آزمایشگاه فناوری وب

پایان‌نامه کارشناسی ارشد

ارائه یک سیستم پیشنهاد استناد مبتنی بر داده‌های پیوندی

فتانه زرین کلام

استاد راهنما: دکتر محسن کاهانی

بهمن ماه ۱۳۹۰


بسم الله الرحمن الرحيم

تقدیم

تقدیم به پدر و مادر عزیزم

به خاطر همه حمایت‌ها و زحمات بی دریغشان

تقدیم به همسر مهربانم

که در لحظه لحظه‌های اجرای این پایان‌نامه، یار و همراهم بود

تقدیر و سپاس

سپاس خدا را به اندازه همه سپاسی که نزدیکترین فرشتگان و گرامی‌ترین بندگان و پسندیده‌ترین ستایش کنندگان او را ستایش کرده‌اند , سپاسی که بر سپاس‌های دیگر برتری داشته باشد، مانند برتری که پروردگار نسبت به آفریدگانش دارد.

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

همچنین در انجام این پایان‌نامه از همکاری اعضای آزمایشگاه فناوری وب[1] بهره‌ی زیادی برده‌ام که در اینجا لازم است از آن‌ها تشکر نمایم

[1] http://wtlab.um.ac.ir

چکیده

حجم فراوان و روبه رشد مقالات منتشر شده بر روی وب، فرآیند تصمیم‌گیری و انتخاب مقالات مرتبط با یک زمینه تحقیقاتی را برای پژوهشگران دشوار کرده است. روش رایجی که اغلب پژوهشگران برای جستجوی اسناد مرتبط با یک زمینه تحقیقاتی استفاده می‌کنند، یافتن کلمات کلیدی و استفاده از موتورهای جستجو می‌باشد. با توجه به این‌که پیدا کردن لیست کلمات کلیدی که دربرگیرنده تمام مقالات یک زمینه باشند کاری دشوار است، با استفاده از این روش، نمی‌توان به خوبی تمامی مقالات مرتبط را پیدا نمود.

یک سیستم پیشنهاد استناد، با دریافت متن ورودی، مقالاتی که باید آن متن به آن‌ها استناد کند را پیشنهاد می‌کند، و بدین ترتیب می‌تواند در یافتن مقالات مرتبط با یک موضوع به پژوهشگر کمک کند.

در حال حاضر سیستم‌های پیشنهاد استناد موجود محدود به پیشنهاد از یک منبع داده محلی می‌باشند، این محدودیت، از آنجاییکه در زمینه کتاب‌شناسی هیچ منبع‌داده‌ای حاوی اطلاعات کامل درباره تمام مقالات نمی-باشد، باعث کاهش کیفیت پیشنهادها می‌شود.

در این پایان‌نامه یک سیستم جدید برای پیشنهاد استناد ارائه شده است که در لایه داده خود از داده‌های پیوندی استفاده می‌کند و الگوریتم پیشنهاد آن مبتنی بر ترکیب شباهت رابطه‌ای و شباهت متنی می‌باشد. ارزیابی-های انجام شده نشان می‌دهد که استفاده از داده‌های پیوندی بعنوان لایه داده بدلیل مزایای آن از جمله انتشار داده‌ها با یک قالب استاندارد و برقراری پیوند بین منابع داده مختلف باعث کاهش پیچیدگی جمع‌آوری داده و غنی شدن لایه داده به دلیل استفاده از چندین منبع می‌شود.

همچنین با توجه به آزمایش‌های انجام شده، معیار شباهت رابطه‌ای پیشنهادی، در تشخیص شباهت مقالات موفق است و استفاده از آن در کنار شباهت متنی می‌تواند ضعف استفاده تنها از شباهت متنی را در پیدا کردن مقالات مرتبط کاهش دهد و در نتیجه سبب بهبود کیفیت سیستم پیشنهاد استناد شود.

کلمات کلیدی: بازیابی اطلاعات، پیشنهاد، استناد، تحلیل استناد، شباهت رابطه‌ای، داده‌های پیوندی

فهرست مطالب

فهرست شکل‌ها

فهرست جدول‌ها

فهرست علائم و اختصارات

SemCiR	Semantic Citation Recommendation
NDCG	Normalized Discounted cumulative gain
RDF	Resource Description Framework.
SPARQL	SPARQL Protocol and RDF Query Language.
ItIF	In-text Impact Factor
DSI	Distance Similarity Index
N3	Notation3
LOD	Linked Open Data
HTTP	Hypertext Transfer Protocol
URI	Uniform Resource Identifier
W3C	World Wide Web Consortium

فصل ۱– مقدمه

در این فصل، ابتدا تعریف مساله آورده شده است که در آن علت وجود سیستم‌های پیشنهاد استناد و چالش‌های آن توضیح داده می‌شود. سپس در بخش ۱-۲، راه‌حل پیشنهادی این پایان‌نامه برای برطرف کردن چالش‌های موجود به اختصار شرح داده می‌شود. در ادامه نیز نوآوری‌های روش‌پیشنهادی و ساختار بقیه مطالب این پایان‌نامه ذکر می‌شود.

۱–۱ تعریف مساله

هر پژوهشگری قبل از شروع کاری جدید در زمینه مورد علاقه خود باید از کارهای انجام شده درباره آن موضوع آگاهی کافی داشته باشد. نداشتن دانش کافی نسبت به کارهای گذشته باعث به نتیجه نرسیدن تلاش-های یک پژوهشگر و یا انجام کاری تکراری می‌شود. با توجه به اهمیت زیاد این موضوع، و نیز رشد روزافزون علم و افزایش تعداد اسناد علمی منتشر شده، نیاز به سیستمی که پژوهشگران را در این امر یاری کند کاملا محسوس است (Bethard et al, 2010; Tang and Zhang, 2009).

امروزه، اغلب پژوهشگران برای یافتن کارهای مرتبط با یک موضوع، از روش‌های رایج، مثل جستجو در گوگل استفاده می‌کنند. ورودی این روش‌ها، اغلب تعدادی کلمه کلیدی، و خروجی آن‌ها اسنادی است که شامل این کلمات کلیدی هستند. بدین ترتیب اگر پژوهشگری در ارتباط با موضوع مورد علاقه خود متنی داشته باشد و کارهای مرتبط با آن را بخواهد، ابتدا باید کلمات کلیدی موجود در متن را استخراج کند. استخراج این کلمات برای پژوهشگری که به تازگی به تحقیق در یک زمینه پرداخته است، کار آسانی نیست. همچنین مشکل دیگر این است که پژوهشگر باید زمان نسبتا زیادی را برای انتخاب گزینه‌های بهتر از بین خروجی‌های موتور جستجو صرف کند.

ضمنا از آنجاییکه هدف بدست آوردن کارهای مرتبط است و نه صرفا کارهایی که شباهت متنی زیادی با متن ورودی دارند، ممکن است تعدادی از کارهای مرتبط در بین جواب‌های جستجو ظاهر نشوند

(Henzinger et al., 2003). علت این امر را می‌توان علاوه بر مشکلات رایج در پردازش متن مثل ابهام‌های زبانی و کلمات هم‌خانواده (Baeza-Yates, 2004)، موارد اشاره شده در زیر دانست:

۱- ممکن است موضوع دو سند دقیقا یکسان باشد، اما از آنجایی که توسط دو نویسنده مختلف نوشته شده‌اند و کلمات مورد استفاده این دو نویسنده متفاوت است، شباهت متنی دو سند کم بوده و در نتیجه استفاده صرف از شباهت متنی باعث می‌شود مرتبط شناخته نشوند.

۲- دو سند مرتبط لزوما دو سند با شباهت متنی زیاد نمی‌باشند، مثلا سندی که درباره زمانبندی پردازش-ها به کمک الگوریتم ژنتیک بحث می‌کند، ممکن است به سندی که ایده کلی الگوریتم ژنتیک را توضیح داده است، شباهت متنی کمی داشته باشد. اما این دو سند کاملا به هم مرتبط می‌باشند، چرا که سند دوم، پایه علمی تکنیک مورد استفاده در سند اول را توضیح می‌دهد. در اینجا نیز استفاده صرف از شباهت متنی قادر به تشخیص ارتباط این اسناد نیست.

راه‌حل برطرف کردن چنین مشکلاتی وجود یک سیستم پیشنهاد استناد است که ورودی آن یک قطعه متن و خروجی آن مقالاتی است که باید در آن متن مورد استناد قرار بگیرند، به عبارتی مقالات مرتبط با آن متن است (Strohman et al., 2007; Bethard et al., 2010; Tang and Zhang, 2009).

لایه داده در سیستم‌های پیشنهاد استناد، منابع داده‌ای می‌باشند که اطلاعات مقالات علمی را منتشر می‌کنند، جمع‌آوری این داده‌ها بعنوان لایه داده یکی از مهم‌ترین چالش‌ها در این سیستم‌ها می‌باشد. چرا که منابع داده مختلف اطلاعات خود را به قالب‌های مختلفی منتشر می‌کنند، در نتیجه برای جمع‌آوری اطلاعات هر یک از این منابع داده باید روال مجزایی اجرا شود.

چالش دیگری که در لایه داده سیستم‌های موجود وجود دارد، این است که از آنجاییکه این منابع داده کاملا جدا از هم می‌باشند و داده‌های آن‌ها به صورت محلی و خصوصی می‌باشد، اگر مثلا مقاله‌ای در یک منبع داده، و مراجع آن در منبع داده دیگری منتشر شود، امکان بازیابی اطلاعات مراجع آن مقاله بسادگی وجود ندارد.

معمولا اطلاعات مقالات مختلف توسط منابع‌داده متفاوتی منتشر می‌شوند، مثلا اطلاعات یک مقاله ممکن است توسط *IEEE*[1] منتشر شود و در *Citeseer*[2] و *DBLP*[3] وجود نداشته باشد، و یا برعکس. حتی اگر اطلاعات یک مقاله توسط دو منبع داده منتشر شود، این منابع ممکن است اطلاعات متفاوتی درباره آن مقاله منتشر کنند، برای مثال کلمات کلیدی یک مقاله در *IEEE* منتشر می‌شود ولی در *DBLP* منتشر نمی‌شود. در نتیجه فرض استفاده تنها از یک منبع داده در زمینه کتاب‌شناسی[4] برای برطرف کردن نیازهای اطلاعاتی سیستم‌های پیشنهاد استناد، یک فرض منطقی نمی‌باشد و باعث می‌شود کیفیت پیشنهادها از نظر کامل بودن نتایج کاهش یابد.

۱-۲ راه‌حل

در این پایان‌نامه برای کاهش ضعف ناشی از استفاده تنها از شباهت متنی در پیشنهاد کارهای مرتبط، دو راهکار ارائه شده است.

۱- استفاده از ویژگی‌های رابطه‌ای مثل لیست مراجع و نویسندگان، در کنار ویژگی‌های متنی

۲- استفاده از متن استناد بعنوان یک ویژگی متنی علاوه بر بقیه ویژگی‌های متنی مثل عنوان و چکیده

برای استفاده از این دو راهکار، ابتدا یک معیار جدید برای محاسبه شباهت رابطه‌ای دو مقاله ارائه شده است. سپس نشان داده می‌شود که استفاده از این معیار در کنار شباهت متنی نقش موثری در تشخیص مقالات مرتبط دارد، و با افزودن متن استناد نیز می‌توان کیفیت سیستم‌های پیشنهاد استناد را افزایش داد.

همچنین در این پایان‌نامه برای تولید سیستمی که به راحتی بر روی هر منبع داده‌ای کار کند و قادر باشد از اطلاعات چندین منبع داده استفاده کند، استفاده از وب داده بجای وب اسناد در لایه داده پیشنهاد می‌شود. استفاده از داده‌های پیوندی بدلیل قالب یکسان آن‌ها در بازنمایی داده‌ها مشکل ناهمگونی داده‌های منابع

[1] http://www.ieee.org/index.html
[2] http://citeseer.ist.psu.edu/index
[3] http://dblp.uni-trier.de/
[4] Bibliography

مختلف را برطرف می‌کند، همچنین بدلیل داشتن پیوندهای معنادار بین داده‌های منابع مختلف امکان استفاده از منابع داده مختلف را ساده می‌کند.

۱–۳ نوآوری‌های سیستم پیشنهادی

نوآوری‌های سیستم پیشنهادی را می‌توان در موارد زیر خلاصه کرد:

۱- ارائه الگوریتمی مبتنی بر ترکیب ویژگی‌های متنی و رابطه‌ای برای پیشنهاد استناد

۲- استفاده از متن استناد بعنوان یک ویژگی متنی و نشان دادن تاثیر آن در سیستم‌های پیشنهاد استناد

۳- ارائه یک معیار شباهت رابطه‌ای برای بدست آوردن شباهت دو مقاله

۴- استفاده از الگوریتم ژنتیک برای وزن‌دهی خودکار به ویژگی‌های رابطه‌ای و نشان دادن میزان تاثیر هر یک از آن‌ها در سیستم پیشنهاد استناد

۵- استفاده از داده‌های پیوندی در لایه داده سیستم پیشنهاد استناد

۶- ارائه یک الگوریتم غنی‌سازی داده‌ها با استفاده از چند منبع داده پیوندی

۱–۴ ساختار پایان‌نامه

سازماندهی مطالب این پایان‌نامه در ادامه توضیح داده شده است. در بخش بعدی با توجه به این که کار انجام شده در این پایان‌نامه به سه زمینه "تحلیل استناد"، "سیستم‌های پیشنهاد مقاله" و "سیستم‌های پیشنهاد دهنده مبتنی بر داده‌های پیوندی" مرتبط است، کارهای مرتبط انجام شده در هر زمینه شرح داده شده است. سپس در فصل سوم، راه‌حل پیشنهادی برای تولید یک سیستم پیشنهاد استناد توضیح داده شده است، در فصل چهارم، جزئیات پیاده‌سازی سیستم پیشنهادی، نحوه ارزیابی و نتایج آن آورده شده است. بخش آخر نیز به بیان نتیجه‌گیری و کارهای آینده اختصاص دارد.

فصل ۲– مروری بر کارهای گذشته

سیستم پیشنهادی این پایان‌نامه، یک سیستم پیشنهاد استناد است که باگرفتن متن ورودی مقالات مرتبط با آن متن را پیشنهاد می‌دهد، در لایه داده این سیستم از داده‌های پیوندی استفاده شده است و الگوریتم آن مبتنی بر روش‌های موجود در زمینه تحلیل استناد می‌باشد. در نتیجه در این فصل کارهای مرتبط موجود در هر یک از زمینه‌های مرتبط با سیستم پیشنهادی یعنی "**تحلیل استناد**[1] "، "**سیستم‌های پیشنهاد مقاله**" و "**سیستم‌های پیشنهاد دهنده مبتنی بر داده‌های پیوندی**"، مورد بررسی قرارگرفته است.

۲–۱ تحلیل استناد

یک استناد در واقع تاییدی است که یک سند از دیگری دریافت می‌کند (Smith, 1981). در حالت کلی، تحلیل استناد به مطالعه ارتباطات بین یک سند و سندی که به آن استناد می‌کند می‌پردازد، در این حوزه از علم مفاهیم مختلفی استفاده می‌شود که در ادامه توضیح داده شده‌اند:

- *گراف استناد*[2]: گرافی که گره‌های آن نشان‌دهنده اسناد و یال‌های آن‌ها نشان‌دهنده استناد بین آن‌ها می‌باشد.
- *تعداد استناد یک سند:* تعداد اسنادی که به آن سند استناد می‌کنند و اغلب بعنوان معیاری برای سنجش کیفیت یک سند استفاده می‌شود، به‌طوری‌که اسنادی که تعداد استناد بیشتری دارند از آنجایی که مورد تصدیق افراد بیشتری قرار گرفته‌اند از کیفیت بالاتری برخوردارند. در (Bornaman,2008) بررسی کاملی از تفسیرهای مختلف معیار تعداد استناد، شرح داده شده است.
- معیار تاثیر یک مجله که توسط گارفیلد در (Garfield, 1972) به‌عنوان معیاری برای ارزیابی کیفیت یک مجله معرفی شده است، برابر است با تعداد دفعاتی که در یک بازه مشخص به طور میانگین مقالات موجود در آن مجله مورد استناد قرار می‌گیرند.

[1] Citation analysis
[2] Citation graph

- استناد مشترک[1]: دو سند دارای استناد مشترک می‌باشند اگر سندی وجود داشته باشد که به هر دو این اسناد استناد کند (Small, 1973). مفهوم استناد مشترک این است که سندهایی که دارای شباهت هستند به احتمال زیاد توسط یک سند مشترک مورد استناد قرار می‌گیرند. از این معیار نیز جهت بدست آوردن میزان شباهت نین دو سند استفاده می‌شود.
- متن استناد: قسمتی از متن یک سند است که در آن به سند دیگری استناد شده است. در تحلیل متن استناد[2] ، کلمات صریح و با محتوایی که نویسنده استناد دهنده برای توصیف کار استناد شده استفاده می‌کند هدف مطالعه قرار می‌گیرد (Small, 1982). استناد در مقاله استناد دهنده خلاصه-ای از سند استناد شده و یا حداقل چیزی است که نویسنده استناد دهنده فکر می‌کند در در کار استناد شده مهم می‌باشد. به‌عبارت دیگر، متن استناد شامل کلمات اصلی کار استناد شده است. در نتیجه برای بدست آوردن میزان شباهت بین یک متن و یک سند می‌توان میزان شباهت آن را با متن‌های استناد آن بدست آورد.

اسمیت در (Smith, 1981) کاربردهای زیادی برای تحلیل استناد نام برده است که یکی از آن‌ها که مرتبط با سیستم پیشنهادی این پایان‌نامه نیز می‌باشد، کاربرد تحلیل استناد در بازیابی اطلاعات است. برای مثال برای بهبود عملکرد الگوریتم‌های دسته‌بندی اسناد در (Couto et al., 2010) از دو معیار استناد مشترک و زوج‌های کتابشناختی استفاده شده است، استروهمن و همکارانش نیز در (Strohman et al, 2007) از معیار استناد مشترک در کنار فاکتوهای دیگر برای پیشنهاد اسناد مرتبط با یک متن استفاده کردند.

با توجه به این ایده که متن استناد شامل کلمات اصلی متن استناد شده است، مطالعات زیادی نیز به طور خاص نقش موثر متن استناد را در بازیابی اطلاعات نشان داده‌اند، برای مثال، اُ کرنل در (O'Connor, 1982)به این موضوع اشاره می‌کند که عبارات اسمی[3] موجود در متن استنادهای یک سند، برای بهبود در بازیابی باید اندیس‌گذاری شوند و به نمایش آن سند اضافه شوند. الجابر و همکارانش در (Aljaber et al., 2010) نقش موثر استفاده از متن استناد را در بالا بردن کارایی خوشه‌بندی اسناد نشان دادند.

[1] Co-citation
[2] Citation context analysis
[3] Noun phrase

ریتچی و همکارانش در (Ritchie, 2008; Ritchie et al., 2008) برای بهبود کارایی بازیابی اسناد، علاوه بر متن آن سند، متن‌های استنادی که در سندهای دیگر به آن سند استناد شده بود را نیز اندیس‌گذاری کردند و نشان‌دادند که کارایی بهبود می‌یابد، همچنین آزمایشاتی جهت بدست آوردن طول متن استناد نیز انجام دادند و نتیجه گرفتند که طول ثابت برای انتخاب متن استناد موثر ترین روش است.

۲-۲ پیشنهاد مقاله

سیستم‌های پیشنهاد دهنده مقالات به کاربران، با توجه به علاقه کاربر و زمینه کاری او از بین مقالات موجود در مجموعه داده خود تعدادی را به ترتیب میزان ارتباط با علاقه‌مندی کاربر به او پیشنهاد می‌دهند، کارهای انجام شده در این زمینه را می‌توان از نظر نحوه گرفتن علاقه‌مندی‌های کاربر به دو دسته تقسیم کرد: (۱) کارهایی که با توجه به پروفایل کاربر علاقه‌مندی‌های او را استخراج کرده و به او پیشنهاد می‌دهند و (۲) کارهایی که متنی را دریافت کرده و کارهای مرتبط با آن متن را پیشنهاد می‌دهند، متن ورودی شامل اطلاعاتی است که که کاربر قصد مطالعه بیشتر در آن زمینه را دارد.

۲-۲-۱ پیشنهاد مقاله مبتنی بر پروفایل کاربر

روش‌های موجود در این دسته در یک مرحله پروفایل کاربر را استخراج کرده و در مرحله دیگر مقالاتی که مشابه علاقه‌مندی‌های کاربر باشد را به او پیشنهاد می‌دهند، در ادامه تعدادی از کارهایی که در این دسته می‌باشند توضیح داده شده است.

باسیو و همکارانش در (Basu et al., 2001) مساله پیشنهاد مقاله را این گونه مطرح کردند که قرار است مقالات یک کنفرانس به اعضای کمیته بازبینی آن داده شود. به طور خاص رویکرد آن‌ها برای پیشنهاد مقاله، تخصیص مقالات به بازبین‌گرها با توجه به خصوصیات آن‌ها می‌باشد. برای مثال مقالات می‌توانند به وسیله عناوین، چکیده و لیست کلمات کلیدی نمایش داده شوند، و اطلاعات بازبین‌گرها با تحلیل مقالات نوشته شده توسط آن‌ها و استخراج اطلاعاتی درباره تخصص آن‌ها بدست می‌آید. بعد از این‌که این اطلاعات بدست آمد، یک ماتریس مقاله-بازبین‌گر تولید می‌شود و هر پیشنهادی می‌تواند با پرس‌جو گرفتن از پایگاه‌داده‌های ماتریس بدست آید. برای مثال پرس و جوی زیر به این منظور نوشته شده است.

SELECT Reviewer.Name, Paper.ID

FROM Paper AND Reviewer

WHERE Reviewer.Descriptor SIM Paper.Abstract

میدلتون و همکارانش نیز در (Middleton et al., 2004) روشی در زمینه پیشنهاد مقالات پژوهشی با هدف جستجو برای مقالات مرتبط پیشنهاد دادند. رویکرد آن‌ها برای رسیدن به هدف خود منطبق کردن هستان‌شناسی‌های[1] مقالات و کاربران است. ویژگی‌های یک مقاله با یک بردار که شامل عنوان‌های وابسته به هستان‌شناسی است ارائه می‌شوند. پروفایل کاربر نیز به کمک ابزارهای تعبیه شده‌ای که رفتارهای کاربر در سیستم را دنبال می‌کنند، استخراج می‌شود. برای مثال یک کاربر یک مقاله خاص، این امکان را دارد که مشخص کند در مورد آن مقاله علاقه‌مند، بدون علاقه و یا بی‌نظر است. یک کاربر همچنین می‌تواند مقالات را با دسته‌بندی‌های از پیش تعریف شده توسط سیستم برچسب‌گذاری کند. در نهایت پروفایل‌های علاقه‌مندی‌های کاربر بدست آمده و پیشنهادات با تکنیک‌های بازیابی اطلاعات بدست می‌آید.

بوگرز و بوسچ در (Bogers and Bosch, 2008) توضیح دادند که چگونه می‌توان از سایت *CiteULike*[2] که یک سایت مدیریت مراجع اجتماعی[3] است، برای پیشنهاد دادن مقاله‌های علمی به کاربران بر اساس مجموعه مراجع‌شان که به‌طور ضمنی نشان‌دهنده پروفایل آن‌ها می‌باشد، استفاده کرد. *CiteULike* یک نمونه خاص از سایت‌های نشانه‌گذاری اجتماعی[4] است. کاربران می‌توانند در این سایت‌ها عضو شوند و بعد منابع مختلفی را که روی وب استفاده می‌کنند را نشانه‌گذاری کنند. مثلا یک نفر آدرس صفحات وبی را که برایش جالب است را در یک سایت نشانه‌گذاری ذخیره می‌کند و برای هر کدام نیز تگ‌هایی انتخاب می‌کند. در نتیجه در آینده می‌تواند در بین این آدرس‌ها به جستجو (با استفاده از تگ‌ها) بپردازد. معمولا در این سیستم‌ها کاربران می‌توانند از نشانه‌گذاری‌های یکدیگر نیز استفاده کنند. کاربران *CiteULike* می‌توانند مجموعه مراجع مورد نظرشان را در آن ذخیره و با استفاده از تگ‌ها سازماندهی نمایند (این مراجع بطور خاص، مقاله های علمی می باشند) سپس می‌توانند مراجع کاربران دیگر را که دارای تگ‌های مشابهی هستند یا دارای شهرت و عمومیت بیشتری هستند

[1] Ontology
[2] http://www.citeulike.org/faq/data.adp.
[3] Social reference management
[4] Social bookmarking

را بازیابی کنند. همین مساله در سطح نویسنده نیز قابل انجام است. یعنی می‌توانند مراجعی را که توسط کاربران دیگر ثبت شده‌اند و نویسنده آن مراجع با نویسنده مراجع مورد نظر کاربر، یکسان است را جستجو کنند.

در (Bogers and Bosch, 2008)، هدف این است که با استفاده از سرویس تحت وب *CiteULike*، بتوان برای کاربر، یک لیست از مراجع مناسب تولید و به او پیشنهاد کرد. بوگر و بوسچ برای رسیدن به این هدف با کمک داده‌های بدست آمده از *CiteULike*، سه الگوریتم مختلف فیلتر همبستگی پیشنهاد کردند: الگوریتم اول، مبتنی بر آیتم است و برای تشخیص مشابهت اشیا از فاصله کسینوسی استفاده می کند، الگوریتم دوم نیز مبتنی بر آیتم است و برای تشخیص مشابهت‌ها از احتمال شرطی استفاده می‌کند. الگوریتم سوم، مبتنی بر کاربر است و از فاصله کسینوسی استفاده می‌کند. در مدل فیلتر همبستگی مبتنی بر کاربر، هر کاربر دارای پروفایلی است که اشیای مورد علاقه‌اش در آن ثبت شده‌اند.

آن‌ها به این نتیجه رسیدند که معمولا روش مبتنی بر آیتم زمانی خوب است که تعداد کاربران به نسبت تعداد اشیا بیشتر باشد و روش مبتنی بر کاربر زمانی خوب عمل می کند که تعداد اشیا از تعداد کاربران بیشتر باشد. در این مقاله با توجه به آزمایشات تجربی انجام شده در مجموع نتیجه گرفته‌اند که الگوریتم مبتنی بر کاربر به میزان قابل توجهی، از دو الگوریتم دیگر بهتر عمل می‌کند و کارایی اش هم قابل قبول می‌باشد.

گوری و پوسی در(Gori and Pucci, 2006) یک الگوریتم پیشنهاد مقالات تحقیقاتی ارائه کردند، که مبتنی بر گراف استناد و خصیصه‌های گشت‌زن تصادفی[1] است. این الگوریتم که *PaperRank* نام دارد می‌تواند مجموعه‌ای از مقالات موجود در یک کتابخانه الکترونیکی که از طریق مراجع به یکدیگر پیوند داده شده‌اند را بررسی کرده و به هر کدام یک امتیاز اولویت[2] نسبت دهد.

در واقع کاری که در (Gori and Pucci, 2006) انجام شده است این است که آن‌ها الگوریتم *pageRank* گوگل را برای هدف خودشان تطبیق داده‌اند. به عبارتی *PaperRank* یک نسخه بایاس شده از الگوریتم *PageRank* است که برای استفاده در سیستم‌های پیشنهاددهنده طراحی شده است. الگوریتم گوگل ایده‌اش

[1] Random walker
[2] Preference score

این است که محاسبه می‌کند که یک کاربر که بطور تصادفی بین لینک‌های صفحات وب جابجا می‌شود، چقدر احتمال دارد که صفحه x را ببیند، یعنی چقدر احتمال دارد که در حین اینکه لینک‌ها را بطور تصادفی دنبال می‌کند، به صفحه x برسد. هرچه میزان این احتمال برای صفحه‌ای بیشتر باشد، آن صفحه در مرتب‌سازی گوگل، اولویت بیشتری می‌گیرد. گوگل برای محاسبه این احتمال از تعداد لینک‌های خروجی صفحات استفاده می‌کند.

در الگوریتم ارائه شده در (Gori and Pucci, 2006) هم ایده‌ای مشابه الگوریتم گوگل استفاده شده است. روش کار بدین ترتیب است که ابتدا گراف استناد مربوط به کل مقالات موجود در پایگاه ایجاد می‌شود. سپس کاربر، یک مجموعه کوچک از مقالات مورد نظرش را بعنوان نمونه انتخاب می‌کند این مقالات پروفایل کاربر را مشخص می‌کنند. بعد سیستم با توجه به این مقالات و استفاده از گراف استناد مقالات مشابه که از پایگاه‌داده‌ها استخراج می‌شود. مشابه الگوریتم گوگل، برای هر مقاله، یک امتیاز مشخص می‌کند. امتیاز مقاله x، مشخص می‌کند که وقتی یک گشت زن تصادفی از یکی از مقالات مورد علاقه کاربر شروع به گشت زنی کند، چقدر احتمال دارد که به مقاله x برسد. هرچه این احتمال بیشتر باشد، امتیاز مقاله x برای پیشنهاد شدن به کاربر افزایش می یابد. برای محاسبه این احتمال و امتیاز از فرمول‌های مربوط به گراف استناد استفاده شده است.

چندراسکان و همکارانش در(Chandrasekaran et al, 2008) و پوداکاتری و همکارانش در (Pudhiyaveetil et al, 2009) سیستمی برای پیشنهاد مقاله با توجه به پروفایل کاربر پیشنهاد دادند، آن‌ها برای پیشنهاد دادن از مشابهت بین مفاهیم موجود در پروفایل کاربر و مفاهیم موجود در مقالات استفاده کردند. تفاوت کار آن‌ها در نحوه ساختن پروفایل کاربر بود. چندراسکان و همکارانش در (Chandrasekaran et al, 2008)برای ساختن پروفایل کاربر از مقالات نوشته شده توسط آن کاربر که در پایگاه‌داده‌های *Citeseer* وجود دارد استفاده کردند، در نتیجه سیستم آن‌ها تنها قادر بود برای نویسندگانی که توسط *Citeseer* شناخته شده بودند پیشنهاداتی داشته باشد، اما سیستم پیشنهادی کوداکاتری و همکارانش از مقالاتی که توسط آن کاربر مشاهده شده بود برای ساختن پروفایل کاربر استفاده کرد در نتیجه دامنه پیشنهاد او نسبت به سیستم قبلی گسترده‌تر بود.

۲-۲-۲ پیشنهاد مقاله مبتنی بر محتوای متن ورودی

هدف سیستم‌هایی که در این دسته قرار می‌گیرند، پیشنهاد مقالاتی است که مرتبط با متنی می‌باشند که به‌عنوان ورودی به سیستم وارد می‌شود، به این سیستم‌ها در اصطلاح "سیستم‌های پیشنهاد استناد" گفته می‌شود. به عبارتی این سیستم‌ها مقالاتی که می‌توانند توسط متن ورودی مورد استناد قرار بگیرند را پیشنهاد می‌دهند در این قسمت ابتدا روش‌های موجود در این سیستم‌ها و سپس نحوه ارزیابی آن‌ها توضیح داده شده است.

۲-۲-۲-۱ سیستم‌های پیشنهاد استناد: روش‌ها

کارهای موجود در این سیستم‌ها خود با توجه به هدفشان در سه دسته قرار می‌گیرند:

دسته اول:

دسته اول، کارهایی می‌باشند که تنها یک متن می‌گیرند و هدف آن‌ها پیشنهاد مقالاتی است که مرتبط با آن متن ورودی می‌باشند، در ادامه کارهای متعلق به این دسته شرح داده شده است:

استروهمن و همکارانش در (Strohman et al., 2007) مساله پیشنهاد مقالات مرتبط را این‌گونه مطرح کردند: کاربری برای متنی که در ارتباط با موضوعی نوشته است، نیاز به سیستمی دارد تا مقالاتی که باید در آن متن به آن‌ها استناد شود را پیشنهاد دهد. آنها برای ایجاد این سیستم از گراف استناد استفاده کردند و ادعا کردند هیچ یک از ویژگی‌های مبتنی بر متن یا ویژگی‌های مبتنی بر استناد به تنهایی خوب عمل نمی‌کنند. ویژگی‌های مبتنی بر متن برای پیدا کردن کارهای مرتبط اخیر مناسب هستند، زیرا در آن‌ها از کلمات مشابه استفاده می‌شود اما برای پیداکردن کارهای مشابه مفهومی از آنجایی که از کلمات متفاوتی استفاده می‌کنند مناسب نیستند و در بدست آوردن اعتبار متن‌ها ناتوان هستند. ویژگی‌های مبتنی بر استناد می‌توانند اعتبار متن‌ها را محاسبه کنند اما از آنجاییکه مقالات اخیر ممکن است مورد استناد قرار نگرفته باشند، دارای ضعیف‌تر می‌باشند. به طور کلی مدل پیشنهادی استروهمن در (Strohman et al., 2007)، شامل دو قدم اصلی است: ۱) انتخاب اسناد و ۲) مرتب کردن آن‌ها برای پیشنهاد.

در قدم اول، ابتدا ۱۰۰ مقاله که بیشترین شباهت متنی با متن ورودی دارند بعنوان مجموعه پایه انتخاب می‌شوند سپس همراه با مراجع خود بعنوان مجموعه کاندید انتخاب می‌شوند و در گام دوم این اسناد بر اساس یک ترکیب خطی از معیارهای ذکر شده در جدول ۲-۱ مرتب می‌شوند.

آنها روش خود را با روش پایه‌ای که تنها از شباهت متنی استفاده می‌کرد مقایسه کردند و نشان دادند که استفاده از ویژگی‌های مبتنی بر استناد در کنار ویژگی‌های مبتنی بر متن می‌تواند مفید باشد، همچنین اثر وجود یا عدم وجود هر یک از ویژگی‌های مبتنی بر استناد را بررسی کردند و به این نتیجه رسیدند که از بین ویژگی‌های مختلف مبتنی بر استناد معیار *Katz* (Liben-Nowell and Kleinberg, 2003) از اهمیت بالاتری برخوردار است.

جدول ۲-۱: لیست ویژگی‌های بکار برده شده در مقاله (Strohman et al., 2007)

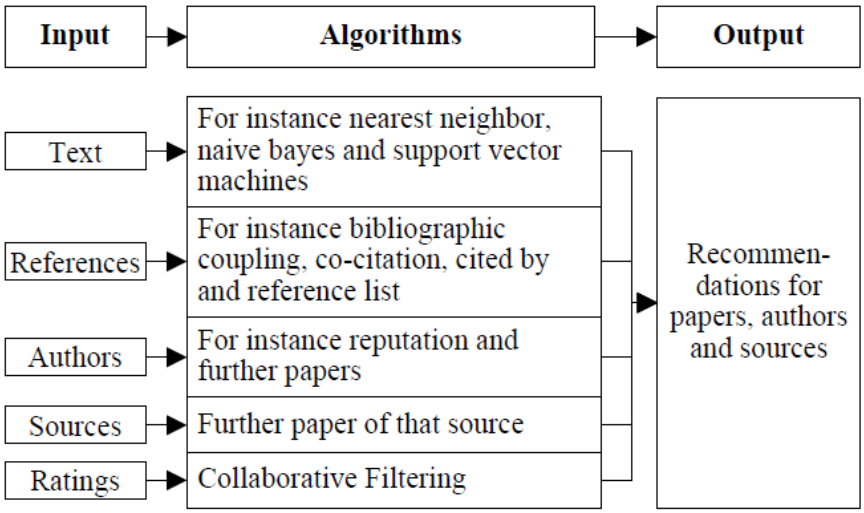
Publication Year	سالی که مقاله منتشر شده است (نرمال شده به‌وسیله کاهش از ۱۹۵۰)
Text Similarity	شباهت متن مقاله و متن ورودی که بر اساس یک کرنل چندجمله‌ای (Lafferty and Lebanon, 2005) است.
Co-citation Coupling	تعداد استنادها مشترک مقاله کاندید و هر مقاله موجود در مجموعه پایه
Same Author	داشتن نویسنده مشترک با متن ورودی
Katz	معیار فاصله در گراف
Citation count	تعداد دفعاتی که یک مقاله توسط بقیه مقالات موجود مورد استناد قرا می‌گیرد

گیپ و همکارانش در (Gipp et al., 2009) به معرفی سیستم *sceinstein* پرداختند و ادعا کردند که این سیستم اولین سیستم سفارش‌دهنده مقاله‌های تحقیقاتی از نوع ترکیبی می‌باشد و می‌تواند یک جایگزین قوی برای موتورهای جستجوی دانشگاهی فعلی باشد. *scienstein*، روش رایج جستجوی مبتنی بر کلمه کلیدی را با تکنیک‌های تحلیل استناد، تحلیل نویسنده[۱]، تحلیل منبع[۲]، نرخ دهی صریح[۳]، نرخ دهی ضمنی[۴]، و همچنین تکنیک‌های جدیدی که تا کنون استفاده نشده‌اند، مانند اندیس تشابه فاصله[۵](*DSI*) و فاکتور تاثیر در

[1] Author analysis
[2] Source analysis
[3] Explicit rating
[4] Implicit ratings
[5] Distance Similarity Index(DSI)

متن[1](*ItIF*) ، ترکیب کرده و در مجموع، کیفیت جستجو را بهبود می‌دهند. عملکرد آن‌ها در این مقاله برای استفاده از هر یک از این تکنیک‌ها در ذیل به اختصار توضیح داده شده است.

عموما سیستم‌های جستجوی دانشگاهی مبتنی بر جستجوی کلمه کلیدی در اسناد می‌باشند در نتیجه با توجه به کلمات هم‌معنی [2] و برخی ابهام‌ها و معانی مختلف لغات، این روش کارایی مناسبی ندارد. همانطور که در شکل ۲-۱ نمایش داده شده است، کاربر *Scienstein* محدود به وارد کردن کلمه کلیدی نیست و می تواند هر کدام از ۶ نوع ورودی را انتخاب کند.



شکل ۲-۱: روش *Scienstein* برای پیشنهاد مقاله (Gipp et al., 2009)

این سیستم پس از پیدا کردن مقالات مرتبط با مقاله ورودی، برای مرتب کردن آن‌ها از دو فاکتور *ICFA* و *ICDA* استفاده می‌کند. *ICFA* بررسی می‌کند که اگر مقاله *X* در مقاله *Y* مورد استناد قرار گرفته، این استناد ها با چه فرکانسی صورت گرفته است (به بیان ساده، چند بار). برای محاسبه این مقدار، در (Gipp et al., 2009) معیار *ItIF* ارائه شده‌است که مثلا *ItIF(X)* در مقاله *Y* برابر است با تعداد دفعاتی که مقاله *X* در مقاله *Y* مورد استناد قرار گرفته، تقسیم بر تعداد کل استنادهایی که در مقاله *Y* صورت گرفته است. هرچه *ItIF(X)* بیشتر باشد، بدین معناست که ارتباط مقاله *X* با مقاله *Y* قابل توجه تر می باشد.

[1] In-text Impact Factor(ItIF)
[2] Synonym

ICDA بدین ترتیب عمل می‌کند که فاصله بین استنادها را بعنوان نشان‌دهنده میزان ارتباط معنایی مقالات در نظر می‌گیرد. ایده این است که اگر دو مقاله *x1* و *x2* هر دو در مقاله *Y* مورد استناد قرار گرفته باشند، هرچه فاصله این دو استناد کمتر باشد، میزان شباهت و ارتباط *x1* و *x2* بیشتر است. در (Gipp et al., 2009) برای محاسبه آن، معیار *DSI* را ارائه شده است که به محاسبه شباهت دو مقاله بر اساس فاصله استناد به آن‌ها، می‌پردازد. اگر استناد به آن دو مقاله، در یک جمله آمده باشد، شباهت شان خیلی زیاد و مقدار *DSI* برابر ۱ است. اگر استناد به آن دو مقاله، در یک پاراگراف باشد، شباهت‌شان زیاد و *DSI* برابر ۰.۵ است

آزمایش‌های اولیه نشان داده که استفاده از معیارهای *ICDA* و *ICFA* کارایی سیستم را بهبود می‌دهد.

در این مقاله علاوه بر اینکه استنادهای داخل مقالات مورد توجه قرار می‌گیرند، کاربر خودش می‌تواند برای مقالاتی که در سیستم مشاهده می‌کند، مقالات مشابهی را پیشنهاد کند و به نوعی مقالات را به مقالات دیگر لینک کند. به این لینک‌ها، پیوندهای همبستگی[1] گفته می شود.

از دیدگاه متن‌کاوی نیز، Scienstein از تکنیک‌های موجود استفاده می‌کند و قابلیت‌های جدیدی هم ارائه می‌کند. مثلا تمام مقالاتی که توسط یک سازمان خاص مورد حمایت قرار گرفته‌اند یا تمام مقالاتی که مربوط به یک پروژه خاص می‌باشند را در یک کلاس قرار دهد.

علاوه بر این‌ها، Scienstein از تگ‌هایی که کاربران بر روی مقالات (مثلا در حین خواندنشان) ایجاد می‌کنند نیز استفاده می‌کند. کاربران می‌توانند مقالات را طبقه‌بندی کنند و این طبقه‌بندی‌ها در تحلیل ها و سفارش‌های سیستم مورد استفاده قرار می‌گیرند.

Scienstein علاوه بر اینکه امکان نرخ‌گذاری صریح را برای کاربران فراهم می‌کند، با مانیتور کردن فعالیت های آنها، نرخ‌گذاری ضمنی را هم پشتیبانی می کند. برای نرخ‌گذاری ضمنی، سیستم فعالیت کاربران را مانیتور می‌کند. ایده این است که مقالات یا قسمت‌هایی از مقالات که کاربر بیشتر روی آنها مکث و بیشتر وقت صرف می‌کند، از اولویت بیشتری برخوردارند، مثلا در مقایسه با مقالاتی که کاربر به محض باز کردن آنها دوباره

[1] Collaborative link

می‌بند. لازم به ذکر است که، مقاله *(Gipp et al., 2009)*، سیستم Scienstein را ارزیابی نکرده است و این سیستم تنها معرفی شده‌است و بیان شده است که این سیستم در حال توسعه است.

بتارد و همکارانش در (Bethard et al, 2010) برای پیشنهاد مقالات، بر اساس ویژگی‌هایی نظیر تاریخ انتشار، اعتبار نویسنده، متن، و موضوع، معیارهای مختلفی برای شباهت تعریف کرده‌اند، آن‌ها شباهت متن ورودی و هر سند را برابر میانگین وزن‌دار هر یک از این شباهت‌ها در نظر گرفته‌اند. سپس اسناد با بیشترین مقدار شباهت را بعنوان استناد پیشنهاد کرده‌اند. در روش پیشنهادی این مقاله، وزن هر ویژگی به کمک یک الگوریتم تکراری محاسبه شده است.

بتارد و همکارانش روش پیشنهادی خود را با روش پایه که تنها از شباهت‌های متنی استفاده می‌کند، مقایسه کردند و نشان دادند که روش آن‌ها کارایی بیشتری دارد.

دسته دوم:

دسته دوم کارهایی می‌باشند که علاوه بر یک متن تعدادی مقاله مرتبط با آن متن را نیز که توسط کاربر مشخص شده است، می‌گیرند و هدف آن‌ها کامل کردن این مقالات می‌باشد.

ودرووف و همکارانش در (Woodruff et al., 2000) برای ایجاد یک پیشنهاددهنده برای خواننده‌هایی با پیش‌زمینه و دانش‌های مختلف، از روش گسترش فعال‌سازی (Salton and Buckley, 1988) استفاده کردند. آن‌ها با ترکیب تحلیل استناد و اطلاعات متنی توانستند به خواننده، مقاله بعدی را که باید از بین ۴۳ مقاله موجود در کتاب الکترونیکی مطالعه کند را پیشنهاد دهند. در این سیستم، سناریوها توسط کاربران با اهداف و تجربیات پژوهشی مختلف ایجاد می‌شود، همچنین لیست مقالاتی که آن‌ها دوست دارند نیز به صورت صریح مشخص می‌شود.

این سیستم برای داشتن کارایی نیاز به دو شرط دارد: ۱) کاربر باید توانایی این را داشته باشد که مقاله ریشه[1] را مشخص کند، چون سیستم پیشنهاد دهنده از آن مقاله شروع خواهد کرد و سپس از آن‌جا مکانیزم

[1] Seed

گسترش فعال‌سازی را بکار می‌گیرد. ۲) کاربر باید انگیزه و هدف مشخصی داشته باشد و قادر باشد متن ورودی به خوبی آماده کند.

سیستم پیشنهاد مقاله‌ای که در (Woodruff et al., 2000) پیشنهاد شده است، برای استفاده دانشجویان مبتدی غیرکاربردی است، چرا که انجام دو شرط بالا توسط آن‌ها به خوبی انجام نخواهد گرفت.

یکی از روش‌های مورد استفاده در این دسته، روش فیلتر همبستگی[1] می‌باشد (Schafe et al., 2007; Adomavicius and Tuzhilin, 2005). در ادامه کارهایی که از این روش در سیستم-های خود استفاده می‌کنند شرح داده شده است.

سین مسنی[2] و همکارانش در (McNee et al. 2002) روش فیلتر همبستگی را به جای اعمال روی شبکه اجتماعی از افراد روی شبکه اجتمایی از مقالات پژوهشی اعمال کردند، در نتیجه توانستند از آن برای کاربرد پیشنهاد مقالات پژوهشی استفاده کنند. لازمه استفاده از این روش، داشتن یک ماتریس نرخ‌گذاری‌شده است، در (McNee et al. 2002) برای ایجاد این ماتریس از گراف استناد استفاده شده است. در نتیجه با استفاده از آن برای ساخت ماتریس نرخ‌گذاری دیگر محدودیت هنگام راه‌اندازی در روش فیلتر همبستگی در این کاربرد وجود ندارد.

در این مقاله به منظور اعمال روش فیلتر همبستگی روی شبکه اجتماعی از مقالات پژوهشی، چندین روش برای ساخت جدول نرخ‌گذاری معرفی شده است به عبارتی با توجه به ساختار جدول نرخ‌گذاری که ردیف-های آن کاربران و ستون‌های آن آیتم‌ها می‌باشند جایگزین‌های مختلفی برای ردیف و ستون آن برای استفاده در این کاربرد پیشنهاد شده است: در روش اول که قابل قیاس با MovieLens و بقیه سیستم‌های پیشنهاد رایج می‌باشد به جای آیتم‌ها از استنادها و به جای کاربران از مردم واقعی[3] که نرخ استنادها را در ماتریس مشخص می‌کنند، استفاده شده است. این روش از گراف استناد استفاده نمی‌کند در نتیجه محدودیت راه اندازی موجود

[1] Collaborative filtering
[2] Sean M.McNee
[3] Real people

در روش فیلتر همبستگی، وجود خواهد داشت. برای حذف این محدودیت، سین مسنی و همکارانش روش‌های دیگری که مستقیما از وب استناد استفاده می‌کنند را معرفی کردند.

روش پیشنهادی بعدی، به‌جای آیتم‌ها از استنادها و به‌جای کاربران از نویسندگان مقالات استفاده می‌کند. در این روش برای نرخ‌دهی به ماتریس هر نویسنده‌ای به کارهایی که به آن‌ها استناد کرده است رتبه می‌دهد. این روش به دلیل استفاده از وب استناد محدودیت راه اندازی را ندارد. این روش دارای مشکل عمومیت[1] است: تعداد زیادی از نویسندگان، مقاله‌هایی در زمینه‌های مختلفی دارند. برای مثال بن شریدرمن مقالات زیادی در زمینه‌های تعامل بین کامپیوتر و انسان، و طراحی واسط کاربر نوشته است، اما مقالاتی نیز در مورد *FORTRAN* و چندین مقاله در ارتباط با میراث یهود و تاریخ نیز دارد. از آنجایی که نویسندگان زیادی مانند بن شریدرمن دارای مقاله‌های فراوان در زمینه‌های مختلف می‌باشند، اگر یک کاربر لیستی از مقالات را با موضوع واسط کاربر بعنوان ورودی ارائه کند و یک پیشنهاد برای مقالات بن شریدرمن دریافت کند که معقول نیز به نظر می‌رسد، تعدادی از مقالات پیشنهاد شده در زمینه واسط کاربر نخواهند بود و بی ربط با خواسته کاربر می‌باشند.

روش پیشنهادی بعدی به جای آیتم‌ها از استنادها و به جای کاربران از مقالات استفاده می‌کند، هر مقاله‌ای نرخ استنادهایی که در فهرست مطالبش وجود دارد را مشخص می‌کند. این روش نیز به دلیل استفاده از وب استناد محدودیت راه اندازی را ندارد، مشکل روش نویسنده-استناد نیز در این روش وجود ندارد. برخلاف روش‌های معمول در فیلتر همبستگی، که یک کاربر در طی زمان می‌تواند به آیتم‌های دیگری نیز نرخ دهد، در این روش هر مقاله بعد از این‌که برای اولین بار در جدول وارد می‌شود از آنجایی که فهرست مراجع آن ثابت است نرخ جدیدی در ماتریس ایجاد نمی‌کند. روش پیشنهادی آخر به جای کاربران و آیتم‌ها از استنادها استفاده می‌کند در نتیجه برای نرخ گذاری هر وارده از ماتریس از معیار استناد همزمان استفاده می‌کند. این معیار با روش فیلتر همبستگی آیتم-آیتم مرتبط است.

سین مسنی و همکارانش در (McNee et al. 2002) چهار الگوریتم مبتنی بر فیلتر همبستگی پیشنهاد کردند، ورودی تمام این الگوریتم‌ها مجموعه‌ای از استنادها و خروجی آن‌ها لیست مرتب شده‌ای از استنادهای

[1] Generality

پیشنهاد شده است. آن‌ها به منظور مقایسه با این الگوریتم‌ها دو روش دیگر که مبتنی بر فیلتر همبستگی نیستند نیز معرفی کردند. در این دو روش از چکیده و عنوان مقالات نیز برای جستجو استفاده می‌شد.

آن‌ها برای ارزیابی کارایی الگوریتم‌های خود از دو روش آزمایش دستی و خودکار استفاده می‌کنند، در آزمایش خودکار یکی از استنادها از مجموعه ورودی حذف می‌شود و قرار است توانایی الگوریتم‌ها برای پیدا کردن استناد حذف شده سنجیده شود. در آزمایش دستی، نویسنده هر مقاله بعنوان داور آن مقاله، سوالاتی از پیش تعیین شده را در ارتباط با هر پیشنهاد سیستم پاسخ می‌گویند تا میزان مرتبط بودن[۱] آن پیشنهادات با مقاله هدف و جدید بودن[۲] مقالات نسبت به مراجع آن مقاله سنجیده شود.

سین مسنی و همکارانش در (McNee et al. 2002) با توجه به نتایج بدست آمده از آزمایشات به این نتیجه رسیدند که با توجه به کاربردهای مختلف باید از روش‌های مختلف استفاده شود. مثلا برای پیدا کردن مراجع جدید برای کامل کردن یک فهرست مطالب کاربران ترجیح می‌دهند که پیشنهادات بیشتر مرتبط باشند در نتیجه پیشنهاد کننده گوگل بهترین روش است و از آنجایی که شکست این روش زیاد است دومین الگوریتم از نظر شبیه ترین یعنی بیزین ساده، پیشنهاد می شود.

روش استفاده شده توسط تورس و همکارانش در (Torres et al., 2004) از آن جهت که تکنیک اصلی آن ۱)تحلیل استناد و استناد همزمان و ۲)روش‌های مبتنی بر محتوای مقاله، می‌باشد، شبیه به (McNee et al. 2002) می‌باشد. تورس برای پیشنهاد مقالات پژوهشی، آن‌ها را به چهار دسته اصلی تقسیم می‌کند : جدید[۳]، معتبر[۴]، معارفه‌ای[۵] و مطالعه اجمالی[۶]. سپس به صورت تجربی نشان داد که برای دسته‌های مختلف از مقالات باید الگوریتم‌های پیشنهاددهی مختلفی استفاده شود. الگوریتم‌های پیشنهاددهی که در این مقاله استفاده شده است عبارتند از فیلتر همبستگی که از تحلیل استناد و استناد همزمان برای پیشنهاد دادن استفاده می‌کند، فیلتر مبتنی بر محتوی که تنها بر اساس چکیده و عنوان مقاله پیشنهاد می‌دهد و فیلتر ترکیبی که ترکیبی از دو روش ذکر شده است.

[1] Relevance
[2] Novelty
[3] Novel
[4] Authoritative
[5] Introductory
[6] Survey

تورس و همکارانش در (Torres et al., 2004)، برای هر دسته از مقالات، الگوریتم‌ها را به ترتیب خوب جواب دادن در آن دسته مرتب می‌کنند. نتایج بدست آمده از آزمایشات آن‌ها از دید مقالات، نتیجه‌گیری شده است و به ارتباط بین نوع کاربران و کلاس مقالات توجهی ندارد. به عبارت دیگر، آن‌ها نشان نمی‌دهند که چه نوعی از مقالات توسط چه نوعی از کاربران بیشتر مورد علاقه هستند و این برای پیشنهاد مقالات پژوهشی از اهمیت بالایی برخوردار است. در حقیقت در ارزیابی‌های انجام شده مشخص شد که افراد حرفه‌ای به اندازه‌ای که دانشجویان از نتایج پیشنهاد این سیستم راضی می‌باشند راضی نیستند و تعداد زیادی از مقالات پیشنهاد داده شده را خود مطالعه کرده‌اند.

دسته سوم:

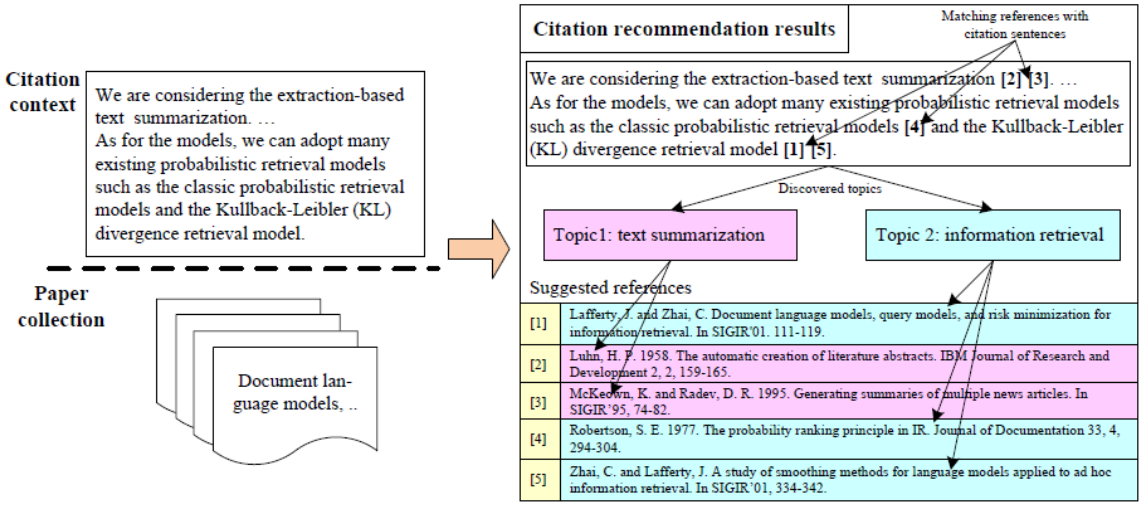
هدف کارهای متعلق به دسته سوم، پیدا کردن مقالات مرتبط با قسمت خاصی از آن متن است، در نتیجه در این سیستم‌ها علاوه بر پیشنهاد مقاله، مکان مورد نظر آن‌ها در متن نیز مشخص می‌شود.

تنگ و ژنگ در(Tang and Zhang, 2009)، برای حل مساله پیشنهاد استناد، از روش‌های ابتکاری استفاده نکردند، آن‌ها به کمک پردازش ماشین بلتزمن محدود به دو لایه[1]، یک فرمول‌بندی برای حل مساله پیشنهاد کردند. دو فرق اساسی بین راه‌حل پیشنهادی آن‌ها و راه‌حل‌های مقالات دیگر که قبل از این مقاله ارائه شده‌اند عبارتند از:۱) در این مقاله نه‌تنها لیستی از مقالاتی که باید استناد شوند را پیشنهاد می‌شود بلکه زیر موضوع‌های موجود در مقاله نیز مشخص می‌شود.۲) مقالات قبلی تنها قادرند به طور کلی مقاله پیشنهاد دهند اما قادر به مشخص کردن مکان استناد در متن نمی‌باشند. در این مقاله جمله‌ای که در متن متعلق به آن مرجع است نیز مشخص می‌شود.

تنگ و ژنگ در (Tang and Zhang, 2009) مساله پیشنهاد استناد را به سه زیر مساله تقسیم کرده‌اند: ۱)کشف عناوین، ۲) پیشنهاد تعدادی مقاله برای هر عنوان، ۳) منطبق کردن مقالات پیشنهاد شده با جملات استناد.

[1] Two-layer Restricted Boltzmann Machine model

برای مثال همانطور که در شکل۲-۲، نشان داده شده است، سیستم ابتدا دو زیر عنوان بازیابی اطلاعات و خلاصه سازی متن[1] را مشخص کرده است، سپس برای عنوان اول دو مقاله و برای عنوان دوم، سه مقاله پیشنهاد کرده است و در نهایت مکان این مراجع در متن مشخص شده است.



شکل۲-۲: مثالی از یک پیشنهاد استناد (Tang and Zhang, 2009)

قبل از توضیح نحوه عملکرد این مقاله در هر یک از زیر مسئله‌های تعریف شده، فرضیات ذیل مشخص شده است:

۱- هر مقاله شامل یک بردار W_d از N_d کلمه می‌باشد به طوری که هر کلمه w_{di} از یک واژگان V کلمه‌ای انتخاب شده است، همچنین هر مقاله دارای یک لیست l_d از L_d مرجع نیز می‌باشد. در نتیجه مجموعه D مقاله به صورت $D = \{(W_1, l_1), \dots, (W_D, l_D)\}$ نمایش داده می‌شود.

۲- مراجعی که در مجموعه مقالات موجود در D وجود ندارند از لیست مراجع هر مقاله حذف می‌شوند. در نتیجه $L=D$.

۳- بعلاوه در نظر گرفته شده است که T عنوان در کل مقالات وجود دارد.

نحوه عملکرد این مقاله برای حل مساله پیشنهاد استناد شامل سه مرحله زیر است:

۱- ابتدا ماشین بولتزمن محدود به دو لایه همان طور که در شکل۲-۳ نشان داده شده است تشکیل می‌شود، این ماشین شامل یک لایه مخفی برای عناوین می‌باشد که دو لایه دیگر را به

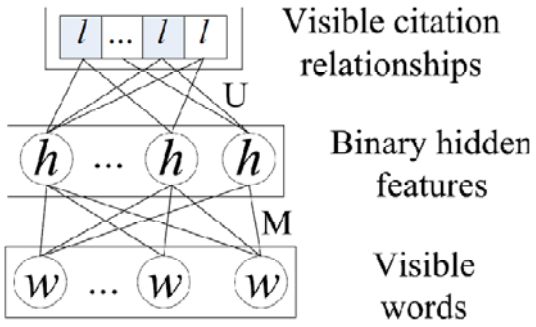
[1] Text summarization

هم متصل می‌کند. دو لایه دیگر عبارتند از: لایه کلمات موجود در مجموعه اسناد که به تعداد کل واژگان یعنی V است و لایه مرجع‌های موجود در مجموعه مقالات که همانطور که ذکر شد به تعداد D است. در این ماشین ماتریس M وزن رابطه‌های بین دو لایه wو h را نشان می‌دهد و ماتریس U وزن رابطه‌های بین h وl را نشان می‌دهد

سپس این ماشین بکمک یک الگوریتم تکراری تعلیم می‌بینند.

۲- در این مرحله ابتدا متن استناد به ماشین تعلیم یافته اعمال می‌شود و عناوین موجود در آن متن پیدا می‌شود، سپس با توجه به هر عنوان خاص بدست آمده، مراجع مربوط به آن موضوع‌ها پیشنهاد می‌شوند.

۳- هدف این مرحله مرتبط کردن مقالات پیشنهاد شده در مرحله قبل با جملات موجود در متن استناد است. برای رسیدن به این هدف، هر مقاله پیشنهاد داده شده بعنوان یک پرس‌وجو در نظر گرفته می‌شود و جمله استناد مرتبط با آن به کمک یک روش بازیابی اطلاعات پیدا می‌شود.



شکل۲-۳: نمایش گرافیکی مدلRBM-CS (Tang and Zhang, 2009)

تنگ و ژنگ نشان دادند استفاده از ماشین بولتزمن می‌تواند به خوبی در حل این مساله کمک کند.

هی و همکارانش در (He et al., 2010) برای حل مساله پیشنهاد استناد یک مدل احتمالی بدون پارامتر برای بدست آوردن رابطه مبتنی بر متن بین یک مقاله و یک متن استناد معرفی‌کردند، سیستم پیشنهادی آن‌ها قادر است در دو سطح سراسری و محلی پیشنهاد دهد.

قبل از توضیح نحوه عملکرد این مقاله در هر یک از سطوح پیشنهاد، تعاریف بکار رفته در این مقاله در زیر آورده شده است: اگر فرض شود d یک سند و D مجموعه اسناد موجود است، در سند d، متن c مجموعه‌ای

از کلمات است. متن سراسری[1]، عنوان و چکیده سند d و متن محلی[2]، متنی است که در اطراف محل استناد قرار دارد. همچنین اگر سند d_1 به سند d_2 استناد کند، متن استناد در d_1، متن خارج از پیوند[3] نسبت به سند d_1، و متن داخل پیوند[4] نسبت به سند d_2 گفته می‌شود.

ورودی سیستم پیشنهادی در این مقاله در سطح سراسری سند d شامل متن سراسری آن سند و مجموعه متن‌های خارج از پیوند آن است (ورودی شامل مراجع مقاله نمی‌شود). و خروجی آن لیستی مرتب شده از استنادهایی است که در D وجود دارند و بعنوان لیست مراجع پیشنهاد می‌شوند.

ورودی سیستم پیشنهاد در سطح محلی یک متن داخل پیوند از یک سند است و خروجی مورد انتظار از سیستم لیست مرتب شده‌ای از مقالات موجود در D و مرتبط با آن متن داخل پیوند است.

هی و همکارانش در (He et al., 2010) برای تولید به سیستم پیشنهادی خود بعد از مرحله پیش-پردازش که در آن متن سراسری و متن محلی خارج از پیوند (۵۰ کلمه قبل و بعد از مکان استناد) آن استخراج می‌شود دو مرحله را انجام دادند: ۱) تولید مجموعه کاندید و۲) مرتب کردن عناصر مجموعه کاندید.

در این مقاله برای بدست آورن مجموعه کاندید دو دسته روش معرفی کرده است که در ذیر شرح مختصری از آن‌ها آورده شده است:

a- روش‌های بی‌توجه به متن: در این روش که به متن محلی توجهی نمی‌شود مجموعه کاندید می‌تواند به روش‌های زیر ایجاد شود.

۱. *GN*: n تا سندی که عنوان و چکیده آن‌ها بیشترین شباهت را با d_1 دارد.

۲. *Auther*: اسنادی که نویسنده مشترک با d_1 دارند.

۳. *CiteHop*: مقالاتی که بوسیله مقالات موجود در مجموعه کاندید که به یکی از دو روش بالا بدست آمده‌اند مورد استناد قرار گرفته‌اند.

[1] Global context
[2] Local context
[3] Out-link context
[4] In-link context

۴. *AuthHop*: اسنادی که به‌وسیله نویسنده‌های موجود در مجموعه کاندید نوشته شده‌اند.

معایب روش‌های بی‌توجه به متن، استفاده از تمام متن برای بدست آوردن شباهت متون با هم کار زمان بری است. استفاده از چکیده و عنوان به‌تنهایی نیز از آنجایی که خیلی از مباحث موجود در مقاله در آن‌ها آورده نشده است جواب دقیقی ایجاد نمی‌کند. همچنین با استفاده از روش‌های مبتنی بر نویسنده از آن‌جا که ممکن است یک نویسنده در زمینه‌های گسترده‌ای فعالیت کرده باشد مقالات بی ربطی را در مجموعه کاندید قرار می‌دهد که در روش آخر نیز یعنی (*AuthHop* و*CiteHop*) به این بی-ربطی‌ها دامن زده می‌شود.

b- روش‌های آگاه از متن: در این روش‌ها مقالاتی که انتخاب می‌شوند با توجه به متن محلی می-باشند. در این‌روش برای هر متن محلی *c* مجموعه کاندید می‌تواند به یکی از روش‌های زیر ایجاد شود:

۵. *LN*: *n* مقاله‌ای که متن داخل پیوند آن‌ها بیشترین شباهت را با *c* دارند.

۶. *LNC*: *n* مقاله‌ای که متن خارج از پیوند آن‌ها بیشترین شباهت را با *c* دارند. (این مقاله‌ها به مقاله‌های بدست آمده از روش قبل استناد می‌کنند).

شش روش ذکر شده در بالا می‌توانند با هم ترکیب شوند و روش‌های جدید بوجود آورند مثلا یکی از ترکیب‌ها می‌تواند*(L100+CiteHop)+G100* باشد.

پس از بدست آوردن مجموعه کاندید برای مرتب کردن مقالات بدست آمده، با توجه به سطح پیشنهاد؛ یعنی پیشنهاد سراسری و پیشنهاد محلی، دو فرمول پیشنهاد کردند که مبتنی بر شباهت‌های متنی بود.

هی و همکارانش در (He et al., 2010) برای ارزیابی سیستم پیشنهادی خود، ابتدا روش‌های ترکیبی مختلفی که برای ایجاد مجموعه کاندید را امتحان کردند و نتیجه گرفتند که LC100+G1000 به یک مجموعه با تعداد مقالات کمتر و دقت بیشتر می‌رسد.

سپس در دو سطح پیشنهاد دهی سیستم خود را آزمایش کردند در سطح سراسری نشان دادند که روش‌پیشنهادی آن‌ها نسبت به روش‌هایی که از قبل وجود دارد از جمله (Strohman et al., 2007) و (Tang and Zhang, 2009)دقت بالاتری دارد همچنین در سطح محلی از آن‌جایی که کاری برای مقایسه وجود نداشت

دو حالتی که ورودی تنها یک متن استناد باشد و وقتی که بعنوان ورودی، خود مقاله هم وارد شود را با هم مقایسه کردند و مشاهده کردند که نتایج حالت دوم بهتر است.

آن‌ها در (He et al., 2011) کار خود را گسترش دادند بطوری که یک مرحله به کار آن‌ها اضافه شد. در (He et al., 2010) ورودی آن‌ها در پیشنهاد متنی بود که قسمت‌هایی که قرار بود برای آن‌ها استناد پیشنهاد شود با علامت "*[?]*" توسط کاربر مشخص شده بود، اما مرحله‌ای که در (He et al., 2011) اضافه شد، سبب شد که ورودی سیستم تنها یک متن باشد و خود الگوریتم مکان‌های مناسب برای استناد دادن را مشخص کند.

۲-۲-۲-۲ سیستم‌های پیشنهاد استناد: معیارهای ارزیابی

بدیهی است که یک سیستم بازیابی اطلاعات زمانی موفق است که تمام اسنادی که مرتبط با یک پرس‌وجو[1] می‌باشند بازیابی شوند و اسناد نامرتبط در لیست بازیابی شده وجود نداشته باشند. ارزیابی پیشنهادات سیستم‌های پیشنهاد استناد بعنوان یک سیستم در زمینه بازیابی اطلاعات اغلب به دو دسته "دستی" و "خودکار" دسته‌بندی می‌شوند:

در ارزیابی به صورت خودکار متن یک مقاله واقعی بعنوان ورودی به سیستم داده می‌شود و خروجی مورد انتظار لیست مراجع آن مقاله می‌باشد؛ ولی در ارزیابی دستی انسان‌ها بعنوان ارزیاب، استنادهای پیشنهاد شده توسط سیستم را ارزیابی می‌کنند.

کارهای مختلف از جمله *(Bethard et al., 2010; Strohman et al., 2007; Tang and Zhang 2009)* برای ارزیابی خودکار سیستم‌های پیشنهادی خود در زمینه سیستم پیشنهاد استناد، از معیارهایی مثل فراخوانی، *P@n* میانگین دقت متوسط (MAP)، (Buckley and Voorhee, 2004) استفاده می‌کنند.

عیب استفاده از این معیارها در ارزیابی خودکار این است که سیستم ممکن است استنادهایی را برای متن یک مقاله ورودی پیشنهاد کند که مرتبط با آن متن باشند، اما به دلیل وجود نداشتن در لیست مراجع آن مقاله نامرتبط شناخته شوند. در نظر گرفتن تنها شرط قرار داشتن در لیست مراجع یک مقاله بعنوان معیار مرتبط بودن به آن، ارزیابی کاملی نمی‌باشد. ممکن است نویسنده یک مقاله، تعدادی از مراجع خود را بدلیل

[1] Query

محدودیت فضا و یا مشابه بودن با دیگر مراجع موجود در مقاله، اشاره نکند و یا حتی ممکن است نویسنده یک مقاله از تعدادی از کارهای مرتبط آگاهی کافی نداشته باشد. هدف سیستم پیشنهاد استناد بهبود و کمک به کار نویسندگان در پیدا کردن کارهای مرتبط است و اگر در ارزیابی این سیستم‌ها مجددا با مراجع یک مقاله مقایسه شود مقایسه کاملی نخواهد بود.

برای ارزیابی چنین خروجی‌هایی در بعضی از کارها از ارزیابی دستی علاوه بر ارزیابی خودکار استفاده شده است (McNee et al., 2002; Strohman et al., 2007; Torres et al., 2004). در ارزیابی دستی، پرشنامه‌هایی فراهم می‌شود و مثلا نویسنده مقاله ورودی بعنوان ارزیاب، مرتبط بودن استنادهای پیشنهاد داده شده برای آن مقاله را ارزیابی می‌کند. مشکل روش‌های دستی محدودیت انجام آن در حجم وسیع و نبودن معیار یکسان برای مقایسه روش‌های مختلف می‌باشد.

برای برطرف کردن این مشکلات در (He et al., 2010; He et al., 2011) معیاری بعنوان *احتمال استناد مشترک*[1] معرفی شد که با استفاده از آن می‌توان برای ارزیابی سیستم از روش‌های خودکار استفاده کرد و تا حدودی مشکلات ناشی از آن‌ها را برطرف کرد.

در این معیار دیگر پیشنهادهایی که در لیست مراجع مقاله ورودی نمی‌باشند نامرتبط شناخته نمی‌شوند و میزان احتمال با هم مورد استناد قرار گرفته شدن آن‌ها با مراجع آن مقاله بعنوان میزان مرتبط بودن آن پیشنهاد با متن ورودی در نظر گرفته می‌شود.

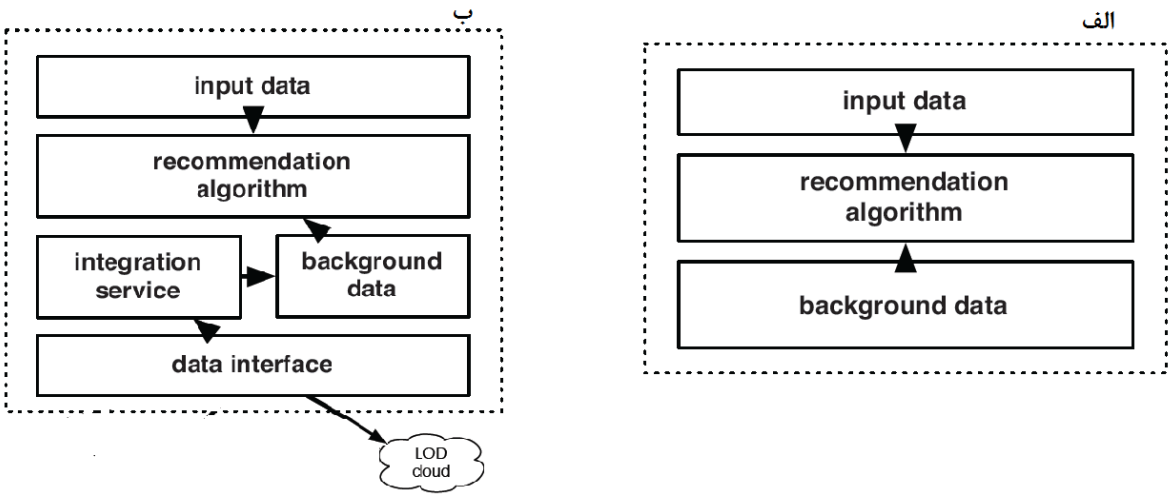
از آنجاییکه مفید بودن سیستم‌های پیشنهاددهنده به ترتیب قرار گرفتن پیشنهادها نیز وابسته است، و استفاده از معیار میانگین دقت متوسط که قادر است سیستم را از این لحاظ ارزیابی کند دارای مشکلات اشاره شده در بالا می‌باشد، در (He et al., 2010; He et al., 2011) از معیار *NDCG* استفاده شده است، در این معیار از آنجاییکه میزان مرتبط بودن یک پیشنهاد به خروجی‌های مورد انتظار دودویی در نظر گرفته نمی‌شود قادر است برای ارزیابی در سیستم‌های پیشنهاد استناد بکار رود.

[1] Co-cited probability

۲-۳ سیستم‌های پیشنهاد دهنده مبتنی بر داده‌های پیوندی

نسل جدیدی از سیستم‌های پیشنهاد دهنده در سال ۲۰۰۹ توسط الکساندر پسنت در (Passant et al., 2009) مطرح شد. در این سیستم‌ها، معماری سیستم‌های پیشنهاد دهنده، از شکل۲-۴ قسمت الف به قسمت ب، تغییر یافت. به عبارتی در دیدگاه جدید، داده‌های پیوندی جایگزین داده‌های پیش-زمینه در سیستم‌های پیشنهاد دهنده فعلی شدند. این ایده در سال ۲۰۱۰ برای پیشنهاد موسیقی مورد استفاده قرار گرفت (Heitmann and Hayes, 2010; Passant et al., 2010).

در (Heitmann and Hayes, 2010) چگونگی رفع مشکلات روش فیلتر همبستگی با استفاده از داده‌های پیوندی بیان شد و سپس برای نشان دادن صحت این ادعا، سیستم پیشنهاد دهنده‌ای برای موسیقی پیاده‌سازی شد. در (Passant et al., 2010) نیز یک سیستم پیشنهاد دهنده موسیقی به نام *dbrec*، با استفاده از داده‌های پیوندی *DBpedia* پیاده‌سازی شد. در این سیستم، از فاصله معنایی داده‌های پیوندی به منظور محاسبه شباهت بین دو داده استفاده شده است و چالش‌های استفاده از داده‌های پیوندی در برنامه‌های کاربردی مشابه توضیح داده شد.



شکل۲-۴: الف) سیستم پیشنهاد دهنده رایج، ب) سیستم پیشنهاد دهنده مبتنی بر داده‌های پیوندی (Heitmann and Hayes, 2010)

شبیر و کلارک نیز در (Shabir and Clarke, 2009) از داده‌های پیوندی بعنوان پایه‌ای برای پیشنهاد منابع آموزشی در محیط‌های آموزش الکترونیکی بهره گرفتند و پیرامون موضوع‌هایی مثل منشاء[۱] و مجوز[۲] داده‌های منتشر شده بر روی *ابر LOD*[۳] بحث کردند.

موضوع داده‌های پیوندی که توسط تیم برنرزلی در سال ۲۰۰۶ مطرح شد (Berners-Lee, 2006). در واقع مجموعه‌ای از تجربیات خوب[۴] برای انتشار داده‌ها بر روی وب، و همچنین ایجاد پیوندهای معنادار بین این داده‌ها می‌باشد. قواعد داده‌های پیوندی که قواعدی برای انتشار داده‌ها بر روی وب می‌باشند توسط تیم برنرزلی به صورت زیر بیان شد:

- استفاده از *URI* ها بعنوان نام برای اشیاء.
- استفاده از *HTTP URI* ها، برای جستجوی این نام‌ها توسط افراد.
- وجود اطلاعات مفید هنگام بازیابی هر *URI*.
- وجود پیوندهایی به *URI*های دیگر، به منظور کشف اطلاعات بیشتر.

مهمترین نمونه تطبیق و تحقق قواعد داده‌های پیوندی را می‌توان در پروژه LOD[۵] مشاهده کرد، که در ژانویه ۲۰۰۷ و تحت حمایت کنسرسیوم W3C آغاز شد. هدف اولیه و همچنین هدف جاری این پروژه آن است که با تشخیص منابع داده موجود تحت گواهی‌های[۶] باز، و با انتشار آن‌ها بر اساس قواعد داده‌های پیوندی، گام-های اولیه و اساسی برای تحقق وب داده را آغاز نماید. برای نمایش حجم و ابعاد موضوعی وب داده حاصل از این پروژه، می‌توان ابر پروژه LOD را مورد توجه قرار داد.

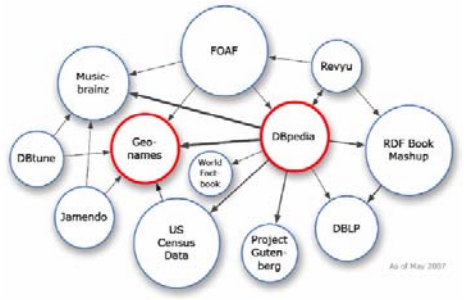
این ابر را می‌توان به شکل یک گراف جهت‌دار در نظر گرفت. هر گره که با یک دایره نشان داده می‌شود نماینده یک مجموعه داده مستقل که بر اساس قواعد داده‌های پیوندی منتشر شده، می‌باشد. یال موجود بین دو گره به این معناست که بین داده‌های موجود در دو مجموعه داده‌های متناظر با آن گره‌ها، پیوند و ارتباط وجود

[1] provenance
[2] Licence
[3] LOD cloud
[4] Best Practice
[5] Linking Open Data Project
[6] License

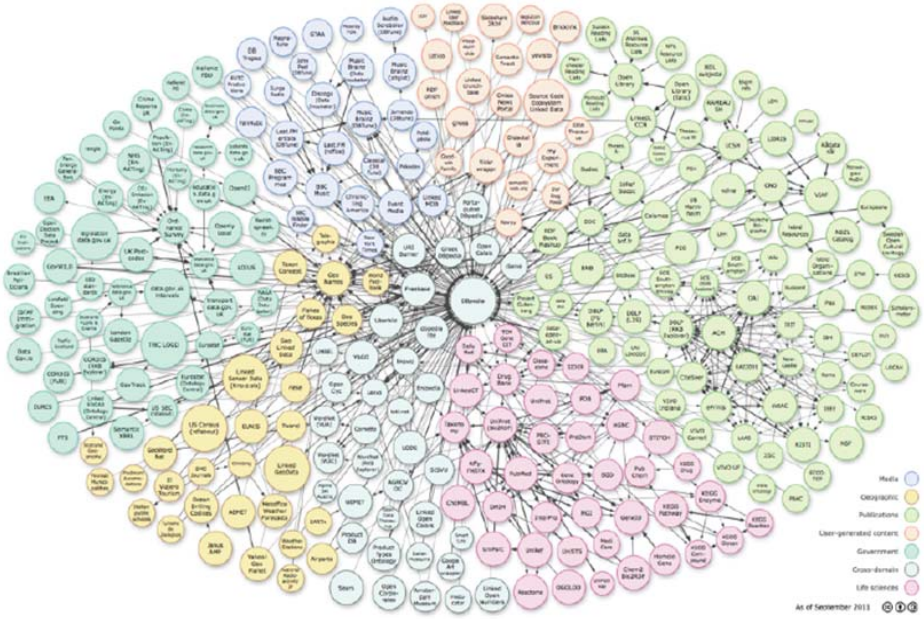
دارد. ضخامت یال‌ها، تخمینی از تعداد این پیوندها ارائه می‌کند. هر چه تعداد پیوندها بیشتر باشد ضخامت یال نیز بیشتر است. یال‌های دوسویه نیز به معنای وجود پیوند از هر یک از دو منبع به یکدیگر می‌باشد

از نظر محتوا، ابر *LOD* دارای تنوع زیادی می‌باشد. بعنوان مثال می‌توان از منابع داده مربوط به مکان-های جغرافیایی، ، شرکت‌های تجاری، انتشارات علمی، فیلم، موسیقی، برنامه‌های تلویزیونی و رادیویی، نام برد.

تعداد منابع‌داده‌ای که داده‌های خود را به صورت داده‌های پیوندی منتشر می‌کنند و تعداد پیوندهای بین منابع‌داده مختلف بشدت در حال گسترش است. مقایسه شکل ۲-۵ و شکل ۲-۶ این رشد ابر *LOD* را از سال ۲۰۰۷ تا ۲۰۱۱ نشان می‌دهد.



شکل ۲-۵: ابر پروژه LOD[1] در may 2007



شکل ۲-۶: ابر پروژه LOD در September 2011

[1] http://richard.cyganiak.de/2007/10/lod/

۲-۴ خلاصه فصل

در این فصل ابتدا زمینه‌های مرتبط با روش پیشنهادی یعنی "تحلیل استناد"، "سیستم‌های پیشنهاد مقاله" و "سیستم‌های پیشنهاد دهنده مبتنی بر داده‌های پیوندی" معرفی شد و سپس کارهای مرتبط با هر یک از این زمینه‌ها به تفکیک شرح داده شد.

همانطور که در بخش ۲-۲ توضیح داده شد، سیستم‌های پیشنهاد مقاله خود بر اساس هدفشان به دو دسته تقسیم می‌شوند، سیستم‌هایی که با توجه به پروفایل کاربر پیشنهاد می‌هند و سیستم‌هایی که صریحا متنی را از کاربر می‌گیرند و بر اساس محتوای آن متن مقالات مرتبط با آن متن را پیشنهاد می‌دهند که سیستم‌های پیشنهاد استناد نام دارند. سیستم پیشنهادی این پایان‌نامه نیز یک سیستم پیشنهاد استناد می‌باشد

در نتیجه در جدول ۲-۲ خلاصه‌ای از کارهای انجام شده در این زمینه، از نظر ورودی، داده‌های پیش-زمینه، تکنیک مورد استفاده در الگوریتم پیشنهاد، خروجی و نحوه ارزیابی، آورده شده است.

جدول ۲-۲: خلاصه‌ای از کارهای انجام شده در زمینه پیشنهاد استناد

	ورودی			داده‌های پیش-زمینه		الگوریتم پیشنهاد						خروجی		ارزیابی	
	متن	متن + لیست جزئی مراجع	متن + مکان‌های استناد	مجموعه داده محلی	منابع داده‌های پیوندی	فیلتر همبستگی	معیار شباهت متنی استفاده از متن استناد در	تحلیل استناد: لیست مراجع	تحلیل استناد: لیست استنادها	تحلیل استناد: استناد مشترک	تحلیل استناد: زوج‌های	پیشنهاد محلی	پیشنهاد سراسری	دستی	خودکار
Woodruff et al., 2000		✓		✓						✓	✓		✓	✓	
McNee et al., 2002		✓		✓		✓		✓					✓	✓	✓
Torres et al., 2004		✓		✓		✓		✓					✓	✓	✓
Strohman et al., 2007	✓			✓				✓		✓			✓	✓	✓
Tang et al., 2009				✓				✓				✓	✓		✓
Gipp et al., 2009	✓			✓				✓	✓	✓	✓		✓		
Bethard et al., 2010	✓			✓			✓		✓				✓		✓
He et al., 2010	✓		✓	✓			✓					✓	✓		✓

He et al., 2011	✓			✓			✓			✓			✓			✓

فصل ۳– روش پیشنهادی

هدف از تولید سیستم ارائه شده در این فصل، یک سیستم پیشنهاد استناد می‌باشد که با گرفتن متن ورودی مقالات مرتبط با آن متن یعنی مقالاتی که باید توسط آن متن مورد استناد قرار بگیرند را پیشنهاد دهد. در ادامه، ابتدا به منظور مقایسه اجمالی سیستم پیشنهادی با نام $SemCiR$[1] و کارهای مرتبط انجام شده در این زمینه، در جدول۳-۱ آورده شده است. سپس در بخش ۳-۱ چارچوب سیستم پیشنهادی و قسمت‌های مختلف آن معرفی شده است.

جدول۳-۱: خلاصه‌ای از جزئیات سیستم پیشنهادی

	ورودی			داده‌های پیش-زمینه		الگوریتم پیشنهاد						خروجی		ارزیابی	
	متن	متن + لیست جزئی مراجع	متن + مکان‌های استناد	مجموعه داده محلی	منابع داده‌های پیوندی	فیلتر همبستگی	معیار شباهت متنی استفاده از متن استناد در	تحلیل استناد: لیست مراجع	تحلیل استناد: لیست استنادها	تحلیل استناد: استناد مشترک	تحلیل استناد: زوج‌های	پیشنهاد محلی	پیشنهاد سراسری	دستی	خودکار
SemCiR	✓				✓		✓	✓	✓	✓	✓		✓		✓

۳–۱ چارچوب سیستم پیشنهادی

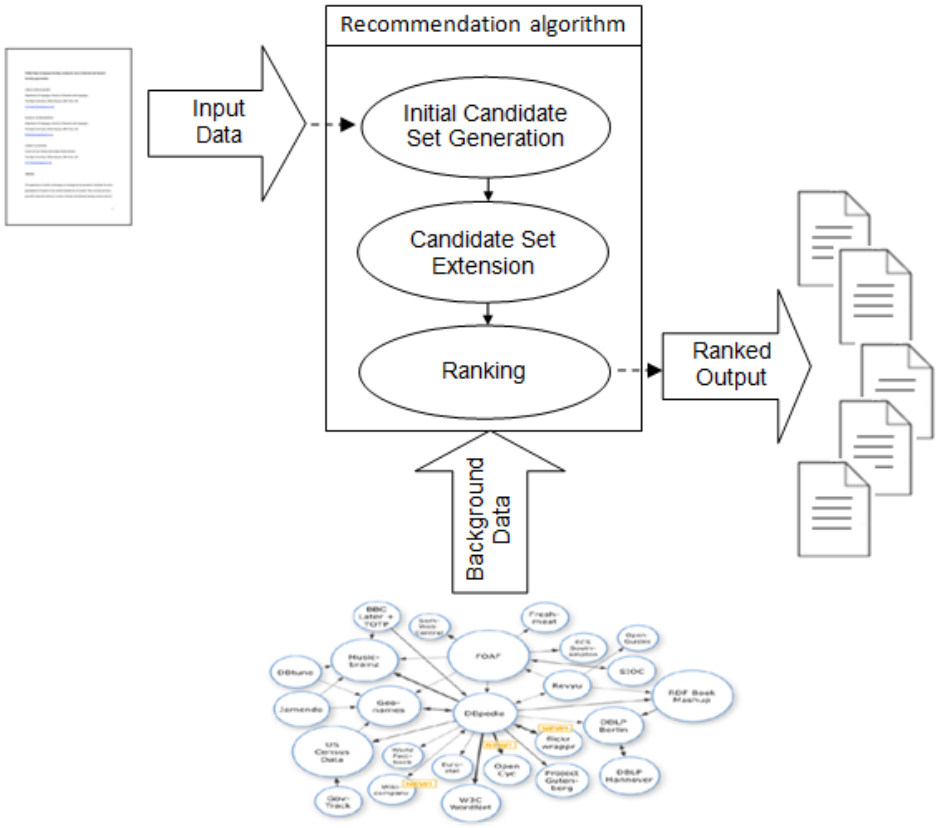
به طور کلی هر سیستم پیشنهاد دهنده از سه بخش اصلی تشکیل شده است (Burke, 2002):

۱- **داده های پیش زمینه**[2]: داده‌هایی که سیستم قبل از شروع به کار دارا می‌باشد.
۲- داده‌های ورودی: داده‌هایی که با هدف پیشنهاد گرفتن، به سیستم وارد می‌شوند.
۳- **الگوریتم پیشنهاد**: بر اساس این الگوریتم سیستم پیشنهاد دهنده از بین داده‌های پیش زمینه به داده ورودی پیشنهاد می‌دهد.

شکل ۳-۱، چارچوب[3] کلی سیستم ارائه شده در این پایان‌نامه را نشان می‌دهد. این سیستم از داده‌های پیوندی بعنوان داده‌های پیش‌زمینه، و از یک الگوریتم جدید که مبتنی بر ترکیب شباهت رابطه‌ای و متنی می-

[1] Semantic Citation Recommendation
[2] Background Data
[3] Framework

باشد، بعنوان الگوریتم پیشنهاد استفاده می‌کند. ورودی سیستم، یک قطعه متن، و خروجی آن لیستی از مقالات مرتبط با آن متن می‌باشد. در ادامه هریک از بخش‌های سیستم پیشنهادی توضیح داده شده است.



شکل ۳-۱: چارچوب سیستم پیشنهادی SemCir

۳-۱-۱ داده‌های پیش زمینه

در لایه داده سیستم پیشنهادی از داده‌های پیوندی استفاده شده است، در این قسمت ابتدا انگیزه استفاده از این داده‌ها در سیستم پیشنهادی، و سپس الگوریتم ارائه شده جهت غنی‌سازی لایه داده به‌کمک داده‌های موجود بر روی ابر *LOD*، شرح داده می‌شود. در نهایت در بخش ۳-۱-۱-۳ ، اطلاعات موجود در داده-های پیش‌زمینه به صورت انتزاعی معرفی می‌شود.

۳-۱-۱-۱ انگیزه استفاده از داده‌های پیوندی

یک بخش مهم در هر سیستم پیشنهاد دهنده‌ای مشخص بودن داده‌های پیش‌زمینه یا لایه داده آن می‌باشد. لایه داده در سیستم‌های پیشنهاد استناد، منابع داده‌ای می‌باشند که اطلاعات مقالات علمی را منتشر

می‌کنند، مانندIEEE، *Citeseer* و *DBLP*. در نتیجه لازمه شروع به کار هر سیستم پیشنهاد استنادی دسترسی به داده‌های موجود در این منابع داده می‌باشد.

در تمامی کارهای موجود در زمینه سیستم‌های پیشنهاد استناد فرض بر این است که داده‌ها به صورت یک مجموعه داده محلی وجود دارد و تمرکز آنها بیشتر بر روی الگوریتم پیشنهاد است، این درحالی است که جمع‌آوری این داده‌ها بعنوان داده‌های پیش‌زمینه یک چالش مهم در این سیستم‌ها می‌باشد.

بعضی از منابع داده سرویسی ارائه می‌کنند که اطلاعات خود را در قالب خاصی در اختیار کاربران قرار می‌دهند، مانند سرویس *OAI*[1] در *CiteseerX*، ولی مشکلی که وجود دارد این است که اولا اطلاعات این فایل‌ها شامل تمامی اطلاعات منتشر شده بر روی صفحات وب آن‌ها نمی‌باشد و در ضمن این سرویس مخصوص یک منبع داده است و ممکن است یک منبع داده دیگر اطلاعات خود را به قالب دیگری در اختیار کاربران قرار دهد و یا اصلا برای یک منبع‌داده چنین سرویسی وجود نداشته باشد، در نتیجه برای جمع‌آوری آن‌ها باید پویشگری[2] نوشته شود که با پردازش متن، از روی صفحات وب اطلاعات را استخراج کند.

چالش دیگری که در لایه داده سیستم‌های موجود وجود دارد، این است که از آنجاییکه این منابع داده کاملا جدا از هم می‌باشند و داده‌های آن‌ها به صورت محلی و خصوصی می‌باشد، اگر مثلا مقاله‌ای در منبع داده *A* منتشر شود و مراجع آن در منبع داده *B*، امکان بازیابی اطلاعات مراجع آن از داخل منبع داده *A* وجود ندارد. تاکید این سیستم‌ها تنها بر روی اطلاعات یک منبع داده، باعث می‌شود که کیفیت پیشنهادها از نظر کامل بودن نتایج کاهش یابد.

در بعضی از زمینه‌ها، مثل پیشنهاد فیلم، مجموعه داده‌ای مثل *IMDB*[3] وجود دارد که شامل اطلاعات تقریبا کاملی در آن زمینه می‌باشند، اما در زمینه کتاب‌شناسی چنین منابع داده‌ای وجود ندارد.

اطلاعات مقالات مختلف توسط منابع‌داده متفاوتی منتشر می‌شوند، اطلاعات یک مقاله ممکن است توسط *IEEE*، منتشر شود ولی در *DBLP* و *ACM* وجود نداشته باشد، و یا برعکس. حتی اگر اطلاعات یک

[1] http://citeseerx.ist.psu.edu/oai.html
[2] crawler
[3] http://www.imdb.com/

مقاله توسط دو منبع داده منتشر شود، مجموعه داده‌های مختلف اطلاعات مختلفی درباره یک مقاله منتشر می-کنند، برای مثال کلمات کلیدی یک مقاله در *IEEE* منتشر می‌شود ولی در *DBLP* منتشر نمی‌شود. در نتیجه، فرض استفاده تنها از یک منبع داده در زمینه کتاب‌شناسی برای برطرف کردن نیازهای اطلاعاتی سیستم‌های پیشنهاد استناد، یک فرض منطقی نمی‌باشد

وجود سیستم پیشنهاد استنادی که قادر باشد در لایه داده خود از چند منبع داده استفاده کند باعث افزایش کیفیت این سیستم‌ها می‌شود، یک راه‌حل ممکن برای ایجاد چنین سیستم‌هایی، تولید یک پویشگر است که اطلاعات این منابع‌داده را جمع‌آوری کند ولی همانطور که اشاره شد از آنجاییکه منابع داده مختلف اطلاعات خود را به قالب‌های مختلفی منتشر می‌کنند برای جمع‌آوری این داده‌ها باید برای هر منبع داده پویشگر متفاوتی نوشته شود، ارتباط دادن این داده‌ها به هم نیز نیازمند روال‌های پیچیده‌ای می‌باشد.

راه‌حل برطرف کردن این مشکلات و داشتن سیستمی که به راحتی بر روی هر منبع داده‌ای کار کند و همچنین قادر باشد از اطلاعات چندین منبع داده استفاده کند، استفاده از وب داده بجای وب اسناد در لایه داده می‌باشد.

وب داده، که نتیجه بکارگیری اصول چهارگانه داده‌های پیوندی می‌باشد (Berners-Lee, 2006)، می-تواند بعنوان یک فضای داده جهانی و واحد استفاده شود. منابع داده مختلف موجود در این فضای داده، از یک قالب استاندارد به نام *RDF*[1]، چارچوب توصیف منابع، برای بازنمایی داده‌هایشان استفاده کرده‌اند، این داده‌ها، ساخت‌یافته می‌باشند و بواسطه استفاده از هستان‌شناسی‌ها، معنای داده‌ها بصورت صریح و قابل فهم برای ماشین بیان می‌گردد. این ویژگی قابل فهم بودن داده‌ها برای ماشین، به‌معنای آن است که امکان بهره‌گیری از توان ماشین در اجرای خودکار فعالیت‌ها افزایش می‌یابد. هر یک از موجودیت‌ها، دارای یک *URI* می‌باشند، این *URI*، در کل فضای داده‌های پیوندی منحصر بفرد است و در صورت بازیابی، اطلاعات مربوط به آن موجودیت در قالب معنایی بازگردانده می‌شود. از مهمترین مزایای داده‌های پیوندی، وجود پیوندها بین داده‌های موجود در

[1] Resource Description Framework

این منابع، می‌باشد. این پیوندها امکان جابجا شدن بین منابع مختلف، بمنظور کشف داده‌های بیشتر را فراهم کرده و کمک می‌کنند تا داده‌های مربوط به یک موجودیت، از منابع مختلف گردآوری شوند.

از دیگر مزایای استفاده از داده‌های پیوندی می‌توان به این مورد نیز اشاره کرد که، با توجه به وجود معمولا یک *SPARQL Endpoint* برای هر منبع داده موجود در فضای داده‌های پیوندی، می‌توان با استفاده از زبان *SPARQL* به جستجو بر روی آن منبع پرداخت، و از آنجایی که منابع داده مربوط به یک حوزه، در خیلی از موارد هستان‌شناسی‌های یکسانی برای توصیف داده‌های خود استفاده می‌کنند، این امکان وجود دارد که یک پرس و جوی *SPARQL* که با توجه به یک هستان‌شناسی خاصی نوشته شده است را بر روی چند منبع مختلف که همگی مربوط به یک موضوع خاص هستند اجرا کرده و بدین ترتیب با یک پرس و جو، به جستجو بر روی چند منبع داده پرداخت و اطلاعات آن‌ها را با قالب یکسان دریافت کرد (Berners-Lee, 2006; Bizer et al., 2009).

به طور کلی، نتیجه طبیعی بکارگیری اصول داده‌های پیوندی آن است که قابلیت دسترس پذیری و اکتشاف داده‌ها بشدت افزایش می‌یابد. این امر می‌تواند در بهبود کارایی سیستم‌های پیشنهاد استناد با توجه به ویژگی انتشار مقالات بر روی منابع داده مختلف و اهمیت لایه داده در سیستم‌های پیشنهاد استناد، بسیار موثر باشد.

در این پایان‌نامه بمنظور بررسی اهمیت استفاده از چند منبع داده در بهبود کیفیت پیشنهادها، یک الگوریتم جدید غنی‌سازی ارائه شده است که با استفاده از منابع مختلف موجود در ابر *LOD*، سعی در برطرف کردن نقائص داده‌ها و غنی کردن لایه داده دارد. این الگوریتم در قسمت بعد معرفی شده است و نتایج آزمایش تجربی آن در بخش ۴-۳-۳-۲ بیان شده است.

۳-۱-۱-۲ الگوریتم غنی سازی داده‌ها به کمک ابر LOD

برای رفع نقص‌های موجود در یک مجموعه داده محلی به کمک داده‌های موجود بر روی ابر *LOD*، *K* منبع‌داده‌های پیوندی $S_i\ (1 \le i \le K)$ موجود در ابر *LOD*، که اطلاعات آن‌ها در زمینه کتاب‌شناسی می‌باشند،

انتخاب شده است. از آنجاییکه ممکن است یک داده بر روی چندین منبع داده پیدا شود به منظور انتخاب یک منبع داده برای غنی‌سازی داده‌ها، برای هر منبع داده خارجی S_i، صفتی به نام $trust_i$ در نظر گرفته شده است که در ابتدا صفر است و مقدار آن با توجه به سازگاریش با مجموعه داده محلی افزایش می‌یابد. به بیان دیگر با فرض صحیح بودن اطلاعاتی که در مجموعه داده‌های محلی دارای نقص نمی‌باشند، هرچه اطلاعات موجود در آن منبع داده خارجی منطبق‌تر با داده‌های موجود در مجموعه داده محلی باشد اطمینان ما برای انتخاب از آن منبع داده خارجی بیشتر می‌شود.

در ادامه، صورت انتزاعی هر مقاله موجود در مجموعه داده محلی آورده شده است، توصیف مشخصه‌های هر مقاله در بخش ۳-۱-۱-۳ آورده شده است، در اینجا، با توجه به این‌که منابع داده مختلفی وجود دارد و قرار است معادل‌های یک مقاله در منابع داده دیگر پیدا شود، دو مشخصه $source_i$ و $equList_i$ نیز در نظر گرفته شده است که در ادامه معرفی شده‌اند.

$$P_i=(source_i, T_i, abs_i, keyList_i, refList_i, citList_i, authList_i, V_i, year_i, equList_i)$$

$source_i$: منبع داده‌ای که مقاله P_i در آن موجود است.

$equList_i$: لیست مقالاتی که در $source_i$ قرار ندارند اما معادل P_i می‌باشند.

در این الگوریتم جهت پیدا کردن معادل هر مقاله در هر یک از منابع خارجی از نام نویسندگان مقالات موجود در مجموعه داده محلی استفاده می‌شود، به طوریکه اگر $authList_i$ لیست نام نویسندگان مقاله P_i $(1\leq i\leq N)$ باشد، لیست نام تمام نویسندگان موجود در مجموعه داده محلی $AuthList$ با توجه به فرمول (۳-۱) بدست می‌آید:

$$AuthList = \bigcup_{i=1}^{N} authList_i \quad (۱-۳)$$

$$M = |AuthList|$$

سپس در دو گام غنی‌سازی داده‌ها انجام شده است، الگوریتم انجام این دو گام به صورت شبه کد در شکل ۳-۲ و شکل ۳-۳ نشان داده شده است.

همانطور که در شکل ۳-۲ مشخص است، در گام اول ابتدا به ازای هر نویسنده‌ای که در مجموعه داده محلی قرار دارد، منبع داده‌های از پیش تعریف شده خارجی به منظور پیدا کردن معادل آن نویسنده جستجو می‌شوند. تابع *findEquiAuthors* برای پیدا کردن معادل هر نویسنده، پرس و جوهای *SPARQL* ای را بر روی نقاط پایانی[1] منبع داده‌های خارجی اجرا می‌کند. این پردازش یک "*co-reference resolution*" (Glaser et al., 2008) می‌باشد که یکی از موضوعات مهم در وب داده است و مورد مطالعه پژوهشگران می‌باشد. سپس برای هر یک از این معادل‌های پیدا شده لیست مقالات آن نویسنده از آن منبع داده استخراج می‌شود و با مقالات موجود در مجموعه داده محلی متعلق به آن نویسنده مقایسه می‌شود، اگر معادل آن در مقالات آن نویسنده پیدا شود، در لیست معادل‌های آن مقاله قرار می‌گیرد.

```
for each author Ai in AuthList {
  localPubs = findPublications(Ai, local)
  for each external source Sj,(1≤j≤K) {
    temp = findEquiAuthors(Ai, Sj)
    for each author Am in temp {
      externalPubs = findPublications(Am, Sj)
      for each publication Px in externalPubs {
        Pmatch = findMatch(Px, localPubs)
        if Pmatch!= -1
           add Px to equListmatch
        else
           add Px to local dataset
      }
    }
  }
}
```

شکل ۳-۲: شبه کد گام اول در الگوریتم غنی سازی داده‌ها

با توجه به شکل ۳-۳، در گام دوم به ازای هر مقاله‌ای که در مجموعه داده محلی قرار دارد، مشخصه-های آن بررسی می‌شود و در صورتی که نقصی در هر یک از مشخصه‌ها وجود داشته باشد اگر معادلی برای آن مقاله پیدا شده باشد، مقادیر موجود در معادل‌های آن مقاله در آن مشخصه بعنوان یک مجموعه کاندید برای مقدار دهی به آن مشخصه در نظر گرفته می‌شود و با توجه به میزان *trust* آن منبع داده یکی از مقادیر انتخاب

[1] Endpoint

می‌شود. در صورتی که آن مشخصه نقصی نداشته باشد از آن به منظور مقدار دهی به *trust* هر منبع داده استفاده می‌شود. به طوریکه اگر مقدار مشخصه آن مقاله و معادلش در منبع داده خارجی یکسان باشد یکی به مقدار *trust* آن منبع داده اضافه می‌شود. هدف از این محاسبه این است که منبع داده‌ای که بیشتر شبیه منبع داده محلی می‌باشد قابل اعتمادتر است.

```
for each publication Pi in the local dataset {
   for each attribute ai,m of Pi {
      init candidate list
      for each Pj in equListi {
      if value of ai,m is missing
         if value of aj,m is not missing
             add pair Pi to candidate list
      else
         if value of aj,m == ai,m
             increase trust of sourcej by 1
      }
      If value of ai,m is missing
         ai,m = selectValue(candidate, m)
   }
}
```

شکل ۳-۳: شبه کد گام دوم در الگوریتم غنی‌سازی داده‌ها

۳-۱-۱-۳ داده‌های پیش‌زمینه از دیدگاه انتزاعی

از دیدگاه انتزاعی، داده‌های پیش‌زمینه شامل اطلاعات N مقاله P_i به صورت زیر می‌باشد:

$P_i = (Id_i, T_i, abs_i, keyList_i, refList_i, citList_i, citctxList_i, authList_i, V_i, year_i)$

Id_i: یک شماره یکتا که به هر مقاله P_i داده می‌شود.

T_i: عنوان P_i

abs_i: چکیده P_i

$keyList_i$: لیست کلمات کلیدی P_i

$refList_i$: لیست مراجع P_i

$citList_i$: لیست مقالاتی که به P_i استناد کرده‌اند.

$citctxList_i$: لیست متن‌های استناد P_i به طوریکه هر متن استناد P_i، قسمتی از متن یک مقاله است که در آن قسمت از متن به P_iاستناد شده است.

$authList_i$: لیست نویسندگان P_i

V_i: کنفرانس یا ژورنالی که P_i در آن ارائه شده است

$year_i$: سال انتشار P_i

از آنجایی که الگوریتم پیشنهاد شامل جستجو بر اساس متن مقالات موجود در مجموعه پیش‌زمینه می‌باشد در نتیجه برای بهبود کارایی الگوریتم پیشنهاد در این قسمت متن هر مقاله که شامل عنوان، چکیده، کلمات کلیدی و لیست متن‌های استناد است اندیس گذاری می‌شود.

۳-۱-۲ داده‌های ورودی

ورودی سیستم پیشنهادی یک قطعه متن می‌باشد و هدف سیستم، پیشنهاد مقالاتی است که مرتبط با متن ورودی می‌باشند. به عبارت دیگر، پیشنهاد مقالاتی که متن ورودی باید به آن‌ها استناد کند.

۳-۱-۳ الگوریتم پیشنهاد

الگوریتم پیشنهادی از دو گام تشکیل شده است: در گام اول تعدادی از مقالات موجود در داده‌های پیش زمینه برای متن ورودی کاندید می‌شوند و در گام دوم این مقالات بر اساس یک الگوریتم رتبه‌بندی مرتب می‌شوند. در ادامه این دو گام توضیح داده شده است.

۳-۱-۳-۱ گام اول: تولید مجموعه کاندید

از آنجایی که داده‌های ورودی تنها از متن تشکیل شده‌اند و مشخصه دیگری مثل لیست مراجع یا لیست نویسندگان ندارند، الگوریتم برای تولید یک مجموعه کاندید اولیه تنها قادر است از ویژگی‌های متنی استفاده کند در نتیجه برای تولید مجموعه کاندید اولیه *initCandSet* شباهت متنی هر مقاله موجود در داده-های پیش‌زمینه و ورودی محاسبه می‌شود *txtSim(input, P_i)* و یک تعداد ثابت C از مقالاتی که بیشترین

شباهت را دارند بعنوان مجموعه کاندید اولیه انتخاب می‌شوند. در این مرحله برای بدست آوردن شباهت متنی از اندیس‌هایی که در قسمت مجموعه داده‌های پیش‌زمینه آماده شده بود، استفاده می‌شود.

مجموعه کاندید اولیه که با توجه به ویژگی‌های متنی بدست آمده است شامل C مقاله از داده‌های پیش‌زمینه می‌باشد، در نتیجه هر یک از آن‌ها علاوه بر ویژگی‌های متنی شامل ویژگی‌های دیگری از جمله لیست نویسندگان، کنفرانس یا ژورنالی که مقاله در آن ارائه شده است و لیست مراجع می‌باشند. از آنجایی که هر یک از این ویژگی‌ها یک نوع ارتباط بین مقالات را مشخص می‌کنند به آن‌ها *ویژگی‌های رابطه‌ای* گفته می‌شود. این ویژگی‌ها می‌توانند برای گسترش مجموعه کاندید اولیه با اضافه کردن مقالاتی که مرتبط با مقالات موجود در مجموعه کاندید اولیه هستند، مورد استفاده قرار بگیرند.

به بیان دیگر، اگر P_i یک مقاله از *initCandSet* باشد، از نظر متنی شبیه متن ورودی است. حال هر مقاله P_j که در *initCandSet* وجود ندارد و مرتبط با P_i است، مرتبط با ورودی در نظر گرفته می‌شود. در نتیجه برای بهبود کیفیت پیشنهادها، مجموعه کاندید با اضافه کردن مقالات مرتبط با مقالات موجود در مجموعه کاندید اولیه، گسترش می‌یابد.

پردازش گسترش مجموعه کاندید با اضافه‌کردن عناصر موجود در *initCandSet* به یک مجموعه تهی به نام *candSet* شروع می‌شود. سپس، برای هر مقاله موجود در *initCandSet* همسایگان آن به *candSet* اضافه می‌شود.

دو مقاله P_i و P_j همسایه در نظر گرفته می‌شوند اگر و تنها اگر حداقل یک ارتباط بین آن‌ها وجود داشته باشد. در روش پیشنهادی ۶ رابطه با نام‌های $R_1, R_2, \ldots, R_6$ بین دو مقاله P_i و P_j با شرایط زیر تعریف شده است:

$R_1: P_i \square citeList_j$

$R_2: P_i \square refList_j$

$R_3: authList_i \cap authList_j \neq \emptyset$

$R_4: V_i = V_j$

R_5: $refList_i \cap refList_j \neq \emptyset$

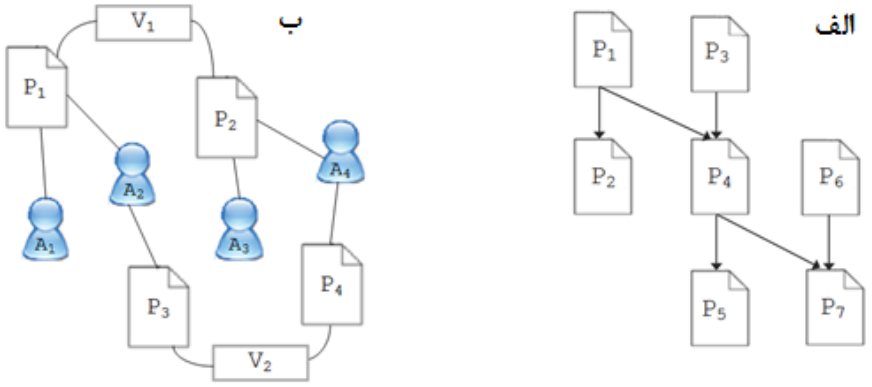
R_6: $citList_i \cap citList_j \neq \emptyset$

رابطه‌های اشاره شده در بالا بر اساس ایده‌های زیر می‌باشند:

وقتی یک مقاله به یک مقاله دیگر استناد می‌کند یا توسط یک مقاله دیگر مورد استناد قرار می‌گیرد، به معنی این است که آن دو مقاله از نظر معنایی با هم مرتبط هستند. این مفاهیم در رابطه‌های R_1 و R_2 نشان داده شده است. از آنجایی که اغلب مقالات نوشته شده توسط یک نویسنده یا مقالات ارائه شده در یک کنفرانس یا ژورنال در یک زمینه می‌باشند رابطه‌های R_3 و R_4 در نظر گرفته شده‌اند. ایده در نظر گرفتن رابطه‌های R_5 و R_6 به ترتیب مبتنی بر دو مفهموم در زمینه تحلیل استناد به نام‌های زوج‌های کتابشناختی و استناد مشترک می‌باشند، این دو مفهوم در بخش ۲-۱ توضیح داده شده‌اند.

شکل ۳-۴، رابطه‌های مختلف بین مقالات را نشان می‌دهد. همانطور که در قسمت "الف" این شکل نشان داده شده است، از مقاله P_4 به P_5 و P_7 رابطه R_1 وجود دارد همچنین از مقاله P_4 به P_1 و P_3 رابطه R_2 وجود دارد. از آنجایی که P_7 مرجع P_4 و P_6 است، از P_4 به P_6، همچنین از P_6 به P_4 رابطه R_5 وجود دارد. در نهایت با توجه به این‌که P_2 و P_4 هر دو توسط P_1 مورد استناد قرار گرفته‌اند، بنابراین رابطه R_6 از P_4 به P_2 و از P_2 به P_4 وجود دارد.

با توجه به شکل ۳-۴، قسمت ب، از آنجایی که P_4 و P_2 دارای نویسنده مشترک A_4 می‌باشند، یک رابطه R_3 از P_4 به P_2 وجود دارد. به علاوه P_4 و P_3 هر دو در کنفرانس یا ژورنال V_1 ارائه شده‌اند در نتیجه از P_4 به P_3 رابطه R_4 وجود دارد.



شکل ۳-۴: الف) گراف استناد، ب) گراف نویسندگان، مقالات و کنفرانس‌ها

۳-۱-۳-۲ گام دوم: مرتب سازی

بعد از تولید مجموعه کاندید، گام مرتب‌سازی انجام می‌شود. در این گام عناصر مجموعه کاندید با توجه به فاصله معنایی آن‌ها از متن ورودی به‌صورت صعودی مرتب می‌شوند، الگوریتم پیشنهادی در این گام برای محاسبه فاصله معنایی از شباهت متنی بدست آمده در گام تولید مجموعه کاندید و یک شباهت رابطه‌ای که در ادامه توضیح آن آورده شده است، استفاده می‌کند.

شباهت رابطه‌ای

با توجه به ۶ ارتباطی که در قسمت ۳-۱-۳-۱ توضیح داده شد، شباهت رابطه‌ای بین هر دو مقاله P_i و P_j به کمک فرمول (۳-۲) تعریف می‌شود:

$$relSim(P_i, P_j) = \sum_{k=1}^{6} w_k F_k(P_i, P_j) \qquad (۳-۲)$$

در فرمول بالا F_k $(1 \le k \le 6)$ یک تابع است که میزان شباهت هر دو مقاله P_i و P_j را با توجه به رابطه متناظر با آن یعنی R_k بدست می‌آورد و w_k وزن تابع F_k را در شباهت رابطه‌ای نشان می‌دهد. تعریف این ۶ تابع در زیر آورده شده است:

$$F_1(P_i, P_j) = \begin{cases} 1 & if\ there\ is\ a\ relation\ R_1\ from\ P_i\ to\ P_j \\ 0 & otherwise \end{cases}$$

$$F_2(P_i, P_j) = \begin{cases} 1 & if\ there\ is\ a\ relation\ R_2\ from\ P_i\ to\ P_j \\ 0 & otherwise \end{cases}$$

$$F_3(P_i, P_j) = \frac{|authList_i \cap authList_j|}{|authList_i \cup authList_j|}$$

$$F_4(P_i, P_j) = \begin{cases} 1 & if\ there\ is\ a\ relation\ R_4\ from\ P_i\ to\ P_j \\ 0 & otherwise \end{cases}$$

$$F_5(P_i, P_j) = \frac{|refList_i \cap refList_j|}{|refList_i \cup refList_j|}$$

$$F_6(P_i, P_j) = \frac{|citList_i \cap citList_j|}{|citList_i \cup citList_j|}$$

در هر یک از توابع F_k، مقدار بازگشتی *0* به معنی وجود نداشتن رابطه R_k و مقدار بازگشتی *1* نشان-دهنده یک ارتباط قوی از نظر R_k می‌باشد.

مقدار برگشتی *1* برای تابع F_1 و F_2 مبتنی بر این ایده است که از آنجایی‌که نویسنده یک مقاله به طور صریح نام مقاله دیگر را در لیست مراجع خود ذکر کرده است، در نتیجه ارتباط قوی‌ای بین دو مقاله وجود دارد و این دو مقاله کاملا مرتبط با هم می‌باشند.

توابع F_3، F_5 و F_6 که به ترتیب برای رابطه‌های R_3، R_5 و R_6 می‌باشند، مبتنی بر ایده یکسانی می-باشند. مثلا در مورد F_3، هر چه نویسندگان مشترک یک مقاله بیشتر باشند، میزان ارتباط آن‌ها بیشتر در نظر گرفته شده است، در حالیکه نسبت نویسندگان مشترک به کل نویسندگان این دو مقاله نیز اهمیت دارد، برای مثال، *دو نویسنده مشترک از سه نویسنده* رابطه‌ی قوی‌تری نسبت به *دو نویسنده مشترک از بین شش نویسنده* می‌باشد. بنابراین رابطه F_3 مستقیما با تعداد نویسندگان مشترک و به صورت معکوس با کل نویسندگان آن دو مقاله در ارتباط است.

مقدار بازگشتی تابع F_4 در صورت وجود رابطه R_4 یعنی ارائه شدن در یک کنفرانس یا ژورنال برابر *1*، و در صورت وجود نداشتن برابر *0* می‌باشد.

فاصله معنایی

به منظور مرتب‌سازی عناصر مجموعه کاندید، فاصله معنایی بین متن ورودی و هر مقاله P_i در مجموعه کاندید به کمک فرمول (۳-۳) بدست می‌آید.

$$semanticDist(input, P_i) = \frac{1}{1+semanticSim(input,P_i)} \qquad (۳-۳)$$

در فرمول بالا، $semanticSim(input,P_i)$ شباهت معنایی بین متن ورودی و مقاله P_i می‌باشد و به کمک فرمول (۳-۴) محاسبه می‌شود.

$$semanticSim(input, P_i) = \frac{1}{C}\sum_{j=1}^{C}\left[relSim(P_i, Q_j) * normTxtSim(input, Q_j)\right] \qquad (۴-۳)$$

در فرمول بالا، Q نشان دهنده لیست مقالات موجود در مجموعه کاندید اولیه $initCandSet$، C نشان-دهنده تعداد عناصر آن و $normTextSim(input, Q_j)$ که در فرمول (۳-۵) تعریف شده است، برابر نرمال شده شباهت متنی، متن ورودی و مقاله Q_j می‌باشد.

$$normTxtSim(input, Q_j) = \frac{txtSim(input,Q_j)}{max_{(1\le k\le C)}\, txtSim(input,Q_k)} \qquad (۵-۳)$$

ایده‌ای که در فرمول (۳-۴) وجود دارد، در شکل ۳-۵ نشان داده شده است.

شکل ۳-۵، یک گراف وزن‌دار جهت دار $G=(V, E)$ است، که در حقیقت نمایشی از خروجی گام اول یعنی تولید مجموعه کاندید می‌باشد.

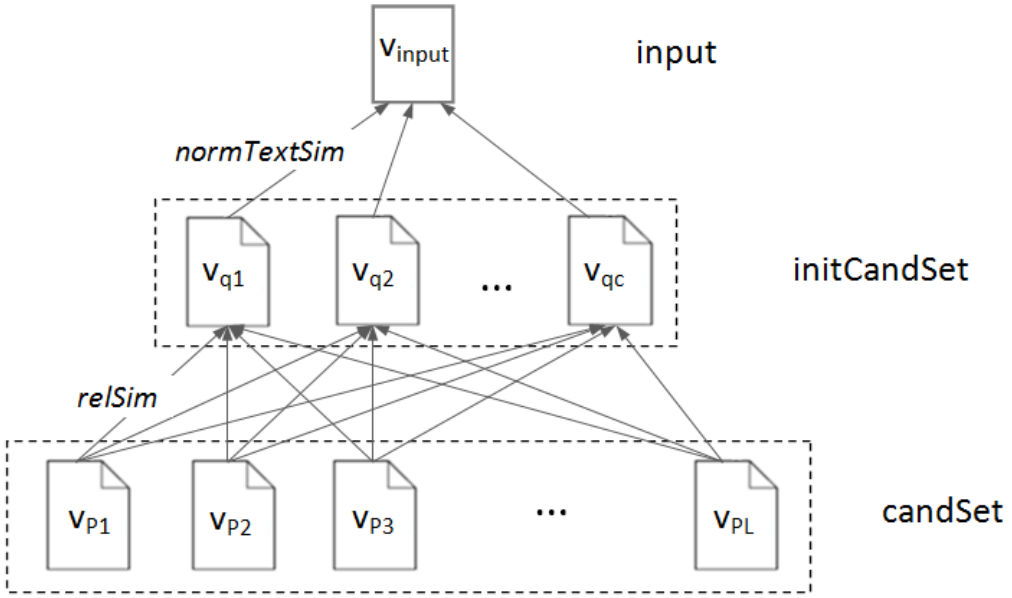
به ازای هر مقاله‌ای که در $initCandSet$ و $candSet$ وجود دارد و همچنین متن ورودی، یک راس در این گراف در نظر گرفته شده است. از آنجایی که هر مقاله Q_j موجود در مجموعه کاندید اولیه بر اساس شباهت متنی‌اش تا ورودی بدست آمده است، یک یال از هر راس نشان دهنده Q_j یعنی v_{qj} به راس ورودی یعنی v_{input} وجود دارد، وزن این یال برابر $normTxtSim(input, Q_j)$ می‌باشد.

بعلاوه، از آنجایی که هر مقاله P_i موجود در مجموعه کاندید $candSet$ همسایه بعضی از مقالات موجود در $initCandSet$ می‌باشد، یالی با وزن $relSim(P_i, Q_j)$ از هر راس نشان‌دهنده مقالات موجود در مجموعه کاندید یعنی v_{pi} به v_{qj} وجود دارد.

برای مرتب‌سازی هر مقاله P_i موجود در *candSet*، باید شباهت آن تا ورودی بدست آید. همانطور که در شکل ۳-۵ نشان داده شده است، یال مستقیمی از v_{pi} به v_{input} وجود ندارد. اما تعداد C مسیر به صورت $[v_{pi}, v_{qj}, v_{input}]$ از v_{pi} به v_{input} وجود دارد.

برای هر مقاله P_i از *candSet*، شباهت آن تا ورودی از طریق C مسیر موجود به وسیله ضرب کردن وزن دو یال موجود در مسیر بدست می‌آید. در نهایت میانگین این C مقدار بعنوان شباهت معنایی P_i و متن ورودی، یعنی $semanticSim(input, P_i)$ در نظر گرفته می‌شود.

دلیل ضرب کردن وزن دو یال موجود در مسیر این است که مقدار $normTxtSim(input, Q_j)$ میزان سهم $relSim(P_i, Q_j)$ را در شباهت معنایی مشخص می‌کند. به بیان دیگر بعنوان وزن تاثیر $relSim(P_i, Q_j)$ در $semanticSim(input, P_i)$ می‌باشد.



شکل ۳-۵: نمایش گرافی نتایج حاصل از گام اول در الگوریتم پیشنهاد

در نهایت عناصر موجود در مجموعه کاندید، با توجه به فاصله معنایی‌شان تا ورودی به صورت صعودی مرتب شده و M عنصر اول پیشنهاد داده می‌شوند.

۳-۲ خلاصه فصل

در این فصل، چارچوب سیستم پیشنهادی در این پایان‌نامه برای کمک به پژوهشگران در پیدا کردن کارهای مرتبط با یک زمینه تحقیقاتی ارائه شده است، سیستم پیشنهادی با گرفتن متن ورودی و استفاده از داده‌های پیوندی بعنوان داده‌های پیش‌زمینه، مقالاتی که باید در آن متن مورد استناد قرار بگیرند را به کمک یک الگوریتم جدید، پیشنهاد می‌دهد.

فصل ۴– پیاده‌سازی و ارزیابی

سیستم پیشنهادی به زبان برنامه‌نویسی *Java* پیاده‌سازی و ارزیابی شده است. در این فصل، ابتدا معیارهای ارزیابی و جزییات آماده‌سازی‌های محیط آزمایش در بخش‌های مختلف سیستم پیشنهادی، توضیح داده شده است، سپس نتایج آزمایشات مختلف انجام شده بمنظور ارزیابی سیستم پیشنهادی آورده شده است.

۴–۱ معیارهای ارزیابی

برای ارزیابی سیستم پیشنهادی از ارزیابی خودکار استفاده شده است و سه معیار ذکر شده در ادامه بعنوان معیارهای ارزیابی کیفیت سیستم پیشنهادی در نظر گرفته شده‌اند.

در هر یک از معیارهای ارزیابی، برای هر مقاله ورودی $Input_i$ $(1 \le i \le N)$، که به سیستم داده می‌شود، لیست استنادهای پیشنهاد داده شده بعنوان $recList_i$ در نظر گرفته شده است:

۴–۱–۱ فراخوانی (recall)

برای هر مقاله ورودی $Input_i$، $recall_i$ به وسیله فرمول (۴-۱) و سپس *recall* کل سیستم به کمک فرمول (۴-۲) بدست می‌آید:

$$recall_i = \frac{|refList_i \cap recList_i|}{|refList_i|} \qquad (۱-۴)$$

$$recall = \frac{1}{N} \sum_{i=1}^{N} recall_i \qquad (۲-۴)$$

۴–۱–۲ احتمال استناد مشترک (Cocited_Probability)

علت در نظر گرفتن معیار احتمال استناد مشترک که در (He et al., 2010) معرفی شده است، در بخش

۲-۲-۲-۲ توضیح داده شد، در این قسمت نحوه محاسبه این معیار برای ارزیابی سیستم پیشنهادی آورده شده است:

برای هر مقاله ورودی $Input_i$، $cocited\_probability_i$ به وسیله فرمول (۴-۳) بدست می‌آید:

$$cocited\_probability_i = \frac{1}{M*L} \sum_{j=1}^{M} \sum_{k=1}^{L} cocited\_probability(P_j, P_k) \quad (۳-۴)$$

در فرمول بالا:

$cocited\_probability\ (P_j, P_k)$ میزان شباهت دومقاله P_j و P_k از نظر احتمال استناد مشترک می‌باشد و با فرمول (۴-۴) محاسبه می‌شود:

$$cocited\_probability(P_j, P_k) = \frac{|citList_j \cap citList_k|}{|citList_j|} \quad (۴-۴)$$

همچنین

$$P_j \in refList_i \quad and\ M = |refList_i|$$

$$P_k \in [recList_i - refList_i] \quad and\ L = |recList_i - refList_i|$$

در نهایت ارزیابی کل سیستم با این معیار از طریق فرمول (۴-۵) بدست می‌آید.

$$cocited\_probability = \frac{1}{N} \sum_{i=1}^{N} cocited\_probability_i \quad (۵-۴)$$

۴-۱-۳ NDCG[1]

معیار *NDCG* همانطور که در بخش۲-۲-۲-۲ توضیح داده شد برای ارزیابی ترتیب قرار گرفتن پیشنهادها در خروجی استفاده می‌شود. مقدار این معیار در مکان i از یک لیست مرتب به کمک فرمول(۴-۶) محاسبه می‌شود:

$$NDCG@i = Z_i \sum_{j=1}^{i} \frac{2^{r(j)}-1}{log(1+j)} \quad (۶-۴)$$

در فرمول بالا $r(j)$ برابر رتبه jامین سند در لیست مرتب شده می‌باشد و Z_i طوری انتخاب می‌شود که مقدار *NDCG* برای یک لیست کاملا مرتب برابر ۱ شود.

[1] Normalized Discounted Cumulative Gain

به‌منظور استفاده این معیار برای سیستم‌های پیشنهاد استناد، در *(He et al.,2010; He et al., 2011)* برای محاسبه رتبه هر پیشنهاد در لیست پیشنهادهای خروجی از معیار احتمال استناد مشترک استفاده شده است، به‌طوریکه برای هر مقاله موجود در داده‌های پیش‌زمینه، احتمال استناد مشترک آن با هر یک مقالات موجود در لیست مراجع مقاله ورودی محاسبه می‌شود و سپس میانگین این مقادیر بعنوان میزان شباهت مقاله موجود در مجموعه داده‌های پیش‌زمینه و مقاله ورودی در نظر گرفته می‌شود.

شباهت هر یک از مقالات موجود در داده‌های پیش‌زمینه و مقاله ورودی محاسبه می‌شود و ماکزیمم این مقادیر R_{max} بعنوان بالاترین رتبه یعنی ۵ در نظر گرفته می‌شود، سپس، برای رتبه دادن مقالاتی که از مجموعه داده‌های پیش‌زمینه در خروجی آمده‌اند، با توجه به شباهت آن‌ها به ورودی و با توجه به بازه‌های زیر رتبه‌های ۴ تا ۰ می‌گیرند:

$$(3R_{max} / 4, R_{max}] , (R_{max} / 2, 3R_{max} / 4] , (R_{max} / 4, R_{max} / 2] , (0, R_{max} / 4] , 0$$

در نهایت ارزیابی کل سیستم با این معیار، میانگین مقادیر بدست آمده برای هر مقاله ورودی خواهد بود.

۴-۲ آماده سازی محیط

به منظور آماده‌سازی محیط آزمایش، کارهای مختلفی در هر یک از بخش‌های معرفی شده در چارچوب سیستم پیشنهادی انجام شده است. در این قسمت آماده‌سازی‌های اولیه در هر بخش توضیح داده می‌شود.

۴-۲-۱ داده‌های پیش‌زمینه

همانطور که در ابتدای فصل ۳ گفته شد، سیستم پیشنهادی از داده‌های پیوندی بعنوان داده‌های پیش-زمینه استفاده می‌کند. منابع داده مختلفی بر روی ابر *LOD* وجود دارد که داده‌های کتاب‌شناسی مربوط به مقالات علمی را منتشر می‌کنند. بعنوان مثال می‌توان به *IEEE*[1]، *ACM*[2]، *Citeseer*[3] و *DBLP*[4] اشاره کرد.

[1] http://ieee.rkbexplorer.com
[2] http://citeseer.rkbexplorer.com
[3] http://acm.rkbexplorer.com
[4] http://dblp.rkbexplorer.com

به منظور استفاده از داده‌های پیوندی بعنوان لایه داده در سیستم پیشنهادی، ابتدا تعدادی از این منابع داده مورد بررسی کیفی قرار گرفتند. نتیجه این بررسی نشان داد اطلاعاتی که توسط این منابع داده بر روی ابر *LOD* منتشر شده است، در مقایسه با اطلاعاتی که همین منابع بر روی وب اسناد منتشر کرده‌اند، از کیفیت پایین‌تری برخوردار است. این ضعف کیفی شامل دو مورد اصلی می‌باشد:

۱. بعضی از اطلاعات مقالاتی که در وب اسناد منتشر شده‌اند، بر روی ابر *LOD* منتشر نشده‌اند. بعنوان مثال، منبع *Citeseer* چکیده و متن استناد مقالات را بر روی وب اسناد منتشر کرده، اما بر روی ابر *LOD* منتشر نکرده است. همچنین اطلاعات مراجع نیز برای برخی مقالات بر روی ابر *LOD* منتشر نشده است، در حالیکه بر روی وب اسناد منتشر شده است.

۲. پیوندهای لازم بین داده‌های منتشر شده توسط منابع مختلف در حد قابل قبولی ایجاد نشده است. بیشتر پیوندهای موجود از نوع *owl:sameAs* می‌باشد که برای بیان اینکه دو موجودیت از دو منبع مختلف، در واقع یک چیز هستند بکار می‌رود. علیرغم اینکه وجود چنین پیوندهایی خیلی مفید است و می‌توان با استفاده از آنها، یک مقاله را در چند منبع پیدا کرد، اما انواع دیگری از پیوندها نیز مورد نیاز می‌باشد. بعنوان نمونه، پیوندهایی برای بیان اینکه مقاله موجود در یک منبع، به مقاله دیگری در منبع دیگر استناد می‌کند.

نکته مهم آن است که هیچ‌یک از دو مشکل فوق، مشکلات ذاتی داده‌های پیوندی نمی‌باشند. بعنوان مثال، اینکه *Citeseer* برخی از داده‌هایی که بر روی وب اسناد منتشر کرده است را بر روی ابر *LOD* منتشر نکرده، به نظر امری مقطعی می‌باشد و با گذر زمان، این کمبود داده‌ها رفع می‌شود. انتظار می‌رود با توجه به اقبال روزافزون محققان به داده‌های پیوندی، این منابع داده، اطلاعات کامل‌تری درباره مقالات بر روی ابر *LOD* منتشر کنند.

همانطور که در بخش ۳-۱-۱-۱ اشاره شد، سیستمی که از داده‌های پیوندی استفاده کند در مقایسه با سیستمی که مبتنی بر یک مجموعه داده محلی باشد مزایای بیشتری دارد. با توجه به همین موضوع، یک مجموعه داده پیوندی آزمایشی برای ارزیابی الگوریتم پیشنهادی ایجاد شده است. دیدگاه این است که در آینده

نزدیک، کیفیت داده‌های پیوندی موجود در منابع مورد نظر بهبود می‌یابد و در آن صورت می‌توان سیستم پیشنهادی را بر روی منابع داده ابر *LOD* اجرا نمود.

تولید مجموعه داده پیش‌زمینه آزمایشی شامل ۳ گام می‌باشد: جمع‌آوری داده‌های موجود در یک منبع داده از وب اسناد، تبدیل آن‌ها به داده‌های پیوندی و پیش‌پردازش متنی، که در ادامه نحوه انجام این مراحل توضیح داده شده است:

۱- جمع‌آوری داده‌های یک منبع داده از وب اسناد:

داده‌های موجود در *CiteSeerX*[1] بعنوان منبع داده در سیستم پیشنهادی استفاده شده است. برای جمع‌آوری این داده‌ها یک پیمایشگر نوشته شده است که از سرویس *OAI* موجود در *CiteSeerX* استفاده می‌کند. پیمایش برای جمع‌آوری داده‌ها از یک مقاله خاص در زمینه داده‌های پیوندی در سال ۲۰۱۰ شروع شد و بصورت پیمایش سطحی گراف استناد تا رسیدن به یک مقدار قابل قبول مقاله ادامه یافت. از آنجاییکه این سرویس تمامی اطلاعات اشاره شده در بخش ۳-۱-۱-۳ را فراهم نمی‌کرد یک پیمایشگر دیگر هم نوشته شد تا بتواند از روی صفحات وب *CiteSeerX* اطلاعاتی مثل کنفرانس یا ژورنالی که مقاله در آن ارائه شده و متن‌های استناد یک مقاله را بازیابی کند، البته کلمه کلیدی، جزء اطلاعات منتشر شده در وب اسناد *CiteseerX* نیز نبود، در نتیجه در آزمایشات، لیست کلمات کلیدی تهی می‌باشد. برای نوشتن پیمایشگری که بتواند اطلاعات را از صفحات وب جمع‌آوری کند از کتابخانه *Htmlunit* استفاده شده است. لازم به ذکر است، برای استخراج بعضی از اطلاعات از وب اسناد، مثل متن استناد روال‌های پیچیده‌ای نوشته شده است.

بعد از جمع آوری اطلاعات ۳۰۰۰۰ مقاله، ابتدا مقالاتی که سال انتشار آن‌ها بعد از ۲۰۰۷ بود بعنوان داده‌های ورودی برای آزمایش سیستم از مجموعه داده حذف شدند، سپس از بین مقالات باقی مانده آن‌هایی که تعداد زیادی از داده‌های آن‌ها بی‌ارزش بود، مثلا مقالاتی که عنوان، چکیده و لیست مراجع نداشتند، حذف شدند. در نهایت حدود ۱۲۰۰۰ مقاله بدست آمد که در پایگاه داده *MySQL* ذخیره شدند.

[1] http://citeseerx.ist.psu.edu

۲- تبدیل داده‌های موجود در پایگاه داده بصورت داده‌های پیوندی:

در این مرحله طبق قواعد داده‌های پیوندی، داده‌های ذخیره شده در پایگاه داده، به داده‌های پیوندی تبدیل شده‌اند. بدین منظور با استفاده از زبان برنامه‌نویسی جاوا و ابزارهای موجود نظیر *Jena*، برنامه‌ای بعنوان *RDFizer* پیاده‌سازی شد که داده‌ها را از پایگاه داده خوانده و آنها را با استفاده از هستان‌شناسی‌ها، به قالب داده‌های پیوندی تبدیل نماید. طبق نخستین مورد از قواعد داده‌های پیوندی، باید به هر یک از موجودیت‌هایی که قرار است اطلاعات آنها منتشر شود، یک *URI* منحصر بفرد اختصاص داده شود. بدین منظور سه الگوی *URI* برای توصیف موجودیت‌های مقاله، نویسنده، و مکان انتشار، به شکل زیر تعریف گردید که در این الگوها، *[PAPER_ID]*، *[AUTHOR_ID]* و *[VENUE_ID]* به ترتیب کلیدهای اصلی جدول‌های مقالات، نویسندگان، و مکان‌های انتشار از پایگاه داده می‌باشند.

http://wtlab.um.ac.ir/linkeddata/semcir/paper/[PAPER_ID]

http://wtlab.um.ac.ir/linkeddata/semcir/author/[AUTHOR_ID]

http://wtlab.um.ac.ir/linkeddata/semcir/venue/[VENUE_ID]

بر اساس دومین قاعده داده‌های پیوندی، وقتی *URI* مربوط به یک موجودیت بازیابی می‌شود باید اطلاعات مفیدی درباره آن موجودیت بازگردانده شود. بدین منظور، اطلاعات هر موجودیت از جدول‌های پایگاه اطلاعات استخراج و با استفاده از هستان‌شناسی‌های مختلفی نظیر *AKT*، *FOAF* و *Dublin Core*، در قالب *RDF* بازنمایی گردید. این بازنمایی شامل تعدادی سه‌گانه *RDF* می‌باشد که هر سه‌گانه، کوچکترین جزء اطلاعات را بیان می‌کند. بعنوان نمونه،شکل ۴-۱ قسمتی از توصیف معنایی یک مقاله را در قالب *N3* نشان می-دهد.

بعد از تولید داده‌های پیوندی مورد نظر، برای ذخیره‌سازی آنها از ابزار *OpenRDF Workbench* استفاده شده است که امکان ذخیره‌سازی داده‌های معنایی و اجرای پرس‌وجوهای *SPARQL* بر روی آنها را فراهم می‌کند.

```
paper:10 rdf:type akt:Article-Reference  ,
      akt:has-title "Under Pressure: Recommendations for Managing a
      Practical Course in Software Engineering",
      dc:creator author:5 ,
      akt:cites-publication-reference paper:36 ,
      akt:cites-publication-reference paper:58 ,
      akt:cites-publication-reference paper:111 .
author:5 rdf:type foaf:Person ,
      foaf:name "Klaus Bergner".
```

شکل ۴-۱: قسمتی از توصیف معنایی یک مقاله در قالب N3

۳- پیش‌پردازش متنی:

مرحله بعد در تولید داده‌های پیش‌زمینه، بعد از جمع‌آوری مجموعه داده‌ها و تبدیل آن‌ها به داده‌های پیوندی، پیش‌پردازش متنی است. این مرحله به کمک کتابخانه *lucene*[1] پیاده‌سازی شده است، *lucene* یک کتابخانه متن‌باز فعال است که امکان جستجو را در برنامه‌های کاربردی جاوا فراهم می‌کند. این کتابخانه از دو بخش اصلی به نام‌های اندیس‌گذاری و جستجو تشکیل شده است. در مرحله پیش‌پردازش متنی، از بخش اندیس‌گذاری استفاده شده است. عملیات این جزء شامل سه گام است که در ادامه توضیح داده می شود (McCandless, 2010):

ساختن سند: در این گام *lucene* داده‌های خام را به واحدهای منطقی‌ای که سند نام دارند تبدیل می‌کند. هر سند از تعدادی فیلد تشکیل شده است که *lucene* با ایجاد اندیس بر روی فیلدهای این اسناد مقادیر آن‌ها را قابل جستجو می‌کند. در آزمایشات، هر مقاله P_i از مجموعه داده محلی به کمک *lucene* به سندی که دارای فیلدهای *ID* و *TEXT* است تبدیل می‌شود، برای هر سند مقدار فیلد *ID* برابر Id_i و مقدار فیلد *TEXT* برابر $text_i$ می‌باشد. لازم به ذکر است که $text_i$ شامل ترکیبی از $\{T_i, abs_i, citctxList_i\}$ می‌باشد که در آزمایشات مختلف متفاوت در نظر گرفته شده است.

[1] http://lucene.apache.org

تحلیل سند: در این گام مشخص می‌شود که چگونه *lucene* باید مقادیر متنی فیلدهای سندها را به رشته‌ای از کلمات تبدیل می‌کند، این کلمات واحدهای واقعی خواهند بود که در فیلدهای متنی اندیس می‌شوند. برای انجام این گام تحلیل‌گری برای *lucene* نوشته شده است که اعداد و ایست واژه‌ها[1] را حذف کند و ریشه کلمات باقی مانده را استخراج کند.

اندیس‌گذاری سند: در این گام، هر سند به کمک *lucene* اندیس می‌شود.

۴-۲-۲ داده‌های ورودی

به منظور ارزیابی سیستم پیشنهادی، مقالات منتشر شده در سال‌های ۲۰۰۸، ۲۰۰۹ و ۲۰۱۰ بعنوان داده‌های ورودی انتخاب شده‌اند.

هر مقاله P_i در مجموعه داده‌های ورودی به شکل زیر خواهد بود:

$$P_i = (T_i, abs_i, refctxList_i, refList_i)$$

$refctxList_i$: لیست قسمت‌هایی از متن مقاله P_i می‌باشد، که در آن‌ها به مقالات دیگر استناد شده است.

سپس از بین مجموعه داده‌های ورودی مقالاتی که دارای شرایط زیر هستند حذف می‌شوند و حدود ۶۰۰ مقاله باقی مانده بعنوان مجموعه داده ورودی انتخاب می‌شوند:

$$(T_i=\emptyset \ OR \ abs_i=\emptyset \ OR \ |refctxList_i|<3 \ \ OR \ |refList_i|<5)$$

علت شرایطی اشاره شده در بالا برای حذف تعدادی از مجموعه ورودی، این است که مقالاتی که متن کافی ندارند و یا مراجع آن‌ها در مجموعه داده محلی وجود ندارد، معیار ارزیابی سیستم قرار نگیرند.

برای ارزیابی سیستم پیشنهادی، T_i, abs_i و $refctxList_i$ هر مقاله بعنوان متن ورودی و $refList_i$ بعنوان خروجی مورد انتظار از سیستم پیشنهاد دهنده در نظر گرفته شده است.

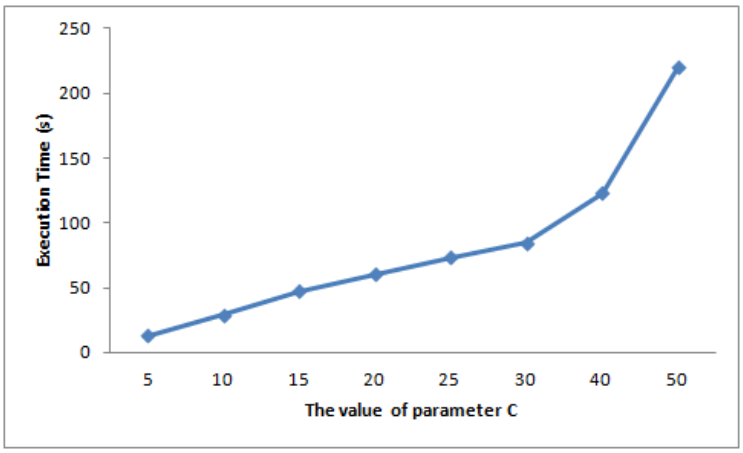
۴-۲-۳ الگوریتم پیشنهاد

[1] Stop word

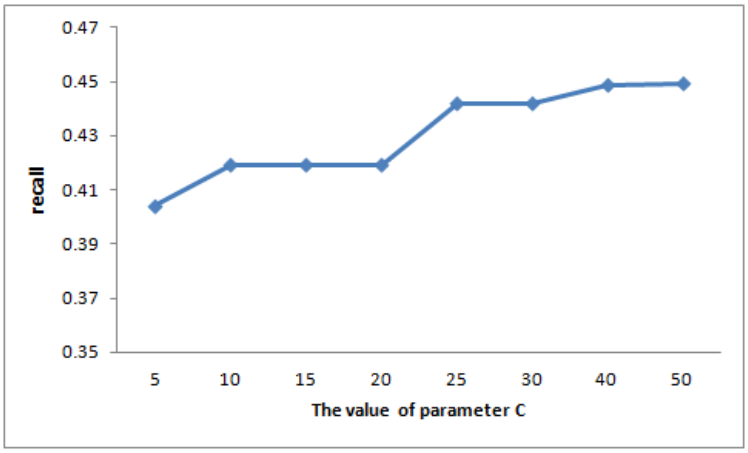
همانطور که در بخش ۳-۱-۳، اشاره شد، در مرحله تولید مجموعه کاندید اولیه از شباهت متنی استفاده شده است. در نتیجه یک پردازش جستجوی مبتنی بر متن برای پیداکردن مقالاتی که شباهت متنی زیادی با متن ورودی دارند نیاز است. برای پیاده‌سازی این مرحله، از بخش جستجوی *lucene* استفاده شده است. به عبارتی متن ورودی بعنوان یک پرس‌وجو در نظر گرفته شده است و به کمک بخش جستجوی *lucene*، از بین مقالاتی که اندیس‌گذاری شده‌اند، C مقاله که بیشترین شباهت متنی را دارند بعنوان مجموعه کاندید اولیه انتخاب می‌شوند. این مجموعه بعنوان ورودی مرحله بعد یعنی گسترش مجموعه کاندید استفاده می‌شود. بعد از تولید مجموعه کاندید، الگوریتم مرتب سازی M مقاله از مجموعه کاندید را پیشنهاد می‌کند. در آزمایشات برای مقدار M بازه $[25, 250]$ انتخاب شد است.

برای مشخص کردن مقدار C، در قسمت تولید مجموعه کاندید اولیه یک آزمایش ساده انجام شده است، یک مجموعه از ۱۰۰ مقاله از بین مجموعه داده‌های انتخاب شده برای داده‌های ورودی به طور تصادفی برای این آزمایش انتخاب می‌شود و الگوریتم پیشنهادی برای مقادیر مختلف C یعنی $\{5, 10, 15, 20, 25, 30, 40, 50\}$ آزمایش شده است. نتایج آزمایشات با توجه به معیارهای ارزیابی و معیار زمان، ارزیابی شده و نتایج آن در شکل ۴-۲ تا شکل ۴-۵ آورده شده است. همانطور که در این شکل‌ها دیده می‌شود، در بازه در نظر گرفته شده برای C، هرچه مقدار C بزرگتر می‌شود زمان بیشتری برای پیشنهاد دادن صرف می‌شود و نتایج نیز با توجه به معیارهای ارزیابی بهتر می‌شوند.

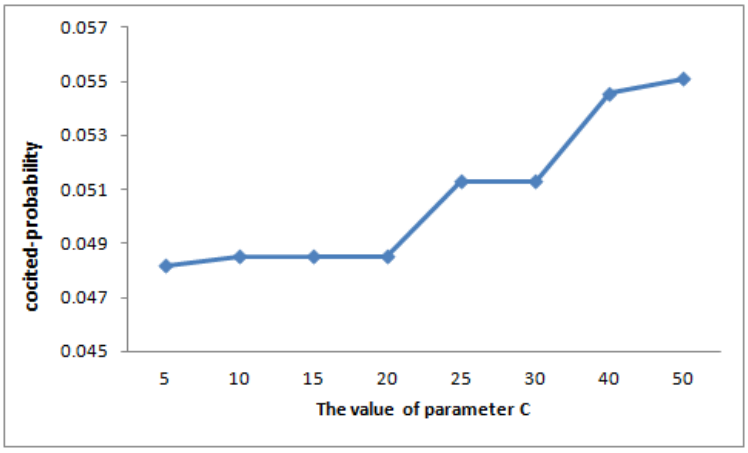
با توجه به شکل ۴-۲ تا شکل ۴-۵ در $C=25$ به مقدار قابل توجهی مقدار سه معیار ارزیابی افزایش یافته است، در حالیکه رشد زمان ثابت است. بنابراین به منظور ایجاد تعادل بین زمان و معیارهای ارزیابی مقدار 25 برای مقدار C در انتخاب مجموعه کاندید اولیه انتخاب شده است.



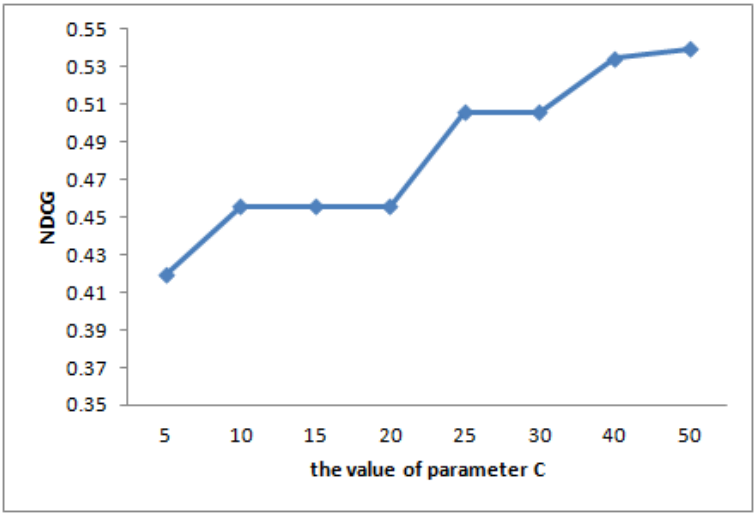
شکل ۴-۲: زمان اجرا برای مقادیر مختلف C



شکل ۴-۳: معیار فراخوانی برای مقادیر مختلف C



شکل ۴-۴: معیار استناد مشترک برای مقادیر مختلف C



شکل ۴-۵: معیار NDCG برای مقادیر مختلف C

در قسمت شباهت رابطه‌ای نیز، به هر یک از توابع در نظر گرفته شده برای رابطه‌ها، یک وزن اختصاص داده شده است که نشان‌دهنده میزان تاثیر آن رابطه در شباهت رابطه‌ای دو مقاله می‌باشد. در آزمایش‌های تجربی صورت گرفته، برای تخصیص وزن‌ها از الگوریتم ژنتیک که یک تکنیک جستجو برای یافتن راه‌حل تقریبی مسائل بهینه‌سازی می‌باشد (Golberg, 1989)، استفاده شده است. مراحل انجام این الگوریتم در ادامه توضیح داده می‌شود.

۴-۱-۳-۱ وزن‌دهی به رابطه‌ها به‌کمک الگوریتم ژنتیک

مرحله اول- ایجاد جمعیت اولیه: وزن‌های توابع به صورت یک کروموزوم در نظر گرفته شده است به طوری که وزن F_k در خانه k ام کروموزوم قرار می‌گیرد. همچنین برای عدد وزن، اعداد بین صفر و یک در نظر گرفته شده است. در نتیجه در این مرحله ابتدا ۵۰ کروموزوم بعنوان جمعیت اولیه تعریف می‌شود و با اعداد بین صفر و یک به صورت تصادفی مقدار می‌گیرد.

مرحله دوم- محاسبه برازندگی افراد جمعیت: برای محاسبه برازندگی هر کروموزم، ۵۰ مقاله بطور تصادفی از بین مقالات مجموعه داده‌های ورودی انتخاب شده‌اند.

ابتدا هر تابع با وزن متناظر خود در کروموزوم، وزن می‌گیرد، سپس برای متن ورودی هر یک از این ۵۰ مقاله، مراحل زیر انجام می‌شود:

a. اجرای الگوریتم پیشنهادی با گرفتن متن ورودی

b. محاسبه مقادیر فراخوانی، احتمال استناد مشترک و *NDCG* برای پیشنهادهای حاصل از خروجی مرحله *a*

c. محاسبه حاصل جمع سه مقدار بدست آمده از مرحله *b*

پس از انجام مراحل ذکر شده در بالا برای هر متن ورودی، میانگین ۵۰ مقدار محاسبه شده، بعنوان میزان برازندگی آن کروموزوم در نظر گرفته می‌شود.

علت جمع‌کردن مقادیر سه معیار در مرحله *c* این است که هدف بدست آوردن وزن‌هایی است که بطور کلی باعث بهبود حاصل جمع سه مقدار می‌شوند.

مرحله سوم- عمل انتخاب[1]. برای ایجاد نسل جدید به‌روش چرخ رولت، احتمال انتخاب هر کروموزم، با توجه به مقدار برازندگی آن محاسبه می‌گردد.

مرحله چهارم- عمل تبادل[2]. در این روش، به صورت تصادفی ژن i در یکی از دو کروموزوم والد انتخاب می‌شود. سپس از ژن i تا آخر کروموزوم، ژن‌های دو کروموزوم والد انتخاب شده جابجا می‌شوند (تبادل یک نقطه‌ای).

مرحله پنجم- عمل جهش[3]. با احتمال ۰.۰۰۲ هر ژن با عدد دیگری در بازه صفر و یک جابجا می‌شود.

مراحل توضیح داده شده در بالا، برای تعداد نسل مشخص، ۲۰، ادامه می‌یابد و کروموزومی که بهترین برازندگی را دارد بعنوان وزن‌های بهینه در نظر گرفته می‌شود.

وزن‌های بدست آمده برای هر تابع F_k بعد از گرد کردن آنها، در زیر آورده شده است:

F_1 F_2 F_3 F_4 F_5 F_6

[1] Selection
[2] Cross over
[3] Mutation

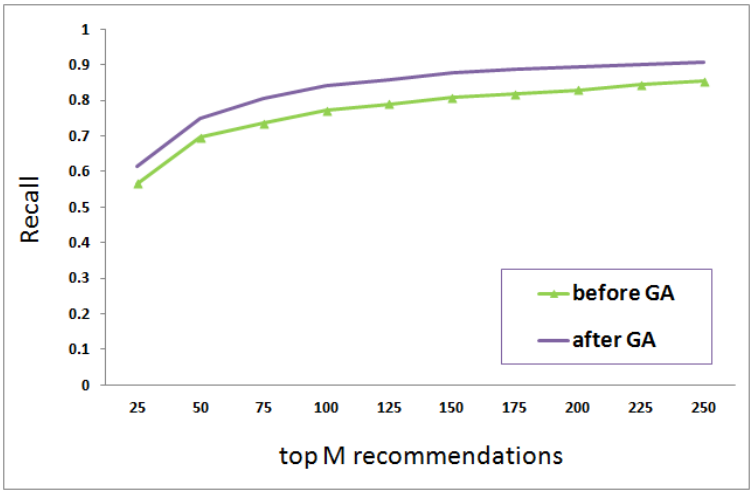
0.8	0.2	0.4	0.2	0.5	0.7

در ادامه ابتدا اهمیت استفاده از وزن برای هر یک از رابطه‌ها نشان داده شده است و سپس یک بررسی تحلیلی، تا حدی منطقی بودن مقادیر وزن‌های بدست آمده توسط الگوریتم ژنتیک را نشان می‌دهد، البته اثبات دقیق این ادعا کار ساده‌ای نیست.

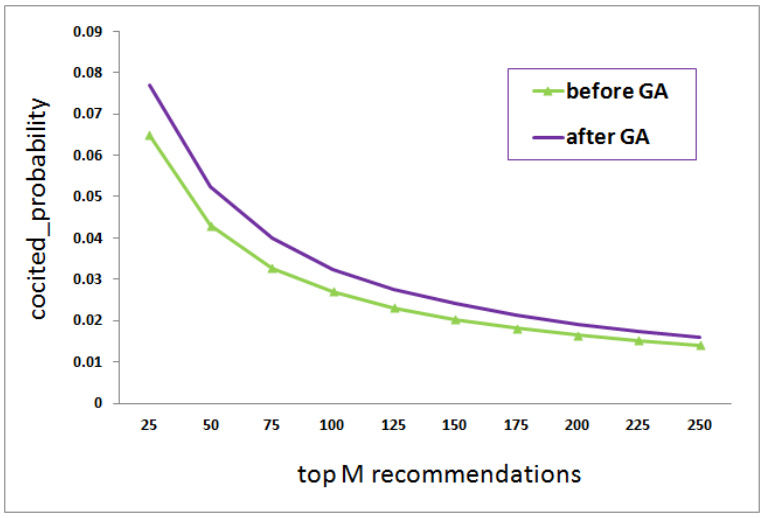
اهمیت در نظر گرفتن وزن برای رابطه‌ها:

در این پایان‌نامه، برای نشان دادن اهمیت در نظر گرفتن وزن‌ها، آزمایشی در نظر گرفته شده است. در این آزمایش ابتدا ۱۰۰مقاله بطور تصادفی از بین مقالات ورودی برای ورودی الگوریتم انتخاب شده‌اند، سپس الگوریتم پیشنهاد یک بار با در نظر گرفتن وزن ۱ برای همه رابطه‌ها و یک بار با در نظر گرفتن وزن‌های بدست آمده از الگوریتم ژنتیک اجرا شده است.

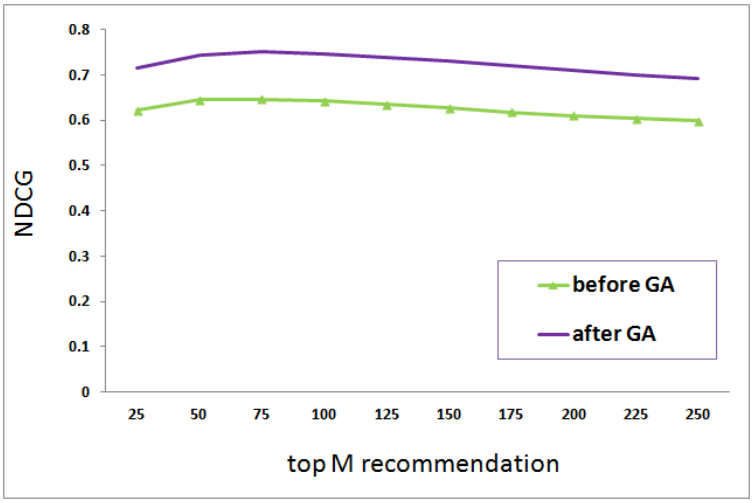
شکل ۴- ۶ تا شکل ۴- ۸ نتایج ارزیابی پیشنهادهای این دو حالت را از نظر سه معیار ارزیابی، نشان می‌دهد. همانطور که در این شکل‌ها مشاهده می‌شود از نظر هر سه معیار ارزیابی، استفاده از وزن برای رابطه‌ها باعث بهبود کیفیت پیشنهادها می‌شود.



شکل ۴- ۶: اهمیت وزن‌دهی به رابطه‌ها از نظر معیار فراخوانی



شکل ۴- ۷: اهمیت وزن‌دهی به رابطه‌ها از نظر معیار استناد مشترک



شکل ۴- ۸: اهمیت وزن‌دهی به رابطه‌ها از نظر معیار NDCG

بررسی تحلیلی مقادیر وزن‌ها:

وزن F_1 از بقیه بیشتر است و مقدار نسبتا بزرگی هم دارد. این امر منطقی است چرا که مقالات موجود در مجموعه کاندید اولیه با متن ورودی شباهت متنی زیادی دارند و در نتیجه اگر اکثر این مقالات به مقاله خاصی استناد کنند، به احتمال زیاد متن ورودی نیز باید به آن مقاله استناد نماید. بهمین دلیل منطقی است که F_1 به نسبت دیگر توابع، از وزن بزرگتری برخوردار باشد.

وزن اختصاص داده شده به تابع F_2 مقدار کوچکی دارد. توجیه آن بدین ترتیب است که اگر مقاله‌ای به بیشتر مقالات موجود در مجموعه کاندید اولیه، که همگی از نظر متنی شبیه متن ورودی هستند، استناد کند،

بین آن مقاله و متن ورودی رابطه‌ای وجود دارد، اما لزوما نمی‌توان گفت که متن ورودی باید به آن مقاله استناد کند. طبیعی است که وزن F_2 باید از وزن F_1 کمتر باشد.

وزن بدست آمده برای F_4 برابر 0.2 است. این مقدار مؤید این دیدگاه است که اگرچه معمولا هر کنفرانس یا ژورنالی بر روی یک موضوع متمرکز است اما معمولا هر موضوعی شامل تعداد زیادی زیرموضوع مرتبط می‌باشد. در نتیجه دو مقاله که در یک کنفرانس یا ژورنال ارائه می‌شوند اگرچه زمینه کلی‌شان یکسان است، اما لزوما زیر موضوع یکسانی ندارند. در نتیجه سهم R_4 در شباهت کلی بین دو مقاله کم می‌باشد.

درمورد وزن F_3 می‌توان گفت اگرچه مقاله‌های یک نویسنده اغلب در یک زمینه است ولی این موضوع در همه حالات صدق نمی‌کند و ممکن است یک نویسنده در زمینه‌های مختلفی کار کند بنابراین وزن این ارتباط در شباهت رابطه‌ای نباید زیاد باشد. البته منطقی است که وزن F_3 از وزن F_2 بیشتر باشد، چرا که احتمال شباهت دو مقاله یک نویسنده، بمراتب بیشتر از احتمال شباهت دو مقاله ارائه شده در یک کنفرانس یا ژورنال می‌باشد.

وزن F_6 نسبتا زیاد است. اگر یک مقاله P وجود داشته باشد که به دفعات زیادی، همزمان با مقالات مجموعه کاندید اولیه مورد استناد قرار گرفته باشد، یعنی بیشتر مقالاتی که به مقالات درون مجموعه کاندید اولیه استناد کرده‌اند، به مقاله P نیز استناد کرده باشند، آنگاه با توجه به اینکه مقالات درون مجموعه کاندید اولیه به متن ورودی شباهت متنی زیادی دارند، به احتمال زیاد می‌توان گفت که متن ورودی باید به مقاله P استناد کند. وزن زیادی که به F_6 اختصاص داده شده با این ایده مطابقت دارد و در نتیجه منطقی بنظر می‌رسد.

وزن F_5 از وزن F_6 کمتر است و از مقدار متوسطی برخوردار است. دلیل این امر را می‌توان اینگونه بیان نمود. اگر یک مقاله P وجود داشته باشد که با مقالات مجموعه کاندید اولیه، تعداد زیادی مرجع مشترک داشته باشد، این بدان معناست که مقاله P به مقالات مجموعه کاندید اولیه شباهت زیادی دارد. بواسطه این شباهت تا حدی می‌توان گفت که متن ورودی باید به مقاله P استناد کند. البته واضح است که میزان این امر در مقایسه با F_6 کمتر است.

۴-۳ ارزیابی سیستم پیشنهادی

ارزیابی سیستم پیشنهادی، در دو سطح انجام شده است ابتدا الگوریتم پیشنهادی مورد ارزیابی قرار گرفته است و سپس استفاده از داده‌های پیوندی در سیستم‌های پیشنهاد استناد، مورد ارزیابی کیفی و کمی قرار گرفته است.

۴-۳-۱ ارزیابی الگوریتم پیشنهاد استناد

در این قسمت ابتدا روش‌هایی که برای ارزیابی الگوریتم پیشنهادی در نظر گرفته شده است، شرح داده می‌شود. سپس در بخش ۴-۳-۲-۲ ، نتایج این ارزیابی‌ها و تحلیل آن‌ها ارائه شده است.

۴-۳-۲-۱ روش‌های مورد مقایسه

در آزمایش‌ها، پنج روش مختلف با هم مقایسه شده‌اند، این روش‌ها از سه جنبه زیر با هم متفاوت می-باشند:

۱) عناصر متنی که از هر مقاله اندیس‌گذاری می‌شوند.

۲) روش تولید مجموعه کاندید

۳) روش مرتب سازی

در ادامه این پنج روش، که خلاصه آنها در جدول ۴-۱ آورده شده است، به تفکیک توضیح داده می-شوند.

جدول ۴-۱: روش‌های مورد مقایسه

روش‌ها	عناصر اندیس‌گذاری شده	روش تولید مجموعه کاندید	روش مرتب سازی
روش۱	$\{T_i, abs_i\}$	textual similarity (Lucene)	textual similarity (Lucene)
روش۲	$\{T_i, abs_i\}$	SemCiR	SemCiR
روش۳	$\{T_i, abs_i, citctxList_i\}$	SemCiR	SemCiR
روش۴	$\{T_i, abs_i, citctxList_i\}$	SemCiR	(He et al., 2010)
روش۵	$\{T_i, abs_i, citctxList_i, refctxList_i\}$	(He et al., 2010)	(He et al., 2010)

روش۱ *(M1)*: در این روش، برای هر مقاله P_i، مجموعه $\{T_i, abs_i\}$ برای اندیس‌گذاری در نظر گرفته شده‌اند و برای تولید پیشنهاد، متن ورودی بعنوان یک پرس‌وجو به بخش جستجوی *lucene* داده می‌شود. به عبارتی، تولید مجموعه کاندید و روش مرتب‌سازی هر دو مبتنی بر شباهت متنی هستند و از شباهت متنی *lucene* استفاده می‌کنند.

روش۲ *(M2)*: در این روش نیز، مجموعه $\{T_i, abs_i\}$ برای هر مقاله P_i، اندیس‌گذاری می‌شوند. اما روش تولید مجموعه کاندید و مرتب‌سازی، روش پیشنهادی این پایان‌نامه که در بخش ۳-۱-۳ توضیح داده شده است، می‌باشند. به عبارتی، در این روش علاوه بر شباهت متنی از شباهت رابطه‌ای نیز استفاده شده است.

روش۳ *(M3)*: در این روش برای نشان دادن تاثیر متن استناد در اندیس‌گذاری، روش تولید مجموعه کاندید و مرتب‌سازی همان روش۲ می‌باشد، اما برای هر مقاله P_i، مجموعه $\{T_i, abs_i, citctxList_i\}$ اندیس‌گذاری می‌شوند.

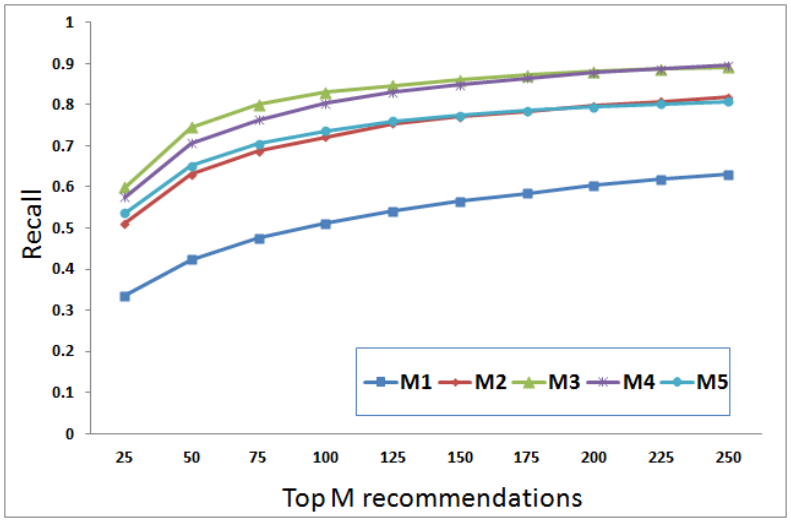
روش۴ *(M4)*: هی و همکارانش در (He et al., 2010)، یک روش مرتب‌سازی در سیستم‌های پیشنهاد استناد ارائه کرده‌اند که مبتنی بر شباهت متنی است و از متن استناد نیز استفاده می‌کند. هی و همکارانش روش خود را با چندین روش قبلی مقایسه کرده و نتیجه گرفته‌اند که این روش از سایر روش‌ها بهتر کار می‌کند. در اینجا، هدف استفاده از روش۴، ارزیابی روش مرتب‌سازی پیشنهادی این پایان‌نامه، از طریق مقایسه آن با کار (He et al., 2010) می‌باشد. در روش۴، برای هر مقاله مجموعه $\{T_i, abs_i, citctxList_i\}$ اندیس‌گذاری می‌شوند. تولید مجموعه کاندید همانند روش۲ است، اما از روش مرتب سازی توضیح داده شده در (He et al., 2010) استفاده می‌شود.

روش۵ *(M5)*: این روش پیاده‌سازی کامل الگوریتم پیشنهادی هی و همکارانش در (He et al., 2010) برای پیشنهاد استناد می‌باشد. هدف استفاده از روش۵، ارزیابی الگوریتم پیشنهادی این پایان‌نامه، از طریق مقایسه آن با کار هی و همکارانش در (He et al., 2010) می‌باشد.

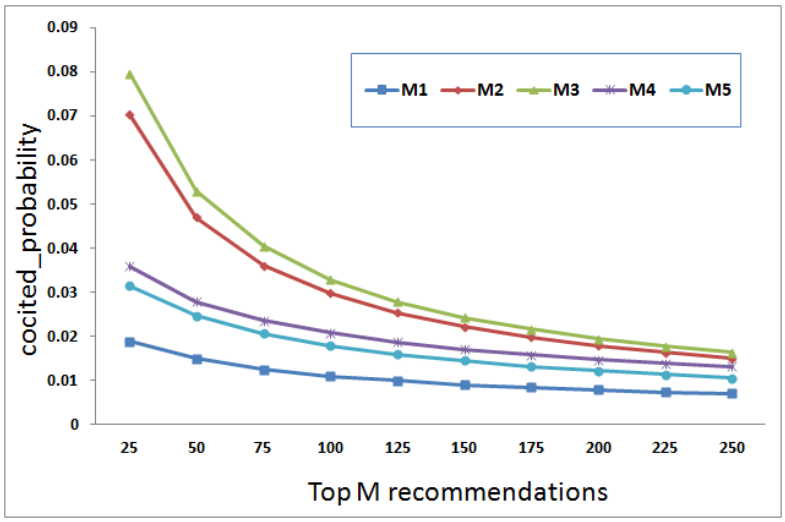
لازم به ذکر است، هی و همکارانش در (He et al., 2010)، برای تولید مجموعه کاندید از بین روش‌های مختلف پیشنهادی خود، با توجه به دو معیار اندازه مجموعه کاندید و پوشش (*coverage*)، روش *LC100+G1000* را انتخاب کرده‌اند، در این روش به ازای هر متن مقاله ورودی P_{input} یعنی $\{T_{input}, abs_{input}, refctxList_{input}\}$، $refctxList_{input}$ به معنی این است برای هر یک از عناصر *LC100* ،۱۰۰ مقاله P_i که *refctxList-i*، بیشترین شباهت را به آن عنصر داشته باشد و *G1000* یعنی ۱۰۰۰ مقاله P_i که عنوان و چکیده‌اش بیشترین شباهت را با متن ورودی داشته باشد. در این روش برای هر مقاله P_i، مجموعه $\{T_i, abs_i, citctxList_i, refctxList_i\}$ اندیس‌گذاری می‌شود.

۴-۳-۲-۲ تحلیل نتایج

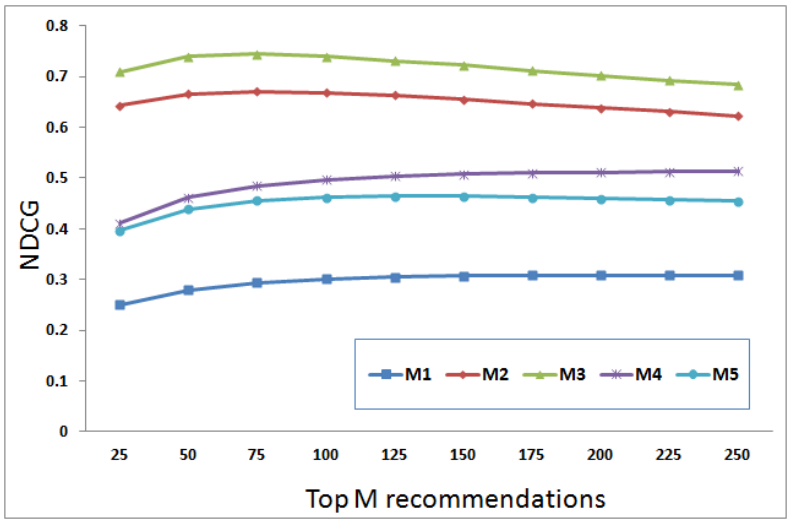
در این قسمت، پنج الگوریتم پیشنهادی که در بخش قبل توضیح داده شد، به کمک سه معیار ارزیابی مورد مقایسه قرار گرفته و نتایج ارزیابی‌ها در شکل ۴-۹ تا شکل ۴-۱۱ آورده شده است.



شکل ۴-۹: مقایسه نتایج از نظر معیار فراخوانی



شکل ۴-۱۰: مقایسه نتایج از نظر معیار استناد مشترک



شکل ۴-۱۱: مقایسه نتایج از نظر معیار NDCG

تفاوت روش۱ و روش۲ در الگوریتم پیشنهاد آنها می‌باشد، در روش۱ تنها از شباهت متنی استفاده شده است اما در روش۲، از ترکیب شباهت متنی و رابطه‌ای استفاده شده است. همان طور که در شکل ۴-۹ تا شکل ۴-۱۱ نشان داده می‌شود، در نتایج حاصل از روش۲ که الگوریتم پیشنهادی این پایان‌نامه است، از نظر هر سه معیار ارزیابی نسبت به روش۱، بهبود قابل ملاحظه‌ای حاصل شده است. با توجه به این مقایسه می‌توان نتیجه گرفت که استفاده از ویژگی‌های رابطه‌ای در کنار ویژگی‌های متنی می‌تواند باعث کاهش ضعف استفاده از شباهت متنی در بدست آوردن مقالات مرتبط با یک متن شود. علت ضعف شباهت متنی در بدست آوردن مقالات مرتبط در بخش ۱-۱ ذکر شده است.

تفاوت روش۲ و روش۳ در استفاده از متن استناد بعنوان یک ویژگی متنی می‌باشد. همانطور که در شکل ۴-۹ تا شکل ۴-۱۱ مشاهده می‌شود، استفاده از متن استناد در روش۳، نقش موثری در پیشنهاد استناد برای یک متن دارد. علت این بهبود در ادامه توضیح داده می‌شود.

$citctxList_i$، لیست متن‌های استناد مقاله P_i، مشخص کننده متن‌هایی از مقالات دیگر می‌باشد که در آن‌ها به P_i استناد شده است، در نتیجه از نظر نویسنده آن مقالات، آن قسمت از متن مقاله با مقاله P_i مرتبط می‌باشد. حال برای بدست آوردن مقالات مرتبط با یک متن، اگر این متن شباهت متنی زیادی با متن‌های استناد مقاله P_i داشته باشد، از آنجایی که متن‌های استناد مقاله P_i با P_i مرتبط می‌باشند در نتیجه آن متن نیز با مقاله P_i مرتبط است. به بیان دیگر استفاده از لیست متن‌های استناد یک مقاله در کنار چکیده و عنوان آن، می‌تواند ضعف روش‌های متنی را در پیدا کردن مقالات مرتبط کاهش دهد و نقش موثری در سیستم‌های پیشنهاد استناد دارد.

تفاوت روش۳ و روش۴ تنها در روش مرتب‌سازی آنها می‌باشد، روش‌مرتب‌سازی در روش۴ از شباهت متنی، و از متن استناد نیز بعنوان یک ویژگی متنی استفاده می‌کند. همانطور که در شکل ۴-۹ تا شکل ۴-۱۱ مشاهده می‌شود روش مرتب‌سازی پیشنهادی این سه از سه معیار ارزیابی نسبت به روش مرتب‌سازی پیشنهادی هی و همکارانش در (He et al., 2010) بهتر عمل کرده است. علت این امر را می‌توان در تفاوت آن‌ها یعنی استفاده از ویژگی‌های رابطه‌ای در روش۳ دانست.

تفاوت روش۴ و روش۵ در روش تولید مجموعه کاندید آن‌ها می‌باشد. به کمک این دو روش می‌توان روش تولید مجموعه کاندید پیشنهادی در این پایان‌نامه را با روش تولید مجموعه کاندید ارائه شده توسط هی و همکارانش در (He et al., 2010) مقایسه کرد. با توجه به شکل ۴-۹ تا شکل ۴-۱۱، روش تولید مجموعه کاندید پیشنهادی این پایان‌نامه با بهره‌گیری از ویژگی‌های رابطه‌ای توانسته است نتایج بهتری را کسب کند. این امر نشان‌دهنده نقش مثبت استفاده از ویژگی‌های رابطه‌ای در کاندید کردن مقالات مرتبط است.

در نهایت مقایسه‌ای نیز بین روش۳ و روش۵ انجام شده است که مقایسه‌ای بین الگوریتم پیشنهادی این پایان‌نامه و الگوریتم پیشنهادی ارائه شده توسط هی و همکارانش در (He et al., 2010) انجام شود. این

مقایسه نیز با توجه به شکل ۴-۹ تا شکل ۴-۱۱، نشان می‌دهد که الگوریتم پیشنهادی این پایان‌نامه به طور کلی از نظر هر سه معیار ارزیابی نسبت به الگوریتم پیشنهادی در (He et al., 2010) منجر به نتایج بهتری شده است.

بطور خلاصه با توجه به پنج مقایسه انجام شده بین روش‌های مختلف، می‌توان به موفق بودن الگوریتم پیشنهادی در پیشنهاد استناد و نقش مثبت متن استناد و استفاده از ویژگی‌های رابطه‌ای در کنار ویژگی‌های متنی برای بدست آوردن کارهای مرتبط اشاره کرد.

۴-۳-۲ ارزیابی استفاده از داده‌های پیوندی

انگیزه استفاده از داده‌های پیوندی در سیستم‌های پیشنهاد استناد در بخش ۳-۱-۱-۱ توضیح داده شد. در ادامه، ابتدا رویکرد مبتنی بر داده‌های پیوندی با رویکرد معمولی مورد مقایسه کیفی قرار گرفته و سپس یک مقایسه کمی انجام می‌شود.

۴-۳-۳-۱ مقایسه کیفی

در جدول ۴-۲، رویکرد مبتنی بر داده‌های پیوندی با رویکرد معمولی، مقایسه کیفی شده است.

جدول ۴-۲: مقایسه کیفی رویکرد مبتنی بر داده‌های پیوندی و رویکرد معمولی

	رویکرد مبتنی بر داده‌های پیوندی	رویکرد معمولی
قالب بازنمایی داده‌ها در منابع داده	مدل داده معنایی (*RDF*)	مدل داده‌های مختلف نظیر *XML*، *HTML*
مکانیزم پرس‌وجو بر روی داده‌ها	اجرای پرس‌وجوی *SPARQL*	مکانیزم‌های مختلف مثل *OAI service*، *web service*، *GUI*، *web API*
طبیعت داده‌ها	داده‌های معنایی قابل پردازش توسط ماشین	داده‌های آماده شده جهت استفاده عامل انسانی
قابلیت استفاده از چند منبع داده	توسط خود قواعد و منابع داده‌های پیوندی فراهم می‌شود	بعهده برنامه کاربردی می‌باشد
ارتباط بین داده‌های منابع مختلف	بطور صریح و قابل پردازش توسط ماشین، در منابع داده‌های پیوندی بیان شده است	منابع داده از هم جدا هستند و بین آنها ارتباط‌های صریح و قابل پردازش توسط ماشین وجود ندارد
قابلیت استخراج دانش	وجود دارد	وجود ندارد
بلوغ داده‌های موجود در منابع داده	برخی از داده‌ها موجود نمی‌باشد	داده‌ها تا حد زیادی موجود می‌باشد

همانطور که از این جدول نتیجه گرفته می‌شود، یکی از مزایای استفاده از داده‌های پیوندی، استفاده آسان از چند منبع داده مختلف و در نتیجه غنی شدن لایه داده‌ها می‌باشد. داده‌های یک سیستم پیشنهاد استناد که مبتنی بر یک مجموعه داده محلی می‌باشد، محدود به استفاده از داده‌های همان مجموعه می‌باشد، و بهره‌گیری از داده‌های مجموعه داده‌های دیگر، به دلیل ناهمگونی نحوه انتشار داده‌ها توسط منابع داده مختلف، با پیچیدگی‌های زیادی روبرو می‌باشد.

با توجه به ماهیت انتشار مقالات بر روی منابع مختلف و انتشار اطلاعات توسط منابع داده متفاوت، استفاده از داده‌های پیوندی سبب می‌شود بواسطه قالب یکسان بازنمایی داده‌ها، و با دنبال کردن پیوندهای صریح و معناداری که بین مجموعه‌داده‌های مختلف وجود دارد، استفاده از داده‌های منابع گوناگون ساده شود.

متاسفانه، در حال حاضر منابع موجود داده‌های پیوندی از کیفیت داده مناسبی برخوردار نمی‌باشند. یعنی منابع داده‌ای که داده‌های‌شان را هم بصورت داده‌های پیوندی و هم بصورت معمولی منتشر کرده‌اند، در نسخه داده‌های پیوندی، داده‌های کمتری را پوشش داده‌اند و برخی از اجزای اطلاعاتی را نیز منتشر نکرده‌اند. در نتیجه در حال حاضر اگر در لایه داده فقط از منابع داده‌های پیوندی استفاده شود لایه داده از غنای مناسبی برخوردار نخواهد بود.

۴-۳-۳-۲ مقایسه کمی

همانطور که در بخش ۴-۲-۱ توضیح داده شد، برای آماده سازی لایه داده، داده‌های منبع داده *CiteseerX* به کمک سرویس آن جمع‌آوری شده و به صورت داده‌های پیوندی بازنمایی شده است. همچنین برای رفع مشکل کمبود داده‌ها، در بخش ۳-۱-۱-۲ یک الگوریتم غنی‌سازی ارائه شد. در ادامه ابتدا نحوه پیاده-سازی این الگوریتم، و سپس نتایج آزمایش‌های مربوطه شرح داده شده است.

در پیاده‌سازی این الگوریتم، به منظور غنی‌سازی مجموعه داده جمع‌آوری شده از *CiteSeerX*، سه منبع داده پیوندی از ابر *LOD* به نام‌های *IEEE*، *DBLP* و *ACM* انتخاب شدند. علت انتخاب این منابع داده از ابر *LOD* را می‌توان در سه مورد زیر خلاصه کرد:

۱- آن‌ها منابع اصلی در زمینه اطلاعات کتاب شناسی در حوزه علوم کامپیوتر و مهندسی می‌باشند.

۲- استفاده از آن‌ها با توجه به اطلاعاتی که منتشر می‌کردند برای غنی سازی داده‌ها امیدوارکننده بود. مثلا *IEEE* کلمات کلیدی و چکیده مقالات را منتشر می‌کرد، *ACM* دسته‌بندی هر مقاله که به صورتی بعنوان کلمات کلیدی کاربرد داشت و *DBLP* شامل تعداد زیادی مقاله می‌شد و از نظر فراداده‌هایی از جمله نام نویسندگان، سال و محل انتشار بسیار غنی بود.

۳- از آنجایی که همه آن‌ها از یک هستان‌شناسی ،*AKT*[1] ، برای انتشار داده‌های خود استفاده می‌کردند و شامل نقاط پایانی برای دسترسی به داده‌های خود می‌باشند، در نتیجه نوشتن یک پرس‌جوی *SPARQL* می‌توانست بر روی هر سه اجرا شود و با توجه به مزایای داده‌های پیوندی نیازی به نوشتن یک پیمایش‌گر خاص برای جمع‌آوری داده‌ها از هر یک از این منابع داده وجود نداشت.

در پیاده‌سازی الگوریتم غنی‌سازی داده‌ها که شبه‌کد آن در شکل ۳-۲ و شکل ۳-۳: نشان داده شده است، تابع *findmatch* از الگوریتم فاصله *Levenshtein* (Vladimir, 1966) به منظور محاسبه شباهت بین عنوان دو مقاله استفاده می‌کند، و اگر شباهت *Levenshtein* عنوان دو مقاله بیشتر از یک حد آستانه باشد این دو مقاله معادل محسوب می‌شوند در این تابع مقدار حد آستانه *0.9* در نظر گرفته شده است.

همچنین قابل ذکر است که با توجه به بررسی کیفی داده‌های موجود در مجموعه داده محلی، معنی *missing* در این الگوریتم برای هر مشخصه از مقاله به صورت مجزا در نظر گرفته شده است. برای مثال، در مورد مشخصه سال انتشار مقاله، داده‌هایی که طولشان کمتر از ۴، و یا مقدارشان برابر "*Null*" یا تهی بودند، نقص داده محسوب شده‌اند. از نظر مشخصه چکیده داده‌هایی با طول کمتر از ۵۰ حرف، نقص داده محسوب می‌شوند، در حالیکه در مورد مشخصه عنوان، طول کمتر از ۱۰ در نظر گرفته شده است.

بعد از اجرای الگوریتم غنی‌سازی تعدادی از نقص داده‌ها برطرف شد که در جدول ۴-۳ نتایج مربوط به مشخصه‌هایی که بهبود یافته‌اند، نشان داده شده است.

جدول ۴-۳: نتایج الگوریتم غنی‌سازی از نظر میزان بهبود مقادیر مشخصه‌های مقالات

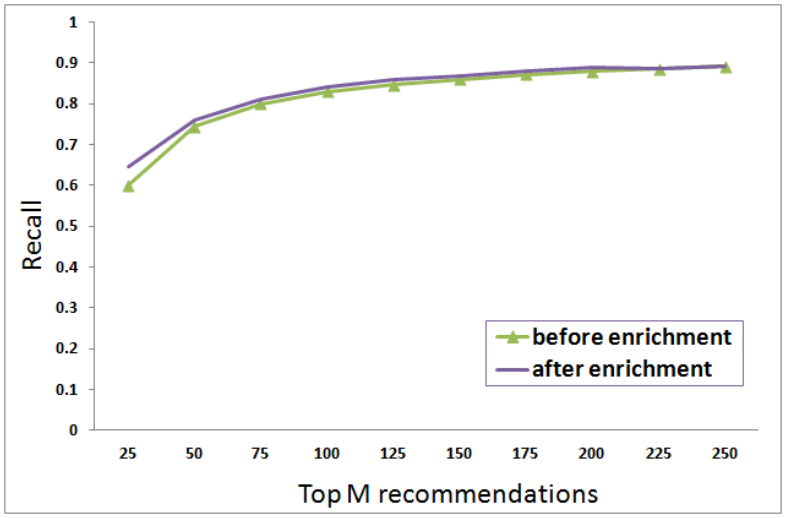
مشخصه‌های مقاله	چکیده	کلمات کلیدی	مکان انتشار	سال انتشار	نویسندگان
بهبود (٪)	۱۵	۳۰	۲۱	۷۸	۲۷

[1] http://www.aktors.org/ontology

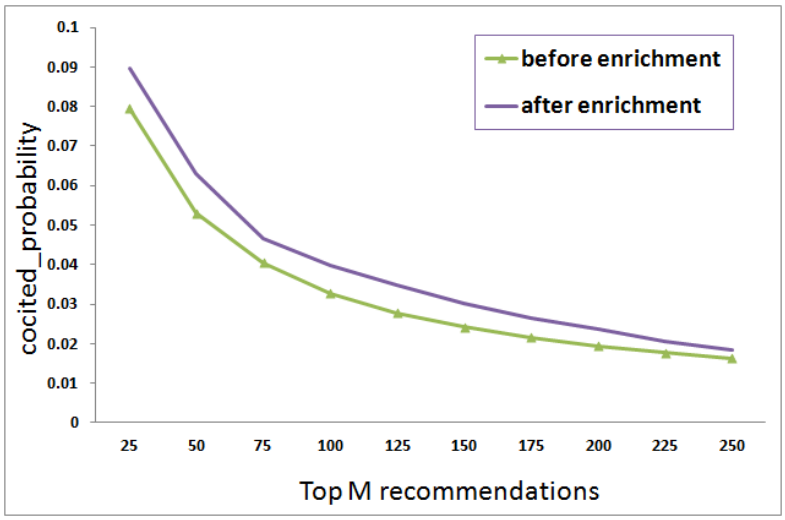
همانطور که در جدول ۴-۳ مشاهده می‌شود میزان بهبود مشخصه سال انتشار، از بقیه مشخصه‌ها بیشتر بوده است و برخی از مشخصه‌ها، مانند عنوان و لیست مراجع مقالات بهبود نداشته است. از آنجا که اساس پیدا کردن مقالات معادل، مقایسه و شباهت عنوان مقالات است در نتیجه اگر در مجموعه داده محلی مقاله‌ای وجود داشته باشد که عنوانش مقدار معتبری نداشته باشد قطعا هیچ معادلی برای آن پیدا نخواهد شد و در نتیجه روال غنی‌سازی مورد نظر، قادر به رفع نقص عنوان مقاله‌ها نمی‌باشد.

درمورد مشخصه‌های لیست مراجع و استنادها نیز علت بهبود نیافتن، این است که داده‌های پیوندی موجود از نظر انتشار اطلاعات مربوط به مراجع و استنادها خیلی ضعیف می‌باشند. علت بهبود قابل ملاحظه سال انتشار، نام نویسنده و محل انتشار این است که منبع داده *DBLP* مقالات زیادی را به صورت داده‌های پیوندی منتشر کرده است و اطلاعات آن‌ها از نظر مشخصه‌های فوق، خیلی غنی هستند. در مورد کلمات کلیدی نیز از آنجاییکه قبلا هیچ کلمه کلیدی وجود نداشت تقریبا برای هر مقاله‌ای که معادلش در *IEEE* یا *ACM* پیدا شده است، بهبود یافته است.

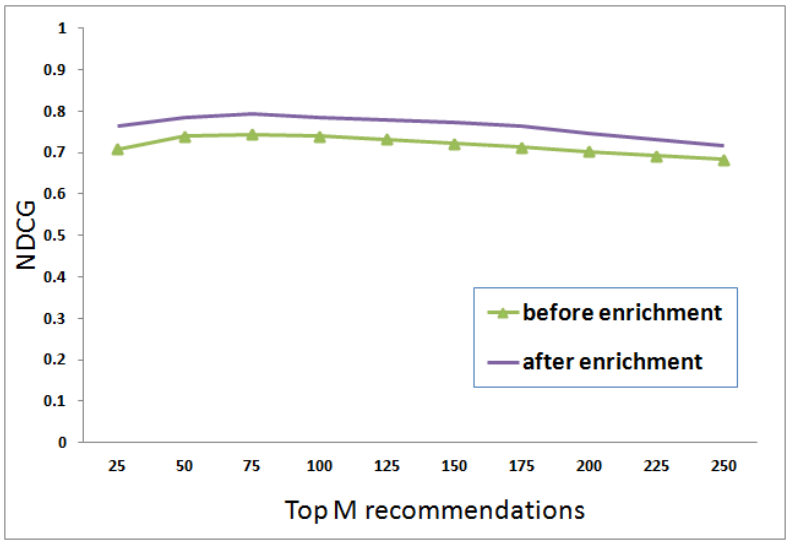
برای نشان دادن میزان تاثیر این بهبود داده‌ها در سیستم پیشنهاد استناد، آزمایش دیگری نیز در نظر گرفته شد که یک بار الگوریتم پیشنهاد استناد روی مجموعه داده‌ها قبل از غنی‌سازی، و یک بار نیز پس از غنی‌سازی اجرا شد. نتایج این آزمایش، در شکل ۴-۱۲ تا شکل ۴-۱۴ از نظر سه معیار ارزیابی آورده شده است.



شکل ۴-۱۲: نتایج حاصل از الگوریتم غنی‌سازی از نظر معیار فراخوانی



شکل ۴-۱۳: نتایج حاصل از الگوریتم غنی‌سازی از نظر معیار استناد مشترک



شکل ۴-۱۴: نتایج حاصل از الگوریتم غنی‌سازی از نظر معیار NDCG

همانطور که در شکل ۴-۱۲ تا شکل ۴-۱۴ نشان داده شده است، استفاده از الگوریتم غنی‌سازی بمنظور رفع نقص‌های موجود در مجموعه داده به کمک داده‌های موجود بر روی ابر LOD باعث بهبود پیشنهادها در $M=25$ از نظر هر سه معیار ارزیابی داشته است، و در مقادیر دیگر M، تنها از نظر معیار استناد مشترک و $NDCG$ بهبود حاصل شده است. با توجه به اینکه بهبود در تعداد M کمتر از اهمیت بالاتری برخوردار است، در حال کلی الگوریتم غنی‌سازی توانسته است کیفیت سیستم پیشنهادی را افزایش دهد، اما این بهبود خیلی زیاد نیست.

این بهبود کم، نشان دهنده تاثیر نداشتن استفاده از مجموعه داده‌های مختلف در سیستم‌های پیشنهاد استناد نمی‌باشد، بلکه علت آن کمبود داده در داده‌های پیوندی و کیفیت نداشتن آن‌ها می‌باشد. همانطور که در جدول ۴-۳، نشان داده شد، داده‌های موجود در ابر LOD نتوانسته‌اند باعث بهبود زیادی در مشخصه‌های تاثیر گذار در الگوریتم پیشنهادی باشند. در نتیجه آن طور که انتظار می‌رفت کیفیت پیشنهادها بهبود نیافت.

۴-۴ خلاصه فصل

در این فصل، جزئیات پیاده‌سازی‌های انجام شده برای تولید سیستم پیشنهادی توضیح داده شد و سپس به منظور ارزیابی این سیستم دو آزمایش در نظر گرفته شد. در آزمایش اول الگوریتم پیشنهاد، با روش‌های دیگر مورد مقایسه قرار گرفت، و در آزمایش دوم، ابتدا یک ارزیابی کیفی از استفاده از داده‌های پیوندی در لایه داده انجام شد و سپس تاثیر الگوریتم غنی سازی داده‌ها که در بخش ۳-۱-۱-۲ توضیح داده شده بود، در سیستم پیشنهادی سنجیده شد.

فصل ۵- نتیجه گیری و کارهای آینده

در این پایان‌نامه، یک سیستم پیشنهاد استناد ارائه شده است که می‌تواند پژوهشگران را در پیدا کردن کارهای مرتبط با یک زمینه تحقیقاتی کمک کند.

چارچوب سیستم پیشنهادی از سه بخش اصلی تشکیل شده است:

۱. داده‌های پیش‌زمینه: مجموعه‌ای از مقالات می‌باشند و در سیستم پیشنهادی، بمنظور سهولت دسترسی به داده‌ها و استفاده از داده‌های چند منبع، از داده‌های پیوندی استفاده شده است.

۲. داده‌های ورودی: ورودی سیستم پیشنهادی یک متن می‌باشد و سیستم با دریافت آن، لیستی از اسنادی که آن متن باید به آنها استناد کند را پیشنهاد می‌کند.

۳. الگوریتم پیشنهاد: در سیستم پیشنهادی یک معیار جدید به نام شباهت رابطه‌ای برای بدست آوردن میزان شباهت دو مقاله معرفی شده است، سپس با ترکیب این معیار و معیار شباهت متنی، یک الگوریتم پیشنهاد استناد ارائه شده است. در شباهت متنی نیز از متن استناد به عنوان یک ویژگی متنی استفاده شده است تا نقش آن در بهبود سیستم‌های پیشنهاد استناد سنجیده شود. لازم به ذکر است وزن هر ویژگی رابطه‌ای در الگوریتم پیشنهادی بصورت خودکار و به کمک الگوریتم ژنتیک بدست آمده است

بر اساس ارزیابی‌های انجام شده نتایج زیر بدست آمده است:

۱- استفاده از ترکیب ویژگی‌های رابطه‌ای و متنی، در مقایسه با استفاده تنها از ویژگی‌های متنی، منجر به نتایج بهتری می‌شود.

۲- متن استناد بعنوان یک ویژگی متنی، نقش مؤثری در بهبود کیفیت سیستم‌های پیشنهاد استناد دارد.

۳- ترکیب ویژگی‌های رابطه‌ای و ویژگی‌های متنی، بهمراه استفاده از متن استناد در ویژگی‌های متنی، در مقایسه با حالتی که فقط از ویژگی‌های متنی بعلاوه متن استناد استفاده شود، و همچنین نسبت به

حالتی که از ترکیب ویژگی‌های رابطه‌ای و ویژگی‌های متنی اما بدون متن استناد استفاده شود، نتایج بهتری تولید می‌کند.

۴- علیرغم مزایای نظری استفاده از داده‌های پیوندی، بعلت آنکه در حال حاضر داده‌های موجود بر روی ابر LOD، در مقایسه با داده‌های موجود بر روی وب اسناد از کیفیت خوبی برخوردار نیستند، استفاده از داده‌های ابر LOD، تأثیر کمی در بهبود نتایج سیستم پیشنهاد استناد دارد. از طرفی با توجه به رشد سریع انتشار داده‌های پیوندی، بنظر می‌رسد که در آینده نزدیک وضعیت بهتر شود. بنابراین یک سیستم پیشنهاد استناد که قادر باشد در لایه داده خود از داده‌های پیوندی استفاده کند می‌تواند از مزایای داده‌های پیوندی بهره‌مند شود.

۵- یکی از ویژگی‌های مهم داده‌های پیوندی وجود ارتباطات صریح و معنادار بین داده‌های منابع مختلف می‌باشد. با توجه به ماهیت انتشار مقالات توسط منابع داده مختلف، استفاده از داده‌های پیوندی می‌تواند در سیستم‌های پیشنهاد استناد باعث غنی شدن داده‌ها و بهبود کیفیت سیستم شود.

۵-۱ کارهای آینده

کارهای آتی که می‌توان در ادامه فعالیت صورت گرفته در این پروژه به آنها پرداخت عبارتند از:

- تولید سیستم پیشنهاد دهنده‌ای که قادر باشد با گرفتن یک متن ورودی که مکان‌های خاصی از آن علامت‌گذاری شده است، استنادهای مناسبی برای آن مکان‌ها پیشنهاد کند. چنین سیستمی می‌تواند به نویسندگان مقالات کمک کند تا برای قسمت‌هایی از متن مقاله خود استنادهای مناسب بدست آورند.
- افزودن قابلیت ارائه توضیحات درباره پیشنهادها، که این امر موجب افزایش اعتماد کاربر به سیستم می‌شود.
- بهبود معیار شباهت رابطه‌ای بگونه‌ای که از اطلاعات بیشتری استفاده کند. بعنوان مثال اطلاعات مربوط به اینکه هر مقاله موجود در داده‌های پیش‌زمینه چند مرتبه به مقاله دیگری

استناد کرده است، یا اینکه این استنادها در کدام قسمت از آن مقاله انجام شده است، می‌تواند مورد توجه قرار گرفته و تأثیر آنها بررسی شود.

- پیاده‌سازی یک نسخه عملیاتی و کاربرپسند از سیستم پیشنهادی ارائه شده برای استفاده کاربران نهایی
- تعریف قواعد معنایی بگونه‌ای که امکان استنتاج بر روی داده‌های پیش‌زمینه فراهم شود. نتایج حاصل از استنتاج می‌تواند با افزودن اطلاعات مفید به داده‌های پیش‌زمینه کارهای الگوریتم پیشنهاد را بهبود بخشد.
- بهبود الگوریتم غنی‌سازی به منظور بهره‌گیری بهتر از داده‌های منابع مختلف

فهرست مراجع

Adomavicius, G., Tuzhilin, A., "*Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions*", Journal of IEEE Transaction Knowledge Data Engineering, Vol. 17, No. 6, pp. 734-749, 2005.

Aljaber, B., Stokes, N., Bailey, J. and Pei, J., "*Document clustering of scientific texts using citation contexts*", Journal of Information Retrieval, Vol. 13, No. 2, pp. 101-131, 2010.

Baeza-Yates, R., "*Challenges in the Interaction of Information Retrieval and Natural Language Processing*", In Proceedings of 5 th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing 2004), Seoul , Corea. Lecture Notes in Computer Science vol. 2945, pp. 445-456, Springer, 2004.

Basu, C., Hirsh, H., Cohen, W., Nevill-Manning, C., "*Technical paper recommendations: a study in combining multiple information sources*", Journal of Artificial Intelligence Research (JAIR), Vol. 1, pp. 231-252, 2001.

Berners-Lee, T., "*Linked Data. Design Issues for the World Wide Web*", http://www.w3.org/DesignIssues/LinkedData.html, 2006.

Bethard, S., Dan Jurafsky, D., "Who should I cite? Learning literature search models from citation behavior", In Proceedings of ACM Conference on Information and Knowledge Management, pp. 609-618, 2010.

Bizer, C., Heath, T., Berners-Lee, T., "*Linked Data - The Story So Far*", Journal of Semantic Web and Information Systems, Vol. 5 No.3, pp. 1-22, 2009.

Bogers, T., Bosch, A., "*Recommending Scientific Articles Using CiteULike*", In Proceedings of the ACM conference on Recommender systems, pp. 287-290, 2008.

Bornmann, L., Daniel, H.D, "*What do citation counts measure? A review of studies on citing behavior*", Journal of Documentation, Vol. 64 No. 1, pp. 45-80, 2008

Burke, R., "*Hybrid Recommender Systems: Survey and Experiments*", Journal of User Modeling and User-Adapted Interaction, Vol. 12 No. 4, pp. 331-370, 2002.

Buckley, C., Voorhees, E. M., "*Retrieval evaluation with incomplete information*", In Proceedings of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2004), Sheffield, UK, pp. 25–32, 2004.

Chandrasekaran, K., Gauch, S., Lakkaraju, P., Luong, H.P, "*Concept-Based Document Recommendations for CiteSeer Authors*", In Proceedings of the 5th international conference on Adaptive Hypermedia and Adaptive Web-Based Systems, July 29-August 01, 2008.

Couto, T., Ziviani, N., Calado, P., Cristo, M., Gonçalves, M., Moura, E.S.D., Brandão, W.C., "*Classifying documents with link-based bibliometric measures*", Journal of information retrieval, Vol. 13 No. 4, pp. 315-345, 2010.

Garfield, E., "*Citation analysis as a tool in journal evaluation*", Journal of Science, Vol. 178 No. 1972, pp. 471-479, 1972.

Gipp, B., Beel, J., Hentschel, C., "*Scienstein: A Research Paper Recommender System*", In Proceedings of the International Conference on Emerging Trends in Computing (ICETiC'09), pp. 309-315, 2009.

Glaser, H., Millard, I., Jari, A., Lewy, T., Dowling, B., "*On coreference and the Semantic web*", In Proceedings of 7'h International Semantic Web Conference (ISWC 2008), Karlsruhe, Germany, 2008.

Goldberg, E.D., "*Genetic Algorithms in Search, Optimization & Machine Learning*", Addison-Wesley, Reading, MA, 1989.

Gori, M., Pucci, A., "*Research Paper Recommender Systems: A Random-Walk Based Approach*", In Proceedings of the 2006 International Conference on Web Intelligence, pp. 778-781, 2006.

He, Q., Pei, J., Kifer, D., Mitra, P., Giles, C.L., "*Context-aware Citation Recommendation*", In Proceedings of the 19th International World Wide Web Conference (WWW), pp. 421–430, 2010.

He, Q., Kifer, D., Pei, J., Mitra,P., Giles, C.L., "*Citation recommendation without author supervision*", In Proceedings of WSDM'11, pp. 755-764, 2011.

Heitmann, B., Hayes, C., "*Using Linked Data to build open, collaborative recommender systems*", In: Proceedings of the AAAI Spring Symposium "Linked Data Meets Artificial Intelligence", pp. 76-81, 2010.

Henzinger, M.R., Motwani, R., Silverstein, C., "*Challenges in web search engines*", In Proceedings of the 18th international joint conference on artificial intelligence, pp. 1573-1579, 2003.

Kessler, M., "*Bibliographic coupling between scientific papers",* Journal of American Documentation, Vol. 14 No. 1, pp. 10–25, 1963.

Lafferty, J., Lebanon, G., "*Diffusion kernels on statistical manifolds*", JMLR, Vol. 6, pp. 129–163, 2005.

Liben-Nowell, D., Kleinberg, J., "*The link prediction problem for social networks*", In Proceedings of the 12th International Conference on Information and Knowledge Management, pp. 556–559, 2003.

McCandless, Michael, Erik Hatcher, and Otis Gospodnetić, "*Lucene in Action*", Second Edition. Manning Press, 2010.

McNee, S., Albert, I., Cosley, D., Gopalkrishnan, P., Lam, S., Rashid, A., Konstan, J., Ried, J., "*On the Recommending of Citations for Research Papers*", In Proceedings of CSCW'02, 2002.

Middleton, S. E., Shadbolt, N. R., De Roure, D. C., "*Ontological user profiling in recommender systems*", Journal of ACM Transactions on Information Systems, Vol. 22 No. 1, pp. 54-88, 2004.

O'Connor, J., "*Citing statements: Computer recognition and use to improve retrieval*", Journal of Information Processing and Management Vol. 18 No. 3, pp. 125–131, 1982.

Pudhiyaveetil, A.K., Gauch S., Luong, H.P., Josh Eno J., "*Conceptual recommender system for CiteSeerX*", In Proceedings of RecSys'2009. pp. 41-244, 2009.

Passant, A., Heitmann, B., Hayes, C., "*Using linked data to build recommender systems*", In Proceedings of RecSys '9 New-York, NY USA, 2009.

Passant, A., "*dbrec - Music Recommendations Using DBpedia*", In Proceedings of International Semantic Web Conference (2), pp. 209-224, 2010.

Ritchie, A., "*Citation context analysis for information retrieval*", PhD thesis, University of Cambridge, 2008.

Ritchie, A., Robertson, S., Teufel, S., "*Comparing citation contexts for information retrieval*", In Proceedings of the 17th ACM conference on Information and knowledge management, pp. 213-222, 2008.

Ritchie, A., Teufel, S., and Robertson. S., "*Using Terms from Citations for IR: Some First Results*", In Proceedings of ECIR, pp. 211–221, 2008.

Salton, G., Buckley, C., "*On the use of spreading activation methods in automatic information*", In Proceedings of SIGIR '88, pp. 147–160, New York, NY, USA, 1988. ACM Press.

Schafer, B., Frankowski, D., Herlocker, J., Sen, S., "*Collaborative filtering recommender systems*", In Brusilovsky, P., Kobsa, A., Nejdl, W., eds., The Adaptive Web: Methods and Strategies of Web Personalization. Lecture Notes in Computer Science, Vol. 4321, Berlin Heidelberg New York, Springer-Verlag, 2007.

Shabir, N. and Clarke, C., "*Using Linked Data as a basis for a Learning Resource Recommendation System*", In 1st International Workshop on Semantic Web Applications for Learning and Teaching Support in Higher Education (SemHE'09), ECTEL'09, Nice, France, 2009.

Small, H., "*Citation context analysis*", In B. Dervin & M. J. Voigt, eds, Progress in Communication Sciences, Vol. 3, Ablex Publishing, pp. 287-310, 1982.

Small, H., "*Co-citation in the scientific literature: A new measurement of the relationship between two documents*", Journal of the American Society of Information Science Vol. 24 No. 4, 265–269, 1973.

Smith, L.C., "*Citation analysis*", Journal of Library Trends, Vol. 30 No. 1, pp. 83-106., 1981.

Strohman, T., Croft, W. B., Jensen, D., "*Recommending citations for academic papers*", In Proceedings of the 30th Annual ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)', ACM Press, pp. 705–706, 2007.

Tang, J., Zhang, J., "*A Discriminative Approach to Topic-Based Citation Recommendations*", In Proceedings of PAKDD'09, 2009.

Torres, R., McNee, S. M., Abel, M., Konstan, J.A., Riedl, J., "*Enhancing digital libraries with TechLens*", In Proceedings of IEEE/ACM Joint Conference on Digital Libraries (ACM/IEEE JCDL'2004), Tuscon, AZ, USA, pp. 228-236, 2004.

Vladimir I. Levenshtein, "*Binary codes capable of correcting deletions, insertions, and reversals*", Journal of Soviet Physics Doklady, Vol. 10 No. 8, 1966.

Woodruff, A., Gossweiler, R., Pitkow, J., Chi, E., Card, S., "*Enhancing a digital book with a reading recommender*", In Proceedings of ACM Conference on Human Factors in Computing Systems (ACM CHI'00), The Hague, The Netherlands, pp. 153-160, 2000.

واژه‌نامه انگلیسی به فارسی

Background data	داده‌های پیش‌زمینه
Bibliographic coupling	زوج کتابشناختی
Bookmarking	نشانه‌گذاری
Citation	استناد
Co_citation	استناد مشترک
Collaborative filtering	فیلتر همبستگی
Crawler	پویشگر
Endpoint	نقطه پایانی
Enrichment	غنی‌سازی
Licence	گواهی
Linked Data	داده‌های پیوندی
Relational similarity	شباهت رابطه‌ای
Semantic distance	فاصله معنایی

Abstarct

Large and growing number of scientific publications available on the web, has made it difficult for researcher to select publications that are most related to their fields of interests. The traditional approach used by most researchers for searching related documents is using popular search engines, like Google, and performing a keyword-based search. Since it is hard to choose the appropriate keywords that include all the publications of a specific domain, the keyword-based approach has the disadvantage that it is very sensitive to the specified keywords and it is possible to simply miss some good results.

A citation recommendation system accepts a text as input, and generates a list of publications that the input text should cite. Therefore, it helps a researcher to find papers related to a specific subject.

Currently, citation recommendation systems are limited to use a single local data source for generating recommendations. This limitation reduces the quality of recommendations, since there is no single bibliographic data source which covers all existing publications.

In this thesis, a new citation recommendation system is presented which uses the Linked Data in its data layer, and a novel recommendation algorithm based on a combination of relational and textual similarity. Experimental evaluation shows that using Linked Data in the data layer reduces the complexities of integrating data from multiple sources, and leads to a richer data layer. This is due to the benefits of Linked Data, e.g. the standard publication format and the explicit typed links between the data of different data sources. Further, evaluation results support the claim that the proposed relational similarity measure is successful in determining similarity of publications, and combining it by textual similarity leads to improving quality of the citation recommendation system. It also remedies the weaknesses of textual similarity in finding related publications.

Keywords: information retrieval, recommendation, citation, citation analysis, relational similarity, Linked Data

Engineering Faculty

Department of Computer Engineering

A Linked Data Based Citation Recommendation System

Fattane Zarrinkalam

Supervisor: Dr. Mohsen Kahani

Dissertation submitted in pursuance of the degree of Master of computer software engineering

January 2012