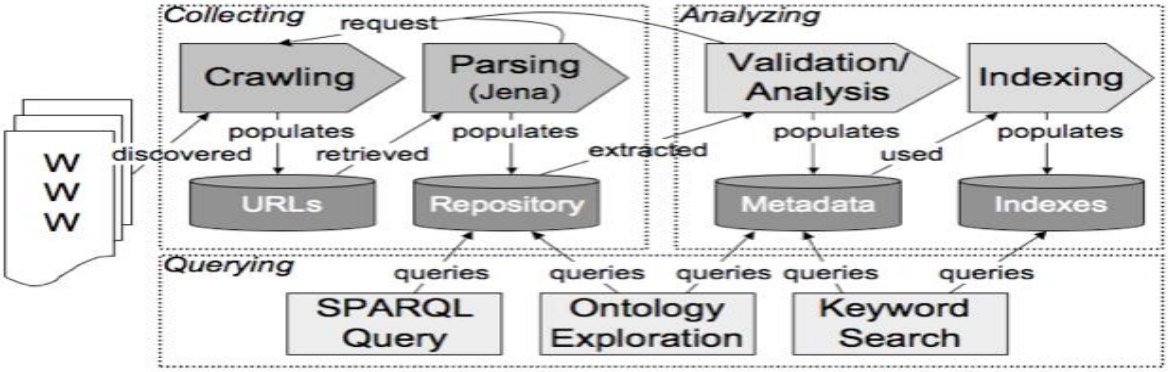
که در آنها cnt − uses(d,t) تعداد دفعات استفاده از عبارت t در سند d است،

و |{d|uses(d,t)}| تعداد کل سندهایی که از عبارت t استفاده کرده‌اند می‌باشد.

5-1-2- واتسون

واتسون دروازه‌ای برای وب معنایی است که با توجه به نیازمندی‌های برنامه‌های کاربردی وب معنایی، طراحی شده است که بصورت پویا دانش را انتخاب، ترکیب و بکار می‌گیرد. سه

فعالیت اصلی که در واتسون انجام می‌شوند عبارتند از: پیمایش وب معنایی، تحلیل و شاخص‌گذاری داده‌ها و آنتولوژی‌های وب معنایی، و فراهم کردن امکان دسترسی کارا به این داده‌ها برای کاربران و برنامه‌های کاربردی. در شکل ۲ لایه‌های موتور جستجوی معنایی واتسون در یک نگاه آورده شده است. هر لایه یکی از سه فعالیت اصلی واتسون را انجام می‌دهد. هر سه لایه روی یک وبسرور قرار دارند که بر روی پایگاه داده رابطه‌ای MYSQL استوار است. تمامی اجزای واتسون براساس جاوا است [Aqu07].



شکل ۲: شمای کلی موتور جستجوی واتسون

2-2-کارهای انجام شده در زمینه‌ی شاخص‌گذاری

در ادامه روش‌های مختلف ارائه شده برای شاخص‌گذاری معرفی شده اند:

1-2-2- یک مدل برای شاخص‌گذاری بهینه

معماری‌های زیادی برای ذخیره و پرسش از حجم زیادی از داده‌های RDF ارائه شده است از جمله FORTH RDF suit [Ale01] که همگی از پایگاه داده رابطه‌ای به عنوان انبار داده استفاده کرده‌اند. در اکثر این سیستم‌ها داده‌های RDF، به تعداد زیادی سه‌تایی تجزیه شده‌اند که مستقیماً در جدول‌های رابطه‌ای یا

جداول هش ذخیره می‌شوند. پردازش پرسش‌های بر اساس عبارت[1] (یعنی یک یا دو بخش از سه‌تایی نامعلوم است و جواب منابعی است که کامل کننده بخش‌های حذف شده هستند) توسط این سیستم‌ها رضایت بخش است. ولی پرسش بر اساس سه‌تایی جزء روش‌های معمول برای پرسش از داده‌های RDF نیستند. به همین دلیل سیستم‌هایی مثل jena سعی در ایجاد جدول شبه رابطه‌ای ویژگی‌ها دارند که هدف گردآوری اطلاعاتی راجع به چندین ویژگی روی لیستی از فاعل‌هاست. این روش همچنین برای پرسش‌هایی که نمی‌توانند تنها از یک جدول ویژگی استخراج شوند، نامناسب است و باید داده‌ها از چندین جدول ترکیب شوند. با این وجود این جداول ساختار شبه رابطه‌ای را روی داده‌های RDF شبه‌ساخت‌یافته اعمال می‌کنند. اعمال ساختار در جایی که واقعاً ساختاری وجود ندارد منجر به نمایش‌های پراکنده با مقادیر زیادی داده تهی در جداول ویژگی تشکیل‌یافته می‌شوند. مدیریت این جداول پراکنده در مقابل جداول فشرده‌تر نیاز به بار محاسباتی زیادی دارد.

2-2-2- ذخیره داده‌های RDF بصورت گراف

تحقیقات زیادی امکان ذخیره داده‌های RDF بصورت گراف را بررسی کرده‌اند. ولی مشکل مقیاس‌پذیری هنوز در این روش‌ها حل نشده است. در [Kim05] یک روش دیگر ارائه شده که مسیر داده‌های RDF را ذخیره می‌کند. ولی این روش در نهایت بر اساس یک پایگاه داده رابطه‌ای کار می‌کند. آن‌ها زیر گراف‌ها را در جداول رابطه‌ای مجزا ذخیره می‌کنند که باز هم برای حجم زیادی از داده‌ها مقیاس‌پذیر نیست. سایر تحقیقات روی اندازه‌گیری شباهت در وب معنایی متمرکز شده‌اند و مشکل مقیاس‌پذیری هنوز وجود دارد.

---

[1] statement based queries

3-2-2- روشهای چندین شاخصی

در[Har05] داده‌های RDF بر اساس چندین شاخص ذخیره می‌شوند. در این روش اطلاعات زمینه هم ذخیره می‌شوند و می‌توان با ایجاد ۶ شاخص همه‌ی ۱۶ الگوی دسترسی[۱] لازم برای چهارگانه‌ها را پوشش داد. می‌توان چهارگانه‌های متناظر با یک الگوی دسترسی را سریعاً بازیابی کرد. این روش مشابه پرسش‌های ساده بر اساس سه‌تایی است و امکان پردازش کارای پرسش‌های پیچیده‌تر را نمی‌دهد.

یک روش چند شاخصی مشابه نیز برای چهارگانه‌ها در [Woo05] پیشنهاد شده است که سه‌گانه‌های RDF را همراه با المان زمینه، بصورت چهارگانه ذخیره می‌کند. ۶ ترتیب متفاوت برای ۴ نوع نود مختلف وجود دارد طوریکه هر مجموعه‌ای از ۱ تا ۴ نود بتوانند برای یافتن یک عبارت[۲] یا مجموعه‌ای از عبارات استفاده شوند. هر کدام از این ترتیب‌ها یک شاخص مرکب را ایجاد می‌کنند و مستقلاً شامل همه عبارات انبار RDF می‌شوند. در این روش از درختان AVL برای ذخیره و سازماندهی المان‌های این شاخص‌ها استفاده شده است و هدف پاسخگویی به پرسشهای ساده بر اساس عبارت[۳] است. ۶ ترتیبی که در این روش لحاظ می‌شوند، همگی یک ترتیب چرخشی دارند و تمام 4!=24 جایگشت ممکن چهارگانه‌ها و 3!=6 جایگشت ممکن سه‌گانه‌ها لحاظ نشده‌اند. بنابراین اگر نود زمینه را لحاظ نکنیم، تعداد شاخص‌های مورد نیاز همان سه ترتیب چرخشی {s,p,o},{p,o,s},{o,s,p} هستند. این شاخص‌ها یک لیست مرتب از فاعل‌های تعریف شده برای یک ویژگی خاص را نمی‌توانند ارائه کنند. بنابراین

---

[۱] access pattern
[۲] statement
[۳] simple statement based queries

این روش نمی‌تواند پرسش‌های پیچیده را بصورت مناسب پاسخگو باشد.

4-2-2- روش دسته‌بندی افقی[1]

این روش در [Aba07] پیشنهاد شده است. در این روش جدول سه‌گانه‌ها به n جدول ۲ ستونی تجزیه می‌شود. بدین ترتیب برای هر ویژگی یک جدول ارائه می‌شود که n تعداد ویژگی‌هاست. هر جدول شامل ستون‌های فاعل و مفعول است. فاعل‌های چند مقداری، فاعل‌هایی که برای یک ویژگی چندین مفعول دارند، توسط چندین سطر در جدول با مقادیر فاعل‌های یکسان نشان داده می‌شوند. هر جدول بر اساس فاعل مرتب شده است و بنابراین فاعل‌های خاص به سادگی قابل بازیابی هستند. برای بازیابی اطلاعات چندین ویژگی یک مجموعه از فاعل‌ها می‌توان از آن استفاده کرد. پیاده‌سازی با یک DBMS ستون‌گرا انجام شده است (این DBMS برای مورد خاص دسته‌بندی عمودی طراحی شده و در مقابل DBMS سطرگرا وجود دارد که در مورد فشرده‌سازی و کارایی کاراست) و برای پردازش مواردی که ویژگی‌ها بصورت متغیرهایی با مقادیر مشخص هستند، مزایای زیادی دارند.

همچنین در [Aba07] پیشنهاد شده است که ستون مفعول نیز می تواند شاخص‌گذاری شود (مثلاً با استفاده از درخت B+ کلاستر بندی نشده) و یا اینکه یک کپی جدید از جدول ایجاد کرد که روی ستون مفعول کلاستربندی شده است. بنابراین یک روش چند شاخصی در یک معماری ویژگی‌گرا برای مدیریت داده در وب معنایی ارائه شده است. در اصل این معماری برای پاسخ به پرسش‌هایی است که مقدار ویژگی مشخص است یا اینکه جستجو به یک سری ویژگی خاص محدود شده است. در حقیقت مهمترین نوآوری [Aba07]، یکپارچه‌سازی جدول‌های دوستونی ویژگی‌ها در یک DBMS

[1] vertical partitioning approach

ستون‌گرا است. عدم کارایی جدول ویژگی‌ها در پرسش‌هایی که مقدار ویژگی نامشخص است یا در زمان اجرا مشخص می‌شود، اثبات شده است.

برای اجرای پرسش‌هایی که مقدار ویژگی آن‌ها مشخص نیست، همه جدول‌های دوستونی باید در پرسش بررسی شوند و نتایج با هم مجتمع شوند (با عبارت‌های اجتماع پیچیده یا الحاق).

بررسی‌های انجام شده در[Aba07]، بر اساس این فرضیه است که برای پرسش‌هایی که ویژگی آن‌ها مشخص نیست ، تنها روی ۲۸ ویژگی خاص از ۲۲۱ ویژگی اجرا شود که این فرضیه در دنیای واقعی غیرمحتمل است و نتایج درستی حاصل نمی‌شود.

به عنوان مثال داده‌هایی را در نظر بگیرید که شامل اطلاعات دانشگاهی ۴ فرد خاص است و همه ویژگی‌ها برای همه فاعل‌ها تعریف نشده‌اند. یک پرسش روی این مجموعه "یک فرد خاص چه روابطی با MIT دارد؟" است. یک پرسش دیگر "یافتن افرادی است که همان ارتباطی را با stanford دارند که یک فرد خاص با Yale دارد.". در این پرسش‌ها ویژگی مشخص نیست و باید اطلاعاتی راجع به چندین ویژگی را جمع‌آوری کرد. علاوه براین بعضی مواقع نیاز به الحاق‌های پیچیده روی لیست فاعل‌های مرتبط با یک مفعول خاص از طریق چندین ویژگی است. در این موارد مقید سازی نه با ویژگی و نه با فاعل است بلکه با مفعول است.

5-2-2- مدل شاخص‌گذاری HEXASTORE

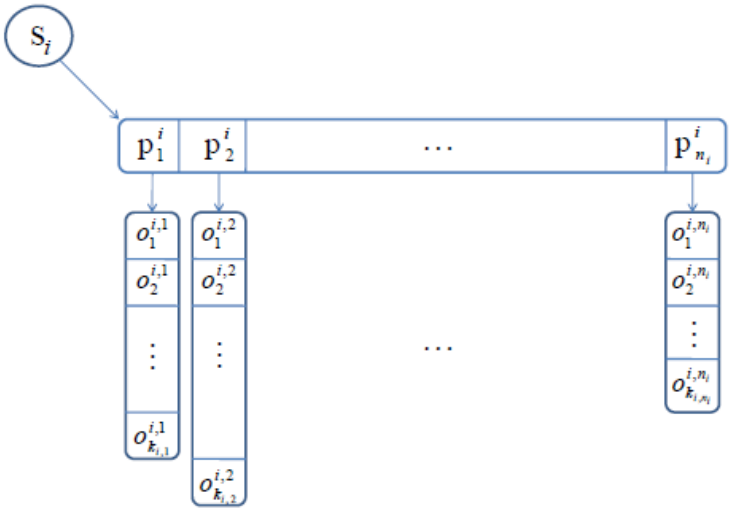
پرسش‌های معمول در وب معنایی ضرورتاً بر اساس ویژگی نیستند بلکه در بعضی مواقع افراد به دنبال روابط بین افراد هستند و اصلاً اینکه روابط از چه نوعی ممکن است باشند، مشخص نشده است. روش‌های فوق الذکر برای ذخیره RDF، با هدف پاسخ‌گویی به چنین سؤالاتی طراحی نشده‌اند. بلکه معمولاً با هدف گردآوری سه‌تایی‌هایی که دارای یک ویژگی خاص هستند یا فاعل خاصی

دارند است. در بعضی موارد کلید هش subject-object وجود دارد که برای شناسایی ویژگی‌ها به کار می‌رود. به هر حال روشی برای ارائه مستقیم قابلیت‌هایی مثل ارائه لیست فاعل‌ها یا ویژگی‌های مربوط به یک مفعول خاص وجود ندارد. چنین کاربردهایی قابل پیاده‌سازی هستند. برای مثال یک مجموعه از ۶ شاخص همه الگوهای دسترسی که در یک پرسش RDF ممکن است وجود داشته باشد را پوشش می‌دهد. بنابراین هرچند شاخص‌گذاری چندگانه ممکن است برای یک جدول پیچیدگی ترکیبیاتی ایجاد کند، ولی در مورد داده‌های RDF عملی است. در [Wei08] روش HEXASTORE ارائه شده است که مشکلات بالا را ندارد.

HEXASTORE هیچ تفاوتی بین المان‌های RDF قائل نمی‌شود و فاعل و مفعول و ویژگی را به یک دید نگاه می‌کند. بنابراین هر کدام یک شاخص خاص خود را خواهند داشت. علاوه براین هر ترتیبی از این سه المان دارای اهمیت است. انبار داده‌ای که همه ۶ شاخص فوق را ذخیره کند، HEXASTORE نامیده می‌شود. هر شاخص در این مدل پیرامون یک المان RDF است و یک اولویت برای دو المان دیگر تعریف می‌کند. در دنیای سه‌گانه‌های RDF همه چیز بر اساس ویژگی نیست، بلکه باید سه‌گانه‌ها را بر اساس فاعل و مفعول نیز در نظر گرفت. در حالت اول برای یک فاعل خاص یک بردار ویژگی‌ها وجود دارد و برای هر المان بردار ویژگی، یک بردار از مقادیر مفعول‌ها وجود دارد. برای یک فاعل و یک ویژگی خاص، لیست مفعول‌ها در شاخص‌گذاری بر اساس فاعل و در شاخص‌گذاری بر اساس ویژگی یکی است. بنابراین تنها یک کپی از چنین لیست‌هایی در معماری شاخص‌ها لازم است. پس در ایجاد آخرین شاخص بر اساس مفعول، برای هر فاعل خاص، لیست از قبل موجود است و برای هر ویژگی خاص نیز لیست از قبل ایجاد شده است. بنابراین ۶ نوع شاخص‌گذاری

داریم مثلاً شاخص SPO، ابتدا بر اساس فاعل دسته‌بندی صورت می گیرد و برای هر فاعل لیستی از ویژگی‌هایش را داریم و برای هر ویژگی لیستی از مفعول‌ها را داریم (شکل ۳). روش ارائه شده در [Aba07] نوعی شاخص‌گذاری بر اساس PSO است.

همچنین به جای ذخیره‌سازی کل رشته یا URI از کلید آن‌ها استفاده می‌کنیم. بنابراین علاوه بر ۶ شاخص با استفاده از شناسه‌های هر المان RDF، یک جدول نگاشت داریم که هرکلید را به المان متناظرش نگاشت می‌کند. همانطور که گفتیم ۳ جفت از شاخص‌ها در این روش ۶ تایی، لیست نهایی یکسانی دارند. بنابراین شاخص SPO و PSO لیست مفعول نهایی یکسانی دارند و الی آخر. در اصل بدترین حالت فضای اشغال شده در HEXASTORE، ۶ برابر فضای لازم برای ذخیره سه‌گانه‌ها در یک جدول سه‌گانه نیست. به این دلیل که کلید هرکدام از منابع دو بار در لیست اول و دو بار در بردار و یک بار در لیست ظاهر می‌شود. در بدترین حالت فرض کنید که در سه‌گانه <si,pj,ok> ، هر شناسه تنها یکبار در مجموعه داده RDF آمده باشند. آنگاه کلید هر کدام از منابع پنج بار در شاخص HEXASTORE ظاهر می‌شود. بنابراین در بدترین حالت فضای HEXASTORE پنج برابر فضای مورد نیاز برای ذخیره سازی کلیدها در جدول سه‌گانه‌هاست. در عمل فضا کمتر است چون هر عنصر بیشتر از یکبار در کل مجموعه داده ظاهر می‌شود. [Wei08]



شکل ۳ : شاخص‌گذاری SPO در HEXASTORE

3-2-کارهای انجام شده در زمینه‌ی کشف ارتباط گروهی در وب اسناد

در [Wu05] ایده‌ی ایجاد مجموعه اولیه صفحات اسپم و توسعه آن برای شناسایی سایر اعضای ارتباط گروهی در وب اسناد، بررسی شده است. مجموعه اولیه صفحات اسپم بر اساس تعداد اعضای مجموعه لینک‌های مشترک ورودی و خروجی صفحات شکل می‌گیرد. سپس در مرحله توسعه، مجموعه پایه برای کشف سایر اعضای ارتباط گروهی توسعه داده می‌شود. بدین ترتیب که اگر یک صفحه به تعداد زیادی از اعضای ارتباطات گروهی اشاره کند، به احتمال زیاد خود این صفحه نیز از اعضای همان ارتباط گروهی است. بعد از شناسایی اعضای ارتباطات گروهی، باید به نحوی ماتریس گراف کلی را تغییر داد که اثر لینک‌های ما بین اعضای اسپم، حذف یا کمرنگ گردد. بدین منظور یک راه حل حذف نودهای مشکوک و تمامی یال‌های مرتبط با آن‌هاست. این جریمه خیلی سخت است و نودهایی که تشخیص اسپم یا تشخیص عدم اسپم در مورد آن‌ها اشتباه بوده حذف می‌شوند و هیچ شانسی برای حضور در رتبه‌بندی ندارند. راه حل بعدی حذف لینک‌های بین اعضای تشخیص داده شده به عنوان ارتباط گروهی است. این راه حل کمی نرم‌تر برخورد می‌کند و

به اعضای ارتباط گروهی، شانس حضور در رتبه‌بندی را می‌دهد. بدین ترتیب نودهای مجموعه اسپم‌ها در الگوریتم‌های رتبه‌بندی حضور دارند تا رتبه‌بندی جدید با حذف اثر ارتباط گروهی انجام شود. در این پایان نامه مفهوم ارتباط گروهی برای وب‌داده‌ها و با هدف گمراه سازی الگوریتم‌های رتبه‌بندی وب‌داده‌ها، تعمیم داده شده است. الگوریتم کشف ارتباط گروهی برای وب داده‌ها معرفی شده و روش‌های مقابله با این نوع اسپم نیز معرفی شده است.

2-4-کارهای انجام شده در زمینه‌ی مدل‌های پایه رتبه‌بندی

در این بخش مدل‌های ابتدایی و اولیه برای رتبه‌بندی معرفی شده‌اند. در تحقیقات انجام شده توسط محققان، نسخه‌های متفاوتی از این مدل‌ها توسعه داده شده است که هر کدام کاربردهای خاصی دارند و این پایان‌نامه به آن‌ها نمی‌پردازد.

2-4-1- متریک‌های اعتماد برای رتبه‌بندی

یک روش برای اندازه‌گیری کیفیت صفحات و رتبه‌بندی آن‌ها، استفاده از متریک‌های اعتماد است. دو راه معمول برای تعیین اعتماد استفاده از سیاست‌ها[1] و شهرت[2] است.

سیاست‌گذاری، تعیین شرایط لازم برای بدست آوردن اعتماد است و می‌توان در شرایط خاص نتایج را پیش‌بینی کرد. در سیاست‌گذاری عملیات تبادل یا بررسی گواهینامه‌ها صورت می‌گیرد (گواهینامه‌ها اطلاعاتی هستند که توسط یک موجودیت خاص صادر شده‌اند و دارای امضای دیجیتال هستند). مثلاً دارنده گواهینامه مدرک دانشگاهی، توسط دانشگاه صادر کننده گواهینامه، تأیید شده است. گواهینامه، دارنده را با دانشگاه و افرادی که در آن زمینه فارغ‌التحصیل شده‌اند

---

[1] policy
[2] reputation

مرتبط می‌کند. بنابراین یک سری سیاست‌هایی برای أخذ گواهینامه‌ها تعریف می‌شوند و سیستم بر اساس سیاست‌ها گواهینامه صادر می‌کند. برای بررسی قابلیت اعتماد نیز یک سری سیاست‌های دیگر تعریف می‌شود. بنابراین موجودیت‌هایی که تمامی گواهینامه‌های لازم برای احراز اعتماد را دارند مورد اعتماد، و بقیه مورد اعتماد نیستند. با توجه به پیچیدگی بحث گواهینامه‌ها و لزوم وجود یک سری قوانین اعتماد سلیقه‌ای، به نظر اعتماد بر اساس سیاست‌گذاری برای کاربردهای عام مثل موتور جستجو غیرممکن است. چون قوانین اعتماد تعریف شده توسط موتور جستجو ممکن است توسط کاربران قابل قبول نباشد و با تعریف سیاست‌ها، موتور جستجو وابسته به سایت‌های خاصی می‌شود و از هدف اصلی خود که صرفاً بازیابی نتایج باکیفیت بدون توجه به سیاست خاصی است، باز می‌ماند.

شهرت یک نوع ارزیابی بر اساس تاریخچه تعاملات مستقیم یا غیر مستقیم موجودیت‌ها با یکدیگر است که به اهداف موتور جستجو نزدیکتر است. نحوه ترکیب این تعاملات و چگونگی استنتاج از این تعاملات متفاوت است. شهرت و گواهینامه، اعتماد را از یک موجودیت به موجودیت دیگر اعطا می‌کنند. ولی هر روش مشکلات خاص خود را دارد که انگیزه تحقیقات در هر حیطه است.

متریک اعتماد در وب یک تکنیک برای پیش بینی اعتماد بین دو موجودیت است که صریحاً میزان اعتماد بین آن‌ها ذکر نشده است. موجودیت‌هایی که قبلاً رابطه تراکنش با هم نداشته‌اند، میزان اعتماد بین آن‌ها مشخص نیست و بنابراین اعتماد بین آن‌ها را باید از طریق انتشار اعتماد پیش بینی کرد. علت انتشار اعتماد اینست که شما به دوست خود اعتماد بیشتری دارید تا به یک غریبه و همچنین احتمالاً به دوست دوست خود

نیز بیشتر از یک غریبه اعتماد خواهید کرد. متریک‌های اعتماد به دو دسته متریک‌های عمومی و متریک‌های اعتماد محلی تقسیم می‌شوند[1].

متریک‌های عمومی، متریک‌هایی هستند که برای هر موجودیت یک رتبه کلی تعیین می‌شود که میزان اعتماد همه موجودیت‌ها به آن موجودیت را مشخص می‌کند. چنین متریک‌هایی در موتورهای جستجو و مواردی که میزان اعتماد به صورت کلی باید اندازه‌گیری شود کاربرد دارد. از جمله متریک‌های عمومی اعتماد می‌توان الگوریتم‌های تحلیل لینک مثل pageRank را نام برد. آنچه مسلم است ورودی به یک متریک اعتماد عمومی، ماتریس مجاورت یک گراف وزن‌دار است که ستون‌های آن موجودیت‌ها و سطرها نیز موجودیت‌ها هستند. خروجی یک بردار است که میزان اعتماد سراسری به هر موجودیت را تعیین می‌کند و به عبارت دیگر موجودیت‌ها را بر اساس میزان اعتماد عمومی به آن‌ها مرتب سازی می‌کند.

در متریک‌های محلی اعتماد باید میزان اعتماد هر موجودیت به سایر موجودیت‌ها جداگانه تعیین و مشخص شود. کاربرد متریک‌های محلی در سیستم‌های توصیه‌گر و مواردی است که یک پروفایل از هر موجودیت در دسترس است که شامل تجربه‌های تراکنش او و معیارهای مهم او می‌باشد. از جمله متریک‌های محلی اعتماد می‌توان advogato و tidalTrust و appleSeed و... را نام برد [Gol06]. آنچه مسلم است ورودی به یک متریک اعتماد محلی ماتریسی است که سطرها و ستون‌های آن موجودیت‌ها هستند و بعضی از اعضای آن تعیین نشده‌اند و خروجی همان ماتریس است که تمامی اعضای آن مشخص شده‌اند.

---

[1] http://www.trustlet.org

۲-۴-۲- الگوریتم تحلیل لینک PageRank

زمانی که گوگل، هنوز یک پروژه دانشگاهی در استنفورد بود، طراحان آن، فرمول اصلی خود را در محاسبه PageRank طبق معادله (۱۱) بیان کرده‌اند. البته امکان دارد که آن‌ها دیگر از این فرمول استفاده نکنند، اما امروزه هم به اندازه کافی دقیق به نظر می‌رسد. سیستم رتبه‌بندی در PageRank، بر اساس الگوریتم حرکت تصادفی است و احتمال اینکه پیمایشگر صفحات، روی هر صفحه باشد را محاسبه می‌کند. این الگوریتم یک لینک از i به j را به عنوان دلیلی بر اهمیت j قلمداد می‌کند. علاوه براین میزان اهمیت انتقالی از i به j، با اهمیت i رابطه مستقیم و با تعداد صفحاتی که i به آن‌ها اشاره می‌کند رابطه غیرمستقیم دارد.

$$r^{k}(j) = \alpha \sum_{\forall i \text{ which links to } j} \frac{r^{k-1}(i)}{|L(i)|} + \frac{(1-\alpha)}{|E|} \qquad (۱۱)$$

در الگوریتم PageRank محاسبه‌ی رتبه‌ها طبق معادله‌ی (۱۱) تا زمانی ادامه پیدا می‌کند که حداکثر تغییر مقادیر رتبه‌ها از یک آستانه کمتر باشد. در این معادله، α یک فاکتور کندکننده[۱] است که مقداری بین ۰ تا ۱ دارد. معمولاً برای α مقداری معادل ۰,۸۵ انتخاب می‌شود. L(i) مجموعه لینک‌های خروجی از i با مقصدی متفاوت از j هستند. E مجموعه کل موجودیت‌ها است. در ابتدا الگوریتم رتبه همه موجودیت‌ها با یک مقدار یکسان مقداردهی می‌شود [Pag98].

۲-۴-۳- الگوریتم تحلیل لینک HITS

HITSیک الگوریتم تحلیل لینک است که "مراکز" و "ذیصلاحان" را رتبه‌بندی می‌کند. این الگوریتم قبل از pageRank ایجاد

---

[۱] Damping factor

[2] hubs

شده است و علت رتبه‌بندی "مراکز" و "ذیصلاحان" به علت دیدگاه خاص ایجاد صفحات در وب اولیه است. در سال‌های آغازین وب صفحات خاصی به نام "مرکز" به صورت دایرکتوری‌های خیلی بزرگ ایجاد شده بود. "مرکز" واقعاً در زمینه اطلاعاتی که نگهداری می‌کرد ذیصلاح نبود، اما به عنوان یک جمع‌بندی از حجم وسیعی از اطلاعات مناسب بود و کاربران را به سمت صفحات ذیصلاح هدایت می‌کرد. به عبارت دیگر یک "مرکز" خوب صفحه‌ای بود که به تعداد بیشتری از صفحات ارجاع می‌کرد و "ذیصلاح" خوب صفحه‌ای بود که توسط تعداد زیادی "مرکز" ارجاع شده بود. این روش به هر صفحه دو نمره اختصاص می‌داد. نمره "صلاحیت" که ارزش محتوا صفحه را نشان می‌دهد و نمره "مرکزیت" که ارزش ارتباطات آن به سایر صفحات را نشان می‌دهد. [Kle99]

در مرحله اول این الگوریتم، مجموعه جواب‌ها به پرسش یافت می‌شود و محاسبات برای رتبه‌بندی، تنها روی این مجموعه جواب‌ها انجام می‌شود. مقادیر مرکزیت و صلاحیت از روی یکدیگر و به صورت بازگشتی تعریف می‌شوند. مقدار صلاحیت مجموع مقادیر مرکزیت صفحات ارجاع کننده به آن است و مقدار مرکزیت مجموع مقادیر صلاحیت صفحاتی است که به آن ارجاع می‌کنند. الگوریتم چندین بار تکرار می‌شود که در هر تکرار مقادیر مرکزیت و صلاحیت هر نود دوباره محاسبه می‌شود. HITS مثل PageRank یک الگوریتم چرخشی است و تا زمانی‌که مقادیر مرکزیت و صلاحیت به یک حالت ایستا برسد تکرار ادامه می‌یابد.

این الگوریتم بر خلاف pageRank در زمان پرسش اجرا می‌شود و نه در زمان ایجاد شاخص. همچنین مقادیر مرکزیت و صلاحیت اختصاص یافته به هر صفحه مختص آن پرسش خاص هستند و HITS تنها روی

---

[1] authority

زیر مجموعه کوچکی از اسناد مرتبط اجرا می‌شود و نه روی کل وب .[Sig05]

فصل 3- روش پیشنهادی

هدف اصلی این پایان نامه پیاده‌سازی تعدادی از فازهای یک موتور جستجوی معنایی با استفاده از ابزارهای با کد باز موجود و بهینه‌سازی الگوریتم رتبه‌بندی مجموعه داده‌ها در موتورهای جستجوی معنایی موجود است. شکل ۴شکل ٤ طرح پیشنهادی برای موتور جستجوی معنایی و فازهای آن را نشان می‌دهد.

در چارچوب پیشنهادی برای موتورهای جستجوی معنایی، فازهای پیمایش و رتبه‌بندی مستقل از پرسش (برای حذف تاثیر ارتباطات گروهی و محاسبه رتبه اولیه)، یکپارچه‌سازی، شاخص‌گذاری و پرسش و رتبه‌بندی وابسته به پرسش، در نظر گرفته شده است. داده‌های خروجی هر فاز به فاز بعدی وارد می‌شوند. شناسایی موجودیت‌های یکسان و تشکیل گراف موجودیت‌های مشابه، فرآیندی مستقل از کلمه کلیدی است. بنابراین بهتر است قبل از شاخص‌گذاری انجام شود تا زمان انتظار کاربر بالا نرود. علاوه بر این از نتایج آماده آن نیز (که روی شاخص ذخیره شده است) در زمان رتبه‌بندی وابسته به پرسش (رتبه‌بندی بر اساس نزدیکی به عبارت جستجو) استفاده می‌شود.

با توجه به اینکه یک موتور جستجوی معنایی با منبع باز در حال حاضر در دسترس نیست، بایستی تعدادی از فازها پیاده‌سازی شوند تا بتوان نتایج الگوریتم رتبه‌بندی پیشنهادی را مشاهده کرد. فاز پیمایش با استفاده از ابزار با کد باز LDSpider روی اطلاعات دارویی LOD، انجام شده است و بدین ترتیب مجموعه داده، آماده شده است. کارکشف ارتباطات گروهی نیز در مرحله قبل از شاخص‌گذاری و در فاز رتبه‌بندی مستقل از پرسش انجام می‌شود. زیرا این فرآیند وابسته به

کلمه کلیدی نیست و می‌تواند مستقلاً انجام شود. هر چند اگر آستانه تشخیص ارتباطات گروهی را در مرحله اول بالاتر ببریم تا نرخ تشخیص نادرست[1] پایین‌تر بیاید، لازم است تا در مرحله بعد از بازیابی نتایج پرسش، نیز به دنبال ارتباطات گروهی وابسته به کلمه کلیدی باشیم. پس از کشف ارتباطات گروهی و کاهش وزن ارتباطات بین اعضای ارتباط گروهی ایجاد شده، ماتریس صلاحیت نام‌گذاری با این تفاوت ایجاد می‌شود که اعضای ماتریس به جای اینکه صرفاً استفاده یا عدم استفاده را نشان دهند، تعداد دفعات استفاده را نشان می‌دهد. در روش پیشنهادی، تعداد دفعات استفاده متقابل اعضای ارتباط گروهی در ماتریس صلاحیت نام‌گذاری تاثیری ندارد. سپس pageRank وزن‌دار روی آن اعمال می‌شود و دامنه‌ها با توجه به فاکتور "استفاده" رتبه‌بندی می‌شوند. از این رتبه، در معادله (۱۲) برای وزن‌دهی ارتباطات استفاده می‌شود. با اعمال مجدد pageRank وزن‌دار رتبه کیفیت نهایی همه مجموعه‌داده‌ها با توجه به فاکتورهای نوع‌لینک و "استفاده" حاصل می‌شود. بعد از رتبه‌بندی مستقل از پرسش، مرحله یکپارچه‌سازی داده‌ها صورت می‌گیرد. این مرحله نمی‌تواند قبل از رتبه‌بندی و کشف ارتباط گروهی صورت گیرد زیرا با یکپارچه‌سازی، URI موجودیت‌ها بازنویسی می‌شود و امکان شناسایی همه اعضای ارتباط گروهی حذف می‌شود. بعد از شناسایی موجودیت‌های مشابه هر شناسه، گراف ارتباطات SameAs بین شناسه‌ها برای رتبه‌بندی شناسه‌ها شکل می‌گیرد و چون اینجا همگی لینک‌ها از نوع SameAs یا مشابه آن است، تنها اصالت لینک در وزن‌دار کردن این گراف مهم است. بدین ترتیب رتبه هر موجودیت در گراف موجودیت‌های

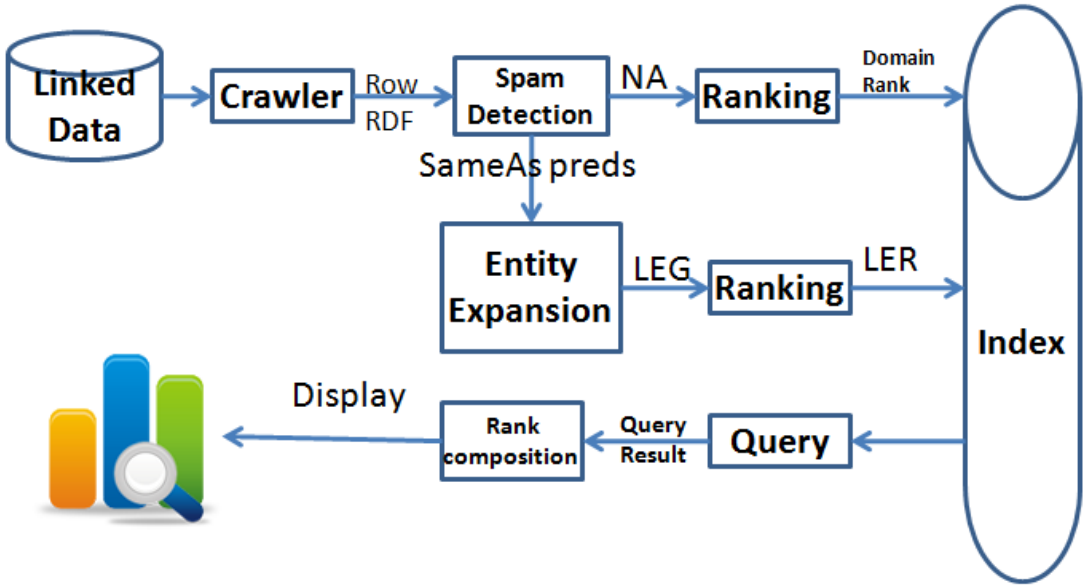
[1] false positive موجودیتهایی که جز ارتباط گروهی اسپم نیستند ولی به اشتباه اسپم تشخیص داده شده‌اند.

مشابه نیز بدست می‌آید. نتایج رتبه‌بندی مرحله قبل و رتبه‌بندی موجودیت‌ها در یکپارچه‌سازی، در شاخص Lucene با ساختار پیشنهادی ذخیره شده است. پرسش با استفاده از توابع کتابخانه‌ای Lucene، نتایج را بازیابی می‌کند و نتایج بازیابی شده با استفاده از رتبه اصالت و رتبه محاسبه شده برای موجودیت در مرحله رتبه‌بندی مستقل از پرسش، مرتب سازی می‌شوند. در اینجا ارتباط[1] مفهومی ندارد چون همگی موجودیت‌ها در جستجوی کلمه کلیدی یا شامل کلمه کلیدی بوده‌اند یا مشابه یکی از موجودیت‌هایی بوده‌اند که کلمه کلیدی را داشته است و تنها معیار مرتب سازی اصالت است. قسمتی از رتبه‌بندی وابسته به پرسش (آنجا که بین موجودیت‌های یکسان در فاز یکپارچه‌سازی رتبه‌بندی انجام می‌شود) در واقع در مرحله آفلاین انجام گرفته است و ما در اینجا فقط نتایج رتبه‌بندی انجام شده را بازیابی می‌کنیم.

در ادامه جزئیات پیاده‌سازی و ابزارهای مورد استفاده در هر فاز با جزئیات بیشتر ارائه شده است.

---

[1] Relevancy



شکل ٤: طرح پیشنهادی برای موتور جستجوی معنایی

1-3-فاز پیمایش

اولین جزء تشکیل دهنده‌ی هر موتور جستجو، پوینده است. هدف پوینده بازیابی همه RDFهای روی وب است. پوینده برنامه‌ای است که داده‌های روی وب را بازیابی و در مکانی که به راحتی توسط موتور جستجوی مربوطه قابل دسترسی باشد، ذخیره می‌کند. پوینده با شروع از تعدادی URI اولیه، محتوای آن‌ها را در دیسک به فرمت چهارگانه ذخیره می‌کند و به صورت بازگشتی، URIهای جدید را برای پیمایش انتخاب می‌کند. پوینده باید میلیون‌ها صفحه و داده‌های موجود در وب را در زمانی کوتاه دانلود کرده و متناوباً تغییرات را کنترل و اطلاعات دانلود شده را بهنگام کند. علاوه بر این پوینده نباید بار زیادی روی وب سایت‌هایی که از آن‌ها بازدید می‌کند، اعمال کند. به نکات زیر در مورد پوینده باید دقت شود:

1. پوینده نباید با حجم زیادی از درخواست‌ها بار زیادی روی سرورهای مقصد اعمال کند و مطابق با سیاست‌های موجود در فایل robots.txt عمل کند.

2. پوینده باید بیشترین تعداد URI را در کمترین زمان ممکن با رعایت مورد ۱ بپیماید.
3. پوینده باید URIهای با کیفیت بالا را در اولویت قرار دهد[۱].

بنابراین طراحی پوینده وب داده‌ها با الهام از پوینده وب اسناد انجام می‌شود و علاوه بر آن باید اسناد RDF/XML را پیمایش کند و فرمت‌های HTML و ... را در صف پیمایش قرار ندهد.

1-1-3- ابزارها

سه ابزار قوی برای پویش وب داده‌ها در دسترس است. ابزار Slug[۲] یک چارچوب ماژولار و قابل پیکربندی فراهم می‌آورد که انعطاف‌پذیری بسیار زیادی برای چگونگی بازیابی و پردازش و ذخیره اطلاعات دارد. Slug با جاوا پیاده‌سازی شده است و از کتابخانه‌های Jena نیز استفاده می‌نماید.

RDF Crawler[۳] بصورت برنامه کنسول و کتابخانه در دسترس است و می‌توان در برنامه جاوا از توابع آن استفاده کرد. پوینده سوم و مشهورترین پوینده، LDSpider[٤] است . هدف پروژه LDSpider ایجاد چارچوب پیمایش برای داده‌های پیوندی است. از آنجا که نیازمندی‌ها و مشکلات پیمایش وب داده‌ها متفاوت با وب معمولی است، نمی‌توان از ابزارهای وب معمولی برای پیمایش استفاده کرد.

می‌توان فایل jar پوینده را در پروژه جاوا استفاده کرد که امکان پیمایش داده‌های پیوندی با شروع از یکURL خاص و با استراتژی تعیین شده و تا سطح معینی را می‌دهد و نتیجه‌ی

---

۱ http://code.google.com/ldspider
۲ http://www.ldodds.com/projects/slug/
۳ http://ontobroker.semanticweb.org/rdfcrawl/
٤ http://code.google.com/p/ldspider/

خروجی فایل چهارگانه‌های پیوندی است که به فرمت سه‌گانه به علاوه اصالت (آدرسی که سه‌گانه از آن اخذ شده است) می‌باشد. البته می‌توان چهارگانه‌ها را با استفاده از دستور[1] sed به سه‌گانه های RDF تبدیل نمود.

از ویژگی‌های این پوینده عبارتند از:

1. مدیر محتوا برای فرمتهای مختلف
   - شامل مدیرهایی برای خواندن انواع فرمت‌ها شامل RDF(XML) و N-Triples و N-Quads
   - مدیر برای ارتباط با سرور Any23 server برای استخراج RDF از سایر فرمت‌ها مثل RDFa
   - طراحی ساده ی واسط برای پیاده‌سازی مدیرهایی که کاربر برای فرمت‌های خاصی ارائه می‌کند
2. استراتژی‌های پیمایش متفاوت
   - اول سطح
   - اول عمق
   - پیمایش دلخواه اطلاعات
3. حوزه پیمایش
   - می‌توان پیمایش را به صفحات با پیشوند حوزه خاص محدود کرد.(محدود سازی اصالت)
4. فرمت خروجی
   - فرمت فایل خروجی می‌تواند RDF/XML یا NQuad باشد.
   - پوینده می‌تواند اطلاعات خروجی را در انبار سه‌گانه با استفاده از SPARQL UPDATE ذخیره کند. یا اینکه از گراف‌های نامدار[2] برای سازماندهی عبارات نوشته شده با توجه به اصالت استفاده کند

---

[1] http://www.sedtutorial.com/

[2] named graph

- امکان شمول اطلاعات اصالت فراهم است

امکان استفاده از LDSpider به دو طریق فراهم است:

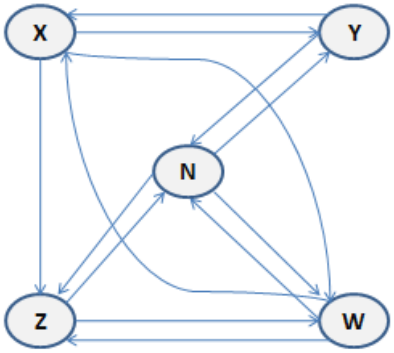
1. از طریق دستور خط فرمان (CLI)
2. از طریق استفاده از کتابخانه در جاوا (API)

در موتور جستجوی پیشنهادی، به دلیل قابلیت‌های فراوان پوینده LDSpider، برای پیمایش اطلاعات دارویی موجود در وب داده‌ها، از آن پوینده استفاده شده است. برای تهیه مجموعه داده اولیه، LDSpider با شروع از مکان "http://wtlab.um.ac.ir/parameters/wtlab/filemanager/searchEngine/drugbank.rdf" و استراتژی پیمایش اول سطح، شروع به پیمایش وب داده‌ها تا سه سطح می‌نماید. بعد از ۱۲ ساعت مجموعه داده اولیه، که شامل سه‌گانه‌ها همراه با اصالت آن‌هاست، برای شاخص‌گذاری آماده شد.

2-1-3- استراتژی تولید اسپم

برای تهیه و تزریق اسپم به این مجموعه داده، دو تکنیک اسپم محتوا و اسپم لینک به کار گرفته شده‌است. نتایج پرسش نیز روی مجموعه داده آلوده شده با انواع اسپم، بررسی شده است. مجموعه داده آلوده با اسپم محتوا را ایجاد نموده‌ایم. در ادامه پنج سه‌گانه اسپم محتوا، که مجموعه داده جعلی را به سایر مجموعه داده‌های مشهور متصل می‌کند، به عنوان اسپم محتوا در نظر گرفته شده‌است. اسپم لینک با دو ساختار متفاوت تزریق شده است. استراتژی اول برای ایجاد اسپم لینک، استفاده از مفهوم "استفاده کردن یک مجموعه داده از مجموعه داده دیگر" به عنوان لینک است و موجودیت‌ها را به عنوان URI مجموعه داده‌ها در نظر می‌گیریم. بدین ترتیب ساختار اسپم در شکل ۵ شکل ۱نشان داده شده است. شمای کلی

اسپم لینک ایجاد شده نیز در شکل ۶ بیان شده است. این اسپم لینک در تست دوم به مجموعه داده تزریق می‌شود.



شکل ۵ : ساختار اسپم لینک تزریقی

<x>: <y>p1<x>,<w>p2<x>,<z>p3<y>

<y>: <x>p4<y>,<n>p5<x>

<n>:<y>p6<w>,<w>p7<n>,<z>p8<n>

<w>: <n>p9<w>,<z>p10<n>

<z>: <n>p11<z>,<w>p12<z>

شکل ٦: شمای چهارگانه‌های اسپم لینک تزریقی

در روش دوم ایجاد اسپم لینک، با تعیین تعداد حوزه‌ها، تعداد دلخواهی چهارگانه از آن حوزه‌ها و حوزه‌های موجود ایجاد شده است، هرچند برای اصالت چهارگانه‌ها فقط می‌توان از حوزه‌های اسپم ایجاد شده استفاده کرد. چون هدف اصلی ارتباط گروهی افزایش تعداد لینک‌های ورودی به همدیگر برای تقویت متقابل است، بنابراین باید ارتباطات دو طرفه بین هر مجموعه داده اسپم با اکثر مجموعه داده‌های اسپم برقرار شود. همچنین در عین حال می‌توان با سایر مجموعه داده‌های موجود نیز ارتباط برقرار شود و به نوعی سایر مجموعه داده‌ها نیز وادار به برقراری ارتباط گردند. این تلاش برای برقراری ارتباط ورودی می‌تواند در قالب فریفتن مجموعه داده هدف و استفاده از موجودیت‌های مجموعه داده اسپم یا برقراری

لینک از طریق ایجاد نظر یا تبانی مجموعه داده‌ها با هم و افزایش اعضای ارتباط گروهی باشد. بررسی‌های انجام شده حاکی از آنست که هر چه تعداد اعضای ارتباط گروهی بیشتر باشد، شانس آن در گمراه سازی الگوریتم تحلیل لینک بالاتر می‌رود. برنامه تولید اسپم که برای این قسمت پیاده‌سازی نموده‌ایم در ضمیمه آمده است.

الگوریتم کشف ارتباطات گروهی برای مثال اول، ابتدا N,W را به عنوان مجموعه پایه اسپم انتخاب می‌کند. اگر آستانه را ۳ انتخاب کنیم، ابتدا Z به مجموعه اسپم اضافه می‌شود و در تکرارهای بعدی X و Y به ترتیب به مجموعه اسپم‌ها اضافه می‌شوند. برای اینکه این حوزه‌ها در نتایج پرسش بازیابی شوند، نام تعدادی از داروها (کلمات کلیدی) را در مجموعه اسپم تکرار می‌کنیم تا این اسپم‌های ایجاد شده، در نتایج اولیه جستجو ظاهرشوند و همچنین تعدادی لینک به حوزه‌های مشهور در حوزه‌های اسپم ،با استفاده از لینک SameAs، ذکر می‌گردد تا در مرحله یکپارچه‌سازی نیز تعداد دیگری از اسپم‌ها به نتایج جستجو اضافه گردند.

به عنوان مثال:

n: <drugbank:drug1> <sameAs> <n:n1>,....

که خاطر نشان می‌کند موجودیت n1 در حوزه جعلی n با موجودیت drug1 در حوزه drugbank یکسان است.

2-3-رتبه‌بندی

رتبه‌بندی در فرآیند جستجو بسیار مهم است و نقش بسیار مهمی در رضایت کاربر و جذب مشتری دارد. زمان انتظار کاربر برای رتبه‌بندی نتایج باید کمینه گردد. علاوه‌براین، داده‌های با کیفیت و نزدیکی بیشتر با عبارت جستجو، باید در رتبه‌های بالاتر ظاهر شوند. برای کاهش زمان انتظار کاربر سعی می‌شود

که اکثر فعالیتهای مرتبط با رتبه‌بندی بصورت آفلاین انجام شود تا زمان انتظار کاربر تا حد ممکن کمینه گردد.
رتبه‌بندی در موتور جستجو دو بار انجام می‌شود. رتبه‌بندی مستقل از پرسش بصورت آفلاین انجام می‌شود و در زمان پاسخگویی موتور جستجو تاثیری ندارد. الگوریتم تحلیل لینک برای شناسایی منابع با کیفیت بالا روی داده‌ها اعمال می‌شوند و نتایج حاصل از رتبه‌بندی مستقل از پرسش روی شاخص ذخیره می‌شود. در زمان انتظار کاربر، رتبه‌بندی وابسته به پرسش انجام می‌شود و داده‌های حاصل از اعمال پرسش روی شاخص بر اساس قابلیت اعتماد (نتایج بازیابی شده رتبه‌بندی مستقل از پرسش) و نزدیکی به پرسش (محاسبه نزدیکی چون وابسته به پرسش است در زمان رتبه‌بندی وابسته به پرسش انجام می‌شود)، مرتب سازی می‌شوند. در اینجا چون جستجوی کلمه کلیدی، موجودیت‌های شامل کلمه کلیدی را باز می‌گرداند، بنابراین موجودیت‌های بازیابی شده همگی مرتبط هستند و رتبه‌بندی مستقل از پرسش نداریم. در واقع بحث نزدیکی، بیشتر در جستجوهای آنتولوژیکی مطرح می‌شود که از حوصله این پایان نامه خارج است.
هدف اصلی این پایان نامه مرتب سازی مجموعه داده‌های وب‌داده‌ها به نحوی است که مجموعه‌داده‌های جعلی و مجموعه‌داده‌های بدون کیفیت اسپم که با هدف گمراه سازی الگوریتم رتبه‌بندی درصدد گرفتن رتبه‌های بالای نتایج جستجو هستند، نتوانند به هدف خود برسند و موتور جستجو نیز زمان کاربران خود را تلف نکند. آنچه مسلم است، درک تکنیکهای تولیدکنندگان اسپم، برای ارزیابی قدرت و ضعف یک الگوریتم رتبه‌بندی ضروری است.

3-2-1- الگوریتم کشف ارتباط گروهی در وب‌داده‌ها

اعضای ارتباط گروهی با توجه به روابط گسترده فیمابین به راحتی قابل کشف هستند. برای کشف اعضای ارتباط گروهی در وب داده‌ها از شبه کد شکل ۷ استفاده می‌شود که با الهام از مفهوم چهارگانه در وب داده‌ها، چگونگی کشف اسپم با توجه به مفاهیم "استفاده کردن" و نوع لینک را نشان می‌دهد.

```
array Domains
Array detectLinkFarm( array Domain)
{
    Array IsLinkfarmMember;
    foreach d in Domain
    {
        beforuse(d)=find the contexts which their subject's domain or their object's domain is d.
        afteruse(d)= find all the domains which are used with context d.
        if(intersection(beforuse , afteruse)>threshold)  mark d as a linkfarm member in  IsLinkfarmMember.
    }
    foreach d in domains
    {
        sum=0;
        foreach A in afteruse(d)
            if(A was marked as a linkfarm member in  IsLinkfarmMember) sum++;
        if (sum>threshold)
            mark d as a linkfarm member in  IsLinkfarmMember.
    }
    return IsLinkfarmMember;
}
```

شکل ۷: شبه کد بکار رفته برای کشف اعضای ارتباط گروهی

3-2-2- الگوریتم رتبه‌بندی پیشنهادی

در وب داده‌ها سه‌گانه‌ها جهت ندارند یعنی با یک گراف جهت‌دار نمی‌توان این سه‌گانه‌ها را مدل کرد. بنابراین باید با استفاده از چهارگانه‌ها و مفهوم "استفاده کردن"، یک گراف جهت‌دار و وزن‌دار آماده کرد و سپس از الگوریتم weighted pageRank برای تعیین رتبه هر دامنه استفاده کرد. در هر چهارگانه S P O C سه لینک ضمنی وجود دارد. دو تا در بر دارنده مفهوم "استفاده کردن" بین S,C وO,C هستند و قدرت آن‌ها ناشی از تعداد دفعات استفاده است. سومی بین S,O، در بردارنده مفهوم لینک است و وزن آن ناشی از فاکتورهای نوع لینک(P) و اصالت

لینک(C) است. بدین منظور برای دو لینکی که شامل مفهوم استفاده کردن هستند از گراف صلاحیت نام‌گذاری که اعضای آن فرکانس استفاده هستند و بر طبق معادله (۱۳) مقداردهی می‌شود، استفاده می‌کنیم. برای لینک سوم از الگوریتم رتبه‌بندی DING با اعمال تغییراتی روی آن استفاده می‌شود. این تغییرات دربردارنده اضافه کردن مفهوم اصالت به تابع وزن‌دهی است. چون وزن یک نوع لینک خاص بین دو مجموعه داده به میزان زیادی به اصالت لینک وابسته است، بنابراین اصالت تمام این ارتباطات طبق فرمول تغییر یافته زیر باید لحاظ گردد.

$$w_{\sigma,i,j} = \sum_{\forall c \in C_{L_{\sigma,i,j}}} W(c) \times LF\left(L_{\sigma,i,j}\right) \times IDF(\sigma) \qquad (۱۲)$$

$$= \sum_{\forall c \in C_{L_{\sigma,i,j}}} W(c) \times \frac{\left|L_{\sigma,i,j}\right|}{\sum_{L_{\tau,i,k}} \left|L_{\tau,i,k}\right|} \times \log \frac{\mathrm{N}}{1 + freq(\sigma)}$$

که در آن $C_{L_{\sigma,i,j}}$ مجموعه اصالت لینک‌های $L_{\sigma,i,j}$ است و W(c) وزن اصالت c است که قبلاً، توسط گراف صلاحیت نام‌گذاری، محاسبه شده است.

هر چند نویسنده معتقد است اگر انتساب وزن‌های هر نوع لینک خاص، توسط طراح آنتولوژی صورت گیرد، همانند کاری که در[Par11] صورت گرفته است، خیلی بهتر و قابل اعتمادتر است. اما اعمال این فرآیند برای کل وب فرآیندی وقت گیر و پرهزینه است، ناهمگونی‌های زیادی را بوجود می‌آورد، تا حد زیادی غیرممکن است و نیاز به فراهم آوردن زیر ساختهای زیادی دارد. زیرا بعضی از لینک‌ها مثلاً SameAs که فرکانس زیادی دارند، بار معنایی و اعتماد زیادی را منتقل می‌کنند. اما این الگوریتم وزن کمی به آن انتساب می‌دهد چون فرکانس آن زیاد است. همان طور که می‌دانیم pageRank توسط ارتباطات

گروهی گمراه می‌شود. بنابراین روی گراف اولیه و قبل از ایجاد گراف صلاحیت نام‌گذاری، ابتدا ارتباطات گروهی با توجه به الگوریتم بخش ۱-۲-۳- کشف می‌شود . نحوه برخورد با ارتباطات گروهی به سه طریق ممکن است:

1. حذف تمامی نودها (دامنه‌ها) شرکت کننده در ارتباطات گروهی
2. حذف ارتباطات مابین نودهای ارتباطات گروهی و عدم حفظ نودها برای مشاهده رتبه نودهای شرکت کننده در ارتباطات گروهی
3. تقلیل وزن ارتباطات مابین ارتباطات گروهی و حفظ نودها برای مشاهده رتبه نودهای شرکت کننده در ارتباطات گروهی

همه مجموعه داده‌های موجود با یک رتبه اولیه مقداردهی می‌شوند. سپس weighted pagerank روی گراف وزندار حاصله اعمال می‌شود تا رتبه نهایی نودها (حوزه‌ها)، با کاهش وزن ارتباطات اعضای ارتباط گروهی، مشخص گردد.

در گراف وزن‌دار حاصل، حذف نودهای مظنون به اسپم و یا حذف یالهای بین این نوع نودها، یک جریمه کاملاً صریح است و ریسک حذف نودهای غیر اسپم که به اشتباه اسپم معرفی شده‌اند، وجود دارد. بنابراین کمی انعطاف پذیرتر با این‌گونه نودها برخورد می‌شود . بدین می‌توان ترتیب وزن سه نوع لینک را کاهش دهیم:

1. لینک‌هایی که یک مجموعه داده مظنون از موجودیت‌های مجموعه داده مظنون دیگر "استفاده" کرده است، تا بصورت متقابل همدیگر را تقویت کنند.
2. لینک‌هایی که یک مجموعه داده خوب از یک مجموعه داده بد (به اشتباه) استفاده کرده است. البته این نوع لینک‌ها

بعضاً نجات دهنده هستند زیرا گاهی اوقات الگوریتم تشخیص ارتباطات گروهی در شناسایی اعضای ارتباطات گروهی دچار مشکل می‌شود.

3. لینک‌هایی که یک مجموعه داده بد از یک مجموعه داده خوب استفاده کرده است. کیفیت اطلاعات مجموعه داده خوب نباید باعث تغییر زیادی در کیفیت مجموعه داده بد شود.

با این وجود میزان تقلیل وزن هر کدام از سه نوع فوق را می‌توان جداگانه تعیین کرد. مثلاً تقلیل وزن لینک‌هایی از نوع ۱ باید بیشتر از سایر موارد باشد. ماتریس صلاحیت نام‌گذاری، با توجه به اعضای ارتباط گروهی بصورت زیر ایجاد می‌شود:

$$a_{i,j} = \begin{cases} \text{useFrequncy}(i,j) * C & \text{if } i \textit{ and } j \textit{ are GOOD} \\ \text{useFrequency}(i,j) * c & \text{if i is GOOD and j is BAD} \\ 1 & \text{if i is BAD and j is GOO} \\ 1 & \text{if i and j are bad} \end{cases} \quad (۱۳)$$

این فرآیند ربطی به کلمه کلیدی ندارد بنابراین ضرورتی برای اعمال آن در مرحله "رتبه‌بندی وابسته به پرسش" نیست. در این پایان نامه روش سوم برای برخورد با ارتباطات گروهی پیاده‌سازی شده است. مقداردهی دو ثابت c,C توسط متخصص دامنه و با توجه به تجربه باید صورت گیرد. در کارهای آینده این موضوع را بیشتر بررسی خواهیم کرد.

با این وجود موتور جستجو می‌تواند الگوریتم کشف ارتباطات گروهی را در دو مرحله زیر انجام دهد:

مرحله اول: بعد از مرحله پیمایش و قبل از شاخص‌گذاری روی گراف دامنه‌ها (حوزه‌ها) می‌توان ارتباطات گروهی را کشف کرد. دقت شود که آستانه تعیین ارتباطات گروهی در این مرحله بزرگتر است. ارتباطات بین اعضای ارتباط گروهی با درجه اهمیت کمتری لحاظ می‌شوند و رتبه‌بندی روی بقیه گراف صورت می‌گیرد.

مرحله دوم: در گراف موجودیت‌های مشابه (محدوده SameAs) مجموعه جواب تنها موجودیت‌هایی هستند که شامل کلمه کلیدی جستجو یا مشابه آن هستند. کشف ارتباط گروهی در این مرحله مزایای زیر را دارد. اولین مزیت آن اینست که کشف ارتباطات گروهی در زمان انتظار کاربر نیست و زمان انتظار افزایش نمی‌یابد. دومین مزیت آن اینست که "ارتباطات گروهی وابسته به زمینه خاص" کشف می‌شوند و در ضمن آستانه تعیین ارتباط گروهی در این مرحله کمتر است. برای مثال بعضی تولیدکنندگان اسپم برای یک کلمه کلیدی خاص یا زمینه خاصی ارتباط گروهی ایجاد می‌کنند تا ضمن حفظ وابستگی صفحات، موتور جستجو را گمراه کنند. این نوع "ارتباطات گروهی وابسته به زمینه" در مرحله اول یا در سایر زمینه‌ها قابل کشف نیستند. همچنین می‌توان آستانه تعیین ارتباط گروهی را بر اساس تعداد مجموعه داده‌های موجود در حوزه تعیین کرد و نه صرفاً یک عدد صریح به عنوان آستانه ذکر شود، زیرا هر چه تعداد مجموعه داده‌ها بیشتر باشد آستانه بالاتر می‌رود. در این آزمایش کشف ارتباط گروهی در مرحله قبل از شاخص‌گذاری انجام می‌شود و به دلیل کوچک بودن سایز مجموعه داده جمع‌آوری شده نسبت به کل وب، آستانه را صریحاً تعیین کرده‌ایم و مقدار آن نیز ۳ در نظر گرفته شده است. در مرحله یکپارچه‌سازی به جای استفاده از رتبه گوگل هر مجموعه داده، از رتبه‌ای که در این مرحله بدست آمده است استفاده می‌کنیم تا بتوانیم اثر ارتباطات گروهی ایجاد شده برای گمراه کردن الگوریتم تحلیل لینک موتور جستجو را حذف کنیم.

بنابراین مراحل اجرای الگوریتم رتبه‌بندی پیشنهادی به قرار زیر است:

1. کشف ارتباطات گروهی و برخورد با آن‌ها روی داده‌های خام اولیه
2. محاسبه رتبه فقط با در نظر گرفتن دو نوع لینک با مفهوم "استفاده کردن" و اعمال الگوریتم pageRank وزندار روی ماتریس تغییر یافته "صلاحیت نام‌گذاری"
3. وزن‌دار کردن گراف با استفاده از نتایج مرحله قبل و معادله (۱۲) و محاسبه رتبه با در نظر گرفتن نوع لینک
4. ذخیره رتبه و چهارگانه‌ها روی شاخص
5. استخراج روابط SameAs
6. تشکیل گراف هر موجودیت با توجه به روابط استخراج شده مرحله قبل و وزن‌دار کردن گراف با استفاده از اصالت
7. رتبه‌بندی محلی تمام شناسه‌های متفاوت یک موجودیت و ذخیره رتبه‌های محلی روی شاخص

در بررسی‌های انجام شده بر روی مجموعه داده BTC ، ۷۸۱ دامنه شناسایی شده است. دامنه‌های ورودی و خروجی به هر دامنه شناسایی شده‌اند. با مشاهده تعداد اعضای مشترک ورودی و خروجی برای هر دامنه مجموعه داده غیرآلوده، حداکثر تعداد اعضای مشترک ۳ است به جز مجموعه داده خاص DBPedia که تعداد اعضای مشترک وروردی و خروجی ۷۳ است. در نتیجه تعداد زیادی لینک دوطرفه در DBPedia وجود دارد که منجر به بالا رفتن رتبه این مجموعه داده می‌شود و در نتایج جستجو نیز این سایت در بالاترین رتبه قرار دارد. بنابراین این نوع لینک‌های دوطرفه باید جریمه شوند زیرا یک نوع ارتباط گروهی هستند. یک روش برخورد با این نوع اسپم، کاهش وزن لینک‌های ورودی و خروجی دوطرفه است (مثلاً کاهش وزن به نسبت معکوس تعداد اعضای اشتراک است). یا اینکه این نوع لینک‌ها حذف گردند که ما روش اول را برگزیده‌ایم.

مطالعه مشابهی روی مجموعه داده داروها که در مرحله پیمایش جمع‌آوری شده (مجموعه داده آلوده نشده)، انجام شده است که آستانه ۳ را برای تشخیص ارتباط گروهی تأیید می‌کند.
همانطورکه مشاهدات انجام شده نشان می‌دهند، تعداد اعضای اشتراک در مجموعه داده‌های نرمال و آلوده نشده حداکثر ۳ است. بنابراین ما نیز آستانه تعیین ارتباطات گروهی را در الگوریتم پیشنهادی ۳ در نظر می‌گیریم. در نتیجه هر نوع ارتباط گروهی که با هدف گمراه سازی موتور جستجو با اشتراک بیشتر از ۳ تشکیل شود به سادگی به عنوان ارتباط گروهی شناسایی و جریمه می‌گردد.
سناریوی جستجو به قرار زیر است. در مرحله قبل از شاخص‌گذاری، رتبه کیفیتی مجموعه داده‌ها و موجودیت‌ها مشخص می‌شود. بدین منظور ابتدا گراف صلاحیت نام‌گذاری دامنه‌های موجود با الگوریتم تغییر یافته رتبه‌بندی می‌شود. رتبه مجموعه داده‌ها فقط با لحاظ کردن لینک‌های "استفاده" بدست می‌آید. سپس برای لحاظ کردن مفهوم لینک‌ها یا نوع لینک (مفهوم ضمنی سوم هر چهارگانه) برای رتبه‌بندی، گراف مجموعه داده را با معادله (۱۲) وزن‌دار می‌کنیم. و با اعمال مجدد Weighted PageRank، رتبه نهایی هر مجموعه داده تعیین می‌شود. برای بدست آوردن رتبه سراسری هر موجودیت برابر رتبه موجودیت در گراف موجودیت‌های مشابه آن، ضربدر مجموع رتبه مجموعه داده‌هایی که از آن استفاده کرده‌اند، طبق معادله (۱۴)، می‌باشد.
برای تعیین رتبه محلی هر موجودیت در گراف موجودیت‌های مشابه آن، گراف روابط SameAs را ایجاد می‌کنیم که وزن لینک‌ها از روی اصالت رابطه SameAs مشخص می‌شود. با اعمال pageRank روی این گراف رتبه محلی هر موجودیت بین موجودیت‌های

مشابه مشخص می‌شود. رتبه نهایی هر موجودیت نیز بصورت زیر محاسبه می‌شود:

رتبه هر موجودیت برابر حاصلضرب رتبه موجودیت در گراف موجودیت‌های مشابه آن ($r(e)$) و مجموع رتبه مجموعه داده‌هایی است که از آن استفاده کرده‌اند. نتایج به ترتیب رتبه‌بندی به کاربر ارائه می‌شود.

$$entityRank(e) = r(e) \times \sum_{\forall D_i which\ use\ e} r(D_i) \qquad (۱۴)$$

3-3-فاز یکپارچه‌سازی

همانطور که در قبل هم اشاره شد، استاندارد RDF با هدف تسهیل یکپارچه‌سازی داده‌های منابع مختلف ایجاد شده است. این یکپارچه‌سازی منوط به تسهیم و استفاده مجدد URIها برای موجودیت‌های خاص در همه منابع است. اما در واقعیت اسناد RDF متفاوت که توسط افراد مختلفی با واژگان متفاوت منتشر شده‌اند، یک موجودیت را با URIهای مختلف معرفی می‌کنند.

در این حالت اگر روی مجموعه داده خام جمع‌آوری شده از وب، پرسشی را اعمال کنیم، نتایج گوناگونی که به یک منبع ارجاع می‌کنند حاصل می‌شود. حتی گاهی اوقات موجودیت‌های یکسان با نام‌های متفاوت وجود دارند که با ارتباط SameAs روی یکسان بودن آن‌ها تاکید شده است. در این موارد اگر گراف موجودیت‌های مشابه برای موجودیت ایجاد نشود، همه موجودیت‌های یکسان بازیابی نمی‌شوند. بنابراین اگر ابزاری برای شناسایی موجودیت‌های یکسان داشته باشیم، نیازی به اجبار استفاده از یک URI یکسان، برای یک موجودیت خاص، در کل وب نیست. هرچند اعمال چنین سیاستی در کل وب امکان پذیر نیست. بنابراین OWL راه‌حل‌های استانداردی برای چنین مسائلی

ارائه داده است. ویژگی OWL:SameAs با هدف معرفی دو موجودیت یکسان ایجاد شده است. این ویژگی خاصیت تقارنی و تعدی و انعکاسی دارد. منابع زیادی روی وب ارتباطات OWL:SameAs را بین موجودیت‌های محلی و یا بین موجودیت‌های معادل بین مجموعه داده های مختلف استفاده کرده اند[Hog-Zim10] .

owl مکانیزم های دیگری برای کشف OWL:SameAsهای ضمنی ارائه کرده است. مثلاً OWL:InverseFunctionalPropertyکلاسی از ویژگیهایی است که مقادیر آن‌ها بصورت یکتا موجودیت را معرفی می‌کنند. یک نمونه معمول از این ویژگی ISBN است که یک کتاب را بصورت یکتا مشخص می‌کند. اگر دو موجودیت یک مقدار ISBN یکتا داشته باشند، یک رابطه OWL:SameAs بین آن‌ها قابل استنتاج است. علاوه بر این روابط OWL:SameAs از روی ویژگی‌های OWL:FunctionalProperty و OWL:MaxCardinality قابل استنتاج هستند. در اینجا ما فقط روی روابط OWL:SameAs صریحاً ذکر شده در داده ها، برای یکپارچه‌سازی تکیه می‌کنیم.

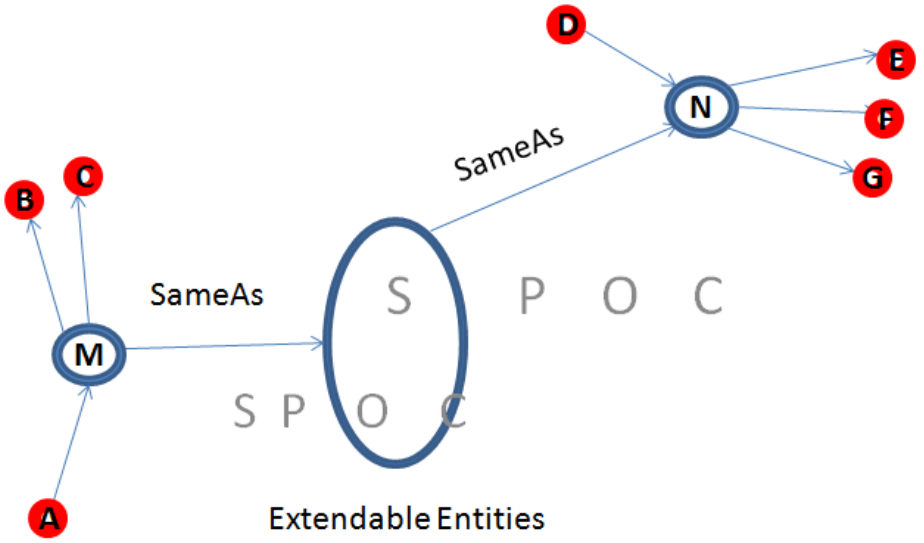
1-3-3- ابزارها و روشها

روش کلی با دو پیمایش روی بدنه کل داده ها انجام می‌شود:

1. ابتدا تمامی عبارات OWL:SameAs از داده ها استخراج می‌شوند (پیمایش اول) و سه‌گانه های SameAs در مکان دیگری ذخیره می‌شوند تا با کاهش حجم آن‌ها بتوان به راحتی روی آن‌ها جستجو انجام داد.
2. برای هر موجودیت، موجودیت‌های متصل به آن با SameAs، شناسایی می‌شوند. بدین منظور، مجموعه اولیه گراف موجودیت‌های متصل شامل موجودیت اولیه است. در هر تکرار، روی داده ها بدنبال موجودیت‌هایی هستیم که با SameAs به این موجودیت وصل شده اند یا این موجودیت با SameAs به آن‌ها وصل شده است. این موجودیت‌ها به مجموعه

اولیه اضافه می‌شوند. این فرآیند تا زمانی تکرار می‌شود که مجموعه اولیه تغییر نکند. در شکل ۸، ابتدا گراف موجودیت‌های مشابه شامل S,O است در تکرار دوم M,N اضافه می‌شوند و در تکرار آخر A,B,C,D,E,F,G به گراف موجودیت‌های مشابه اضافه می‌شوند.

3. برای موجودیت‌های متصل (کلاسهای معادل) یک شناسه کانونی انتخاب می‌شود.
4. بدنه اصلی داده‌ها برای بار دوم پیمایش می‌شود و شناسه‌ها با شناسه کانونی خود جایگزین می‌شوند. شناسه‌های مسند و اشیا رشته‌ای ثابت[1] بازنویسی نمی‌شوند.



شکل ۸ :فاز توسعه SameAs

یک مشکل این روش اینست که موجودیت‌های عضو گراف موجودیت‌های مشابه بعد از یکپارچه‌سازی هیچ تمایزی با همدیگر نخواهند داشت و همگی بعد از تشکیل گراف موجودیت‌های مشابه، تحت عنوان یک شناسه یکتا[2] شناسایی می‌شوند. این در حالی است که URIهای مربوط به یک موجودیت به میزان متفاوتی قابلیت

[1] Literal
[2] canonical Identifier

اعتماد دارند و این صرفاً از روی اصالت قابل تشخیص نیست. به عنوان مثال اصالت صفحه شخصی یک فرد ممکن در کل وب رتبه بالایی نداشته باشد و از رتبه یک سایت عمومی مثل Wikipedia کمتر باشد ولی در شناسه‌های مربوط به آن فرد بالاترین رتبه را داشته باشد. بنابراین رتبه آن موجودیت باید در گراف موجودیت‌های مشابه آن محاسبه شود. بدین منظور در مرحله رتبه‌بندی مستقل از پرسش در گراف موجودیت‌های مشابه هر موجودیت، یک pageRank وزن‌دار روی گراف محلی اعمال می‌شود. در این گراف فاکتور نوع لینک حذف می‌شود و فقط اصالت لینک‌ها تعیین کننده وزن لینک است. نتایج رتبه‌بندی اعضای گراف موجودیت‌های مشابه در یک شاخص جداگانه که شامل شناسه موجودیت و شناسه کانونی آن و رتبه محلی موجودیت می‌باشد، ذخیره می‌شوند. بدین ترتیب در هنگام رتبه‌بندی وابسته به پرسش این نتایج نیازی به محاسبه مجدد ندارند و به سادگی بازیابی می‌شوند.

4-3-استنتاج

ماژول استنتاج وظیفه استنتاج داده‌های جدید از روی داده‌های موجود و قوانین مفهومی تعریف شده برای آنتولوژی‌ها را دارد. این استنتاج باید بتواند میزان اعتماد به داده‌های استنتاج شده را از روی منابع استنتاج محاسبه نماید.

با استفاده از زبان‌های OWL و RDFS می‌توان اطلاعات موجودیت‌ها را با استفاده از اطلاعات ساختاری ترکیب کرد و کلاس‌ها و ویژگی‌ها را توصیف کرد تا این اطلاعات به سادگی برای عامل نرم‌افزاری قابل درک باشد. بدین ترتیب عامل‌های نرم‌افزاری قادر خواهند بود با استفاده از قوانین تعریف شده، عملیات استنتاج را روی داده‌ها انجام دهند.

در اطلاعات داوریی، قانون خاصی برای استنتاج اطلاعات جدید وجود ندارد. مفهوم استنتاج در وب داده‌ها باید با استفاده از انواع قوانین استنتاج روی انواع آنتولوژی‌های موجود، بررسی شود و تمامی سه‌گانه‌ها به داده‌ها اضافه شوند. قابلیت اعتماد به هر سه‌گانه استنتاج شده وابسته به قابلیت اعتماد به عناصر "مقدم" و آنتولوژی و قانون آن است.
در مطالعه موردی مجموعه داده داروها، قانون استنتاج خاصی وجود ندارد و اطلاعات ناشی از استنتاج‌های سایر آنتولوژیها، مفید نیست. بنابراین این فاز در این مطالعه موردی چندان بررسی نشده است.

3-5-شاخص‌گذاری

ساده ترین راه برای جستجوی اسناد، جستجوی پشت سر هم آن‌ها و جستجوی عبارات مورد نظر در محتوای ترتیبی اسناد است. این روش برای تعداد زیادی سند جوابگو نیست. بنابراین نیاز به یک روش سریعتر برای جستجو روی مجموعه بزرگتری از اسناد است. برای جستجوی سریع در مجموعه بزرگی از اسناد، ابتدا باید آن‌ها را به فرمتی تبدیل کنیم که امکان جستجوی سریع روی آن‌ها فراهم باشد. این فرآیند تبدیل، شاخص‌گذاری نامیده می‌شود. شاخص چیزی جز یک ساختار داده خاص که امکان جستجوی سریع روی آن فراهم است، نیست. شاخص در Lucene یک ساختمان داده با فرمت خاص است که در مجموعه‌ای از فایل‌ها ذخیره می‌شود. توضیحات کامل در بخش ۲-۲- ذکر شده است.
برای شاخص‌گذاری از قابلیت‌های موتور جستجوی با کد باز Lucene استفاده شده است. ایجاد شاخص در Lucene با استفاده از فراخوانی تابع indexfiles صورت می‌گیرد باید این تابع به نحوی تغییر کند که چهارتایی‌ها را شاخص‌گذاری کند. بدین منظور ساختار شاخص پیشنهادی دارای ۴ فیلد برای بخشهای فاعل و

مسند و مفعول و زمینه یک چهارگانه است که هر سند شاخص دارای این فیلدهاست. در واقع در Lucene همان المانی که می‌خواهیم هنگام جستجو بازیابی شود، باید بصورت Lucene Document وارد شاخص شود. اینجا با هر پرسش نیاز است تمامی چهارگانه‌های مرتبط با آن بازیابی شوند. بنابراین هر چهارگانه یک Lucene Document را تشکیل می‌دهد که شامل ۴ فیلد مجزا برای هر بخش است. بدین ترتیب قادر خواهیم بود روی همه فیلدها مستقلاً جستجو انجام دهیم. یک شاخص دیگر دامنه‌ها و رتبه محاسبه شده آن‌ها توسط الگوریتم پیشنهادی را ذخیره می‌کند. شاخص بعدی برای ذخیره موجودیت‌های محلی و رتبه آن‌ها در گراف موجودیت‌های مشابه است.

3-6-پردازش پرسش

جستجوی اطلاعات شاخص شده، فرآیند جستجو به دنبال کلمات کلیدی یا رتبه مورد نیاز در شاخص و بازیابی موجودیت‌ها یا سندهایی است که مورد نیازند یا شامل آن کلمات کلیدی هستند. اگر شاخص بصورت توزیع شده ذخیره شده است، فرآیند پردازش پرسش نیز باید بصورت توزیع شده انجام شود و سپس نتایج توزیع شده یکپارچه شوند. دقت کنید که نتایج حاصل از این فاز رتبه‌بندی نشده‌اند و بعد از بازیابی باید از روی رتبه‌های بازیابی شده، رتبه نهایی هر موجودیت را محاسبه کرد و نتایج را به ترتیب به کاربران ارائه کرد.

برای پردازش پرسش نیز از تابع indexSearcher در موتور جستجوی Lucene استفاده کرده‌ایم.

3-7-رتبه‌بندی وابسته به پرسش

در رتبه‌بندی وابسته به پرسش نیز کاملاً شبیه رتبه‌بندی مستقل از پرسش است ولی تفاوت اصلی روی گراف پایه‌ای است که

رتبه‌بندی روی آن اعمال می‌شود. رتبه‌بندی وابسته به پرسش روی گراف نتایج حاصل از پرسش یا گراف توسعه یافته آن(بعد از استنتاج و یکپارچه‌سازی) انجام می‌شود و از نتایج رتبه‌بندی مستقل از پرسش برای رتبه‌بندی استفاده می‌نماید. زمان اجرای رتبه‌بندی وابسته به پرسش باید تا حد ممکن کمینه گردد زیرا در زمان انتظار کاربر است. در روش پیشنهادی رتبه‌بندی وابسته به پرسش (رتبه‌بندی اعضای گراف موجودیت‌های مشابه) در زمان آفلاین انجام شده است. با توجه به اینکه جستجو بر اساس کلمه کلیدی است، تمام نتایج بازیابی شده یا شامل کلمه کلیدی هستند و یا صریحاً شباهت آنها با موجودیتهای شامل کلمه کلیدی ذکر شده است. بنابراین رتبه‌بندی وابسته به پرسش بصورت مفهوم عام آن، یعنی محاسبه نزدیکی با عبارت پرسش، قبلاً انجام شده است و در این فاز تنها بازیابی محاسبات انجام شده شکل می‌گیرد. اما با فراهم شدن سایر انواع پرسش نیز باید این فاز مورد بازبینی قرار گیرد.

فصل 4- پیاده‌سازی و ارزیابی

برای ارزیابی موتور جستجوی پیشنهادی، الگوریتم رتبه‌بندی DING و الگوریتم رتبه‌بندی SWSE با استفاده از ماتریس صلاحیت نام‌گذاری پیاده‌سازی شده‌اند. بنابراین می‌توان نتایج رتبه‌بندی آن‌ها را با نتایج رتبه‌بندی الگوریتم پیشنهادی مقایسه کرد. از آنجا که تزریق اسپم‌های ایجاد شده به شاخص موتورهای جستجوی فعلی صرفاً برای مشاهده نتایج رتبه‌بندی و تست روش جدید، عملی غیراخلاقی و زمانبر است و این موتورهای جستجو نیز کد باز نیستند، بنابراین تنها روش ممکن برای مقایسه، پیاده‌سازی الگوریتم‌ها و مقایسه نتایج است. الگوریتم رتبه‌بندی DING پیاده‌سازی شده است و از اینجا[1] قابل دسترس است. پیاده‌سازی الگوریتم رتبه‌بندی SWSE با استفاده از صلاحیت نام‌گذاری نیز از اینجا[2] قابل دسترس است.

ماتریس وزن در الگوریتم DING با پیچیدگی مکانی n*n*m است که n تعداد دامنه‌های متفاوت و m تعداد انواع متفاوت لینک است. با توجه به اینکه تعداد دامنه‌ها و تعداد لینک‌های متمایز خیلی زیاد است بنابراین حجم این ماتریس خیلی بزرگ خواهد شد ولی باید در حافظه نگهداری شود . خوشبختانه بررسی‌های به عمل آمده حاکی از تنک[3] بودن این ماتریس است و بنابراین آن را در یک آرایه بصورت کلید-مقدار نگهداری می‌کنیم. که کلید، ترکیبی از سه شاخص سطر و ستون و نوع‌لینک است و مقدار، وزن انتسابی است.

---

[1] http://wtlab.um.ac.ir/parameters/wtlab/filemanager/searchEngine/ding.java

[2] http://wtlab.um.ac.ir/parameters/wtlab/filemanager/searchEngine/swse.java

[3] sparse

4-1-سناریوی طراحی ارتباط گروهی

برای هر الگوریتم رتبه‌بندی، روش طراحی ارتباط گروهی خاصی باید اعمال شود. در اینجا روش طراحی اسپم برای گمراه سازی هر کدام از الگوریتمهای رتبه‌بندی مشهور و اسپم‌های طراحی شده معرفی شده اند.

4-1-1- ارتباط گروهی برای گمراه سازی الگوریتم رتبه‌بندی DING

با توجه به فرمول انتساب وزن در معادله (۱۲)، اگر فرکانس رخداد یک مسند کم باشد مخرج کسر دوم کوچکتر و مقدار آن بزرگتر می‌شود. بنابراین اسپم را یکبار با مسندهایی که رخداد آن‌ها کمتر است و یکبار با مسندهای جعلی که اصلاً در مجموعه داده نیستند ایجاد می‌کنیم. در کسر اول صورت جزئی از مخرج است بنابراین هر چه تعداد سایر مسندهای خروجی کمتر باشند کسر بزرگتر خواهد شد. این اصل نیز در تولید اسپم رعایت شده است. اسپم‌های ایجاد شده در مرحله تست مربوطه معرفی و از آدرس ارائه شده قابل دسترسی هستند.

4-1-2- ارتباط گروهی برای گمراه سازی الگوریتم رتبه‌بندی SWSE

با توجه به معادله (۴) اگر مجموعه‌داده‌های اسپم در دفعات زیادی از موجودیت‌های همدیگر استفاده کنند، هر بار استفاده، باعث تقویت رتبه مجموعه داده مورد استفاده می‌شود. برای تولید اسپم پیشنهادی این تکنیک نیز به کار رفته است.

4-2-آزمایش اول

4-2-1- طراحی و تزریق اسپم

در مرحله اول تعدادی اسپم محتوا به مجموعه داده اولیه داروها تزریق می‌شود. بدین منظور دامنه www.fake.org ایجاد شده

است و در این دامنه تعدادی موجودیت جعلی از دامنه ww.fake.org اعلام کرده‌اند که مشابه یک موجودیت مشهور هستند و یا اینکه برای موجودیت‌های مشهور برچسب‌ها و ویژگی‌های جعلی منتشر کرده‌اند. این فایل آلوده از اینجا[1] قابل دسترس است. نتیجه رتبه‌بندی الگوریتم DING در این مرحله دقیقاً انجام شده است و نتایج از جدول ۱ قابل مشاهده است. سپس همین فایل آلوده به الگوریتم رتبه‌بندی SWSE ارائه شده است و نتایج رتبه‌بندی الگوریتم SWSE نیز از جدول ۲ قابل مشاهده است.

[1] http://wtlab.um.ac.ir/parameters/wtlab/filemanager/searchEngine/smalldrugfordingtest.nq

جدول۱ : رتبه‌بندی DING بعد از تزریق اسپم محتوا

Domain	Rank
bio2rdf.org	0.442505
129.128.185.122	0.251544
dbpedia.org	0.176824
www.drugbank.ca	0.045172
en.wikipedia.org	0.03535
www.rxlist.com	0.029367
www.w3.org	0.008019
www4.wiwiss.fu-berlin.de	0.006392
www.uniprot.org	0.004825
fake.org	1.05E-06

جدول ۲- رتبه‌بندی SWSE بعد از تزریق اسپم محتوا

Domain	Rank
www.w3.org	0.166664
www4.wiwiss.fu-berlin.de	0.026189
dbpedia.org	0.01611
www.uniprot.org	0.01611
bio2rdf.org	0.01611
129.128.185.122	0.01611
www.drugbank.ca	0.01611
en.wikipedia.org	0.01611
www.rxlist.com	0.01611
fake.org	0.002372

نتایج رتبه‌بندی دامنه‌ها در الگوریتم رتبه‌بندی پیشنهادی نیز بصورت جدول ۳ است:

جدول ۳- رتبه‌بندی روش پیشنهادی بعد از تزریق اسپم محتوا

Domain	Rank
bio2rdf.org	777.7737
129.128.185.122	444.0772
dbpedia.org	313.927
www.drugbank.ca	82.33484
en.wikipedia.org	64.83432
www.rxlist.com	53.54001
www.w3.org	14.86574
www4.wiwiss.fu-berlin.de	11.09228
www.uniprot.org	9.309607
fake.org	0.014838

2-2-4- نتیجه گیری آزمایش اول

طبق بررسی‌های به عمل آمده و نتایج جدول‌های ۱و۲و۳، نتیجه می‌گیریم که هر سه الگوریتم به اسپم محتوا حساس هستند و آن را به درستی تشخیص می‌دهند.

4-3-آزمایش دوم

4-3-1- طراحی و تزریق اسپم

آزمایش دوم تست اسپم لینک یا ارتباطات گروهی است. بدین منظور یک ارتباط گروهی با اندازه ۴ ایجاد می‌کنیم که از اینجا[1] قابل دسترسی است. رتبه‌بندی DING در این مورد نیز قابل بررسی است. در رتبه‌بندی اولیه نتایج DING در جدول ۴ آمده است. همانطور که مشهود است دامنه www.fake.org که یکی از دامنه‌های ارتباط گروهی است رتبه بالاتری از دو دامنه غیراسپم www4.wiwiss.fu-berlin.de و www.uniprot.org دارد. اما در دور دوم تنظیم رتبه‌بندیهای DING، این اشتباه توسط DING اصلاح می‌شود و نتایج بعد از دور دوم، بصورت جدول ۵ همگرا شده است. نتایج رتبه‌بندی SWSE از مجموعه داده آلوده در جدول ۶ آمده است. همانطور که انتظار می‌رفت، الگوریتم رتبه‌بندی SWSE توسط پیوستگی ارتباطات گروهی تزریق شده، گمراه شده است و دامنه‌های جعلی بالاترین رتبه‌ها را اخذ نموده‌اند. در انتها فایل آلوده برای رتبه‌بندی به الگوریتم پیشنهادی ارائه شده است و نتایج رتبه‌بندی بصورت جدول ۷ است.

4-3-2- نتیجه گیری آزمایش دوم

با توجه به مشاهدات انجام شده، الگوریتم رتبه‌بندی DING در مقابل ارتباطات گروهی کوچک تزریقی مقاوم بوده است و آن‌ها را به درستی تشخیص داده‌است ولی تعداد تکرارهای لازم برای همگرایی الگوریتمDING، با افزایش سایز ارتباط گروهی افزایش می‌یابد. اما الگوریتم SWSE، همانطور که انتظار می‌رفت هیچ

[1] http://wtlab.um.ac.ir/parameters/wtlab/filemanager/searchEngine/linkfarmspam4.txt

گونه مکانیزم پیشگیرانه برای مقابله با اسپم گروهی ندارد و به سادگی با افزایش تعداد لینک‌های ورودی گمراه می‌شود.

جدول ۴: رتبه‌بندی DING بعد از تزریق سلسله پیم لینک با اندازه ۴ تکرار اول

Domian	Rank
bio2rdf.org	2101.379
129.128.185.122	1194.286
dbpedia.org	839.3002
www.drugbank.ca	214.1324
en.wikipedia.org	167.5204
www.rxlist.com	139.2081
www.w3.org	38.48582
www.faken.org	36.6431
www.fakew.org	34.2846
www4.wiwiss.fu-berlin.de	30.37194
www.uniprot.org	22.80327
www.fakez.org	10.55358
www.fakey.org	3.248632
www.fakex.org	3.248632
fake.org	0.003301

جدول ۵: رتبه‌بندی DING بعد از تزریق سلسله پیم لینک با اندازه ۴ تکرار دوم

Domain	Rank
bio2rdf.org	2098.214
129.128.185.122	1192.487
dbpedia.org	828.412
www.drugbank.ca	213.8099
en.wikipedia.org	167.2681
www.rxlist.com	138.9984
www.w3.org	38.43269
www4.wiwiss.fu-berlin.de	30.34906
www.uniprot.org	22.76892
www.fakex.org	0.751035
www.faken.org	0.741516
www.fakew.org	0.364824
www.fakez.org	0.128398
www.fakey.org	0.012898
fake.org	0.00323
www.w3.org	0.000876

جدول ۶: رتبه‌بندی SWSE بعد از تزریق سلسله پیم لینک با اندازه ۴

Domain	Rank
fake.org	0.297017
www.fakex.org	0.115944
www.faken.org	0.093344
www.fakey.org	0.088278
www.fakew.org	0.068561
www.fakez.org	0.062708
www4.wiwiss.fu-berlin.de	0.041312
www.rxlist.com	0.038824
en.wikipedia.org	0.032412
www.drugbank.ca	0.028589
129.128.185.122	0.012213
bio2rdf.org	0.010067
www.uniprot.org	0.009621
dbpedia.org	0.008556

جدول ۷: رتبه‌بندی روش پیشنهادی بعد از تزریق سلسله پیم لینک با اندازه ۴

Domain	Rank
bio2rdf.org	534.4987
129.128.185.122	305.1813
dbpedia.org	213.3416
www.drugbank.ca	56.59444
en.wikipedia.org	44.56518
www.rxlist.com	36.79457
www.w3.org	10.21828
www4.wiwiss.fu-berlin.de	7.625585
www.uniprot.org	6.397596
www.faken.org	0.237435
www.fakex.org	0.147879
www.fakew.org	0.053366

www.fakez.org	0.028163
www.fakey.org	0.021987

fake.org	0.010184

4-4-آزمایش سوم

در این آزمایش اسپم تزریقی را با هدف گمراه سازی DING ، طبق ۱-۱-۴- طراحی کرده ایم.

4-4-1- آزمایش سوم مرحله اول

در مرحله اول از **یک** مسند جدید برای همه اعضای ارتباط گروهی استفاده کرده ایم که فایل اسپم ایجاد شده از اینجا[۱] قابل دانلود است. نتایج رتبه‌بندی DING در تکرارهای آن، در جدولهای ۸و۹و۱۰و۱۱ قابل مشاهده است.

همانطور که مشاهده شده است، الگوریتم رتبه‌بندیDING، ارتباط گروهی شامل یک نوع لینک جدید و با تعداد اعضای ۵۰ را شناسایی کرده است.

4-4-2- آزمایش سوم مرحله دوم

یکبار دیگر از **چندین** مسند جدید در ارتباط گروهی استفاده کرده‌ایم که فایل اسپم ایجاد شده از اینجا[۲] قابل دانلود است. و نتایج رتبه‌بندی DING نیز در جداول ۱۲و۱۳و۱۴و۱۵ آمده است و حاکی از آنست که این نوع اسپم توانسته است DING را گمراه سازد. نتایج رتبه‌بندی SWSE نیز توسط ارتباط گروهی با چندین مسند و با یک مسند جدید، گمراه می‌شود. نتایج رتبه‌بندی SWSE در جدول ۱۶ آمده است. در انتها رتبه‌بندی الگوریتم پیشنهادی از ارتباط گروهی با چند مسند جدید به صورت جدول ۱۷ است.

---

[۱] http://wtlab.um.ac.ir/parameters/wtlab/filemanager/searchEngine/RandomlinkFarm50.nq

[۲] http://wtlab.um.ac.ir/parameters/wtlab/filemanager/searchEngine/RandomlinkFarm50diffpred.nq

3-4-4- نتیجه گیری آزمایش سوم

DING روی ارتباطات گروهی با سایز کوچک و یک مسند جدید، بعد از چندین تکرار موفق عمل کرده است. اما DING برای همین آزمایش با چندین مسند، گمراه شده است. بنابراین در آزمایش بعدی تعداد مسند جدید بیشتر و مجموعه داده با سایز بزرگتر را بررسی کرده ایم.

جدول۸: رتبه‌بندی DING بعد از تزریق اسپم لینک با یک مسند جدید تکرار اول

Domain	Rank
bio2rdf.org	121524.1
129.128.185.122	69066.21
dbpedia.org	48537.12
www.drugbank.ca	12383.27
en.wikipedia.org	9687.679
www.rxlist.com	8050.4
www.w3.org	2194.788
www4.wiwiss.fu-berlin.de	1756.436
www.uniprot.org	1318.687
www.d2.org	867.4174
www.d3.org	670.7569
www.d4.org	326.8728
www.d0.org	288.3543
www.d1.org	123.1016
fake.org	0.19026

جدول ۹ : رتبه‌بندی DING بعد از تزریق اسپم لینک با یک مسند جدید تکرار دوم

Domain	Rank
bio2rdf.org	4002.099
129.128.185.122	2274.528
dbpedia.org	1598.454
www.drugbank.ca	407.8152
en.wikipedia.org	319.0423
www.rxlist.com	265.1218
www.w3.org	72.34873
www4.wiwiss.fu-berlin.de	57.84381
www.d2.org	44.77054
www.uniprot.org	43.42842
www.d3.org	34.79096
www.d4.org	17.13124
www.d0.org	14.95693
www.d1.org	6.480504
fake.org	0.006277

جدول۱۰: رتبه‌بندی DING بعد از تزریق اسپم لینک با یک مسند جدید تکرار سوم

Domain	Rank
bio2rdf.org	131.7921
129.128.185.122	74.90356
dbpedia.org	52.64091
www.drugbank.ca	13.43213
en.wikipedia.org	10.50857
www.rxlist.com	8.732276
www.w3.org	2.41E+00
www.d2.org	2.190183
www4.wiwiss.fu-berlin.de	1.90472
www.d3.org	1.7763
www.uniprot.org	1.430828
www.d4.org	0.966926
www.d0.org	0.773475
www.d1.org	0.375042
fake.org	2.18E-04

جدول ۱۱ : رتبه‌بندی DING بعد از تزریق اسپم لینک با یک مسند جدید تکرار چهارم

Domain	Rank
bio2rdf.org	3.69E+06
129.128.185.122	2097198
dbpedia.org	1473832
www.drugbank.ca	376018.4

en.wikipedia.org	294166.7
www.rxlist.com	244450.7
www.w3.org	66640.34
www4.wiwiss.fu-berlin.de	53334.26
www.uniprot.org	4.00E+04
www.d2.org	16714.27

جدول۱۲: رتبه‌بندی **DING** بعد از تزریق اسپم لینک با چند مسند جدید تکرار اول

Domain	Rank
bio2rdf.org	131.7921
129.128.185.122	74.90356
dbpedia.org	52.64091
www.drugbank.ca	13.43213
en.wikipedia.org	10.50857
www.rxlist.com	8.732276
www.d3.org	6.13E+00
www.d4.org	3.576905
www.d0.org	3.329958
www.w3.org	2.414197
www4.wiwiss.fu-berlin.de	1.90472
www.uniprot.org	1.430828
www.d2.org	1.170137
www.d1.org	0.453857
fake.org	2.18E-04

جدول۱۳: رتبه‌بندی **DING** بعد از تزریق اسپم لینک با چند مسند جدید تکرار پنجم

Domain	Rank
bio2rdf.org	1.12E+08
129.128.185.122	6.37E+07
dbpedia.org	4.48E+07
www.d3.org	2.46E+07
www.d4.org	1.42E+07
www.d0.org	1.23E+07
www.drugbank.ca	1.14E+07
en.wikipedia.org	8932382
www.rxlist.com	7422754
www.d2.org	4530339
www.w3.org	2023527
www4.wiwiss.fu-berlin.de	1619497

www.d3.org	12918.41
www.d4.org	6289.253
www.d0.org	5553.691
www.d1.org	2367.403
fake.org	5.776906

www.d1.org	1491325
www.uniprot.org	1215876
fake.org	175.4156

جدول ۱٤: رتبه‌بندی **DING** بعد از تزریق اسپم لینک با چند مسند جدید تکرار دهم

Domain	Rank
www.d3.org	9.60E+13
bio2rdf.org	9.53E+13
www.d4.org	5.56E+13
129.128.185.122	5.41E+13
www.d0.org	4.79E+13
dbpedia.org	3.80E+13
www.d2.org	1.77E+13
www.drugbank.ca	9.71E+12
en.wikipedia.org	7.59E+12
www.rxlist.com	6.31E+12
www.d1.org	5.83E+12
www.w3.org	1.72E+12
www4.wiwiss.fu-berlin.de	1.38E+12
www.uniprot.org	1.03E+12
fake.org	1.49E+08

جدول ۱۵: رتبه‌بندی **DING** بعد از تزریق اسپم لینک با چند مسند جدید تکرار بیستم

Domain	Rank
www.d3.org	1.47E+27
www.d4.org	8.49E+26
www.d0.org	7.32E+26
www.d2.org	2.70E+26
www.d1.org	8.90E+25
bio2rdf.org	6.88E+25
129.128.185.122	3.91E+25
dbpedia.org	2.75E+25
www.drugbank.ca	7.02E+24

en.wikipedia.org	5.49E+24
www.rxlist.com	4.56E+24
www.w3.org	1.24E+24

جدول ۶: رتبه‌بندی SWSE بعد از تزریق ۵ پم لینک با اندازه ۵۰ و مسند جدید

Domain	Rank
www.w3.org	0.114581
www.d4.org	0.062247
www.d0.org	0.062247
www.d2.org	0.062247
www.d3.org	0.062247
www.d1.org	0.0514
www4.wiwiss.fu-berlin.de	0.018005
fake.org	0.016304
dbpedia.org	0.011075
www.uniprot.org	0.011075
bio2rdf.org	0.011075
129.128.185.122	0.011075
www.drugbank.ca	0.011075
en.wikipedia.org	0.011075
www.rxlist.com	0.011075

www4.wiwiss.fu-berlin.de	9.95E+23
www.uniprot.org	7.47E+23
fake.org	1.08E+20

جدول ۷: رتبه‌بندی روش پیشنهادی بعد از تزریق ۵ پم لینک با اندازه ۵۰ و مسند جدید

Domain	Rank
bio2rdf.org	400.7644
129.128.185.122	305.6677
dbpedia.org	216.1277
www.drugbank.ca	56.69572
en.wikipedia.org	44.64441
www.rxlist.com	36.85125
www.w3.org	10.23479
www4.wiwiss.fu-berlin.de	7.628382
www.uniprot.org	6.406328
www.d3.org	5.753264
www.d4.org	4.503335
www.d0.org	4.215745
www.d2.org	1.462207
www.d1.org	0.557099
fake.org	0.010202

5-4-آزمایش چهارم

آزمایش چهارم ایجاد یک ارتباط گروهی بزرگتر و تصادفی تر است. بدین منظور تولید کننده اسپم طراحی شده است که تعدادی دلخواه دامنه اسپم ایجاد می‌کند و سپس به تعداد دلخواهی سه‌گانه تصادفی بین این حوزه‌ها ارتباطات متعددی(مسندهای جدید و مسندهای موجود) ایجاد می‌کنند. نوع لینک برقرار کننده‌ی ارتباطات نیز، قابل تعیین است.

1-5-4- آزمایش چهارم مرحله اول

در مرحله اول ۲۰۰ سه‌گانه از ۱۰ حوزه اسپم با استفاده از لینک‌های از انواع مختلف و بصورت تصادفی ایجاد کرده‌ایم.

فایل اسپم ایجاد شده از اینجا[1] قابل دانلود است. این فایل اسپم را در مجموعه داده جمع‌آوری شده تزریق می‌کنیم و مجموعه داده آلوده برای رتبه‌بندی دامنه‌های موجود در آن وارد DING می‌شود.

نتایج رتبه‌بندیهای DING در اینجا قابل تأمل است. رتبه‌بندی نتایج، طی تکرارهای انجام شده در در جدول‌های ۱۸و۱۹و۲۰و۲۱ آمده است.

[1] http://wtlab.um.ac.ir/parameters/wtlab/filemanager/searchEngine/linkspamfull.nq

جدول ۸ : رتبه‌بندی DING بعد از تزریق اسپم لینک با اندازه ۲۰۰تکرار اول

Domain	Rank
bio2rdf.org	2089.498
129.128.185.122	1188.881
dbpedia.org	833.7112
www.drugbank.ca	213.4083
en.wikipedia.org	166.9539
www.rxlist.com	138.5859
www.d6.org	100.1664
www.d1.org	59.86851
www.d2.org	48.08923
www.w3.org	38.37581
www4.wiwiss.fu-berlin.de	30.30836
www.d0.org	29.56271
www.uniprot.org	22.5794
www.d3.org	21.83718
www.d7.org	12.22954
www.d5.org	10.88887
www.d4.org	9.617987
www.d8.org	9.57716
www.d9.org	4.498859
www.fake.org	0.003282

جدول ۱۹ :رتبه‌بندی DING بعد از تزریق اسپم لینک با اندازه ۲۰۰ تکرارپنجم

Domain	Rank
bio2rdf.org	5.34E+10
129.128.185.122	3.04E+10
dbpedia.org	2.13E+10
www.d6.org	5.98E+09
www.drugbank.ca	5.45E+09
en.wikipedia.org	4.26E+09
www.rxlist.com	3.54E+09
www.d1.org	3.50E+09
www.d2.org	2.77E+09
www.d0.org	1.68E+09
www.d3.org	1.29E+09
www.w3.org	9.67E+08
www4.wiwiss.fu-berlin.de	7.74E+08
www.d5.org	7.50E+08
www.d7.org	7.08E+08
www.d8.org	5.86E+08
www.uniprot.org	5.77E+08
www.d4.org	4.05E+08
www.d9.org	2.03E+08
fake.org	83547.45

جدول ۲۰: رتبه‌بندی DING بعد از تزریق ۵ پم لینک با اندازه ۲۰۰ تکرار دهم

Domain	Rank
bio2rdf.org	4.50E+16
129.128.185.122	2.56E+16
dbpedia.org	1.80E+16
www.d6.org	9.82E+15
www.d1.org	5.74E+15
www.drugbank.ca	4.60E+15
www.d2.org	4.55E+15
en.wikipedia.org	3.60E+15
www.rxlist.com	2.98E+15
www.d0.org	2.75E+15
www.d3.org	2.12E+15
www.d5.org	1.23E+15
www.d7.org	1.16E+15
www.d8.org	9.62E+14
www.w3.org	8.15E+14
www.d4.org	6.64E+14
www4.wiwiss.fu-berlin.de	6.53E+14
www.uniprot.org	4.86E+14
www.d9.org	3.33E+14
fake.org	7.04E+10

جدول ۲۱: رتبه‌بندی DING بعد از تزریق ۵ پم لینک با اندازه ۲۰۰ تکرار پانزدهم

Domain	Rank
www.d6.org	2.28E+84
www.d1.org	1.33E+84
www.d2.org	1.06E+84
www.d0.org	6.39E+83
www.d3.org	4.93E+83
www.d5.org	2.86E+83
www.d7.org	2.70E+83
www.d8.org	2.24E+83
www.d4.org	1.54E+83
www.d9.org	7.73E+82
bio2rdf.org	6.88E+81
129.128.185.122	3.91E+81
dbpedia.org	2.75E+81
www.drugbank.ca	7.03E+80
en.wikipedia.org	5.50E+80
www.rxlist.com	4.56E+80
www.w3.org	1.25E+80
www4.wiwiss.fu-berlin.de	9.98E+79
www.uniprot.org	7.43E+79
fake.org	1.08E+76

در الگوریتم رتبه‌بندی پیشنهاد شده در این پایان نامه، ابتدا ارتباط گروهی شناسایی می‌شوند. زیرا هر دامنه جعلی ۹ یا ۱۰ دامنه مشترک در بین دامنه‌های ورودی و خروجی دارد. برای ایجاد ماتریس صلاحیت نام‌گذاری لینک‌های بین اعضای ارتباط گروهی و سایر اعضای ماتریس طبق معادله (۱۳) مقداردهی می‌شوند. سپس بر اساس معادله (۱۲)، لینک‌ها وزن‌دار می‌شوند و سپس الگوریتم پیشنهادی برای محاسبه رتبه اعمال می‌شود. در نهایت رتبه‌بندی الگوریتم پیشنهادی در جدول ۲۲ آورده شده است. نتایج رتبه‌بندی الگوریتم SWSE، روی مجموعه داده آلوده شده با اسپم لینک با تعداد اعضای ۵ و ۵۰

سه‌گانه در جدول ۲۳ آمده است. علت مشابه شدن مقادیر رتبه‌ها در جدول ۲۳ اینست که آرایه صلاحیت فقط استفاده یا عدم استفاده دو حوزه از هم را نشان می‌دهند و تعداد استفاده را بیان نمی‌کنند در حالیکه فرکانس استفاده یکی از فاکتورهای اصلی برای رتبه‌بندی است.

جدول ۲۲: رتبه‌بندی روش پیشنهادی بعد از تزریق لینک با اندازه ۲۰۰

Domain	Rank
bio2rdf.org	31.92569
129.128.185.122	19.21594
dbpedia.org	14.48065
www.drugbank.ca	4.871831
www4.wiwiss.fu-berlin.de	4.487979
en.wikipedia.org	4.032452
www.rxlist.com	3.171212
www.w3.org	1.332953
www.uniprot.org	0.80188
www.d4.org	0.27296
www.d8.org	0.210585
www.d2.org	0.199939
www.d6.org	0.190853
www.d7.org	0.1856
www.d5.org	0.163599
www.d3.org	0.153463
www.d0.org	0.136611
www.d1.org	0.113658
www.d9.org	0.101992
fake.org	0.007167

جدول ۲۳: رتبه‌بندی SWSE بعد از تزریق لینک با اندازه ۲۰۰

Domain	Rank
www.w3.org	0.0873
www.d2.org	0.051435
www.d4.org	0.051435
www.d8.org	0.051435
www.d6.org	0.04712
www.d5.org	0.047016
www.d3.org	0.046009
www.d1.org	0.045234
www.d7.org	0.042673
www.d0.org	0.039809
www.d9.org	0.035567
www4.wiwiss.fu-berlin.de	0.013718
fake.org	0.012422
dbpedia.org	0.008438
www.uniprot.org	0.008438
bio2rdf.org	0.008438
129.128.185.122	0.008438
www.drugbank.ca	0.008438
en.wikipedia.org	0.008438
www.rxlist.com	0.008438

2-5-4- آزمایش چهارم مرحله دوم

نویسندگان Sindice در مقاله [Del10] ادعا کرده‌اند برای پیشگیری از اسپم لینک، ابتدا چهارگانه‌های شبه اسپم (چهارگانه‌هایی که دامنه فاعل و مفعول آن‌ها متفاوت از اصالت است) را حذف می‌کنند. در مرحله دوم آزمایش چهارم، برای تست این موضوع اسپم دیگری، با حذف شبه اسپم‌ها، ایجاد شده است. نتایج

رتبه‌بندی در سه مرحله همگرا شده و اعضای ارتباط گروهی را در بالا قرار داده است. نتایج در جداول ۲۴و ۲۵و ۲۶و ۲۷ حاکی از آنست که این نوع اسپم هم DING راگمراه ساخته است.

جدول ۳: رتبه‌بندی DING بعد از تزریق اسپم لینک با اندازه ۲۰۰ و حذف چهار گانه‌های شبه اسپم درتکرار دوم

Domain	Rank
fake.org	0.00479
www.uniprot.org	33.12724
www4.wiwiss.fu-berlin.de	44.12305
www.w3.org	55.18788
www.d6.org	84.55706
www.rxlist.com	202.2345
www.d2.org	209.1502
en.wikipedia.org	243.365
www.drugbank.ca	311.0807
www.d7.org	545.2404
www.d1.org	735.3786
www.d4.org	768.6864
www.d8.org	1169.515
dbpedia.org	1219.296
129.128.185.122	1735.003
www.d9.org	2130.501
www.d5.org	2918.004
bio2rdf.org	3052.788
www.d3.org	3341.999
www.d0.org	4991.593

جدول ۴: رتبه‌بندی DING بعد از تزریق اسپم لینک با اندازه ۲۰۰ و حذف چهار گانه‌های شبه اسپم درتکرار اول

Domian	Rank
fake.org	1.69E-04
www.d6.org	0.421085
www.d2.org	0.999455
www.uniprot.org	1.091583
www4.wiwiss.fu-berlin.de	1.45286
www.w3.org	1.841632
www.d7.org	2.576187
www.d1.org	3.476869
www.d4.org	3.85981
www.d8.org	5.52927
www.rxlist.com	6.661232
en.wikipedia.org	8.016319
www.d9.org	10.13403
www.drugbank.ca	10.24641
www.d5.org	13.90333
www.d3.org	16.0139
www.d0.org	23.76944
dbpedia.org	40.15426
129.128.185.122	57.13559
bio2rdf.org	100.5288

جدول ۲: رتبه‌بندی DING بعد از تزریق سلپم لینک با اندازه ۲۰۰ و حذف چهل گانه‌های شبه سلپم در تکرار چهارم

Domain	Rank
www.d0.org	2.19E+08
www.d3.org	1.47E+08
www.d5.org	1.28E+08
www.d9.org	9.35E+07
www.d8.org	5.13E+07
www.d4.org	3.37E+07
www.d1.org	3.23E+07
www.d7.org	2.39E+07
www.d2.org	9182851
www.d6.org	3713589
bio2rdf.org	2814786
129.128.185.122	1599737
dbpedia.org	1124235
www.drugbank.ca	286825.8
en.wikipedia.org	224389.5
www.rxlist.com	186466.3
www.w3.org	50833.08
www4.wiwiss.fu-berlin.de	40683.23
www.uniprot.org	30543.9
fake.org	4.406611

جدول ۳: رتبه‌بندی DING بعد از تزریق سلپم لینک با اندازه ۲۰۰ و حذف چهل گانه‌های شبه سلپم در تکرار سوم

Domain	Rank
fake.org	0.145132
www.uniprot.org	1005.891
www4.wiwiss.fu-berlin.de	1339.805
www.w3.org	1674.18
www.rxlist.com	6140.824
en.wikipedia.org	7389.736
www.drugbank.ca	9445.926
www.d6.org	17724.42
dbpedia.org	37023.99
www.d2.org	43828.46
129.128.185.122	52683.52
bio2rdf.org	92698.25
www.d7.org	114226.6
www.d1.org	154068.6
www.d4.org	160975.7
www.d8.org	245083
www.d9.org	446409.5
www.d5.org	611386.7
www.d3.org	700193.7
www.d0.org	1045842

و بعد از این هم الگوریتم همگرا می‌شود. دقت کنید که علت قرار گرفتن fake.org در قعر نتایج اینست که این نوع اسپم، از نوع اسپم محتوا است و لینک ورودی ندارد.

3-5-4- نتیجه‌گیری آزمایش چهارم

مشاهدات نشان می‌دهد با افزایش تعداد اعضای ارتباط گروهی و نوع لینک‌های جدید، الگوریتم تطبیق شده DING که بر پایه الگوریتم pageRank است، گمراه می‌شود. زیرا با افزایش تعداد اعضای ارتباط گروهی تعداد لینک‌های ورودی نیز افزایش می‌یابد. و این برای pageRank گمراه کننده است به همین دلیل است که در مراحل نهایی آزمایش ، سایز مجموعه داده که بزرگتر شده، تعداد لینک‌های ورودی هم بیشتر شده است و

اعضای ارتباط گروهی رتبه‌های بالاتری اخذ کرده‌اند. آنچه مسلم است pageRank نیز در مقابل ارتباط گروهی گمراه می‌شود و به همین دلیل است که سیستم‌های بهینه‌سازی موتور جستجو گاهی اوقات موفق به گمراه سازی گوگل می‌شوند و هزینه‌های جریمه را باید بپردازند.

به نظر نویسنده الگوریتم DING چون توسعه ای بر الگوریتم pageRank است خاصیت‌های آن را نیز به ارث می‌برد. پس بهتر است قبل از اعمال رتبه‌بندی ارتباطات گروهی را شناسایی کرده و تاثیر ارتباطات آن‌ها را حذف کنیم. همانطور که در روش پیشنهادی عمل شده است.

6-4-بررسی صحت پیاده‌سازی الگوریتم DING

بدین منظور نتایج رتبه‌بندی ارائه شده از الگوریتم DING در مقاله [Tou09]روی مجموعه‌داده ارائه شده[1] با نتایج رتبه‌بندی الگوریتم پیاده‌سازی شده، مقایسه شده است. بدین منظور از ضریب رابطه spearman[2]، روی رتبه‌بندی حاصل از دو الگوریتم استفاده شده است. همانطور که می دانیم مقدار این ضریب قدرت رابطه بین دو متغییر، که اینجا رتبه‌بندی‌های دو الگوریتم هستند، استفاده شده است. و مقدار آن بین ۱(یک رابطه مثبت کامل) و ۱- (یک رابطه معکوس کامل) متغییر است.

نتایج رتبه‌بندی DING بر طبق [Tou09] و رتبه‌بندی این دیتاست توسط الگوریتم پیاده‌سازی شده توسط نویسنده، بصورت جدول۲۸ است. ضریب ارتباط spearman بین این دو رتبه‌بندی برابر ۱

---

[1] http://sw.deri.org/2009/02/DING/example-void-collection.ttl.

[2] spearman correlation coefficient test

است که حاکی از یک رابطه مثبت کامل دارد. بنابراین صحت پیاده‌سازی الگوریتم DING تأیید می‌شود.

جدول ۱ : رتبه‌بندی DING بر طبق [Tou09] و نتایج رتبه‌بندی الگوریتم پیشنهادی

Rank	1	2	3	4	5	6
DING [Tou09]	DS4 (0.18)	DS1 (0.14)	DS11 (0.12)	DS13 (0.091)	DS3 (0.081)	DS10 (0.074)
Implemented DING	DS4 (0.31872)	DS1 (0.28094)	DS11 (0.22456)	DS13 (0.10903)	DS3 (0.03487)	DS10 (0.02453)

فصل 5- نتیجه گیری و کارهای آینده

با موفقیت یک ایده‌ی جدید، کاربران زیادی به سمت استفاده از آن می‌رود و با سوء استفاده از این موفقیت می‌توان سود تجاری هنگفتی بدست آورد. تولیدکنندگان اسپم نیز با چنین هدفی بدنبال سوء استفاده از موفقیت "وب داده‌ها" هستند. با ظهور الگوریتم‌های تحلیل لینک، تکنیک اسپم محتوا به تنهایی چندان کارساز نیست. بنابراین تولیدکنندگان اسپم با استفاده از تکنیک اسپم لینک یا ارتباط گروهی به دنبال گمراه‌سازی موتورهای جستجو هستند. ضرورت این موضوع آنجا بارزتر می‌شود که تحلیل داده‌های حاصل از جستجو نیز دیگر با عامل انسانی نیست و نتایج جستجو مستقیماً برای تحلیل وارد برنامه کاربردی می‌شوند. در این پایان نامه یک الگوریتم جدید برای رتبه‌بندی مستقل از پرسش مجموعه‌داده‌های باز پیوندی ارائه شده است. از آنجا که فاز یکپارچه‌سازی و رتبه‌بندی بصورت ابزار با کد باز وجود نداشت این دو فاز جداگانه با روشهای بیان شده، پیاده‌سازی شده‌است. هدف اصلی، رتبه‌بندی مجموعه داده‌ها برای کشف مجموعه‌داده‌های اسپم و جعلی است. بدین منظور همه روابط ضمنی و صریح چهارگانه که امکان انتقال اعتماد از طریق آنها میسر است، بکار گرفته شده‌اند.

متریک‌های محلی اعتماد در موتور جستجو قابل استفاده نیستند، چون برای استفاده از متریک محلی نیاز به ثبت تراکنش‌های کاربر است و بصورت معمول کاربر در موتور جستجو لاگین نمی‌کند و مایل نیست تاریخچه جستجوهای او ثبت شود. بهرحال یکی از کارهای آینده بکارگیری متریک‌های محلی برای موتورهای جستجوی شخصی است.

برای بدست آوردن رتبه محلی موجودیت، رتبه موجودیت در گراف موجودیتهای مشابه آن لحاظ شده است. روشهای موجود، رتبه محلی موجودیت در مجموعه داده محلی آن را بکار برده اند. بنابراین محاسبه اهمیت یک موجودیت از روی شهرت مجموعه داده‌ی منبع آن و شهرت خود موجودیت در بین موجودیتهای مشابه آن انجام می‌شود. بهرحال تست و بررسی جزئیات بیشتر این ایده در کارهای آینده انجام خواهد شد.

همانطور که در نتایج تستها بررسی کردیم، الگوریتم‌های تحلیل لینک مشهور، در برابر اسپم لینک ضعیف عمل می‌کنند. در تست اول تنها اثر اسپم محتوا، با تزریق یک دامنه اسپم محتوا که شامل تعدادی اسپم برای موجودیت‌های مشهور بود، نشان دادیم که هر سه الگوریتم رتبه‌بندی در برابر اسپم محتوا حساس هستند و آن را به درستی تشخیص می‌دهند. در تست دوم اسپم لینک با ۴ دامنه جعلی به مجموعه داده، تزریق شد و نتایج رتبه‌بندی نشان داد که SWSE به سادگی توسط اسپم لینک گمراه می‌گردد ولی DING و روش پیشنهادی، هر دو در مقابل تست دوم مقاوم بودند. مرحله اول تست سوم با هدف گمراه سازی DING و تنها با یک نوع مسند جدید و اسپم محتوای کوچک بررسی شد. DING و روش پیشنهادی هر دو مقاوم بودند. لازم به ذکر است که به تدریج تعداد تکرارهای لازم برای همگرایی DING افزایش می‌یابد. در مرحله دوم از تست سوم بالاخره تولیدکننده اسپم موفق به گمراه سازی DING شد. اسپم لینک شامل چندین مسند بود و سایز مجموعه داده ارتباط گروهی ۵۰ بود . روش پیشنهادی با موفقیت آن را کشف کرده است. در مرحله اول از تست چهارم یک ارتباط گروهی با اندازه ۲۰۰ و چندین مسند به مجموعه داده تزریق کردیم و نتایج حاکی از آنست که الگوریتم رتبه‌بندی DING گمراه

گردیده است. الگوریتم پیشنهادی این تست را هم با موفقیت پشت سر گذاشت. در مرحله دوم از تست چهارم ارتباط گروهی را بعد از حذف شبه اسپم به دیتاست تزریق کردیم، ولی الگوریتم رتبه‌بندی DING مجددا گمراه شده است. الگوریتم پیشنهادی این تست را نیز با موفقیت پشت سر گذاشت و اعضای ارتباط گروهی را در قعر نتایج رتبه‌بندی قرار داد. لازم به ذکر است که SWSE در تمامی تستهای ارتباط گروهی گمراه شده است و نتایج گمراه شدن آن نیز دور از انتظار نبوده است. زیرا استفاده متقابل گسترده در ارتباط گروهی به راحتی ماتریس صلاحیت نام‌گذاری را برای اعضای ارتباط گروهی پر می‌کند و الگوریتم رتبه‌بندی نیز به سادگی گمراه می‌گردد. نتایج تستها بصورت خلاصه در جدول۲۹ آمده است. علامت + نشان‌دهنده موفقیت آمیز بودن اسپم و - نشان‌دهنده موفقیت آمیز بودن الگوریتم رتبه‌بندی و شکست تولید کننده اسپم است.

جدول ۲: مقایسه نهایی سه الگوریتم رتبه‌بندی

	آزمایش۱	آزمایش۲	آزمایش۳ مرحله۱	آزمایش۳ مرحله۲	آزمایش۴مرحله۱	آزمایش۴مرحله۲
SWSE	-	+	+	+	+	+
DING	-	-	-	+	+	+
روش پیشنهادی	-	-	-	-	-	-

در روش پیشنهادی اسپم لینک در مرحله قبل از ایجاد شاخص و با آستانه ثابت کشف می‌شود. یکی از پیشنهادات برای کارهای آینده اضافه شدن کشف اسپم لینک در مرحله بعد از جستجو است. زمانیکه نتایج فقط در یک حوزه خاص هستند و اینجا احتمال کشف اسپم لینک‌های خاص (مربوط به یک حوزه خاص) بیشتر است. یکی دیگر از کارهای آینده اضافه کردن فاز

رتبه‌بندی بر اساس پرسش برای انواع پرسشهای دیگر (علاوه بر پرسش بر اساس کلمه کلیدی، مثلاً پرسشهای آنتولوژیکی و سایر انواع پرسشها) است. برای این منظور باید تابعی برای ارزیابی نزدیکی عبارت پرسش با نتایج بازیابی شده پیدا کرد و رتبه‌بندی نهایی علاوه بر کیفیت، فاکتور نزدیکی عبارت جستجو با نتایج را نیز بررسی نماید. بدین ترتیب در مورد سایر انواع پرسش هم می‌توان نتایج رتبه‌بندی را مقایسه کرد. در این پایان نامه مقاوم سازی الگوریتم رتبه‌بندی مجموعه داده‌ها در مقابل اسپم صورت گرفته است. در آینده قصد داریم، تأثیر نتایج رتبه‌بندی گراف موجودیت‌های مشابه برای رتبه‌بندی موجودیت‌های حاصل از پرسش کلمه کلیدی را نیز بررسی نماییم.

فصل 6- منابع

[Hog-Har11] A. Hogan, A. Harth, J¨. Umbrich, S. Kinsella, A. Polleres and S. Decker, "SEARCHING AND BROWSING LINKED DATA WITH SWSE: THE SEMANTIC WEB SEARCH ENGINE", Journal of Web Semantics: Science, Services and Agents on the World Wide Web 2011. In Press, Corrected Proof.

[Hog11] A. Hogan, "Exploiting RDFS and OWL for Integrating Heterogeneous, Large-Scale, Linked Data Corpora". Ph.D thesis 2011.

[Del10] R. Delbru, N.Toupikov, M. Catasta, G. Tummarello and S. Decker,"Hierarchical Link Analysis for Ranking Web Data",In Proc of 7th Extended Semantic Web Conference, ESWC 2010.

[Par11] H. Park, S. Rho, J. Park,"A Link-Based Ranking Algorithm for Semantic Web Resources: A Class-Oriented Approach Independent of Link Direction ". in IGI Global Journal of Database Management 2011.

[McC10] M. McCandless, E. Hatcher, and O. Gotpodnetic, "Lucene In action, Second Edition", Manning Publications 2010.

[Ore08] E. Oren, R. Delbru,M. Catasta,R. Cyganiak, G. Tummarello, "Sindice.com: A document-oriented lookup index for open linked data", International Journal of Metadata, Semantics and Ontologies,2008

[Din04] L. Ding, T. Finn, et.al. Signorini, "Swoogle: a search and metadata engine for the semantic web", in Proc of the thirteenth ACM international conference on Information and knowledge management, 2004.

[Wu05] B. Wu and B. D. Davison, " Identifying Link Farm Spam Pages", in Proc *Proceedings* of the 14th International World Wide Web Conference, 2005.

[Sig05] A. Signorini, "A Survey of Ranking Algorithms", in Proc IEEE Symposium on Security and Privacy, 2005.

[Art07] D. Artz and Y. Gil. "A Survey of Trust in Computer Science and the Semantic Web",Journal of Web Semantics: Science, Services and Agents on the World Wide Web, 2007.

[Aba07] D.J. Abadi, A. Marcus, S.R. Madden, K.Hollenbach, "Scalable Semantic Web Data Management using vertical partitioning" . In VLDB,2007

[Wei08] C. Weiss, P. Karras, A. Bernstein, "HEXASTORE: Sextuple Indexing for Semantic Web Data Management", VLDB ‘08, August 2430,2008, Auckland, New Zealand.

[Hog-Zim10] A.Hogan, A. Polleres, J. Umbrich and A. Zimmermann, "Some entities are more equal than others:statistical methods to consolidate Linked Data?", In Proceedings of the Workshop on New Forms of Reasoning for the Semantic Web: Scalable & Dynamic (NeFoRS2010), CEUR, 2010.

[Cas10] C. Castillo, D. Donato, L. Becchetti, P. Boldi, M. Santini and S. Vigna., *"Web Spam Detection"*, in *proceedings* of KDD 2010.

[Tum08] G. Tummarello, R. Delbru, and E. Oren, “Sindice.com: Weaving the Open Linked Data”, In Proceedings of the International Semantic Web Conference (ISWC 2008), pp. 552-565..

[Del09] R. Delbru, "SIREn: Entity retrieval system for the web of data", In Proceedings of the 3rd *Symposium on Future* Directions in Information Access (FDIA), 2009.

[Ale01] S. Alexaki, V. Christophides, G. Karvounarakis,D. Plexousakis, and K. Tolle. "The ICS-FORTH RDFSuite: Managing voluminous RDF description bases". In SemWeb, 2001.

[Kim05] Y. Kim, B. Kim, J. Lee, and H. Lim. "The path index for query processing on RDF and RDF Schema". In ICACT, 2005.

[Har05] A. Harth and S. Decker. "Optimized index structures for querying rdf from the web". In LA-WEB, 2005.

[Woo05] D. Wood, P. Gearon, and T. Adams. Kowari: A platform for Semantic Web storage and analysis. In XTech, 2005.

[Aqu07] d’Aquin, M., Baldassarre, C., Gridinoc, L., Sabou, M., Angeletou, S., & Motta, E. "Watson: Supporting next generation Semantic Web applications". In Proceedings of the IADIS International Conference WWW/Internet 2007.

[Pag98] L. Page, S. Brin, R. Motwani, T. Winograd, "The PageRank Citation Ranking: Bringing Order to the Web", Tech. rep., Stanford Digital Library Technologies Project (1998).

[Kle99] J. M. Kleinberg, "Authoritative Sources in a Hyperlinked Environment", Journal of the ACM 46 (5)(1999)

[Hwa06] H. Hwang, V. Hristidis, Y. Papakonstantinou, "ObjectRank: A System for Authoritybased Search on Databases" In Proceedings of the Thirtieth international conference on Very large data bases - Volume 30 (2006), pp. 564-575.

[Tou09] N. Toupikov, J¨. Umbrich, R. Delbru, M. Hausenblas,G. Tummarello, "DING! Dataset Ranking using Formal Descriptions" In LDOW2009.

[Gol06] J. Golbeck, "Trust on the world wide web: a survey," *Found. Trends Web Sci.*, vol. 1, no. 2, pp. 131-197, 2006.

فصل 7- واژه نامه

authority	ذیصلاح
Certificate	گواهینامه
consolidation	یکپارچه‌سازی
context	زمینه
crawling	پیمایش
gather	جمع آوری
hubs	مرکز
link farm	ارتباطات گروهی
Mutual reinforcement	تقویت متقابل
object	مفعول
Policy	سیاست
predicate	مسند
Provenance	اصالت
query	پرسش
Ranking	رتبه‌بندی
reasoning	استنتاج
Reputation	شهرت
run	اجرا
scatter	توزیع
search	جستجو
subject	فاعل
Trust	اعتماد
web of data	وب داده‌ها
web of document	وب اسناد

فصل 8- پیوست: شبه کدها

8-1-کد جاوا برای ایجاد اسپم لینک

```
public String createLinkFarm() {
    try {
      int number = 10;
      //add the fake domains into the available subject domains and avaialbale subjects domains
      FileWriter sfstream = new FileWriter("D://data/subjects.txt", true);
      BufferedWriter sout = new BufferedWriter(sfstream);
      FileWriter ofstream = new FileWriter("D://data/objects.txt", true);
      BufferedWriter oout = new BufferedWriter(ofstream);
      for(int k=0;k<number;k++){
      sout.write("<http://www.d"+k+".org/d> \n");
      oout.write("<http://www.d"+k+".org/d> \n");
      }
      //Close the output stream
      sout.close();
      oout.close();
      //read all objects domains into the objlist array
      //read all subjects domains into the subjlist array
      ArrayList subjlist=new ArrayList();
      ArrayList objlist=new ArrayList();
      BufferedReader subr=new BufferedReader(new FileReader(new File("D://data/subjects.txt")));
      BufferedReader objr=new BufferedReader(new FileReader(new File("D://data/objects.txt")));
      String element ;
      while((element = subr.readLine()) != null)
      {
        subjlist.add(element);
      }
      while((element = objr.readLine()) != null){
        objlist.add(element);
      }
      //read all linktypes into linktype array
      ArrayList linktype=new ArrayList();
      BufferedReader br=new BufferedReader(new FileReader(new File("d://data/linktypeding.txt")));
      String line;
      while ((line = br.readLine()) != null) {
        linktype.add(line);
      }
      // create the fake quads writer
      String filepath = "D://Data/RandomlinkFarm200diffpred.nq";
      BufferedWriter bw = new BufferedWriter(new FileWriter(new File(filepath)));
      int size=200;
      Random r = new Random();
      int subj, obj, p;
      for (int i = 0; i < size; i++) {
```

```
            subj = r.nextInt(number);
            obj = r.nextInt(objlist.size());
            p = r.nextInt(linktype.size());
            //different methods to create fake quads
            //subject is a fake domain, preicate is a fake predicate, object is a fake domain, context is the same as subject
            /*1*/bw.write("<http://www.d"+subj+".org/d> "+"<http://www.fakepred"+r.nextInt(number)+".org/fake>
"+"<http://www.d"+r.nextInt(number)+".org/d> "+"<http://www.d"+subj+".org/d> ."+"\n");
            //subject is a random subject(fake or real), predicate is a real predicate, object is a random object(fake or real), context is a fake domain
            /*2*/bw.write(subjlist.get(subj).toString()+" "+linktype.get(p).toString()+" "+objlist.get(obj).toString()+"
<http://www.d"+r.nextInt(number)+".org/> .\n");
        }
        bw.flush();
        bw.close();
        return filepath;
    } catch (Exception e) {
        return "";
    }
  }
```

Faculty Of Engineering

Computer Engineering Group

A New Dataset Ranking Algorithm for Semantic Search Engine to Resist Web Spam

Soheila Dehghanzadeh

Under Suopervision of Dr. Mohsen Kahani

Dissertation submitted in pursuance of the degree of Master of computer software
September 2011

Abstract:

With the advent of semantic web and proliferation of semantic data, understanding of web data and providing machine understandable data by software agents is strongly demanded. It is obvious that human being desirable search result is different than what is expected by software agents. As the web going to be understandable by software agents, web applications, especially search engines, should adapt themselves accordingly.

The Linked Open Data project has made lots of semantic data available on the web. which is process able by humans and machines. In order to be able to use this vast amount of semantic information, they should be searchable so that humans and machines can locate them. Therefore second generation of semantic web applications needs an efficient access point that take into account the semantic nature of this knowledge.

As search engines are the main gates to web data and knowledge, human and machines should be able to find their search results using them. So an emergent need for a semantic search engine for human and machines appears.

Considering the successful initiation of web of data and Due to the large and ever increasing financial gains that can be gathered from high search engine ratings, there is no doubt that a significant amount of human and machine resources are devoted to artificially inflating the rankings of certain web pages and trying to bypass the ranking algorithm of search engine. These illegal attempts are done by spammers. The ranking method of a search engine is responsible for detecting and combating web spam.

we are inspired by theses emergent need of spam detection in semantic search engine. In this thesis we investigated all the available ranking methods in web of documents and we are intended to propose an new ranking method to resist web spam.

We test Different Spamming methods against famous dataset ranking methods such as DING which used by Sindice and naming authority matrix which is use by SWSE. However the results shows that they are vulnerable against some kind of link spam.

The proposed method for dataset ranking, first detects semantic link farms and penalize them. A new link weighting method has be proposed to weight the links. Finally weighted pageRank is applied. As the results shows, the injected spam has been appeared at the lowest ranks. The author believes that In order to harvest all the implicit and explicit concept of a quad, both the DING and Naming authority matrix should be employed. The proposed method is a combination of both and it is using the spam detection methods.

The main contribution of this thesis is a new dataset ranking method which is resistant against spam and use all the explicit and implicit meaning of quads. To evaluate the proposed method, we have gathered all the drugs related quads form LOD. Four kind of spam has been injected gradually into the gathered dataset. As the results prove, the proposed method is successful to detect any kind of injected spam.

keywords: Semantic Web, Semantic Search Engine, Content Spam, Link Spam, RDF, Ranking, Trust Metric, Link Analysis, Quad, Provenance.