پایان‌نامه کارشناسی ارشد

استخراج قوانین انجمنی از جریان‌های داده معنایی

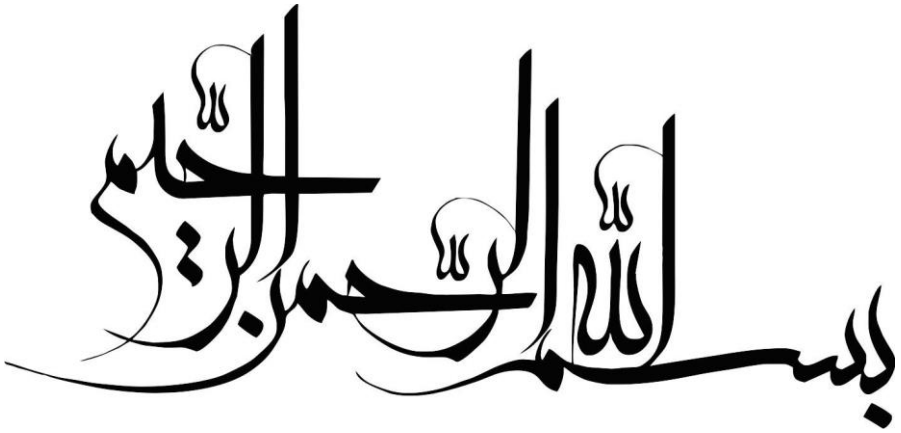
نگارش :

اشرف السادات حیدری یزدی

استاد راهنما:

جناب آقای پروفسور محسن کاهانی

شهریور ۱۳۹۲


بسم الله الرحمن الرحيم

تقدیر و سپاس

منّت خدای را عز و جل که طاعتش موجب قربت است و به شکر اندرش مزید نعمت. بدین وسیله بر خود لازم می‌دانم که از زحمات بی‌دریغ استاد گرامی، جناب آقای پروفسور کاهانی و اعضای آزمایشگاه فناوری وب دانشگاه فردوسی مشهد که راهنمایی‌های ارزشمندشان در تمام مراحل انجام این پایان‌نامه راه‌گشای من بوده است، تشکر نمایم. همچنین از تلاش‌های بی‌وقفه و پشتیبانی‌های خانواده عزیزم کمال سپاس‌گزاری را دارم.

چکیده

روز به روز در کاربردهای بیشتری از قبیل مدل‌سازی ترافیک، شبکه‌های حسگر و دنبال کردن در کاربردهای نظامی، پردازش داده‌های آنلاین و غیره، شاهد تولید هر روزه حجم عظیمی از داده‌های جریان‌دار هستیم. کشف دانش بهینه از این جریان‌های داده، یک حوزه تحقیقاتی فعال در داده‌کاوی با کاربردهای متفاوت بوجود آورده است. برخلاف داده‌های ایستای موجود در پایگاه داده‌های سنتی، داده‌های جریان‌دار اغلب به صورت پیوسته با سرعت بالا و با حجم زیاد و همچنین با توزیع داده متغیر دریافت می‌شود. از طرفی مقادیر آنتولوژی‌ها و حاشیه‌نویسی‌های معنایی موجود بر روی داده‌ها نیز به طور پیوسته در حال افزایش می‌باشد. این نوع داده‌های پیچیده و ناهمگن چالش‌های جدیدی را در حوزه تحقیقاتی داده‌کاوی بوجود آورده است. اصلی‌ترین چالش در این مسأله، نیاز فناوری‌های کاوش قوانین انجمنی به تراکنش‌ها می‌باشد، در حالی که در داده‌های معنایی تعریف دقیقی از تراکنش وجود ندارد. کارهای مشابهی که در این زمینه انجام گردیده‌اند، اغلب با کمک کاربر تراکنش‌ها را برای داده‌های معنایی تعریف کرده و سپس به کمک الگوریتم‌های داده‌کاوی سنتی اقدام به کاوش قوانین انجمنی از این تراکنش‌ها می‌نمایند و لازمه این کار تسلط کاربر به ساختار داده‌های معنایی و دامنه کاربرد مورد نظر می‌باشد. لذا روشی مورد نیاز است که انسجام معنایی را در تمام مراحل کار برقرار سازد و تعریفی کلی و یکپارچه برای تراکنش‌ها و سایر مفاهیم مرتبط ارائه دهد. بنابراین در این تحقیق به ارائه روشی جهت رفع چالش‌های ذکر شده و امکان‌پذیر ساختن پردازش حجم وسیع داده‌های معنایی جریان‌دار موجود و کشف و ذخیره‌سازی قوانین انجمنی جدید در سطح معنایی بالاتر با استفاده از غنای معنایی مفاهیم موجود در آنتولوژی و بهره‌وری از دیگر فناوری‌های معنایی پرداخته شده است.

کلید واژه‌ها - فناوری‌های وب معنایی، حاشیه نویسی معنایی، آنتولوژی، جریان‌های داده، داده کاوی، کاوش قوانین انجمنی

فهرست مطالب

فهرست جداول

فهرست اشکال

فهرست علائم و اختصارات

SSN	Semantic Sensor Network
ARM	Association Rule Mining
C-SPARQL	Continuous SPARQL
RDFS	Resource Description Framework Schema.
RDF	Resource Description Framework.
SWRL	Semantic Web Rules Language
SPARQL	SPARQL Protocol and RDF Query Language.

فصل ۱– مقدمه

۱–۱– تعریف مسأله

دنیای امروز دنیای اطلاعات است و کشف دانش جدید از داده‌های موجود کمک بسیاری جهت پیشبرد اهداف سازمان‌های مختلف می‌باشد. به علاوه روز به روز در کاربردهای بیشتری از قبیل شبکه‌های حسگر، کاربردهای نظامی، پردازش داده‌های آنلاین و غیره، تولید هر روزه حجم عظیمی از داده‌های جریان‌دار صورت می‌گیرد. کشف دانش بهینه از این جریان‌های داده، یک حوزه تحقیقاتی فعال در داده‌کاوی با کاربردهای متفاوت بوجود آورده است. از طرفی مقادیر آنتولوژی‌ها و حاشیه‌نویسی‌های معنایی موجود بر روی داده‌ها نیز به طور پیوسته در حال افزایش می‌باشد. این نوع داده‌های پیچیده و ناهمگن نیز چالش‌های جدیدی را در حوزه تحقیقاتی داده‌کاوی بوجود آورده است. بنابراین در این تحقیق به رفع چالش‌های ذکر شده و امکان‌پذیر ساختن پردازش حجم وسیع داده‌های معنایی جریان‌دار موجود و کشف و ذخیره‌سازی قوانین انجمنی جدید در سطح معنایی بالاتر با استفاده از غنای معنایی مفاهیم موجود در آنتولوژی و بهره‌وری از دیگر فناوری‌های معنایی پرداخته شده است.

در سال‌های اخیر توجه بیشتری به ترکیب دو زمینه تحقیقاتی وب معنایی و داده کاوی صورت گرفته است. با کمک استانداردسازی زبان‌های آنتولوژی RDF(/S) و OWL ، وب معنایی توسعه بیشتری یافته و حجم منابع داده‌ای که شامل حاشیه‌نویسی‌های معنایی هستند، دارای روندی رو به رشد می‌باشد. این ترکیب در اغلب کارهای انجام شده، در قالب یادگیری آنتولوژی و کاوش وب بوده است. اما کار کمی در زمینه کاوش و استخراج دانش از خود داده‌های معنایی صورت گرفته است. کاوش داده‌های معنایی فواید بسیاری را برای جوامع تحقیقاتی خاص دامنه‌ای که داده‌های آنها اغلب پیچیده و ناهمگن است و حجم زیادی دانش در قالب آنتولوژی و حاشیه‌نویسی‌های معنایی موجود است، به بار خواهد داشت.

از طرف دیگر روز به روز در کاربردهای بیشتری شاهد حجم عظیمی از داده‌های جریان‌دار می‌باشیم. کشف دانش بهینه از این جریان‌های داده، یک حوزه تحقیقاتی فعال در داده‌کاوی با کاربردهای متفاوت بوجود آورده

است. برخلاف داده‌های ایستای موجود در پایگاه داده‌های سنتی، داده‌های جریان‌دار اغلب به صورت پیوسته با سرعت بالا و با حجم زیاد و همچنین با توزیع داده متغیر دریافت می‌شوند. این ویژگی‌های داده‌های جریان‌دار مسائل جدیدی را بوجود می‌آورد که لازم است در توسعه تکنیک‌های کاوش قوانین انجمنی از داده‌های جریان‌دار مورد توجه قرار گیرند. به دلیل این ویژگی‌های منحصر به فرد داده‌های جریان‌دار، تکنیک‌های داده‌کاوی سنتی که نیاز به چندین بار مرور کل مجموعه داده دارند، نمی‌توانند به طور مستقیم برای کاوش داده‌های جریان‌دار مورد استفاده قرار گیرند. زیرا این کاوش معمولاً باید با یک مرور و زمان پاسخ سریع صورت گیرد.

بنابراین می‌توان گفت مسأله اصلی که این تحقیق مطرح می‌کند، پردازش و کاوش داده‌های جریان‌دار معنایی که حجم وسیعی از آنها در کاربردهای مختلف موجود است و استخراج قوانین انجمنی از آنها به گونه‌ای است که بتوان با پیش‌بینی برخی روابط به تصمیم‌گیری نهایی در سیستم‌های هوشمند کمک کرد. اهمیت این مسأله از حجم عظیم این گونه داده‌ها در کاربردهای گوناگون، حساسیت این داده‌ها به زمان و استفاده از مفاهیم مستتر در داده‌های معنایی با وجود مشکلاتی که ناهمگنی این داده‌ها به آنها منجر می‌شود، ناشی می‌گردد. علاوه بر این با استفاده از غنای معنایی داده‌ها به قوانین انجمنی مفهومی‌تری دست یافته‌ایم که این امر تا کنون در کارهای مشابه تحقق نیافته است.

۲-۱- چالش‌های مسأله

تحقیق مورد نظر ترکیبی از مباحث مختلف از جمله داده‌های معنایی، داده کاوی و استخراج قوانین انجمنی و همچنین داده‌های پویای جریان‌دار می‌باشد. به طور کلی می‌توان گفت که مسأله استخراج قوانین انجمنی در حال حاضر مسأله‌ای چالش برانگیز است. برای مثال در حالتی که تعداد بسیاری از مجموعه آیتم‌ها وجود داشته باشد، یک چالش اصلی چگونگی شمارش، کشف و ذخیره‌سازی مجموعه آیتم‌های پرتکرار می‌باشد. از سوی دیگر ترکیب دو زمینه تحقیقاتی وب معنایی و داده کاوی اغلب در قالب یادگیری آنتولوژی و کاوش وب صورت گرفته است و کار کمی در زمینه کاوش بر روی خود داده‌های معنایی صورت گرفته است. داده‌های معنایی بسیار با مجموعه داده‌هایی که با الگوریتم‌های داده کاوی سنتی کاوش می‌شوند، متفاوتند. بنابراین چالش‌های اصلی از لحاظ کاوش داده‌های معنایی عبارتند از:

- الگوریتم‌های داده کاوی سنتی با مجموعه داده‌های تشکیل شده از تراکنش‌ها که هر کدام شامل زیرمجموعه‌ای از آیتم‌هاست، کار می‌کنند. اما در یک مخزن داده معنایی حاشیه‌نویسی شده به زبان OWL محورهای آنتولوژی وجود دارند که دامنه مفهومی را توصیف می‌کنند و همچنین حاشیه نویسی‌های معنایی

که به عنوان اعلان‌ها[1] بیان می‌شوند و نمونه داده‌ها را از طریق مشخصه‌هایی که با آنتولوژی سازگارند، مرتبط می‌سازند. نحوه معمول ارائه این اعلان‌ها، سه‌گانه‌هایی شامل فاعل، گزاره و مفعول می‌باشند.

• OWL با استفاده از منطق توصیفی (DL[2]) بوجود آمده است. منطق توصیفی رسمی کننده و قالب دهنده ارائه دانش با مشخصه‌های رسمی معین و معانی می‌باشد. بنابراین داده‌های حاشیه‌نویسی شده از یک ساختار ثابت تبعیت نمی‌کنند. حتی نمونه داده‌های متعلق به یک کلاس ممکن است ساختارهای متفاوتی داشته باشند که این مسأله باعث بوجود آمدن ناهمگنی در ساختار می‌شود.

• از آنجایی که منطق توصیفی با معانی رسمی تعریف می‌شوند، قابلیت‌های استنتاج باید جهت کار با دانش ضمنی بکار گرفته شوند.

• در مقایسه با دیگر روش‌های داده کاوی بر روی داده‌های معنایی مانند خوشه‌بندی و کلاس‌بندی نمونه داده‌های معنایی، کارهای خیلی کمتری بر روی کاوش قوانین انجمنی از روی نمونه‌داده‌های معنایی صورت گرفته است. تکنیک‌های ارائه مورد نیاز در رابطه با قوانین انجمنی کاملاً با آنچه در روش‌های خوشه‌بندی و کلاس‌بندی مورد نیاز است، متفاوت است. در قوانین انجمنی تراکنش‌ها را داریم که مشاهده‌ای از رخداد مجموعه‌ای از آیتم‌ها می‌باشد. در واقع چالش اصلی کار با داده‌های معنایی شناسایی تراکنش‌های مورد نظر و آیتم‌ها از این داده‌های ذاتاً ناهمگن و شبه ساخت‌یافته می‌باشد.

از طرفی دیگر، علاوه بر این که در حال حاضر کاوش مجموعه آیتم‌های پرتکرار و استخراج قوانین انجمنی به خصوص در رابطه با داده‌های معنایی به طور کلی یک مسأله چالش برانگیز است، جریان‌های داده به دلیل ویژگی‌های منحصر به فردشان، چالش‌های جدید زیادی را به کاوش مجموعه آیتم‌های پرتکرار می‌افزایند.

اخیراً نرخ تولید داده‌ها در برخی از منابع داده‌ها سریع‌تر از هر زمانی شده است. این تولید سریع جریان‌های پیوسته اطلاعات، قابلیت‌های ذخیره‌سازی، محاسبات و ارتباطات را در سیستم‌های محاسباتی به چالش کشیده است. ذخیره‌سازی، پرس‌وجو و کاوش این مخازن داده‌ها، کارهای بسیار چالش برانگیزی از لحاظ محاسبات می‌باشد. این چالش‌ها عبارتند از:

• مدیریت جریان پیوسته داده‌ها - این مسأله یک مسأله مدیریت داده‌ها می‌باشد. سیستم‌های مدیریت پایگاه‌داده سنتی دارای قابلیت کار با چنین داده‌های پیوسته‌ای با نرخ بالا نیستند. در نتیجه در بسیاری از موارد، ذخیره‌سازی تمام داده‌ها در رسانه ماندگار عملی نمی‌باشد و در دیگر موارد، دستیابی تصادفی به داده‌ها در چندین زمان کاری بسیار پرهزینه می‌باشد. چالش اصلی از کشف مجموعه آیتم‌های پرتکرار در

[1] Assertion
[2] Descriptive logic

زمانی که داده‌ها تنها یک بار می‌توانند مورد دستیابی قرار بگیرند، ناشی می‌شود. تکنیک‌های اندیس‌گذاری، ذخیره‌سازی و پرس‌وجوی جدید جهت مدیریت این جریان‌های اطلاعاتی متوقف نشدنی و متغیر مورد نیاز است.

• حداقل‌سازی مصرف انرژی - مقادیر بسیاری از جریان‌های داده در محیط‌هایی با منابع محدود تولید می‌گردند. شبکه‌های حسگر یک نمونه معمول از این محیط‌ها می‌باشند. این دستگاه‌ها دارای باتری‌هایی با طول عمر کوتاه می‌باشند. طراحی تکنیک‌هایی که مصرف انرژی را بهینه سازند، یک مسأله بحرانی است. بنابراین ارسال تمام داده‌های جریان‌دار تولید شده به یک سایت مرکزی علاوه بر مسأله عدم مقیاس‌پذیری می‌تواند از لحاظ مصرف انرژی غیربهینه باشد.

• نیازمندی به حافظه غیر محدود - دیگر ویژگی جریان‌های داده این است که داده‌ها غیر محدودند اما حافظه‌ای که بتواند جهت کشف و یا نگهداری مجموعه آیتم‌های پرتکرار بکار رود، محدود می‌باشد. تکنیک‌های یادگیری ماشین، منبع اصلی الگوریتم‌های داده‌کاوی می‌باشند. اکثر روش‌های یادگیری ماشین به پایداری داده‌ها در حافظه در طول اجرای الگوریتم تحلیل نیازمندند. به دلیل مقادیر عظیم جریان‌های تولید شده، مسأله طراحی تکنیک‌های بهینه‌سازی فضا که بتوانند تنها یک مرور یا کمتر بر روی جریان ورودی داشته باشند، بسیار با اهمیت است. دیگر نتیجه داده‌های نامحدود این است که آیا پرتکرار بودن یک مجموعه آیتم به زمان بستگی دارد یا خیر. چالش اصلی از استفاده از حافظه محدود برای کشف مجموعه آیتم‌های پرتکرار پویا از داده‌های نامحدود ناشی می‌شود.

• انتقال نتایج داده کاوی - ارائه ساختار دانش یکی دیگر از مسائل تحقیقاتی مهم می‌باشد. پس از استخراج مدل‌ها و الگوها به طور محلی از منابع تولید جریان‌های داده، انتقال این ساختارها به کاربر بسیار دارای اهمیت است.

• مدل‌سازی تغییرات نتایج در طول زمان - در بعضی موارد، کاربر به نتایج کاوش جریان‌های داده علاقمند نیست اما چگونگی تغییر این نتایج در طول زمان برای او دارای اهمیت می‌باشد. برای مثال، تعداد خوشه‌های ایجاد شده توسط جریان‌های داده در طول زمان تغییر می‌کند. این تغییر ممکن است برخی تغییرات در پویایی جریان ورودی را نشان دهد. پویایی جریان‌های داده در بسیاری از کاربردهای تحلیلی بسته به زمان مانند نظارت ویدیویی، پاسخ‌دهی اورژانسی، بهبود پس از بلایای طبیعی و غیره، با تغییرات در ساختارهای دانش تولید شده مشاهده می‌شود.

• پاسخ‌دهی بلادرنگ - از آنجایی که کاربردهای جریان‌های داده معمولاً وابسته به زمان هستند، نیازمندی‌هایی در رابطه با زمان پاسخ‌دهی در آنها وجود دارد. برای برخی از سناریوهای محدود، الگوریتم‌هایی که آهسته‌تر از نرخ ورود داده‌ها عمل می‌کنند، غیرمفید خواهند بود.

۳-۱- ساختار پایان نامه

در فصل دوم این رساله پس از بیان مفاهیم اولیه اساسیِ تحقیق ارائه شده، معرفی کارهای انجام شده مرتبط با زمینه‌های مورد بحث در این تحقیق صورت گرفته و چالش‌های موجود شرح داده شده‌اند. سپس کارهای مشابه انجام شده در رابطه با کاوش قوانین انجمنی از داده‌های معنایی مورد بررسی و تحلیل جزئی‌تر قرار گرفته است.

سپس در فصل سوم پس از مقدمه‌ای کلی در رابطه با ضرورت پرداختن به مسأله مطرح در رابطه با روش پیشنهادی، ابتدا به معرفی روش پیشنهادی پرداخته می‌شود. در این معرفی معماری کلی و اجزای سیستم پیشنهادی تشریح گردیده و ابزارها و الگوریتم‌های بکار رفته بیان خواهند شد. سپس مراحل عملیاتی کردن مدل پیشنهادی و پیش‌نیازهای آن مطرح می‌گردند. در انتها چالش‌های موجود در کارهای مرتبط به طور مختصر گفته شده و نحوه پاسخگویی روش پیشنهادی به آنها بیان گردیده و سپس نوآوری‌های روش پیشنهادی مورد بحث قرار خواهند گرفت.

در ادامه در فصل چهارم به شرح پیاده‌سازی مدل پیشنهادی در حوزه داده‌های هواشناسی پرداخته شده و با اجراهای مختلف سیستم بر روی بخش‌های مختلف داده‌های معنایی جریان‌دار منتشر شده در این حوزه کاربردی، عملیاتی شدن مدل با برنامه نویسی معنایی نشان داده می‌شود. پس از توصیف مراحل آماده‌سازی از قبیل آماده‌سازی مجموعه داده‌ها و آماده سازی آنتولوژی کاربرد و آنتولوژی مفاهیم کاوش قوانین انجمنی و همچنین ارائه ورودی و خروجی‌های سیستم پیشنهادی، در نهایت به ارزیابی کیفی و کمی روش پیشنهادی با توجه به معیارهای استاندارد معرفی شده پرداخته خواهد شد.

در انتها، فصل پنجم نتیجه‌گیری و برخی از کارهای آینده جهت بهبود روش پیشنهادی را ارائه می‌دهد.

فصل ۲.

فصل ۲– مرور ادبیات موضوع

۱–۲– مقدمه

در این فصل پس از معرفی کلی پیشینه کارهای انجام شده در رابطه با این تحقیق در مقدمه، به معرفی مفاهیم اولیه از جمله داده‌کاوی، قوانین انجمنی، فناوری‌های وب معنایی و ارتباط این مفاهیم با یکدیگر پرداخته و سپس به تشریح کارهای مشابه و تحلیل و بررسی مشکلات آنها پرداخته می‌شود.

پیشینهٔ این تحقیق دارای ابعاد گوناگونی از قبیل الگوریتم‌های داده‌کاوی و استخراج قوانین انجمنی، کاوش داده‌های پویای پیوسته یا جریان‌های داده و همچنین کاوش داده‌های معنایی ساخت‌یافته ناهمگن می‌باشد. در رابطه با هر یک از این ابعاد کارهای مختلفی از جنبه‌های مختلف و با استفاده از تکنیک‌های متفاوت صورت گرفته است.

الگوریتم‌های کاوش قوانین انجمنی معمولاً دارای دو گام اصلی می‌باشند. اولین گام کشف مجموعه آیتم‌های پرتکرار می‌باشد. مرحله دوم، مرحله ایجاد قوانین انجمنی است. اغلب تحقیقات به طور عمده بر روی گام اول یعنی چگونگی کشف بهینه تمام مجموعه آیتم‌های پرتکرار در جریان‌های داده تمرکز دارند [Mal11].

الگوریتم Apriori یک الگوریتم قدرتمند برای کاوش مجموعه آیتم‌های پرتکرار برای کشف قوانین انجمنی می‌باشد که معمولاً به عنوان پایه‌ای برای سایر الگوریتم‌های کشف قوانین انجمنی مطرح می‌شود. ویژگی Apriori این است که هر زیرمجموعه از مجموعه آیتم‌های پرتکرار خود نیز باید پرتکرار باشد. هدف اصلی یافتن مجموعه آیتم‌های پرتکرار است که مجموعه‌هایی از آیتم‌ها هستند که حداقل درجه پشتیبانی را دارا می‌باشند. در [Pin12] یک الگوریتم قوانین سازگاری جدید برای کشف قوانین انجمنی پیشنهاد داده شده است. این الگوریتم به کاربران اجازه می‌دهد که داده‌ها را بدون داشتن دانش دامنه کاوش نمایند. نتایج مشاهده شده در مقایسه با عملکرد قوانین انجمنی پایه بدون قواعد سازگاری قابل توجه است. [Bob12] نیز یک الگوریتم ماتریس بولی جدید برای کاوش داده‌های فضایی مؤثر بر اساس قوانین انجمنی ارائه نموده است. بازدهی استخراج قوانین انجمنی فضایی در مقایسه با الگوریتم Apriori معمولی قابل توجه است. این الگوریتم تعداد زمان‌های اسکن پایگاه داده تراکنش را کمتر نموده و تعداد مجموعه آیتم‌های کاندیدا را نیز کاهش می‌دهد [Sur12].

الگوریتم‌های کاوش مجموعه آیتم‌های پرتکرار که بر روی جریان‌های داده کار می‌کنند و در مرز دانش قرار دارند، در سه دسته مدل پنجره تقسیم‌بندی می‌شوند. این مدل‌های پنجره عبارتند از: پنجره راهنما[1] پنجره نمناک[2] و پنجره لغزنده[3] [Raj12] که در بخش ۲-۳ به تفصیل توصیف می‌گردند.

بخشی از تحقیقات بر روی کاوش داده‌های معنایی در زمینه برنامه نویسی خطی استقرایی (ILP[4]) صورت گرفته است که منطق اساسی کد شده در داده‌ها را برای یادگیری مفاهیم جدید مورد بهره‌برداری قرار می‌دهد. یک نمونه از این کار در [Lis05] آمده است. هرچند مشکل دوباره نویسی مجموعه داده‌ها در قالب‌های رسمی برنامه‌نویسی منطقی وجود دارد و اغلب این راه‌کارها برخلاف الگوریتم‌های آماری قابلیت شناسایی برخی مفاهیم پنهان را دارا نمی‌باشند.

قوانین انجمنی تعمیم‌یافته نیز جهت لحاظ طبقه بندی آیتم برای کاوش قوانین انجمنی ارائه شده است. ایده اصلی این روش گسترش مجموعه آیتم‌ها با تمام پیشینیان هر آیتم، سپس محاسبه مجموعه آیتم‌های پرتکرار و در نهایت فیلتر کردن قانون‌هایی است که شامل یک آیتم و برخی از پیشینیان آن آیتم می‌باشند. این روش از قوانین تکراری جلوگیری می‌نمایند. [Gia10] دو الگوریتم کاوش قوانین انجمنی ارائه می‌نماید. هر دو از این جهت که از تولید هزینه‌بر مجموعه کاندیداها و تعمیم قوانین جلوگیری می‌کنند، بهینه می‌باشند. در [Tse08] به مسأله بروزرسانی بهینه قوانین انجمنی تعمیم‌یافته کشف شده در حالی که طبقه‌بندی آیتم تغییر یافته اند، پرداخته شده است.

برخی دیگر از مطالعات الگوریتم‌های یادگیری ماشین آماری را جهت کار مستقیم با آنتولوژی‌ها و نمونه داده‌های مرتبط با آنها توسعه داده‌اند [Fan09] [Gar08] [Blo07]. در [Fan09] از دانش بدست آمده از آنتولوژی برای تعیین وزن‌های سنجش شباهت استفاده شده است. هرچند این ارائه‌ها برای کاوش قوانین انجمنی مناسب نیستند، زیرا نیازمند دستیابی به تعریف مجموعه آیتم‌ها و تراکنش‌ها از داده‌های معنایی هستند.

موضوع مرتبط دیگر با این پژوهش، درخت کاوش و داده‌های ساخت‌یافته بوسیله گراف می‌باشد. زیردرخت پرتکرار و کاوش گراف در این راستا قرار دارند و هدف آنها شناسایی زیرساختارهای پرتکرار در مجموعه داده‌های پیچیده می‌باشد. اگرچه این الگوریتم‌ها به هدف یافتن روابط محتوایی مورد نظر در گراف‌های RDF(/S) و OWL دست نیافته‌اند. زیرا این روش‌ها زیرساختارهای نحوی پرتکرار را در نظر داشته‌اند نه محتواهای مرتبط

[1] landmark window
[2] damped window
[3] sliding window
[4] Inductive logic programming

معنایی پرتکرار. کار با گراف‌های RDF(/S) و OWL علاوه بر پرداختن به ساختار گرافی آن به استنتاج برای استخراج دانش ضمنی نیز نیاز دارد. [Bec10] و [Mar10] از دانش موجود در آنتولوژی جهت فیلتر کردن و تنظیم قوانین کشف شده استفاده نموده‌اند.

در رابطه با نحوه تعریف تراکنش‌ها نیز کارهایی با هدف تحلیل مجموعه داده‌های ناهمگن در قالب XML انجام شده است. به خصوص در [Xu05] نشان داده شده است که XQuery مناسب‌ترین زبان پرس‌وجو برای استخراج داده از مخازن داده ناهمگن XML نیست، زیرا نیاز به آگاهی کاربر از ساختارهای مستندها وجود خواهد داشت. معانی پایین‌ترین پیشینیان مشترک (LCA) می‌تواند جهت استخراج داده‌های مرتبط معنادار به صورت انعطاف‌پذیرتر بکار گرفته شوند. [Nap08]، [Nie09] و [Tag10] نیز به رفع مشکلات LCA پرداخته‌اند.

هدف برخی دیگر از کارهای مرتبط افزودن بعضی قابلیت‌های کشف دانش به SPARQL از طریق توسعه گرامر آن بوده است [Neb10] [Koc07] [Kie08]. [Neb10] گرامر SPARQL را جهت تعریف الگوهای قوانین انجمنی توسعه داده است. این الگوها به سیستم اجازه می‌دهد که تنها بر روی ویژگی‌های مورد نظر تمرکز کرده و در نتیجه تعداد و طول تراکنش‌های ایجاد شده کاهش یابد. این کار در [Neb12] توسعه داده شده و تکمیل شده است. [Nar11] نیز از زبان پرس و جوی SPARQL برای ایجاد تراکنش‌ها کمک گرفته و مدلی جهت تعامل بهینه با داده‌های پیوندی موجود در وب و در نتیجه انجام عملیات داده‌کاوی بر روی آنها ارائه نموده است. در [Tak07] نیز روشی برای کاوش قوانین انجمنی برای موتور جستجوی تطبیقی ارائه شده است که رفتارهای کاربران قبلی را بازتاب می‌دهد. Log های نحوه بازیابی کاربران در قالب مدل RDF توصیف شده اند و قوانین انجمنی که نحوه موفق بازیابی را بازتاب می‌دهند از آنها استخراج می‌گردند.

در این فصل به بیان مفاهیم اولیه مورد نیاز و بررسی دقیق‌تر برخی از این کارها پرداخته می‌شود.

۲-۲- داده کاوی و کاوش قوانین انجمنی

امروزه با گسترش سیستم‌های بانک اطلاعاتی و حجم بالای داده‌های ذخیره شده در این سیستم‌ها، به ابزاری نیازمندیم تا بتوانیم داده‌های ذخیره شده را پردازش نموده و اطلاعات حاصل از این پردازش را در اختیار کاربران قرار دهیم.

این یک مسأله پذیرفته شده است که فرآیند داده‌کاوی می‌تواند الگوهای فراوانی را از داده‌های مورد نظر استخراج نماید. از مهمترین وظایف در داده‌کاوی، فرآیند کشف مجموعه آیتم‌های پرتکرار و قوانین انجمنی می‌باشد. الگوریتم‌های بهینه بسیاری در مقالات در رابطه با کشف مجموعه آیتم های پرتکرار و قوانین انجمنی موجود است. مسائل سازگاری با کاربرد در روند داده‌کاوی در سال‌های اخیر مورد توجه بیشتری قرار گرفته‌اند.

قوانین انجمنی قابل اطمینان، باعث بهبود در سیستم‌ها بوده و جامعه داده‌کاوی از مدت‌ها پیش اهمیت کشف این قوانین را تأیید نموده است.

چندین کاربرد تجاری وجود دارند که از کشف مجموعه آیتم‌های پرتکرار و قوانین انجمنی از مخزن داده‌ها سود می‌برند. در ادامه به گزارشی مفهومی از انواع روش‌های موجود برای کاوش مجموعه آیتم های پرتکرار و قوانین انجمنی با توجه به مسائل کاربردی و بهینگی ارائه می‌گردد.

پیشرفت‌های اخیر در زمینه علوم اطلاعاتی، دیجیتال‌سازی داده‌ها در مقیاس بزرگ، برجسته‌سازی پایگاه داده‌های دیجیتالی و مخازن عظیم داده‌ها را به دنبال داشته است. در نتیجه لازم است مکانیزم‌هایی جهت مدیریت بهینه مقادیر زیادی از داده‌های متوالی و استخراج فوری دانش مفید بر اساس آن داده‌ها توسعه یابند. فناوری داده‌کاوی به عنوان راهی برای شناسایی الگوها از مقادیر عظیم داده‌ها پدید آمده است. داده‌کاوی که با عنوان کشف دانش در پایگاه داده‌ها نیز شناخته می‌شود، با عنوان "استخراج غیر بدیهی اطلاعات ضمنی ناشناخته در گذشته و بالقوه مفید" تعریف می‌گردد [Sha09].

۱-۲-۲- تعاریف اولیه

داده‌کاوی:

داده‌کاوی یک فرآیند وسیع از آنالیز مقادیر عظیم داده‌ها و گزینش اطلاعات مرتبط و مفید از میان آنها می‌باشد. این عبارت به استخراج یا کاوش دانش از مقادیر وسیع داده‌ها اشاره دارد [Jai07]. داده‌کاوی گام‌های ذیل را در بر می‌گیرد: پایش و یکپارچه‌سازی داده‌های بدست آمده از منابع داده‌ای همچون پایگاه داده‌ها و فایل‌ها، کاوش دانش مورد نیاز و سرانجام ارزیابی و ارائه دانش. این عملیات در شکل شماره ۲-۱ مشخص شده‌اند. در داده‌کاوی یادگیری قوانین انجمنی یکی از متداول‌ترین روش‌ها جهت شناسایی روابط مورد نظر میان داده‌های ذخیره شده در منابع عظیم داده‌ها می‌باشد.

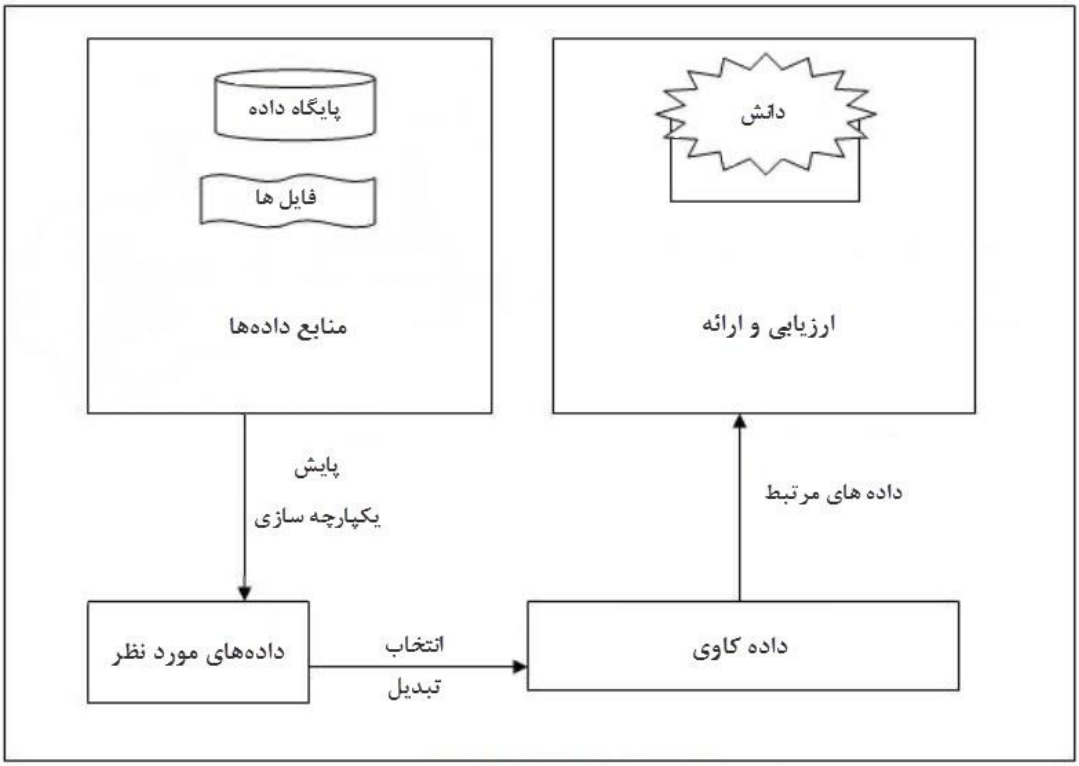
قوانین انجمنی:

قانون انجمنی در داده کاوی به معنای یافتن روابط ناشناخته بین داده‌ها و کشف قوانین میان آنهاست. Agrawal قوانین انجمنی را برای سیستم های POS[1] در سوپرمارکت ها معرفی نمود [Agr94].

یک قانون به عنوان مفهومی به شکل A=>B به طوری که $A \cap B \neq \emptyset$ تعریف می‌شود. سمت چپ قانون با عنوان مقدم نام برده می‌شود و سمت راست قانون را تالی می‌گویند. برای مثال قانون {Onions, Potatoes}=>{Beef} که در رابطه با داده‌های فروش یک سوپرمارکت یافت می‌شود، نشان می‌دهد که اگر یک

[1] Point of Sale

مشتری پیاز و سیب‌زمینی را با هم بخرد، احتمالاً گوشت نیز می‌خرد. اینچنین اطلاعاتی برای تصمیم‌گیری درباره فعالیت‌های خرید مفید خواهد بود. قوانین انجمنی در بسیاری از کاربردهای دیگر از قبیل کاوش استفاده از وب، تشخیص نفوذ و بیوانفورماتیک نیز می‌توانند مورد استفاده قرار گیرند.



شکل ۲-۱: فرآیند داده کاوی [Sur12]

درجه پشتیبانی[1]

فرض می‌کنیم $I = \{ i_1, i_2, i_3, \ldots, i_m \}$ مجموعه‌ای از آیتم هاست. T مجموعه‌ای از تراکنش‌هاست که با این آیتم‌ها مرتبطند و هر تراکنش یک شناسه به نام T_{ID} دارد. در صورتی که قانون انجمنی A=>B وجود داشته باشد به گونه ای که $A \in I$ و $B \in I$ که A را فرضیه مقدم و B را نتیجه می‌نامند، درجه پشتیبانی S به صورت نسبت تراکنش‌هایی در مجموعه داده که شامل مجموعه آیتم‌های مورد نظر می باشند، تعریف می‌شود.

Support(X=>Y) = Support (XUY) = P(XUY)

[1] Support

ضریب اطمینان[1]

با فرض‌های گفته شده ضریب اطمینان به صورت یک احتمال شرطی تعریف می‌گردد.

Confidence (X=>Y) = Support (XUY) / Support(X) = P(Y/X)

۲-۲-۲- مروری بر الگوریتم های کشف قوانین انجمنی

الگوریتم Apriori:

الگوریتم Apriori [Agr93] یک الگوریتم قدرتمند برای کاوش مجموعه آیتم‌های پرتکرار برای کشف قوانین انجمنی بولی می‌باشد. مجموعه آیتم‌های پرتکرار، مجموعه‌هایی از آیتم‌ها هستند که درجه پشتیبانی بزرگتر مساوی با حداقل درجه پشتیبانی را دارا می‌باشند. هدف اصلی الگوریتم یافتن مجموعه آیتم‌های پرتکرار است. یک زیرمجموعه از یک مجموعه آیتم پرتکرار نیز لازم است یک مجموعه آیتم پرتکرار باشد تا بتوان مجموعه آیتم‌های پرتکرار را با کاردینالیتی از ۱ تا k به صورت تکراری پیدا نمود و از این مجموعه آیتم‌های پرتکرار جهت ایجاد قوانین انجمنی استفاده نمود.

Apriori از یک روش تکراری برای یافتن مجموعه عناصر پرتکرار استفاده می‌کند به این صورت که برای یافتن (k+1)-itemset ها از k-itemset ها استفاده می‌کند. ابتدا 1-itemset ها پیدا می‌شوند که با L_1 نمایش داده می‌شوند . L_1 برای یافتن L_2 که 2-itemset ها هستند استفاده می‌شود و همین‌طور این فرآیند ادامه دارد تا هیچ k-itemset یافت نشود. یافتن L_k نیاز دارد تا کل پایگاه یکبار پیمایش شود.

برای تولید سطح به سطح مجموعه‌های پرتکرار یک ویژگی مهم به نام ویژگی Apriori به صورت زیر معرفی می‌شود که فضای جستجو را کاهش می‌دهد.

ویژگی Apriori : تمام زیرمجموعه‌های ناتهی یک مجموعه پرتکرار خود پرتکرار هستند.

از این ویژگی می‌توان این چنین استنتاج کرد که اگر یک مجموعه I حداقل درجه پشتیبانی را نداشته باشد آنگاه I پرتکرار نیست. اگر عنصر A به مجموعه I اضافه شود آنگاه مجموعه حاصل نمی‌تواند دارای فراوانی بیش از I باشد. پس این مجموعه یا همان $I \cup A$ پرتکرار نیست.

[1] Confidence

Algorithm: **Apriori.** Find frequent itemsets using an iterative level-wise approach based on candidate generation.
Input: Database, D, of transactions; minimum support threshold, *min_sup*.
Output: L, frequent itemsets in D.
Method:

```
(1)  L1 = find_frequent_1-itemsets(D);
(2)  for (k = 2; Lk−1 ≠ φ; k++) {
(3)      Ck = apriori_gen(Lk−1, min_sup);
(4)      for each transaction t ∈ D { // scan D for counts
(5)          Ct = subset(Ck, t); // get the subsets of t that are candidates
(6)          for each candidate c ∈ Ct
(7)              c.count++;
(8)      }
(9)      Lk = {c ∈ Ck|c.count ≥ min_sup}
(10)}
(11)return L = ∪k Lk;

procedure apriori_gen(Lk−1:frequent (k − 1)-itemsets;
        min_sup: minimum support threshold)
(1)  for each itemset l1 ∈ Lk−1
(2)      for each itemset l2 ∈ Lk−1
(3)          if (l1[1] = l2[1]) ∧ (l1[2] = l2[2]) ∧ ... ∧ (l1[k − 2] = l2[k − 2]) ∧ (l1[k − 1] < l2[k − 1]) then {
(4)              c = l1 ⋈ l2; // join step: generate candidates
(5)              if has_infrequent_subset(c, Lk−1) then
(6)                  delete c; // prune step: remove unfruitful candidate
(7)              else add c to Ck;
(8)          }
(9)  return Ck;

procedure has_infrequent_subset(c: candidate k-itemset;
        Lk−1: frequent (k − 1)-itemsets); // use prior knowledge
(1)  for each (k − 1)-subset s of c
(2)      if s ∉ Lk−1 then
(3)          return TRUE;
(4)  return FALSE;
```

شکل ۲-۲: شبه کد الگوریتم Apriori [Agr94]

حال می‌خواهیم ببینیم که در این الگوریتم L_{k-1} چگونه در یافتن L_k استفاده می شود. یک فرآیند دو مرحله‌ای دنبال می‌شود که شامل دو عمل پیوند[1] و هرس‌کردن[2] است:

۱. **مرحله پیوند**: در این مرحله برای یافتن L_k ، مجموعه‌ای از k-itemset ها با پیوند L_{k-1} با خودش حاصل می‌شود. این مجموعه را با C_k نمایش می‌دهیم. پیوند بین دو L_{k-1} را به صورت $L_{k-1} \bowtie L_{k-1}$ نشان می‌دهیم. این پیوند در صورتی انجام می‌شود که k-2 عنصر از این دو مجموعه با هم یکسان باشند. در واقع مجموعه عناصری از L_{k-1} که تنها در یک عنصر متفاوت هستند با هم پیوند می‌یابند. این تعریف از پیوند یک مجموعه با خودش هم جلوگیری می‌کند.

۲. **مرحله هرس‌کردن** : C_k شامل L_kها است و اعضای آن ممکن است پرتکرار باشند یا نباشند. اما تمام k-itemsetهای پرتکرار در C_k هستند. یک پیمایش پایگاه برای مشخص کردن تکرار کاندیداهای C_k با

[1] Join
[2] Prune

توجه به درجه پشتیبانی s، L_k را نتیجه می‌دهد. ممکن است C_k بسیار بزرگ باشد که در نتیجه به محاسبات بالا نیاز داریم. برای کم کردن اندازه C_k ویژگی Apriori به این صورت به کار می رود: هر (k-1)-itemset که پرتکرار نیست نمی‌تواند یک زیرمجموعه k-itemset های پرتکرار باشد. بنابراین اگر هر (k-1)-subset از یک کاندیدای k-itemset در L_{k-1} نباشد آنگاه کاندیدای مربوطه پرتکرار نیست و می‌تواند از C_k حذف شود.

شبه کد الگوریتم Apriori و روال‌های مربوطه‌اش در شکل شماره ۲-۲ آمده است. گام ۱ از آن L_1 ها را پیدا می‌کند. در گام‌های ۲ الی ۱۰، L_{k-1} برای تولید C_k به منظور یافتن L_k استفاده می‌شود. روال apriori_gen کاندیداها را تولید می‌کند و از ویژگی Apriori برای حذف آنهایی که یک زیرمجموعه غیرپرتکرار دارند استفاده می‌شود (گام ۳).

الگوریتم کاوش ماتریس انجمنی:

[Wan09] یک پایگاه داده تراکنش را به صورت $D=B^{(0)}.I$ توصیف می‌کند که D مجموعه تراکنش‌ها و I مجموعه آیتم را مدل می‌کند و ماتریس $B^{(0)}$ ماتریس انجمنی آیتم تراکنش است. این المان‌ها می‌توانند به صورت زیر تعریف شوند. اگر تراکنش I با آیتم j انجمن شود، المان متناظر ۱ خواهد بود و در غیر اینصورت ۰ است.

$$D_i = \sum_{j=1}^{M} B_{ij}{}^{(0)} I_j \qquad i = 1,2, \dots, N$$

که در آن i=1,2,…,N یک مجموعه تراکنش و j=1,2,…,M یک مجموعه آیتم است. تکرار j امین آیتم که در کل ماتریس تراکنش ظاهر می‌شود با L_i مشخص می‌گردد.

$$L_j = \sum_{i=1}^{N} B_{ij}{}^{(0)}$$

عملیات اولیه با افزودن $B^{(0)}$ مجموعه‌های آیتم بر روی ماتریس اعمال می‌شوند تا ماتریس جدید $B^{(1)}$ بدست آید به گونه‌ای که شناسایی جفت‌های پرتکرار آیتم‌ها آسان باشد. پس از آن بر اساس حداقل مقدار درجه پشتیبانی، به ترکیبات بیشتری از آیتم‌ها گسترش داده می‌شود. با برقراری درجه پشتیبانی حداقل، می‌توان

تکرار رخداد گروهی آیتم‌ها را بدست آورد. این مقاله به این صورت توانسته است شکل گیری مجموعه‌های آیتم کاندید الگوریتم Apriori را با عملیات ماتریس ابتدایی محقق نماید.

کاوش قانون انجمنی با استفاده از قوانین انسجام:

[Pin12] یک الگوریتم قوانین سازگاری جدید برای قوانین انجمنی پیشنهاد داده است. این الگوریتم که بر اساس قوانین سازگاری می‌باشد، به کاربران اجازه می‌دهد که داده‌ها را بدون داشتن دانش دامنه کاوش نمایند. نتایج مشاهده شده در مقایسه با عملکرد قوانین انجمنی پایه بدون قواعد سازگاری قابل توجه است.

یک الگوریتم ماتریس بولی برای کاوش قوانین انجمنی:

[Bab12] یک الگوریتم ماتریس بولی جدید برای کاوش داده‌های فضای مؤثر بر اساس قوانین انجمنی ارائه نموده است. بازدهی استخراج قوانین انجمنی فضایی در مقایسه با الگوریتم Apriori معمولی قابل توجه است. این الگوریتم تعداد زمان‌های اسکن پایگاه داده تراکنش را کمتر نموده و تعداد مجموعه آیتم‌های کاندیدا را نیز کاهش می‌دهد [Sur12].

۳-۲-۲- مقایسه برخی از الگوریتم‌های کاوش قوانین انجمنی

جدول شماره ۲-۱ مقایسه برخی از الگوریتم‌های کشف مجموعه آیتم‌های پرتکرار و قوانین برای پایگاه داده‌های بزرگ را نشان می‌دهد. این مقایسه بر اساس زمان اجرای صرف شده برای هر یک از این الگوریتم‌ها صورت گرفته است. این جدول مقادیر اندازه فایل، میانگین اندازه تراکنش‌ها، میانگین اندازه مجموعه آیتم‌ها، حداقل مقدار آستانه و زمان اجرای الگوریتم را برای هر الگوریتم نمایش می‌دهد.

جدول ۲-۱: مقایسه زمان‌های اجرای الگوریتم‌های کشف قوانین انجمنی مختلف [Sha09]

نام الگوریتم	اندازه فایل (کیلوبایت)	اندازه متوسط تراکنش‌ها	اندازه متوسط مجموعه آیتم‌ها	آستانه	زمان (ثانیه)
Two-Phase [Liu05]	۱۰۰۰	۲۰	۶	۱	۷۵۰
FUFM [Pod07]	۱۰۰	۱۰	۴	۰,۵	۲۵۰
AprioriHybrid [Agr94]	۱۰۰	۱۰	۴	۰,۷۵	۰,۷۵

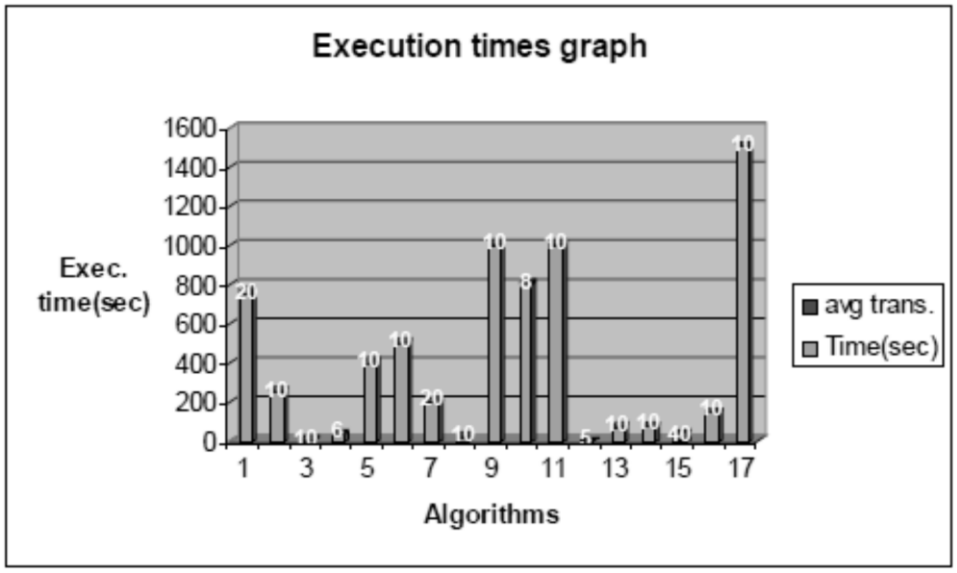
EFSM [Li05]	۱۰۰	۶	۴	۰,۴	۴۰
DSM-FI [Li04]	۲۰۰۰	۱۰	۵	۰,۰۱	۴۰۰
CTU-Mine [Erw07]	۱۰	۱۰	۱۰	۰,۵	۵۰۰
THUI [Chu08]	۱۰۰	۲۰	۶	۱	۲۰۰
Transaction [Son06]	۱۰۰	۱۰	۴	۱	۲۲
Parallel Apriori [Ye06]	۱۰۰	۱۰	۴	۰,۰۰۵	۹۹۸
Inter-transaction [Yu08]	۸۰۰۰	۸	۶	۰,۰۱	۸۰۰
WAR [Wan00]	۱۰۰۰	۱۰	۴	۰,۱	۱۰۰۰
CTU-PRO [Erw07]	۱۰۰	۵	۵	۰,۵	۵
FHARM [Sul06]	۱۰۰	۱۰	۴	۰,۳	۷۵
FWARM [Sul08]	۱۰۰	۱۰	۴	۱	۸۰
Apriori_Brave [Bod03]	۱۰۰	۴۰	۱۰	۰,۰۵	۸,۳
PRICES [Wan04]	۱۰۰	۱۰	۴	۵	۱۵۰
Combined Approach [Don03]	۱۰۰	۱۰	۴	۵	۱۵۰

جدول شماره ۲-۲ نیز برخی الگوریتم‌ها را از لحاظ مصرف حافظه مقایسه می‌نماید.

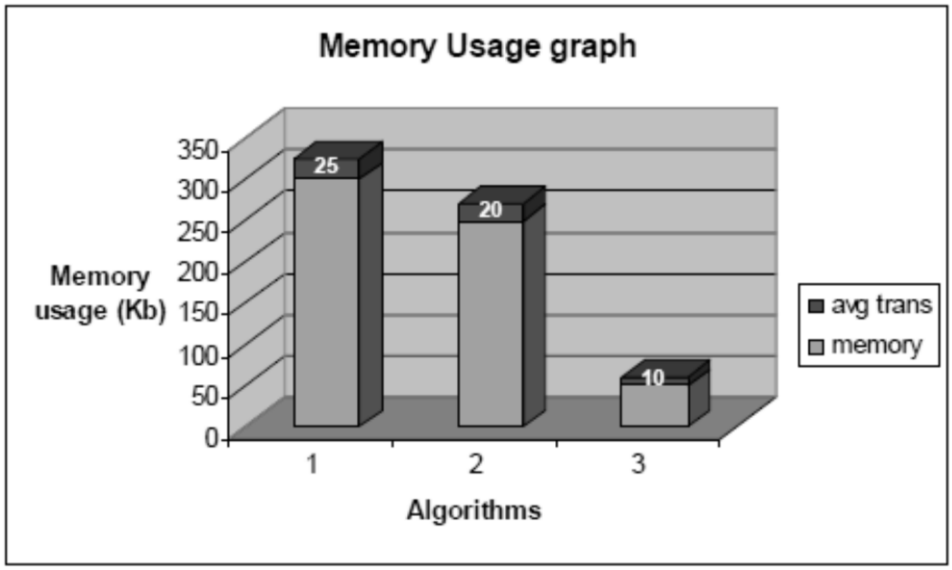
جدول ۲-۲: مقایسه مصرف حافظه الگوریتم‌های کشف قوانین انجمنی مختلف [Sha09]

نام الگوریتم	اندازه فایل (کیلوبایت)	اندازه متوسط تراکنش‌ها	اندازه متوسط مجموعه آیتم‌ها	آستانه	مصرف حافظه
H-mine [Pei07]	۱۰	۲۵	۱۵	۱	Kb ۳۰۰
THUI [Chu08]	۱۰۰	۲۰	۶	۱	Kb۲۵۰
DSM-FI [Li04]	۲۰۰۰	۱۰	۵	۰,۰۱	Kb۵۰
Combined Approach [Don03]	۱۰۰	۱۰	۴	۵	Mb۱۳

شکل‌های شماره ۲-۳ و شماره ۲-۴ نیز مقایسه این الگوریتم‌ها را به ترتیب از لحاظ زمان اجرا و مصرف حافظه به صورت نموداری نمایش می‌دهند. [Sha09]



شکل ۲-۳: مقایسه زمان‌های اجرای الگوریتم‌های کشف قوانین انجمنی مختلف [Sha09]



شکل ۲-۴: مقایسه مصرف حافظه الگوریتم‌های کشف قوانین انجمنی مختلف [Sha09]

۳-۲- داده‌کاوی بر روی جریان‌های داده

روز به روز کاربردهای بیشتری از قبیل مدلسازی ترافیک، شبکه‌های حسگر، دنبال کردن در کاربردهای نظامی، پردازش داده‌های آنلاین و غیره، حجم عظیمی از داده‌های جریان‌دار را هر روزه ایجاد می‌نمایند. کشف دانش بهینه از این جریان‌های داده، یک حوزه تحقیقاتی فعال در داده‌کاوی با کاربردهای متفاوت بوجود آورده است. برخلاف داده‌های ایستای موجود در پایگاه داده‌های سنتی، داده‌های جریان‌دار اغلب به صورت پیوسته با سرعت بالا و با حجم زیاد و همچنین با توزیع داده متغیر دریافت می‌شوند. این ویژگی‌های داده‌های جریان‌دار مسائل جدیدی را بوجود می‌آورد که لازم است در توسعه تکنیک‌های کاوش قوانین انجمنی از داده‌های جریان‌دار مورد توجه قرار گیرند. به دلیل این ویژگی‌های منحصر به فرد داده‌های جریان‌دار، تکنیک‌های داده‌کاوی سنتی که نیاز به چندین بار مرور کل مجموعه‌داده دارند، نمی‌توانند به طور مستقیم برای کاوش داده‌های جریان‌دار مورد استفاده قرار گیرند زیرا این کاوش معمولاً باید با یک مرور و زمان پاسخ سریع صورت گیرد.

یک جریان داده، یک دنباله مرتب از آیتم‌هاست که با ترتیب زمانی دریافت می‌شوند. برخلاف داده‌های ایستای موجود در پایگاه داده‌های سنتی، جریان‌های داده [Guh01] پیوسته، غیرمحدود بوده و معمولاً دارای سرعت بالا و توزیع داده متغیر با زمان می‌باشند.

از آنجایی که تعداد کاربردهای کاوش جریان‌های داده به سرعت رو به رشد است، نیاز به استخراج قوانین انجمنی از جریان‌های داده در حال افزایش است. برای اغلب کاربردهای جریان‌های داده، نیاز به کشف الگوهای تکراری و قوانین انجمنی احساس می‌شود. برخی از مهمترین این کاربردها عبارتند از: نظارت بر عملکرد، نظارت بر تراکنش‌ها و کاوش شبکه حسگر.

۱-۳-۲- دسته بندی جریان داده‌ها

جریان‌های داده را می‌توان به دو کلاس جریان‌های آفلاین و جریان‌های آنلاین دسته‌بندی نمود. جریان‌های آفلاین با ورودی توده‌ای منظم [Sin02] مشخص می‌شوند. در میان نمونه‌های بالا، ایجاد گزارش‌ها بر اساس جریان‌های صورت عملیات وب می‌تواند کاوش جریان‌های داده آفلاین به حساب بیاید زیرا اکثر گزارش‌ها بر اساس داده‌های صورت عملیات در یک دوره زمانی معین بدست آمده اند. نمونه‌های دیگر از جریان‌های آفلاین شامل پرس‌وجوها بر روی به‌روزرسانی مخزن داده‌ها و یا دستگاه‌های پشتیبان می‌باشد. پرس‌وجوهایی که بر این جریان‌ها صورت می‌گیرد، می‌توانند آفلاین انجام شوند.

جریان‌های آنلاین با داده‌های به‌روزرسانی شده بلادرنگی که یکی یکی در طول زمان بدست می‌آیند، مشخص می‌گردند. از میان مثال‌های بالا، پیش‌بینی تخمین پرتکرار جریان‌های بسته‌های اینترنت یکی از کاربردهای کاوش جریان‌های داده آنلاین به حساب بیاید زیرا جریان‌های بسته‌های اینترنت یک فرآیند بلادرنگ یک بسته یک بسته است. نمونه‌های دیگر از جریان‌های آنلاین، نشانگرهای سهام، اندازه‌گیری‌های شبکه و داده‌های حسگر می‌باشند. این عملیات لازم است به صورت آنلاین انجام شده و با سرعت سریع پرس‌وجوهای آنلاین همخوانی داشته باشد. زمانی که این پرس‌وجوها دریافت می‌شوند، پردازش شده و سپس بلافاصله حذف می گردند. به علاوه، برخلاف جریان‌های داده آفلاین، پردازش توده‌ای داده‌ها [Jia06] برای جریان‌های داده آنلاین ممکن نمی‌باشد.

تحقیقات بر روی جریان‌های داده در حدود سال ۲۰۰۰ آغاز گردید و در آن زمان چندین سیستم مدیریت جریان‌های داده (مانند پروژه‌های Brandeis AURORA، Cornell COUGAR و Stanford STREAM) بوجود آمدند تا چالش‌های جدید در کاربردهایی چون نظارت بر ترافیک شبکه، مدیریت داده‌های تراکنش‌ها، نظارت بر جریان‌ها کلیک وب، شبکه‌های حسگر و غیره را حل نمایند.

از آنجایی که کاوش قوانین انجمنی، نقش مهمی در این کاربردهای جریان داده‌ها بازی می‌کند، همراه با توسعه سیستم‌های مدیریت جریان داده‌ها، توسعه الگوریتم‌های کاوش قوانین انجمنی برای جریان‌های داده نیز به یک موضوع تحقیقاتی مهم تبدیل شده است.

۲-۳-۲- مسائل عمومی در زمینه کاوش قوانین انجمنی جریان‌های داده

الگوریتم‌های کاوش قوانین انجمنی معمولاً دارای دو گام اصلی می‌باشند. اولین گام کشف مجموعه آیتم‌های پرتکرار می‌باشد. در این مرحله تمام مجموعه آیتم‌هایی که شرط آستانه درجه پشتیبانی را رعایت می‌کنند، کشف می‌شوند. مرحله دوم، مرحله ایجاد قوانین انجمنی است. در این مرحله، بر اساس مجموعه آیتم‌های پرتکرار کشف شده در مرحله اول، قوانین انجمنی که در شرط درجه اطمینان صدق می‌کنند، کشف می‌شوند. از آنجایی که گام دوم یعنی کشف قوانین انجمنی می‌تواند به طور بهینه به روش رو به جلو صورت گیرد، اغلب تحقیقات به طور عمده بر روی گام اول یعنی چگونگی کشف بهینه تمام مجموعه آیتم‌های پرتکرار در جریان‌های داده تمرکز دارند.

در حال حاضر کاوش مجموعه آیتم‌های پرتکرار به طور کلی یک مسأله چالش برانگیز است. برای مثال بر طبق مسأله انفجار ترکیبی، ممکن است تعداد بسیاری از مجموعه آیتم‌ها وجود داشته باشد و یک چالش اصلی چگونگی شمارش، کشف و ذخیره‌سازی مجموعه آیتم‌های پرتکرار می‌باشد. جریان‌های داده به دلیل ویژگی‌های منحصر به فردشان، چالش‌های جدید زیادی را به کاوش مجموعه آیتم‌های پرتکرار می‌افزایند.

اخیراً نرخ تولید داده‌ها در برخی از منابع داده‌ها سریع‌تر از هر زمانی شده است. این تولید سریع جریان‌های پیوسته اطلاعات، قابلیت‌های ذخیره‌سازی، محاسبات و ارتباطات را در سیستم‌های محاسباتی به چالش کشیده است. ذخیره‌سازی، پرس‌وجو و کاوش این مخازن داده‌ها، کارهای بسیار چالش برانگیزی از لحاظ محاسبات می‌باشد. کاوش جریان‌های داده با استخراج ساختارهای دانش ارائه شده در مدل‌ها و الگوها از جریان‌های متوقف نشدنی اطلاعات سروکار دارد.

کاوش جریان‌های داده یک زمینه مطالعاتی پرمخاطب است که چالش‌ها و مسائل تحقیقاتی را به جوامع داده‌کاوی و کاوش پایگاه‌داده‌ها [Jia06] معرفی نموده است. برخی از این چالش‌های جدید عبارتند از: مدیریت جریان پیوسته داده‌ها، حداقل‌سازی مصرف انرژی، نیازمندی به حافظه غیر محدود، انتقال نتایج داده کاوی، مدل‌سازی تغییرات نتایج در طول زمان، پاسخ‌دهی بلادرنگ و نمایش نتایج داده‌کاوی.

۳-۳-۲- مرور مقالات مرتبط

در ادامه تعدادی از الگوریتم‌های کاوش مجموعه آیتم‌های پرتکرار، مجموعه آیتم‌های حداکثر پرتکرار [Gou01] و یا مجموعه آیتم‌های بسته پرتکرار [Pas99] که بر روی جریان‌های داده کار می‌کنند و در مرز دانش قرار دارند، را مرور می‌نماییم. این الگوریتم‌ها را می‌توان بر اساس مدل پنجره‌ای که دارند، در سه دسته قرار داد. این مدل‌های پنجره عبارتند از: پنجره راهنما، پنجره نمناک و پنجره لغزنده. هر مدل پنجره همچنین می‌تواند بر اساس مبتنی بر زمان یا مبتنی بر شمارش بودن دسته‌بندی شود. بنا بر تعداد تراکنش‌هایی که در هر زمان به‌روزرسانی می‌شوند، الگوریتم‌ها در یکی از دو دسته به‌روزرسانی در هر تراکنش یا به‌روزرسانی در هر دسته نیز قرار می‌گیرند. علاوه بر این، می‌توان الگوریتم‌های کاوش را به دو رده دقیق و تقریبی دسته بندی نمود. همچنین رده‌بندی الگوریتم‌های تقریبی می‌تواند بر اساس نتایجی که برمی‌گردانند صورت گیرد: رویکرد نادرست-مثبت[1] یا رویکرد نادرست-منفی.[2] رویکرد نادرست-مثبت [Lee05] [Man02] [Li04] [Gia04] [Cha04] [Cha03] مجموعه‌ای از مجموعه‌آیتم‌ها را برمی‌گرداند که شامل تمام مجموعه‌آیتم‌های پرتکرار می‌باشد، اما این مجموعه همچنین شامل برخی مجموعه‌آیتم‌های غیرپرتکرار نیز می‌باشد. در حالی که رویکرد نادرست-منفی [Yu04] مجموعه‌ای از مجموعه‌آیتم‌ها را برمی‌گرداند شامل هیچ یک از مجموعه‌آیتم‌های غیرپرتکرار نمی‌باشد، اما این مجموعه برخی مجموعه‌آیتم‌های پرتکرار را نیز در بر ندارد. نیاز است مسائل مختلفی که در رابطه با مدل‌های پنجره مختلف و ذات الگوریتم‌ها وجود دارد، مورد بررسی قرار گیرند. بنابراین در ادامه قواعد اصولی چندین الگوریتم را شرح داده و مزایا و معایب آنها را در رابطه با جریان‌های داده تحلیل می‌نماییم.

۱-۲-۲-۲- کاوش مبتنی بر پنجره راهنما

در مدل پنجره راهنما، کاوش الگوی پرتکرار بر اساس مقادیر میان یک برچسب زمانی مشخص به نام راهنما و زمان حاضر صورت می‌گیرد.

الگوریتم BST شمارش اتلافی:

[Man02] الگوریتم شمارش اتلافی را برای محاسبه یک مجموعه تقریبی از مجموعه‌آیتم‌های پرتکرار بر روی کل تاریخچه یک جریان، ارائه می‌دهند. بنابراین یک مکانیزم تخمین در الگوریتم شمارش اتلافی [Agr94] پیشنهاد شده است. این الگوریتم بر اساس ویژگی Apriori [Agr95] عمل می‌کند. الگوریتم شمارش اتلافی،

[1] false-positive
[2] false-negative

مجموعه‌آیتم‌های پرتکرار را از روی جریان‌های داده کاوش می‌کند، در صورتی که یک خطا با حداکثر مقدار قابل قبول ε، علاوه بر یک حداقل ضریب پشتیبانی θ، بدست آید. اطلاعاتی که در رابطه با نتایج کاوش قبلی تا آخرین بخش عملیات انجام شده موجود است، در ساختار داده‌ای با نام لتیس[1] نگهداری می‌شود.

مزایا و معایب:

یک ویژگی بارز الگوریتم شمارش اتلافی این است که خروجی آن مجموعه‌ای از مجموعه‌آیتم‌هاست که موارد زیر را گارانتی می‌کند:

- تمام مجموعه‌آیتم‌های پرتکرار در خروجی ظاهر می‌شوند.
- هیچ کدام از مجموعه‌آیتم‌ها که تکرار واقعی آن زیر $(\sigma - \varepsilon)N \approx 0$ است، در خروجی نخواهند بود.
- تکرار محاسبه شده یک مجموعه‌آیتم کمتر از تکرار واقعی آن تا حداکثر εN می‌باشد.

اگرچه استفاده از یک آستانه پشتیبانی حداقل شده ε، برای کنترل کیفیت تخمین نتیجه کاوش به فاجعه منجر می‌شود. هرچه مقدار ε کوچکتر باشد، دقت تخمین بالا می‌رود اما تعداد زیر-مجموعه‌آیتم‌های پرتکرار تولید شده بیشتر می‌شود، که این کار هم به فضای حافظه بیشتر و هم به قدرت پردازش CPU بیشتر نیاز دارد. هرچند اگر ε به σ برسد، پاسخ‌های نادرست-مثبت بیشتری در نتیجه خواهد بود. زیرا تمام زیر-مجموعه‌آیتم‌های پرتکرار که تناوب محاسبه شده آنها حداقل $(\sigma - \varepsilon)N \approx 0$ می‌باشد، در خروجی قرار می‌گیرند. در حالی که تکرار محاسبه شده زیر-مجموعه‌آیتم‌های پرتکرار می‌توانند کمتر از تکرار واقعی آنها با حداکثر مقدار Nσ باشند. این مسأله در برخی از دیگر الگوریتم‌های کاوش [Lee05] [Li04] [Gia04] [Cha04] [Cha03] نیز موجود است. این الگوریتم‌ها از یک آستانه درجه پشتیبانی حداقل شده برای کنترل دقت نتیجه کاوش استفاده می‌کنند.

الگوریتم StreamMining:

StreamMining [Jin05] یک الگوریتم کاوش مجموعه آیتم‌های پرتکرار درون هسته‌ای بر مبنای الگوریتم BTS می‌باشد. StreamMining از یک مجموعه کاهش داده شده از مجموعه‌های پرتکرار ۲ آیتمی استفاده می‌نماید. علاوه بر این، ویژگی Apriori را جهت کاهش تعداد مجموعه آیتم‌های i تایی که $i > 2$ بکار می‌برد و یک محدوده برای مثبت نادرست‌ها بوجود می‌آورد. این الگوریتم یک الگوریتم با مرور واحد برای کاوش مجموعه

[1] lattice

آیتم‌های پرتکرار در یک محیط جریان‌دار است. این الگوریتم پارامتری به نام ε را به عنوان ورودی می‌گیرد. این الگوریتم تک مروره با استفاده از درجه پشتیبانی دلخواه سطح θ، تمام مجموعه آیتم‌هایی که با سطح تکرار θ رخ می‌دهند را گزارش نموده و هیچ یک از مجموعه آیتم‌هایی که تکرار آنها کمتر از $\theta(\varepsilon - 1)$ باشد را شامل نمی‌شوند. در این فرآیند نیازمندی به حافظه متناسب با $1/\varepsilon$ افزایش پیدا می‌کند.

مزایا و معایب:

این الگوریتم اولین الگوریتمی است که به هیچ ساختار حافظه خارج از هسته نیاز ندارد و در عمل بسیار بهینه از لحاظ حافظه می‌باشد. یک استثنا این است که مجموعه داده‌ها با طول متوسط خیلی بزرگ برای مجموعه آیتم‌ها، به خوبی پردازش نمی‌شوند. دانش اضافی در رابطه با مجموعه آیتم‌های پرتکرار حداکثر در چنین مواردی مورد نیاز خواهد بود و پرس‌وجوهای حساس به زمان در آن در نظر گرفته نشده‌اند.

۲-۲-۲-۲- الگوریتم‌های مبتنی بر پنجره راهنمای نمناک

در پنجره‌های نمناک، آخرین پنجره‌ها مهم‌تر از دیگر پنجره‌ها می‌باشند. به عبارت دیگر، تراکنش‌های قدیمی‌تر سهم کمتری را در رابطه با تکرارهای مجموعه آیتم‌ها دارا می‌باشند.

الگوریتم estDec:

این الگوریتم یک الگوریتم مبتنی بر پنجره راهنمای نمناک برای کاوش داده‌های جریان دار می‌باشد. روش estDec [Cha03] هر تراکنش در یک جریان داده را یکی یکی بدون ایجاد هرگونه کاندیدایی بررسی می‌نماید و تعداد رخداد یک مجموعه آیتم را در تراکنش‌هایی که تا به حال ایجاد شده‌اند، با استفاده از یک ساختمان داده درخت پیشوند دنبال می‌کند.

estDec یک لتیس برای ثبت مجموعه آیتم‌های ذاتاً پرتکرار و تعداد آنها نگهداری می‌کند و لتیس را برای هر تراکنش جدید به تناسب به‌روزرسانی می‌کند. با ثبت اطلاعات اضافی برای هر مجموعه آیتم p که نقطه زمانی آخرین تراکنش در هر زمان که شامل p می‌شود می‌باشد، الگوریتم تنها نیاز دارد تعدادها را برای مجموعه آیتم‌هایی که زیرمجموعه‌های تراکنش‌های به تازگی دریافت شده هستند، به‌روزرسانی نمایند. این کار تعدادهای آنها را با فاکتور ثابت d کاهش می‌دهد و سپس یکی به آنها می‌افزاید.

مزایا و معایب:

استفاده از یک نرخ تنزلی تأثیر اطلاعات قدیمی و منسوخ یک جریان داده را بر روی نتیجه کاوش کاهش می‌دهد. هرچند تخمین تکرار یک مجموعه آیتم از تکرار زیرمجموعه آن می‌تواند یک خطای بزرگ را بوجود بیاورد و این خطا ممکن است در تمام مسیر میان زیرمجموعه‌های ۲ تایی تا ابرمجموعه‌های n تایی منتشر شود، در حالی که حد بالا خیلی سست باشد. بنابراین فرموله کردن یک محدوده خطا بر روی تکرار محاسبه شده برای مجموعه آیتم‌های نتیجه شده مشکل است و تعداد زیادی نتایج نادرست-مثبت برگردانده می‌شود. زیرا تکرار محاسبه شده برای یک مجموعه آیتم ممکن است بسیار بزرگتر از تکرار واقعی آن باشد. علاوه بر این به‌روزرسانی هر تراکنش ورودی (به جای یک دسته از آنها) ممکن است قابل مدیریت در جریان‌های با سرعت بالا نباشد.

:Instant

Instant [Mao07] یک الگوریتم تک فازه برای یافتن آیتم‌ها با بیشترین تکرار در یک جریان داده است. این الگوریتم تمام مجموعه آیتم‌های حداکثر را برای یک جریان داده ذخیره می نماید. با استفاده از یک شمارش درجه اطمینان حداقل از پیش تعیین شده ∂، که $1 \geq \partial$، $\partial - 1$ آرایه مجزا وجود دارد. هر آرایه U[I] که $\partial - 1 \geq I \geq 1$، مجموعه آیتم های غیرپرتکرار حداکثر را که مقادیر درجه پشتیبانی آنها همانند I هستند، ذخیره می نماید. به علاوه آرایه F تمام آیتم‌ها با بیشترین تکرار را بدون توجه به درجه پشتیبانی هر یک از این آیتم ها نگهداری می نماید.

مزایا و معایب:

یک آیتم با بیشترین تکرار (MFI) جدید می تواند به محض این که تبدیل به یک مجموعه آیتم پرتکرار شود، نمایش داده شود. هرچند این الگوریتم چندین مشکل اساسی دارد. اولین مشکل این است که این الگوریتم تعدادی از آرایه ها را برای نگهداری از تمام MFI ها و مجموعه آیتم های غیرپرتکرار حداکثر بکار می‌گیرد. بنابراین مقدار فضای حافظه مورد نیاز بسیار زیاد است. علاوه بر این، از آنجایی که هیچ گونه مکانیزم بررسی ابرمجموعه یا زیرمجموعه بهینه برای یک MFI جدیداً شناسایی شده وجود ندارد، تعداد مقایسه‌ها میان مجموعه آیتم‌ها افزایش می‌یابد و مقدار فضای حافظه مورد نیاز زمانی که طول متوسط تراکنش‌ها بزرگتر می‌شود، به سرعت افزایش می‌یابد. اگرچه صحت و کامل بودن مجموعه حاصل از MFIها می‌تواند برای هر تراکنش ورودی

تضمین شود، الگوریتم باید برای این کار هم زمان پردازش و هم مصرف حافظه را فدا کند. بنابراین این الگوریتم برای جریان‌های داده آنلاین مناسب نمی‌باشد.

۳-۲-۲-۲- الگوریتم‌های مبتنی بر پنجره لغزنده

مفهوم پنجره لغزنده این است که تحلیل جریان‌های داده اخیر بیشتر مورد توجه کاربر است. بنابراین تحلیل جزئی بر روی آیتم‌های داده اخیر و نسخه‌های خلاصه قدیمی‌ترها صورت می‌گیرد. این ایده در بسیاری از تکنیک‌ها بکار رفته است. در مدل پنجره لغزنده، کشف دانش بر روی تعداد ثابتی از آیتم‌های داده اخیراً ایجاد شده که هدف داده کاوی می باشند، صورت می‌گیرد.

estWin:

الگوریتم estWin [Cha03] برای یافتن مجموعه آیتم‌های پرتکرار اخیر به صورت تطبیق یافته با یک جریان داده تراکنشی آنلاین با پارامترهای درجه پشتیبانی حداقل $Smin \in (0,1)$، درجه پشتیبانی قابل توجه $Ssig \in (0, Smin)$ و اندازه پنجره w، بکار می‌رود. مجموعه آیتم‌های قابل توجه در حافظه اصلی با استفاده از یک ساختمان داده درخت وابسته به واژه نگاری نگهداری می‌شود [Yu04] [Tan02] [cha03]. تأثیر اطلاعات در یک تراکنش قدیمی که خارج از محدوده پنجره لغزان قرار می‌گیرد، با کاهش تعداد رخداد هر مجموعه آیتم که در تراکنش ظاهر می‌شود، حذف می‌شود. الگوریتم estWin می‌تواند جهت یافتن تغییرات اخیر دانش موجود در یک جریان داده بکار رود. محدوده اخیر مورد نظر یک جریان داده با اندازه یک پنجره لغزنده مشخص می‌شود.

مزایا و معایب:

تعداد کل مجموعه آیتم‌های مشاهده شده در حافظه اصلی با بکارگیری تکنیک‌های تزریق تأخیری[1] و هرس کردن حداقل می‌شود. این تعداد همچنین در شناسایی تغییرات اخیر در یک جریان داده مفید است. اندازه پنجره لغزنده می‌تواند به تناسب نیازمندی کاربر با توجه به محدوده تغییرات اخیر تغییر نماید. این روش توازن قابل انعطافی میان زمان پردازش و دقت کاوش برقرار می‌نماید. در صورتی که در آینده نیازی به تحلیل اطلاعات رد شده قدیمی پیدا شود، این اطلاعات قابل دسترسی نخواهند بود. در این روش هیچ ساختمان داده خلاصه‌ای برای تراکنش‌های قدیمی نگهداری نمی‌شود.

[1] delayed-insertion

الگوریتم‌های TransSW و TimeSW:

MFI-TransSW [Li09] جهت کاوش آنلاین و افزایشی مجموعهٔ مجموعه آیتم‌های پرتکرار در یک پنجره لغزنده با استفاده از ارائه آیتم‌ها به صورت دنباله بیت مؤثر بکار می‌رود. بر اساس چارچوب MFI-TransSW یک الگوریتم تک‌مروره توسعه داده شده با نام MFI-TimeSW [Chi04] (کاوش مجموعه آیتم‌های پرتکرار در یک پنجره لغزنده حساس به زمان) برای کاوش مجموعهٔ مجموعه آیتم‌های جریان‌های داده واقع در یک پنجره لغزنده حساس به زمان بکار می‌رود.

در الگوریتم MFI-TimeSW برای هر آیتم X در پنجره لغزنده حساس به تراکنش فعلی TimeSW یک دنباله بیت با w بیت که با Bit(X) نشان داده می‌شود، ساخته می‌شود. اگر یک آیتم X در i امین تراکنش TimeSW فعلی باشد، i امین بیت Bit(X)، ۱ می‌شود. در غیر اینصورت ۰ خواهد بود. این فرآیند تبدیل دنباله بیت نام دارد. الگوریتم MFI-TimeSW شامل سه فاز: مقداردهی پنجره، لغزش پنجره و تولید مجموعه آیتم‌های پرتکرار می‌باشد.

تفاوت‌های اصلی میان MFI-TransSW و MFI-TimeSW در واحد پردازش داده، تبدیل دنباله بیت یک واحد زمانی، تعداد تراکنش‌های لغزنده و آستانه تکرار پویای مجموعه آیتم‌ها می‌باشد. در الگوریتم MFI-TransSW مقدار ثابت s.w آستانه تکرار مجموعه آیتم‌ها می‌باشد. s آستانه درجه پشتیبانی حداقل مشخص شده توسط کاربر در محدوده [0,1] و w اندازه پنجره لغزنده است. هرچند در الگوریتم MFI-TimeSW، مقدار آستانه تکرار s. |TimeSW| است که جمله دوم آن یک مقدار پویاست.

مزایا و معایب:

MFI-TransSW یک الگوریتم تک مروره بهینه برای کاوش مجموعهٔ مجموعه آیتم‌های جریان‌های داده با استفاده از پنجره لغزنده حساس به تراکنش است. ارائه دنباله بیت مؤثر از آیتم‌ها جهت افزایش عملکرد MFI-TransSW توسعه داده شده است. نه تنها MFI-TransSW و MFI-TimeSW نتایج کاوش را با دقت بالایی نتیجه می‌دهند، بلکه به طور قابل توجهی سریعتر اجرا شده و حافظه کمتری نیز به نسبت دیگر الگوریتم‌های موجود مانند SWF و Moment [Cha04] مصرف می‌نمایند. این الگوریتم‌ها هم که از پنجره لغزنده برای کاوش مجموعه آیتم‌های پرتکرار در جریان‌های داده استفاده می‌نمایند، همه مجموعه آیتم‌های پرتکرار که تعداد آنها زیاد است را نگه می‌دارند. در عوض می‌توان تنها مجموعه آیتم‌های پرتکرار بسته را ذخیره نمود که با این کار

فضای مورد نیاز کاهش خواهد یافت. همچنین این الگوریتم‌ها از مفهوم تولید کاندیدای الگوریتم Apriori استفاده می‌نمایند. این الگوریتم می‌تواند با بکارگیری روش‌هایی که تولید کاندیدا ندارند، بهبود یابد [Mal11].

۲-۴- فناوری‌های وب معنایی و جایگاه آن‌ها در داده‌کاوی

در سال‌های اخیر توجه بیشتری به ترکیب دو زمینه تحقیقاتی وب معنایی و داده کاوی صورت گرفته است [Stu06]. با کمک استانداردسازی زبان‌های آنتولوژی RDF/(S) و OWL، وب معنایی بهتر درک شده و مقادیر حاشیه‌نویسی‌های معنایی دارای روندی رو به رشد می‌باشد. این ترکیب اغلب در قالب یادگیری آنتولوژی [Bui05] و کاوش وب بوده است. اما کار کمی در زمینه کاوش بر روی خود داده‌های معنایی صورت گرفته است. کاوش داده‌های وب معنایی فواید بسیاری را برای جوامع تحقیقاتی خاص دامنه‌ای که داده‌های آنها اغلب پیچیده و ناهمگن است و حجم زیادی دانش در قالب آنتولوژی و حاشیه‌نویسی‌های معنایی موجود است، به بار خواهد داشت. بنابراین لازم است طریقه ترکیب نمونه داده‌های در قالب آنتولوژی را با تراکنش‌ها به منظور پردازش آنها با الگوریتم‌های قوانین انجمنی، مورد بررسی بیشتر قرار دهیم.

۱-۴-۲- مفاهیم اولیه

وب معنایی، یک ابتکار با هدف بهبود وضعیت فعلی شبکه جهانی وب و رفع مشکلات آن می‌باشد که ایده اصلی آن استفاده از اطلاعات قابل پردازش توسط ماشین است. وب معنایی پیشنهاد می‌کند که علاوه بر اطلاعاتی که به ایجاد و بازیابی یک سند توسط مخاطبان انسان کمک می‌کند، اطلاعاتی در مورد محتوا نیز ذخیره شود. این کار به مراتب پردازش توسط ماشین‌ها را آسان‌تر می‌کند، به این اطلاعات اضافه شده، ابرداده صریح[1]گفته می‌شود که بخش معنای داده‌ها را در خود دارد [Ant08].

۱-۱-۴-۲- آنتولوژی

آنتولوژی یک حوزه از سخن را بطور رسمی توصیف می‌کند. به طور معمول، آنتولوژی شامل یک لیست محدود از اصطلاحات و روابط بین آنهاست که مفاهیم مهم را در یک دامنه، مشخص می‌کند و فهم مشترکی را ارائه می‌دهد [Gru94] [Gru93]. و همچنین باعث سهولت در اشتراک دانش[2] و استفاده مجدد[3] می‌شود [Gua98]. عناصر تشکیل‌دهنده آنتولوژی‌ها شامل مفاهیم، خصوصیات آن‌ها و ارتباطات بین مفاهیم می‌باشد.

[1] Explicit metadata
[2] Knowledge sharing
[3] reuse

۲-۱-۳-۲- قالب استاندارد RDF برای توصیف داده‌ها

XML یک ابرزبان جهانی برای تعریف نشان‌گذار[1]است. با این وجود، XML ابزاری را برای افزودن معنا به داده‌ها ارائه نمی‌کند. برای مثال، هیچ معنای خاصی مرتبط با تگ‌های تعبیه شده وجود ندارد؛ هر برنامه‌ای، تگ‌های تعبیه شده را به روش خاصی تفسیر می‌کند. بنابراین زبان دیگری جهت استفاده در وب معنایی بوجود آمده که RDF [Man04] نام دارد و زبانی جهت تشریح مفاهیم است و حاوی توضیحاتی است، به نحوی که داده‌ها را قابل درک برای ماشین‌ها می‌سازد. هر عبارت در RDF بصورت سه‌تایی فاعل[2] گزاره[3]و مفعول[4]بیان می‌شود.

زبان RDF به کاربران اجازه می‌دهد منابع را با استفاده از واژگان خودشان توصیف نمایند. RDF در مورد هیچ دامنه کاربردی خاصی فرضیات ندارد و نیز معنایی از هیچ دامنه‌ای تعریف نمی‌کند. تمام این‌ها وابسته به کاربر است و کاربر باید آن را در شمای RDF (RDFS[5]) انجام دهد. RDFS یک زبان آنتولوژی ابتدایی است که امکان مدلسازی را فراهم می‌کند. مفاهیم کلیدی RDFS عبارتند از کلاس، روابط زیر کلاسی، ویژگی، روابط زیرویژگی و محدودیت‌های برد و دامنه [Bri04].

با توجه به اینکه در این تحقیق، داده‌ها از نوع سنسوری و داده‌های جریان‌دار[6]می‌باشند، به این معنی که زمان یک پارامتر مهم است و داده‌ها بایستی به همراه برچسب زمانی ذخیره شوند. بنابراین باید راهی برای ورود زمان به سه‌تایی‌های RDF پیدا کرد. این موضوع یکی از زمینه‌های تحقیقاتی در دنیای وب معنایی می باشد و هر روزه راه حل‌های متفاوتی برای این مسئله ارائه می‌شود. به عنوان مثال از ایده‌های اولیه که پیشنهاد شد، بکار بردن برچسب زمانی هر عبارت سه‌تایی، همراه با فاعل یا گزاره یا مفعول است و تحقیقاتی برای گسترش این ایده و طراحی زبان پرس و جویی برای آن انجام شده است [Gut07] [Pug08].

اگر داده‌های معنایی به صورت چهارگانه[7]درنظر گرفته شوند، با این فرض که قسمت چهارم آن پارامتر زمان باشد، آنگاه داریم: $[(sub_i, pred_i, obj_i) , T_i]$ که در حقیقت همان سه‌تایی‌های قبلی به‌اضافه زمان(T) می باشد.

C-SPARQL[1] یک زبان جدید برای پرس و جوهای پیوسته (مرتبط با زمان) بر روی جریان‌هایی از داده‌های RDF به شکل چهارگانه‌ها می‌باشد. از آنجایی که جریان‌های داده نامتناهی هستند، نیاز است برای پرس و جوی C-SPARQL یک پنجره[2]تعریف شود تا بازه زمانی مورد نظر مشخص شود [Bar10].

---

[1] markup
[2] subject
[3] predicate
[4] object
[5] RDF Schema
[6] Stream data
[7] quad

طبق تعریف، نوع داده‌های ذخیره شده برای پشتیبانی از پرس و جو به زبان C-SPARQL، دنباله ای مرتب شده از دوتایی‌های (سه گانه RDF و برچسب زمانی) است، به طوری که زمان کاهش نمی‌یابد یا به عبارت دیگر برای دو برچسب بلافاصله T_i و T_{i+1} داریم: $T_i <= T_{i+1}$.

۳-۱-۴-۲- استنتاج معنایی

در سیستم‌های معنایی، پایگاه دانش ترکیبی از یک آنتولوژی و داده‌های RDF حوزه کاری موردنظر می‌باشد. در چنین شرایطی استدلال‌گر که توانایی استنتاج حقایق را از حقایق موجود در آنتولوژی دارد، بخشی ضروری است. زیرا دانش جدید در همین بخش تولید می‌شود. به علاوه، استدلال‌گر مدل آنتولوژی را بازرسی کرده و ثبات، رضایت‌بخشی و طبقه‌بندی مفاهیم آن را بررسی می‌کند تا به مدلی سازگار دست یابیم که هیچ ناهماهنگی در آن وجود ندارد.

در RDF و RDF Schema، هر سه‌گانه مانند (a, P, b) را می‌توان با استفاده از یک حقیقت[۳]به شکل P(a,b) بیان نمود و همچنین هر تعریف نمونه مانند type(a,C) را که نشان میدهد a یک نمونه از کلاس C است می‌توان به شکل C(a) نمایش داد و برای نشان دادن اینکه کلاسی مانند C زیر مجموعه کلاس دیگری مانند D است نیز می‌توان به شکل $C(X) \rightarrow D(X)$ درنظر گرفت. برای بیان زیر ویژگی‌ها نیز می‌توان به همین صورت عمل نمود. در نهایت محدودیت بر روی دامنه و برد مقادیر را نیز می‌توان توسط منطق هرن[۴]بیان نمود. به عنوان مثال عبارت $P(X,Y) \rightarrow C(X)$ نشان می‌دهد که نمونه‌های کلاس C دامنه ویژگی P را تشکیل می‌دهند.

علاوه بر این می‌توان استنتاج مبتنی بر قانون داشت که برای این منظور ابتدا باید قوانین را با یک زبان استاندارد برای تعریف قوانین معنایی مانند[۵]SWRL [Hor04] نمایش داد.

در محیط‌های بلادرنگ با داده‌های جریان‌دار، به استنتاج جریان‌دار[۶]نیاز است. طبق تعریف استنتاج جریان‌دار، استنتاج منطقی در زمان واقعی روی داده‌های جریان‌دار می باشد. سیستم‌های معنایی بلادرنگ، با این نوع استنتاج روی داده‌های معنایی روبه رو هستند [Bar10] [Val09].

یکی از موارد اصلی برای محقق شدن استنتاج جریان‌دار معنایی، نگهداری[۱]پایگاه دانش می باشد، بدین معنی که وقتی حقیقتی اضافه شد یا حذف شد، اطلاعات موجود چطور به روز رسانی شوند و یا اینکه

---

[1] Continuous SPARQL
[2] window
[3] fact
[4] horn
[5] Semantic Web Rules Language
[6] Stream reasoning

سه‌تایی‌هایی که زمان آنها انقضا شده است را از پایگاه دانش حذف کند و همچنین سه‌تایی‌هایی که بواسطه این سه‌تایی استنتاج شده اند را حذف نماید. مورد دیگر اینکه اگر چندین کپی از یک سه‌تایی یکسان رسیده و در پایگاه دانش ذخیره شده است، باید تمام موارد به جز آن که زمان انقضای بالاتری دارد را حذف نماییم.

۲-۴-۲- چالش‌های موجود در رابطه با کاوش داده‌های معنایی

داده‌های معنایی بسیار با مجموعه داده‌هایی که با الگوریتم‌های داده کاوی سنتی کاوش می‌شوند، متفاوتند. به علاوه در مقایسه با دیگر روش‌های داده کاوی بر روی داده‌های معنایی از قبیل خوشه‌بندی و کلاس‌بندیِ نمونه داده‌های معنایی [Fan09] [Blo07]، کارهای خیلی کمتری بر روی کاوش قوانین انجمنی از روی نمونه‌داده‌های معنایی صورت گرفته است. تکنیک‌های ارائه مورد نیاز در رابطه با قوانین انجمنی کاملاً با آنچه در روش‌های خوشه‌بندی و کلاس‌بندی مورد نیاز است، متفاوت است. در قوانین انجمنی تراکنش‌ها را داریم که مشاهده‌ای از رخداد مجموعه‌ای از آیتم‌ها می‌باشد. در واقع چالش اصلی کار با داده‌های معنایی شناسایی تراکنش‌های مورد نظر و آیتم‌ها از این داده‌های ذاتاً ناهمگن و شبه ساخت‌یافته می‌باشد. بنابراین بیشترین استفاده ممکن از دانش بوجود آمده توسط آنتولوژی‌ها به گونه‌ای که تراکنش‌ها به آسانی قابل تعریف و کاوش باشند، بسیار مهم و اساسی است. راه‌حل‌های بسیار مختلفی برای ایجاد آیتم‌ها و تراکنش‌ها از داده‌های معنایی وجود دارد که به ماهیت معانی و کاربرد بستگی پیدا می‌کند.

در نتیجه لازم است که مسأله ایجاد تراکنش‌ها و کاوش قوانین انجمنی در داده‌های معنایی تعریف گردد و راه حلی برای استخراج بهینه آیتم‌ها و تراکنش‌ها به طریقی که برای الگوریتم‌های کاوش قوانین انجمنی مناسب باشند، ارائه گردد. در بخش بعد نمونه‌هایی از کارهای مشابه که به کاوش داده‌های معنایی و به خصوص استخراج قوانین انجمنی از آنها پرداخته اند مورد بررسی و تحلیل قرار می‌گیرند.

۲-۵- کارهای مشابه

تا کنون کارهای معدودی در رابطه با کاوش قوانین انجمنی از داده‌های معنایی صورت گرفته است [Ben12]. بسیاری از کارهای انجام شده، روی لینک‌های بدون معنا کار می‌کنند [Zha09]. در ادامه چند مورد از روش‌های کاوش قوانین انجمنی از داده‌های وب معنایی و داده‌های پیوندی را توضیح داده و هر یک را

[1] maintenance

جداگانه تحلیل خواهیم نمود. سپس نگرش‌های کاوش گراف را معرفی می‌کنیم و در نهایت نیز به بررسی خلاءهای مطالعات انجام شده در مورد کاوش قوانین انجمنی از داده‌های وب معنایی خواهیم پرداخت.

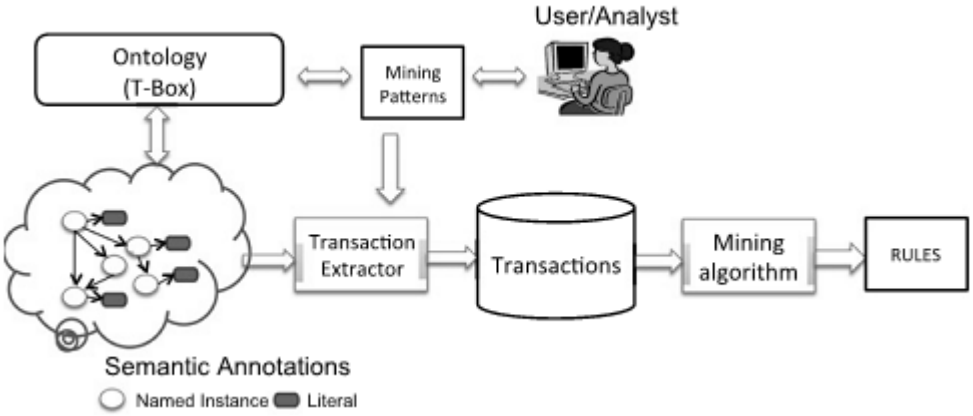
۱-۵-۲- ایجاد تراکنش از داده‌های وب معنایی با کمک الگوی کاوش

۱-۱-۴-۲- معرفی

این کار از لحاظ محتوی مشابه ترین کار به تحقیق مورد نظر بوده و پایه اصلی کار می‌باشد. عنوان مقاله اصلی استخراج شده از این کار "پیدا کردن قوانین انجمنی در داده‌های وب معنایی" [Neb12] است. متدولوژی استفاده شده بدین صورت است که ابتدا به کمک یک الگوی کاوش که کاربر آن را مشخص نموده است، تراکنش‌ها از روی داده‌های معنایی ساخته می‌شوند و سپس یک الگوریتم داده‌کاوی سنتی مانند Apriori، اقدام به کشف مجموعه آیتم‌های پرتکرار و در نهایت ایجاد قوانین انجمنی می‌نماید. این الگوی کاوش در قالب SPARQL توسعه یافته بیان شده است. در این کار بحث اصلی بر روی پیمایش داده‌ها (گراف‌ها) و ساخت تراکنش‌ها می‌باشد.

۲-۱-۴-۲- فرآیند کار

شکل ۲-۵ مرور شماتیکی فرآیند کلی را به تصویر می کشد. کاربر یک الگوی کاوش با نحو SPARQL گسترش داده شده را به صورت دستی مشخص می‌کند. سپس استخراج‌کننده تراکنش قادر به شناسایی و ساخت تراکنش‌ها با توجه به الگوی از قبل مشخص شده خواهد بود. در نهایت، مجموعه ای از تراکنش‌های بدست آمده توسط الگوریتم کاوش الگوی سنتی پردازش می‌شوند. این الگوریتم، قوانین انجمنی با قالب معین شده در الگوی کاوش را با درجه پشتیبانی و درجه اطمینان بیشتر از آنچه کاربر وارد نموده است، می‌یابد.



شکل ۲-۵: فرآیند کلی کشف قوانین انجمنی از داده‌های وب معنایی [Neb12]

در این سیستم قابلیت نمایش داده‌های معنایی به صورت سه‌تایی (فاعل، گزاره و مفعول) وجود دارد. گزاره، یک رابطه میان فاعل و مفعول است. بر همین اساس می‌توان داده‌های معنایی را به گراف تبدیل نمود، به طوری که فاعل و مفعول بیانگر رأس‌های گراف و گزاره‌ها بیانگر یال‌های بین در رأس باشند. بدین ترتیب این کار به داده‌های معنایی به صورت یک گراف نگاه می‌کند.

هر تراکنش دو بخش اصلی دارد. یکی آیتم‌های تراکنش و دیگری شناسه تراکنش یا TID. در این کار، کاربر بایستی تعیین کند که چه چیزهایی (آیتم‌هایی) در تراکنش‌ها قرار بگیرند و این آیتم‌ها بر چه اساس گروه‌بندی شده و تراکنش‌ها را بسازند. بدین منظور کاربر یک الگوی کاوش به نام Q که دارای سه قسمت می‌باشد را معین می‌کند.

Q = (Target Concept, Context Concept, Features)

Target Concept: بیان می‌کند که اطلاعات مرتبط به چه چیزی باید استخراج شود. مثلاً اطلاعات مرتبط با یک بیمار یا یک پزشک. در واقع این بخش از الگو نقاط شروع جستجو در گراف را نشان می‌دهد.

Context Concept: بیانگر معیار ساخت تراکنش است. به عبارت دیگر TID را مشخص می‌کند.

Algorithm 1. Generation of item transactions from instance transactions

Require: An instance transaction T, the query pattern Q, the ontology O, the instance store IS, and the composition triples IS_Q.
Ensure: An item transaction T'.
$T' = \emptyset$
i_C be the instance belonging to $context(Q)$ within the composition path associated to T in IS_Q.
for all $i \in T$ **do**
  if $O \cup IS \models MSC(i) \sqsubseteq C_i \wedge C_i \in features(Q)$
$\wedge\, MSC(i) \sqsubseteq (C_i \sqcap \exists DTP)$ **then**
    for all $(i, DTP, value) \in IS$ **do**
      update T' with the item $MSC(i_C).MSC(i).DTP \rightarrow value$
    end for
  else
    update T' with the item $MSC(i_C).MSC(i).i$
  end if
end for
return T'

شکل ۲-۶: شبه کد فرآیند ایجاد تراکنش‌ها [Neb12]

Features: نشان‌دهنده آیتم‌هایی است که در یک تراکنش قرار می‌گیرند. نیازی نیست تمام ویژگی‌ها در ساخت تراکنش و قوانین انجمنی حاصل حضور داشته باشند. در این بخش می‌توان بیش از یک مقدار وارد نمود.

یک تراکنش تشکیل شده است از مجموعه‌ای از آیتم‌هایی که به طور همزمان در نمونه داده‌های متعلق به context(Q) رخ داده اند. این آیتم‌ها به صورت پویا از نمونه‌داده‌های متعلق به features(Q) و چارچوب مشخص‌شده توسط کاربر بوجود می‌آیند. ابتدا تراکنش‌های نمونه داده‌ها با محاسبه سه‌گانه‌های ترکیبی (با استفاده از تبدیل اول عمق) بدست می‌آیند. سپس این سه‌گانه‌های ترکیبی با توجه به فاعل و مسیرشان گروه‌بندی می‌شوند. آخرین مرحله که در شبه کد فوق مشاهده می‌شود نیز تراکنش‌های آیتم‌ها را استخراج می‌نماید.

به عبارت دیگر، به منظور ساخت تراکنش، گراف بایستی به ازاء تمام رأس‌هایی که از نوع Target Concept هستند پیمایش شود. پیمایش از یک رأس شروع شده و تا جایی ادامه می‌یابد که به رأس‌هایی برسد که آن رأس‌ها حداقل یکی از ویژگی‌های موجود در بخش Features را داشته باشند. سپس مقدار ویژگی برای قرار گرفتن در تراکنش نگهداری می‌شود. سپس نوبت به گروه‌بندی مقدار ویژگی‌ها و ساخت تراکنش می‌رسد.

همیشه یک مسیر بین Target Concept و رأسی که یکی از مقادیر Features را داشته باشد وجود دارد (در این مسیر فقط رأس‌ها در نظر گرفته می‌شوند). به این مسیرها اصلاحاً مسیر تجمعی گفته می‌شود. از این مسیرهای تجمعی به عنوان TID استفاده می‌شود. بدین منظور بایستی پس از پیدا کردن تمامی مسیرهای تجمعی، آن‌ها را اصلاح نمود. بدین صورت که رأس‌های یک مسیر تجمعی را از رأس شروع (Target Concept) تا جایی که به یک رأس از نوع Context Concept برسد نگهداری نموده و بقیه رأس‌های باقیمانده بعدی از مسیر تجمعی حذف می‌شوند. حال مقدار ویژگی‌هایی که مسیر تجمعی یکسانی دارند، یک تراکنش را تشکیل می‌دهند. در نهایت یک الگوریتم داده‌کاوی سنتی مانند Apriori، قوانین انجمنی را از این تراکنش‌ها استخراج می‌کند.

در این روش نیاز است کاربر نوع الگوهایی که علاقمند به دریافت آنها از مخزن می‌باشد را مشخص کند. از آنجا که حاشیه نویسی‌های معنایی در قالب RDF/(S) و OWL می‌باشند، SPARQL با عبارت جدیدی که تعریف یک الگوی کاوش را ممکن می‌سازد، گسترش داده شده است. این نحو از توسعه داده کاوی مایکروسافت (DMX) که یک توسعه SQL برای کار کردن با مدل‌های داده کاوی در سرویس‌های تحلیلی مایکروسافت SQL سرور می‌باشد، الهام گرفته است. قالب تعریف شده در این گرامر SPARQL توسعه داده شده در شکل ۲-۷ نشان داده شده است.

```
Query          ::=   Prologue( SelectQuery | ConstructQuery | DescribeQuery | AskQuery | MiningQuery )
MiningQuery    ::=   CREATE MINING MODEL' Source  '{'
                        Var 'RESOURCE'  'TARGET'
                         ( Var ( 'RESOURCE' | 'LITERAL' )
                           'MAXCARD1'? 'PREDICT'? 'CONTEXT'? )+ '}'
                        DatasetClause* WhereClause UsingClause MeasuresClause
UsingClause    ::=  'USING'  SourceSelector BrackettedExpression
MeasuresClause ::=  'ADD MEASURE' measure +
```

شکل ۲-۷: گرامر SPARQL توسعه داده شده [Neb12]

مثال:

در این مثال کاربر Patient را به عنوان مفهوم مورد نظر تحلیل انتخاب نموده است. مجموعه ویژگی‌هایی که تراکنش‌های شامل diagnosed diseases، prescribed drugs و damage index را در رابطه با بیمار افزایش می‌دهد، مورد نظر خواهد بود. در نهایت، تراکنش در سطح گزارش ساخته خواهد شد. این به این معناست که تراکنش‌ها ویژگی‌های میان گزارش‌ها را شامل نمی‌شوند و تنها ویژگی‌های در درون هر یک از گزارش‌ها را شامل می‌شوند.

اطلاعات شکل ۲-۹ که در مورد یک بیمار به نام PNT_XY21 است را در نظر بگیرید. این اطلاعات در قالب گراف نمایش داده شده است. فرض نمایید که کاربر الگوی کاوش خود را به صورت زیر وارد نماید:

$$Q = (Patient, Report, \{Disease, Drug, Report \sqcap \exists\ damageIndex\})$$

این الگوی کاوش در قالب SPARQL توسعه یافته به صورت زیر است:

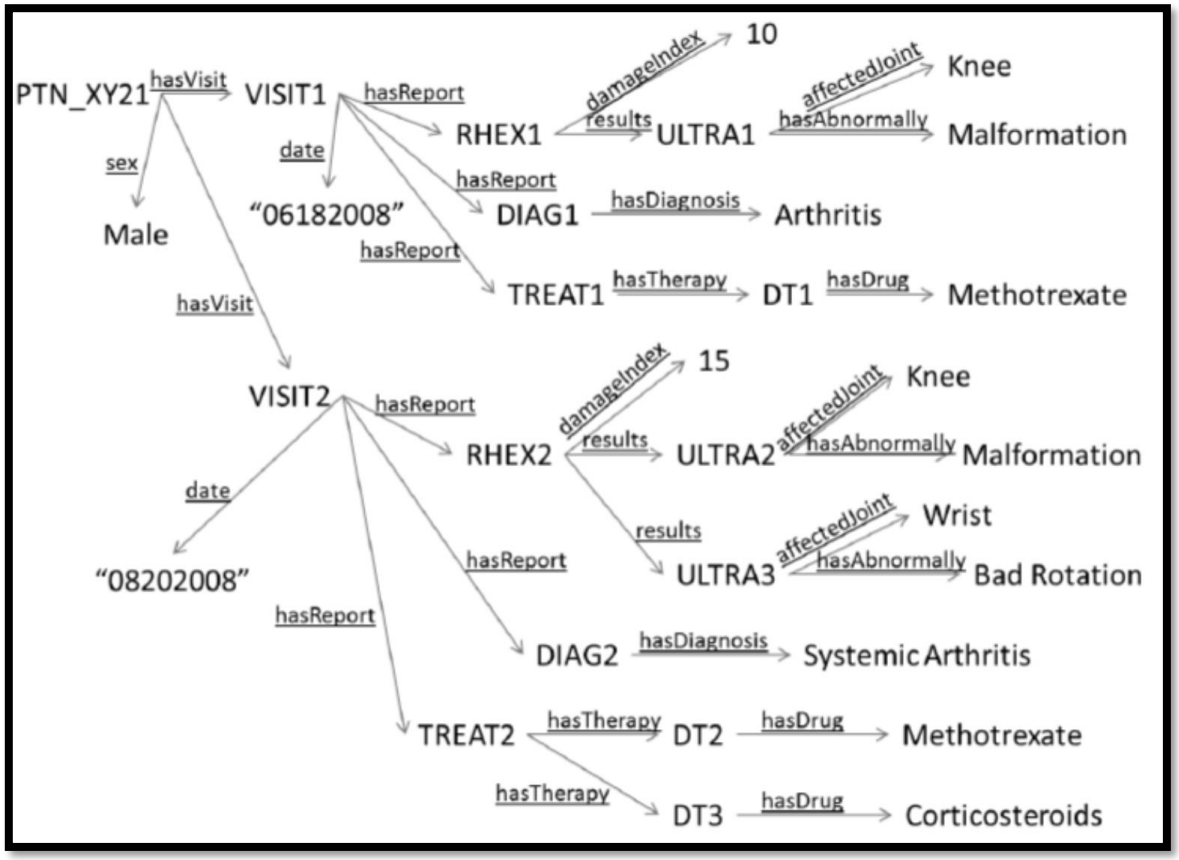
```
CREATE MINING MODEL <Dataset Path>
{
        ?patient        RESOURCE    TARGET
        ?drug           RESOURCE
        ?jadi           LITERAL
        ?disease        RESOURCE    PREDICT
        ?report         RESOURCE    CONTEXT
}
WHERE
{
        ?patient        rdf:type        Patient.
        ?drug           rdf:type        Drug.
        ?disease        rdf:type        Disease.
        ?report         rdf:type        Report.
        ?report         damageIndex     ?jadi.
}
```

شکل ۲-۸: الگوی کاوش نمونه در قالب SPARQL توسعه یافته [Neb12]

این الگوی کاوش بیان می‌کند که اطلاعات بیمار (Patient) بایستی استخراج شود. لذا جستجو بایستی از آن رأس شروع شود. عناصری که تراکنش را می‌سازند عبارتند از نام بیماری (Disease)، نام داروها (Drug) و میزان

شدت بیماری (DamageIndex). همچنین توجه فرمایید که به عنوان مثال اگر یک بیمار فقط نام بیماری و نام دارو را داشته باشد و مقدار شدت بیماری را نداشته باشند، فقط مقادیر نام بیماری و نام دارو در تراکنش قرار می‌گیرند. پس از یافتن مسیرهای تجمعی، در هر مسیر تجمعی، فقط رأس‌هایی نگهداری می‌شوند که بین دو رأس Patient و Report قرار گرفته باشند.

شکل زیر اطلاعات یک بیمار را در قالب سه‌تایی‌هایی که یک گراف را تشکیل می‌دهند، نشان می‌دهد.



شکل ۲-۹: اطلاعات مربوط به یک بیمار در قالب گراف [Neb12]

برخی از سه‌تایی‌های شکل فوق در جدول زیر آمده است.

جدول ۲-۳: اطلاعات مربوط به یک بیمار در قالب سه‌تایی‌ها [Neb12]

Subject	Predicate	Object
PTN_XY21	hasVisit	Visit1
PTN_XY21	hasVisit	Visit2
PTN_XY21	sex	Male
Visit1	hasReport	RHEX1
Visit1	hasReport	DIAG1
Visit1	hasReport	TREAT1
Visit1	date	"06182208"
RHEX1	damageIndex	10
RHEX1	result	ULTRA1
…	…	…

بر اساس الگوی وارد شده نتیجه کاوش این گراف جدول می‌باشد. دقت نمایید که RHEXn و TREATn رأس‌هایی از نوع Report هستند (به ستون Aggregation Path توجه کنید). بر همین اساس مسیرهای تجمعی اصلاح می‌شوند که نتایج آن در جدول نشان داده شده است. سپس تراکنش‌ها ساخته می‌شوند که نهایی در جدول مشاهده می‌شود. در نهایت این تراکنش‌ها به یک الگوریتم کاوش قوانین انجمنی سنتی مانند الگوریتم Apriori داده می‌شوند.

جدول ۲-۴: نتیجه کاوش گراف شکل ۲-۹ [Neb12]

Subject	Aggregation Path	Object	Feature
PTN_XY21	(VISIT1,RHEX1,ULTRA1)	Malformation	Disease
PTN_XY21	(VISIT1,RHEX1)	RHEX1	Report⊓ ∃damageIndex
PTN_XY21	(VISIT1,TREAT1,DT1)	Methotrexate	Drug
PTN_XY21	(VISIT2,RHEX2,ULTRA2)	Malformation	Disease
PTN_XY21	(VISIT2,RHEX2)	RHEX2	Report⊓ ∃damageIndex
PTN_XY21	(VISIT2,RHEX2,ULTRA3)	Bad rotation	Disease
PTN_XY21	(VISIT2,TREAT2,DT2)	Methotrexate	Drug
PTN_XY21	(VISIT2,TREAT2,DT3)	Corticosteroids	Drug
…	…	…	…

جدول ۲-۵: نتیجه کاوش گراف شکل ۲-۹ به همراه شماره تراکنش و مسیر تجمعی اصلاح شده [Neb12]

TID	Subject	Aggregation Path	Object	Feature
1	PTN_XY21	(VISIT1,RHEX1)	Malformation	Disease
1	PTN_XY21	(VISIT1,RHEX1)	RHEX1	Report⊓ ∃damageIndex
2	PTN_XY21	(VISIT1,TREAT1)	Methotrexate	Drug
3	PTN_XY21	(VISIT2,RHEX2)	Malformation	Disease
3	PTN_XY21	(VISIT2,RHEX2)	RHEX2	Report⊓ ∃damageIndex
3	PTN_XY21	(VISIT2,RHEX2)	Bad rotation	Disease
4	PTN_XY21	(VISIT2,TREAT2)	Methotrexate	Drug
4	PTN_XY21	(VISIT2,TREAT2)	Corticosteroids	Drug
…	…	…	…	…

جدول۲-۶: تراکنش‌های نهایی [Neb12]

TID	Object
1	{Malformation, RHEX1}
2	{Methotrexate}
3	{Malformation, RHEX2, Bad rotation}
4	{Methotrexate, Corticosteroids}

۳-۱-۴-۲- تحلیل

همان‌طور که مشاهده شد، تمرکز اصلی این روش بر روی ساخت تراکنش از داده های وب معنایی است که بدین منظور به شدت نیازمند دخالت کاربر نهایی است؛ زیرا همان‌طور که در بخش چالش ها ذکر شد، در داده‌های وب معنایی، مفهوم دقیقی از تراکنش وجود ندارد.

در مجموع می‌توان ویژگی‌های زیر را برای این مقاله در نظر گرفت:

۱- عدم کاوش مستقیم قوانین: روش ارائه شده، کاملا وابسته به یکی از الگوریتم های داده کاوی سنتی است، زیرا تنها کاری که این روش را انجام می دهد، استخراج تراکنش از داده های وب معنایی است.
۲- دخیل نمودن کاربر نهایی: در این روش به منظور استخراج تراکنش ها، به شدت نیازمند دخالت کاربر نهایی هستیم. این کاربر بایستی دقیقا مشخص نماید که تراکنش‌ها بر چه اساسی ساخته شوند. لازمه این کار این است که کاربر نهایی با ساختار آنتولوژی و دامنه مورد استفاده آشنایی کامل داشته و به زبان SPARQL و مفاهیم وب معنایی تسلط کامل داشته باشد. در صورتی که کاربران نهایی معمولا متخصصان علوم کامپیوتر نیستند.
۳- عدم تولید قوانین جامع و کلی: از آنجا که بایستی دقیقا مشخص نماییم که قوانین انجمنی مربوط به کدام نوع موجودیت (Target Concept) بایستی ساخته شود، در این کار امکان تولید قوانین جامع و کلی وجود ندارد و قوانینی که تولید می شوند، تنها به موجودیتی خاص مربوط می‌شوند.
۴- ابهام در الگوی کاوش: به گفته خود نویسندگان مقاله، گاهی ابهاماتی در نوشتن الگوی کاوش ایجاد می‌گردد که این ابهام مربوط به آیتم‌هایی می‌باشد که یک تراکنش را می‌سازند. زیرا ممکن است موجودیت‌های مختلف صفتی هم نام اما با معانی مختلف داشته باشند. راه حل این مشکل، تعریف دقیق مفهوم مورد نظر در SPARQL و در بخش WHERE می باشد که خود این امر بسیار مشکل است.
۵- استفاده از منطق: در پیاده سازی این کار، به خوبی از مفاهیم آنتولوژی (OWL) و منطق (ILP) استفاده شده است. مثلا اگر گفتیم که جستجو از راس های Patient شروع شود و کلاس A نوعی از Patient باشد و B نیز زیر کلاسی از A باشد، جستجو از راس های B نیز شروع خواهد شد.
۶- عدم استفاده از مفاهیم موجود در آنتولوژی و وجود نداشتن انسجام معنایی در تمام مراحل

۲-۵-۲- استخراج تراکنش از داده های وب معنایی با کمک گروه بندی سه تایی ها

۱-۲-۵-۲- معرفی

همان‌طور که در بخش چالش‌ها ذکر شد، یکی از مشکلات وب معنایی، عدم وجود تعریفی دقیق از تراکنش‌ها می‌باشد. اگر به طریقی تراکنش‌ها بوجود بیایند، الگوریتم‌های داده کاوی سنتی به راحتی می‌توانند قوانین انجمنی را استخراج نمایند. این مقاله نیز همانند مقاله قبلی تلاش می کند تا تراکنش‌ها را از داده‌های وب معنایی استخراج کند. نام این مقاله "پیکربندی متن و هدف برای کاوش داده‌های RDF" [Zia11] است.

۲-۲-۵-۲- روش انجام کار

در کار قبلی مشاهده شد که چگونه الگوریتم ارائه شده با دخالت شدید کاربر، به طور هوشمندانه اقدام به ایجاد تراکنش‌ها می‌کند. روشی که در این مقاله به کار گرفته شده، دخالت کاربر را کمتر می‌کند. اما تراکنش‌های تولید شده بسیار کلی می باشند.

همان‌طور که گفته شد عبارت‌های RDF در قالب سه تایی فاعل، گزاره و مفعول بیان می‌شوند. از طرفی نیز بیان گردید که هر تراکنش دو وجه مهم دارد یکی آیتم‌های تراکنش و دیگری شماره تراکنش یا TID. منظور از شماره تراکنش این است که کدام آیتم‌ها یک تراکنش را بسازند، (در عمل آیتم‌ها با TID یکسان یک تراکنش را تشکیل می‌دهند).

روش کار در این مقاله بدین صورت است که از یکی از مقادیر فاعل، گزاره و مفعول به منظور شناسه تراکنش یا همان TID استفاده می‌کند. از این مقدار تحت عنوان متن نام برده می‌شود. سپس از مقادیر متغیرهای یکی از دو مورد باقیمانده به عنوان عناصر تراکنش استفاده می‌گردد. از این مقادیر نیز تحت عنوان هدف نام برده می‌شود. در نهایت مقدار باقیمانده سوم دور ریخته می‌شود.

جدول ۲-۷، شش حالت مختلف از ترکیب دو تا از مقادیر فاعل، گزاره و مفعول را به همراه کاربرد هر کدام از روش‌ها نشان می‌دهد. به عنوان مثال، اگر عناصر فاعل بر اساس گزاره هایشان (رفتارها) گروه بندی شده باشند، فاعل‌های با رفتار یکسان در یک گروه قرار می گیرند و این امر می تواند در خوشه بندی کاربرد داشته باشد.

جدول۲-۷: شش ترکیب مختلف متن و هدف [Zia11]

	متن	هدف	مورد کاربرد
۱	فاعل	گزاره	کشف شِما (Schema Discovery)
۲	فاعل	مفعول	آنالیز سبد (Basket Analysis)
۳	گزاره	فاعل	خوشه بندی
۴	گزاره	مفعول	کشف محدوده (Range Discovery)
۵	مفعول	فاعل	خوشه بندی موضوعی (Topical Clustering)
۶	مفعول	گزاره	انطباق شِما (Context)

۳-۲-۵-۲- تحلیل

در مجموع می‌توان ویژگی‌های زیر را برای این مقاله در نظر گرفت:

۱- در نظر نگرفتن یکی از عناصر سه تایی فاعل، گزاره و مفعول: همان‌طور که ذکر شد، در این روش از یکی از مقادیر فاعل، گزاره و مفعول به منظور ساخت تراکنش و یکی دیگر از دو مقدار باقیمانده به عنوان آیتم‌های تراکنش استفاده می شود. مقادیر مورد باقیمانده سوم اصلا در نظر گرفته نمی‌شوند. این امر باعث می شود که مقدار زیادی از اطلاعات از دست برود.

۲- نامفهوم بودن قوانین انجمنی تولید شده: درست است که در جدول، کاربرد هر یک از ترکیب‌های مختلف ذکر شد؛ اما این کاربردها روی خود تراکنش‌ها است نه روی قوانین تولید شده. مورد خوشه‌بندی که در مورد آن توضیح داده شد را در نظر بگیرید. این حالت بیان می کند که آیتم‌های (فاعلی) یک تراکنش، چون رفتاری (گزاره) شبیه به یکدیگر داشته اند، در یک تراکنش قرار می‌گیرند (در اصل یک خوشه ساخته اند) و حتی چیزی در مورد خود رفتار (گزاره) بیان نمی‌کنند؛ علاوه بر این اصلا مقدار مفعول در نظر گرفته نشده است. حال اگر قوانین قوانین از روی این تراکنش‌ها بدست آید، فقط اطلاعاتی در مورد الگوی تکرار آیتم‌های (فاعلی) می‌دهد و هیچ اطلاعات مفیدی در مورد خوشه‌بندی ارائه نمی‌دهد.

۳- عدم استفاده از مفاهیم موجود در آنتولوژی و وجود نداشتن انسجام معنایی در تمام مراحل.

۳-۵-۲- نرم‌افزار کاوش داده‌های پیوندی

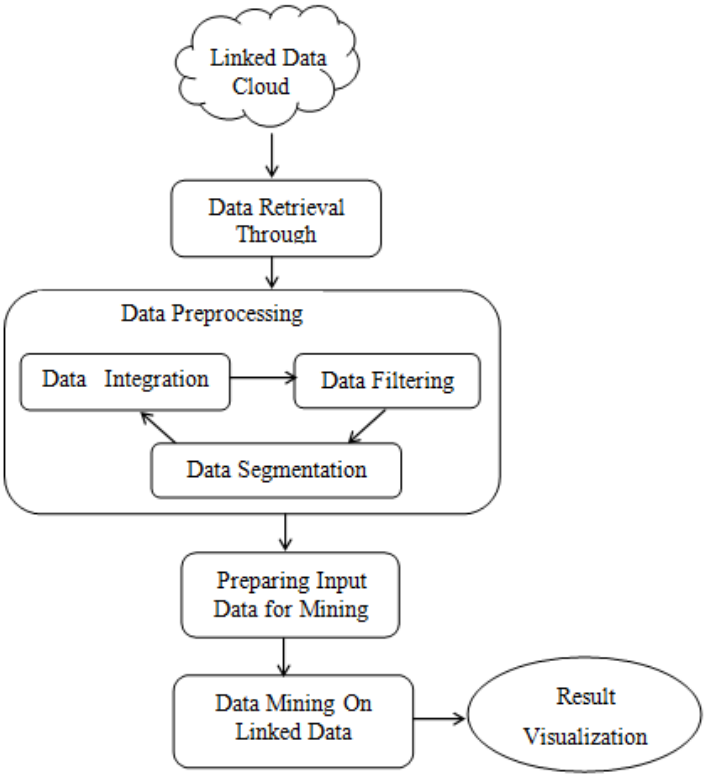
۱-۳-۵-۲- معرفی

در تعریفی که از داده های پیوندی شد، بیان گردید که هدف داده های پیوندی، اتصال مجموعه داده‌های وب معنایی به یکدیگر است. این گروه مقاله ای تحت عنوان " LIDDM: یک سیستم داده کاوی برای داده های پیوندی" [Nar11] منتشرکردند که در آن ابزاری به منظور داده‌کاوی (کلاس‌بندی، خوشه‌بندی و کاوش قوانین انجمنی) از داده های پیوندی ارائه گردیده است.

۲-۴-۳-۲- روش انجام کار

در این مقاله، نرم‌افزاری تحت عنوان LIDDM معرفی شده است که توانایی انجام عملیات داده‌کاوی را روی داده‌های پیوندی دارد. فرآیند کاری این نرم‌افزار در شکل ۲-۱۰ نشان داده شده است.

بسیاری از فرآیندهای معرفی شده در شکل، فرآیندهای عمومی هستند که در KDD[1] وجود داشته و از آنها استفاده می‌شود. این فرآیندهای عمومی روی داده های با ساختار جدولی (تعدادی سطر و ستون) کار می‌کنند. لذا تنها کاری که بایستی انجام شود این است که داده های استخراجی از داده‌های پیوندی (چندین منبع داده) به نحوی به داده با ساختار جدولی تبدیل شوند.



شکل ۲-۱۰: فرآیند کاری LIDDM [Nar11]

نحوه استخراج داده ها:

اولین گام در استفاده از این نرم افزار، استخراج داده‌های مورد نیاز است. به منظور استخراج این داده‌ها بایستی از زبان SPARQL استفاده نمود. بدین منظور ابتدا کاربر منبع داده مورد نظر خود را تعیین می‌کند. به منظور اجرای دستورات SPARQL بایستی آدرس نقطه نهایی[2] منبع داده مورد نظر را داشت. پس از پیدا نمودن این آدرس، بایستی دستور SPARQL مورد نظر را به گونه‌ای اجرا نمود تا داده‌های مورد نظر استخراج شوند.

[1] Knowledge Discovery and Data Mining
[2] Endpoint

خروجی دستورات SPARQL به صورت جدولی (تعدادی سطر و ستون) قابل مشاهده است و دیگر نیازی به تبدیل خروجی این دستور به داده های با ساختار جدولی وجود ندارد.

از آنجا که این نرم افزار روی داده های پیوندی کار می‌کند، این امکان وجود دارد که داده‌های مورد نظر از چندین منبع داده استخراج شوند. بدین منظور بایستی دستور SPARQL را روی چندین منبع داده اجرا کرده و نتایج به دست آمده از هر اجرا را با یکدیگر ترکیب نمود.

ترکیب داده های استخراج شده:

پس از استخراج داده‌ها از چندین منبع داده‌های متفاوت، بایستی آنها را به گونه ای با یکدیگر ترکیب کرد. برای این کار بایستی بین دو مجموعه داده استخراج شده، یک ستون مشترک (از نظر مقادیر متناظر، مثلا کد کشور یا نام کشور) وجود داشته باشد تا بر اساس آن بتوان این دو مجموعه داده را با یکدیگر ادغام نمود.

لازمه این امر این است که منابع داده مختلف، صفتی یکسان با مقادیر یکسان از یک موجودیت داشته باشند. در نرم افزار طراحی شده در این پژوهش، پس از استخراج داده از دو منبع داده، بایستی شماره دو ستونی که بر اساس آن لازم است دو منبع داده به یکدیگر متصل شوند، توسط کاربر مشخص شود. همچنین این قابلیت وجود دارد که نتایج استخراج شده را به وسیله یک ستون مشترک به یکدیگر وصل نمود یا اینکه آن را در انتهای نتایج قبلی قرار داد.

۲-۴-۳-۳- تحلیل

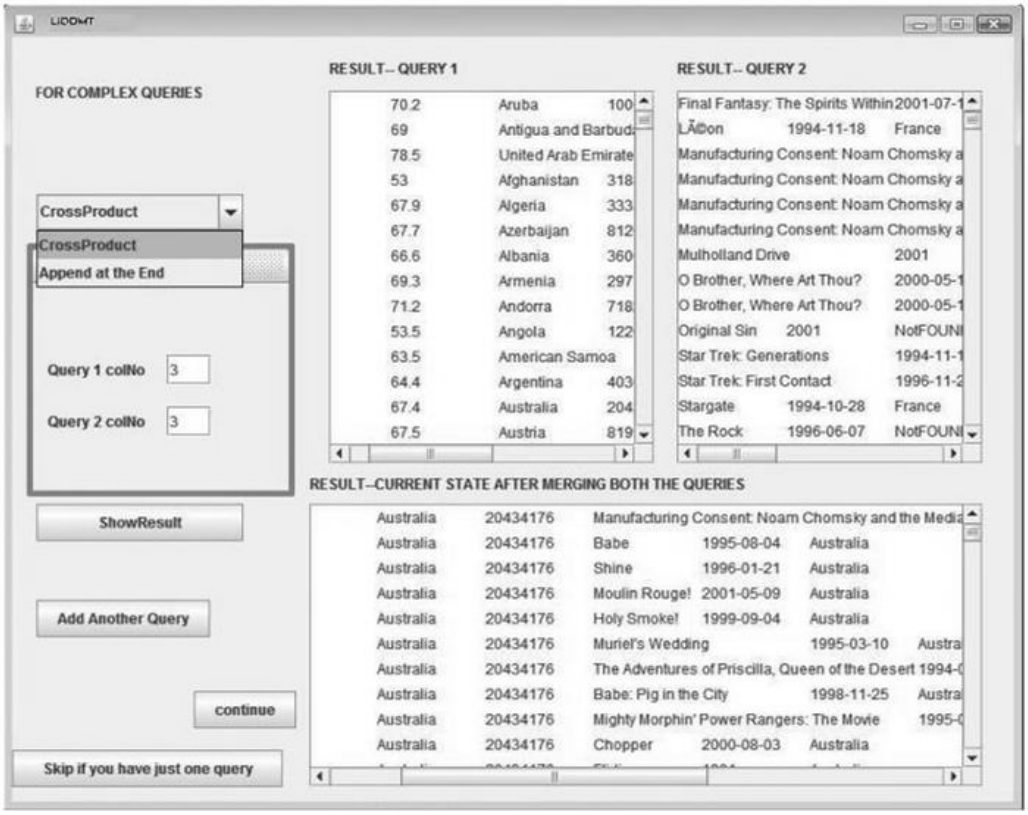
در مجموع می توان ویژگی های زیر را برای این مقاله در نظر گرفت:

۱- وب معنایی و داده‌های پیوندی: این نرم افزار هم قادر است عملیات داده کاوی را روی یک منبع داده وب معنایی انجام دهد و هم قادر است با داده‌های پیوندی کار کند.
۲- پراکندگی داده‌ها: داده‌های وب معنایی و به طبع داده‌های پیوندی، پراکندگی بسیار زیادی دارند. معمولا بسیار کم رخ می دهد که بتوان ستونی را بین دو مجموعه داده پیدا کرد که مقادیر متناظر یک موجودیت در دو منبع داده با یکدیگر برابر باشند. لذا در این روش معمولا زمانی که لازم است دو مجموعه داده با یکدیگر ترکیب شوند، در پیدا کردن دو ستون متناظر بسیار مشکل بوجود خواهد آمد.
۳- درگیر نمودن کاربر نهایی: به منظور استخراج داده‌ها، کاربر نهایی بایستی دستورات SPARQL مناسب را وارد نماید. لازمه این کار این است که: ۱) کاربر به این زبان تسلط داشته باشد. ۲) آدرس نقطه

نهایی مجموعه داده ها را داشته باشد. ۳) با ساختار آن مجموعه داده و آنتولوژی مربوط به آن آشنایی کامل داشته باشد.

۴- فراگیر بودن: این نرم افزار قادر است تمامی عملیات داده کاوی به همراه فرآیندهای مورد نیاز پیش پردازش و پس پردازش و نیز مصورسازی نتایج را به خوبی انجام دهد.

۵- عدم استفاده از مفاهیم موجود در آنتولوژی و وجود نداشتن انسجام معنایی در تمام مراحل یکی از مشکلات این کار است.



شکل ۲-۱۱: نمایی از نرم افزار LIDDM [Nar11]

۴-۵-۲- کاوش گراف/ درخت

کارهای زیادی روی یافتن قوانین انجمنی از داده های با ساختار گرافی و ساختار درخت انجام شده است [Gos05] [Chi05] [Viv10] [Kur01]. در این بخش، جزئیات روش‌های ارائه شده بیان نمی‌گردد. فقط در همین اندازه که نگرش‌های مختلفی در کاوش قوانین انجمنی از گراف و درخت وجود دارد، اما همه آنها در دو امر مشترکند:

۱) منبع داده مورد استفاده، داده‌های سنتی مانند پایگاه داده‌های رابطه‌ای است که پس از خواندن اطلاعات، الگوریتم اقدام به ساخت گراف یا درخت می‌نماید. یعنی درخت‌ها و گراف‌ها دقیقا بر اساس تراکنش‌ها ساخته می‌شوند.
۲) برخی روش‌ها به ازاء هر تراکنش یک زیر گراف یا یک زیر درخت می‌سازند و عمل کاوش بر اساس جستجوی زیر گراف یا زیر درخت انجام می‌گیرد. برخی نیز به ازاء هر موجودیت تنها یک راس می‌سازند که مثلاً یال‌های گراف یا درخت بیانگر تعداد تکرار دو موجودیت می‌باشد.

به هر حال، گرافی که از داده‌های وب معنایی به دست می‌آید، ساختاری متفاوت دارد و الگوریتم‌های فوق برای کاوش آنها مناسب نیستند. این الگوریتم‌ها در اصل جایگزینی برای الگوریتم Apriori به منظور یافتن قوانین انجمنی از تراکنش‌ها می باشند.

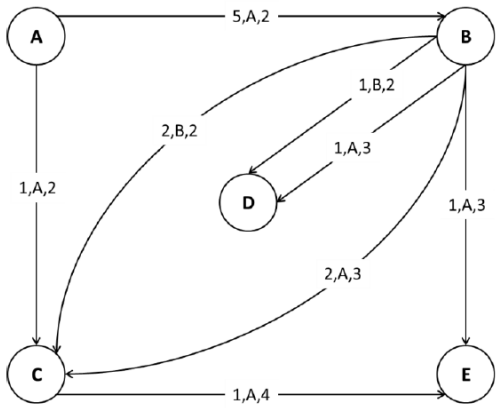
در ادامه در دو مثال بیان می شود که از مقالات ذکر شده استخراج شده‌اند. در اینجا چگونگی کاوش این درخت‌ها و گراف‌ها اهمیت ندارد. بلکه نکته مهم این است که این گراف‌ها و درخت‌ها از همان داده‌هایی تغذیه می‌کنند که الگوریتم سنتی Apriori تغذیه می‌کند و لذا این روش‌ها برای کار با داده‌های وب معنایی مناسب نیستند.

مثال اول، تراکنش‌ها را به گراف تبدیل می‌کند. اطلاعات جدول ۲-۸ را در نظر بگیرید.

جدول۲-۸: مثال از تراکنش ها به منظور تبدیل به گراف [Kur01]

Tid	List of Items
T001	A,B,C
T002	B,D
T003	B,C
T004	A,B,D
T005	A,C
T006	B,C
T007	A,B
T008	A,B,C,E
T009	A,B,C

گراف حاصل از تراکنش های این جدول، در شکل ۲-۱۲ نشان داده شده است.



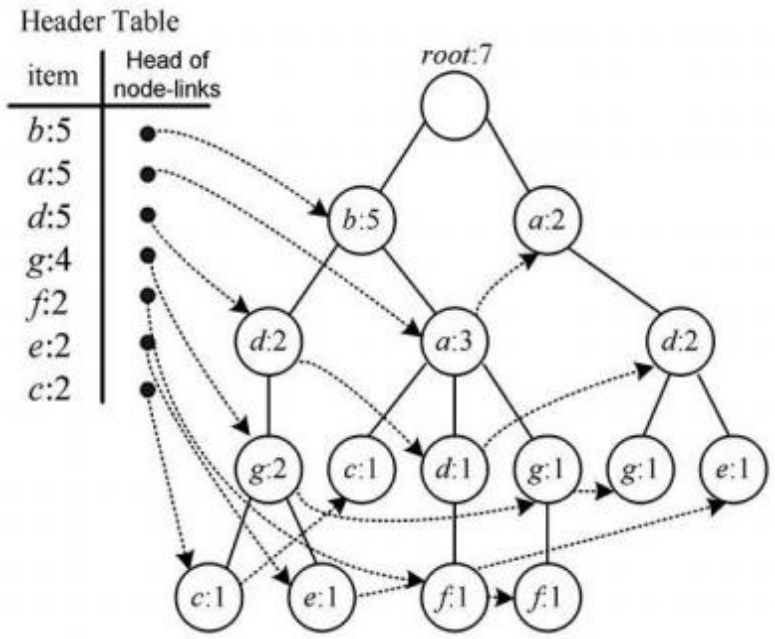
شکل۲-۱۲: گراف حاصل از تراکنش های جدول۲-۸ [Kur01]

مثال دوم، تراکنش‌ها را به درخت تبدیل می‌کند. اطلاعات جدول ۲-۹ را در نظر بگیرید.

جدول ۲-۹: مثال از تراکنش ها به منظور تبدیل به درخت [Gos05]

Tid	List of Items
T001	a,d,e
T002	b,a,f,g,h
T003	b,a,d,f
T004	b,a,c
T005	a,d,g.k
T006	b,d,g,c,i
T007	b,d,g,r,j

درخت حاصل از تراکنش‌های این جدول، در شکل۲-۱۵ نشان داده شده است.



شکل ۲-۱۳: درخت حاصل از تراکنش های جدول ۲-۹ [Gos05]

مستقل از روش کاوش این گراف‌ها و درخت‌ها، نکته مهم این است که این گراف‌ها و درخت‌ها از روی تراکنش‌ها ساخته می شوند، در حالی که در داده‌های وب معنایی، مفهومی به نام تراکنش وجود ندارد [Ram12].

۲-۵-۵- مقایسه کارهای مشابه

در بخش ۲-۵ برخی از پژوهش‌های مشابه که به کاوش قوانین انجمنی از داده‌های معنایی می‌پردازند، توصیف شدند. در جدول ۲-۱۰ مقایسه مختصر این کارها ارائه شده است. در روش پیشنهادی سعی شده است جهت برآورده‌سازی قابلیت‌هایی در جدول که در هیچ یک از کارهای ارائه شده وجود ندارند نیز راه‌کارهایی ارائه گردد.

جدول ۲-۱۰: مقایسه کیفی کارهای مشابه

قابلیت	روش ارائه شده در [Neb12]	روش ارائه شده در [Zia11]	روش ارائه شده در [Nar11]
قابلیت استخراج دانش جدید از داده‌های معنایی	+	+	+

عدم ناهمگنی ساختاری	+ با حذف معانی در مرحله ساخت تراکنش	+ با حذف یکی از عناصر سه‌تایی	+ با تبدیل داده‌ها به جدول‌های رابطه‌ای
جامع بودن	–	–	+
کار با داده‌های پیوندی	–	–	+
استفاده از منطق	+	–	–
عدم دخالت شدید کاربر	–	+	–
مفهوم‌سازی حوزه مدنظر	–	.–	–
فرمت استاندارد برای توصیف داده‌ها در تمام مراحل	–	–	–
وجود ابرداده برای تراکنش‌ها و قوانین انجمنی	–	–	–
دستیابی به قوانین با سطح مفهومی بالاتر	–	–	–
قابلیت کار با داده‌های جریان‌دار	–	–	–

۶-۲- خلاصه فصل

این فصل دربردارنده مفاهیم پایه‌ای است که در این تحقیق به آنها نیاز می‌باشد. در ابتدای فصل به معرفی مفهوم داده کاوی، کاوش قوانین انجمنی و انواع الگوریتم‌های کاوش قوانین انجمنی پرداخته شد و روی الگوریتم Apriori که پایه الگوریتم‌های کاوش قوانین انجمنی است نیز با جزئیات بحث شد. سپس مدل‌های کلی استخراج قوانین انجمنی از جریان‌های داده بررسی شده و نمونه‌هایی از هر کدام بیان شد.

در ادامه بحث، وب معنایی و فناوری‌های آن مورد بحث قرار گرفت و بیان شد که هر فقره اطلاعاتی در وب معنایی از سه بخش فاعل، گزاره و مفعول تشکیل می‌شود که این بخش‌ها توسط آنتولوژی‌ها توصیف می‌گردند.

سپس در انتها چند مورد از کارهای مشابه در رابطه با کاوش قوانین انجمنی از وب معنایی و داده‌های پیوندی معرفی شده و مورد تحلیل و مقایسه قرار گرفتند و نیز به بیان اصول کاوش قوانین انجمنی از گراف و درخت پرداخته شده است.

فصل ۳. فصل ۳- روش پیشنهادی

۱-۳- مقدمه

در فصل گذشته مفاهیم اولیه مرتبط و همچنین برخی از کارهای مشابه انجام شده به همراه تحلیل آنها بیان گردید. در این فصل ابتدا روش ارائه شده جهت کاوش جریان‌های داده معنایی و استخراج قوانین انجمنی از آنها به صورت کلی معرفی شده و سپس به بررسی جزئیات مراحل آن خواهیم پرداخت.

با توجه به چالش‌های ذکر شده در فصل قبل بیشترین استفاده ممکن از دانش بوجود آمده توسط آنتولوژی‌ها به گونه‌ای که تراکنش‌ها به آسانی قابل تعریف و کاوش باشند، بسیار مهم و اساسی است. راه‌حل‌های بسیار مختلفی برای ایجاد آیتم‌ها و تراکنش‌ها از داده‌های معنایی وجود دارد که به ماهیت معانی و کاربرد بستگی پیدا می‌کند. در نتیجه لازم است که مسأله ایجاد تراکنش‌ها و کاوش قوانین انجمنی در داده‌های معنایی تعریف گردد و راه حلی برای استخراج بهینه آیتم‌ها و تراکنش‌هایی معنایی به طریقی که برای الگوریتم‌های کاوش قوانین انجمنی مناسب باشند، ارائه گردد.

در این تحقیق جهت حذف پیچیدگی‌های داده‌های ناهمگن معنایی در راستای استفاده از غنای معنایی داده‌ها و به منظور تبدیل داده‌ها به فرمت قابل پردازش و سازگار با الگوریتم‌های قوانین انجمنی، از مفاهیم موجود در آنتولوژی کاربرد و برچسب‌های زمانی داده‌ها استفاده می‌گردد. همچنین لازم است در کنار این مساله جریان‌دار بودن این داده‌ها و محدودیت‌های مرتبط با آن از قبیل محدودیت‌های زمانی و پیچیدگی‌های مرتبط با مدیریت جریان پیوسته داده‌ها، در نظر گرفته شود. بدین ترتیب که مفاهیم مرتبط با کاوش قوانین انجمنی از قبیل آیتم‌ها و تراکنش‌ها در آنتولوژی تعریف شده و داده‌های معنایی بدون حذف معنایشان در این قالب‌های از پیش تعریف شده درآیند.

در کارهای معدود قبلی که بر روی استخراج قوانین انجمنی از داده‌های معنایی تمرکز داشته‌اند، محتوای خاص و الگوهای مورد نظر به صورت دستی وارد شده و با پرس‌وجو بر روی مخزنی از داده‌های غیر پویا، تبدیل به فرمت قابل قبول برای الگوریتم‌های داده کاوی صورت می‌گرفته است. اما در این تحقیق استخراج تراکنش‌های معنایی به صورت اتوماتیک و تنها بر اساس برچسب‌های زمانی رویدادها صورت می‌گیرد و همچنین با توجه به جریان پیوسته داده‌ها با استفاده از پرس‌وجوی پیوسته و مکانیزم‌های پنجره‌ای قابلیت کار با داده‌های

جریان‌دار که در حال حاضر مخاطبان زیادی داشته و به طور وسیع در کاربردهای مختلف وجود دارند و نیاز پرداختن به آنها در زمینه داده کاوی احساس می‌شود، فراهم می‌آید. به علاوه انتظار می‌رود قوانین انجمنی بدست آمده با توجه به استفاده از مفاهیم و روابط موجود در آنتولوژی و استفاده از این معانی دارای مفاهیم کلی‌تر و انتزاعی‌تری به نسبت قوانین انجمنی معمول در داده کاوی سنتی بوده و در تصمیم‌گیری‌ها بسیار مفیدتر واقع شوند.

به طور کلی می‌توان گفت هدف اصلی این تحقیق رفع چالش‌های ذکر شده و امکان‌پذیر ساختن پردازش حجم وسیع داده‌های معنایی جریان‌دار موجود و کشف و ذخیره‌سازی قوانین انجمنی جدید با استفاده از مفاهیم موجود در آنتولوژی و قوانین اولیه موجود می‌باشد.

برای امکان‌پذیر ساختن کاوش داده‌های معنایی جریان‌دار موجود و کشف قوانین انجمنی جدید با استفاده از یک روش معنایی بهینه لازم است در مرحله اول طریقه تبدیل نمونه داده‌های جریان‌داری که در قالب آنتولوژی قرار دارند به تراکنش‌های معنایی آماده برای داده کاوی به منظور پردازش این داده‌ها با استفاده از الگوریتم‌های استخراج قوانین انجمنی معنایی، مورد بررسی قرار گیرد. روش مورد نظر با توجه به مساله جریان‌دار بودن و حاشیه نویسی معنایی داده‌ها با دیگر روش‌هایی که تا کنون ارائه شده اند، متفاوت خواهد بود.

الگوریتم‌های کشف قوانین انجمنی با مجموعه داده‌های تشکیل شده از تراکنش‌ها که هر کدام شامل زیرمجموعه‌ای از آیتم‌ها می‌باشند، کار می‌کنند. برای تبدیل داده‌های معنایی جریان‌دار موجود به این تراکنش‌ها می‌توان از زبان‌های پرس‌وجوی معنایی پیوسته مانند C-SPARQL[1] استفاده نمود یا به گونه‌ای شیوه کار این ابزارها را شبیه‌سازی نمود. البته یکی از موارد قابل توجه، معنایی بودن این تراکنش‌ها می‌باشد که تحقیق مورد نظر از این لحاظ با کارهای گذشته متفاوت است. منظور از معنایی بودن تراکنش‌ها این است که هر تراکنش در قالب تعریف شده در آنتولوژی قرار می‌گیرد و شامل آیتم‌هایی است که رویدادهای رخ داده در یک موقعیت زمانی را به صورت سه‌تایی‌های RDF نشان می‌دهند. همچنان خودکارسازی و محدود نمودن دخالت کاربر انسانی جهت جلوگیری از خطاهای انسانی، در این مرحله مورد توجه قرار می‌گیرند. سپس برای رسیدن به قوانین کلی‌تر و بامعناتر از قوانینی که از آنتولوژی کاربرد استخراج می‌شود جهت ایجاد تراکنش‌های معنادار کمک خواهیم گرفت. به علاوه در این تبدیل است که با استفاده از یک فرآیند پیش‌پردازش داده‌های مقداری به آیتم‌های غیرمقداری تغییر داده می‌شوند. در واقع این پیش‌پردازش مقادیر محتمل را دسته‌بندی کرده و داده‌ها در مرحله ایجاد تراکنش‌ها با استفاده از این فرآیند کلی‌تر بیان می‌شوند تا در مرحله ایجاد قوانین به طور مناسبی مورد استفاده قرار گیرند.

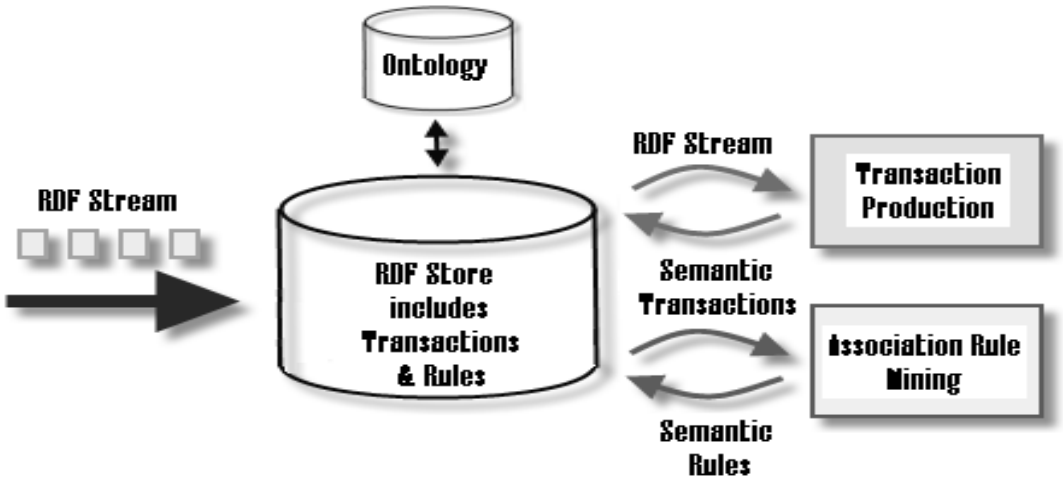
[1] Continuous SPARQL

در گام بعد الگوریتم‌های کشف آیتم‌های پرتکرار جهت استخراج قوانین انجمنی اجرا می‌شوند. در این مرحله از الگوریتم پایه Apriori استفاده شده و این الگوریتم بنا به کاربرد مورد نظر بهینه می‌شود تا بتواند با داده‌های معنایی کار کند. همچنین مقادیری از قبیل درجه پشتیبانی و درجه اطمینان باید برای انواع نمونه داده‌ها تعیین گردند.

در مرحله نهایی لازم است نحوه تبدیل نتایج الگوریتم‌های داده کاوی به قوانین در قالب آنتولوژی و افزودن آنها به منبع قوانین تعیین گردد. این تبدیل نیز با استفاده از قابلیت‌هایی چون قالب‌های مشخص شده در آنتولوژی و با کمک ابزارهایی چون زبان‌های پرس‌وجوی معنایی صورت می‌گیرد. در نتیجه انتظار می‌رود قوانینی مفهومی‌تر که به بهبود و هوشمندسازی تصمیم‌گیری منجر خواهند شد، قابل دستیابی باشند.

۲-۳- معرفی مدل پیشنهادی

معماری کلی سیستم پیشنهادی در شکل ۳-۱ نشان داده شده است. در این سیستم ابتدا جریان‌های داده معنایی که به صورت آفلاین از ابر داده پیوندی (LOD) استخراج شده اند، به سیستم وارد شده و در مخزن معنایی سیستم (RDF Store) ذخیره می‌گردند. در سیستم دو نوع آنتولوژی موجود است که انسجام معنایی را در تمامی بخش‌ها حفظ نموده و در مراحل مختلف کار از آنها استفاده می‌گردد. نوع اول آنتولوژی کاربرد است که بنا بر نوع داده ورودی در سیستم قرار می‌گیرد. آنتولوژی نوع دوم آنتولوژی کاوش قوانین انجمنی (ARM) می‌باشد که در این کار طراحی و پیاده‌سازی گردیده است و مفاهیم و روابط مورد نیاز در رابطه با کاوش قوانین انجمنی را پوشش می‌دهد.



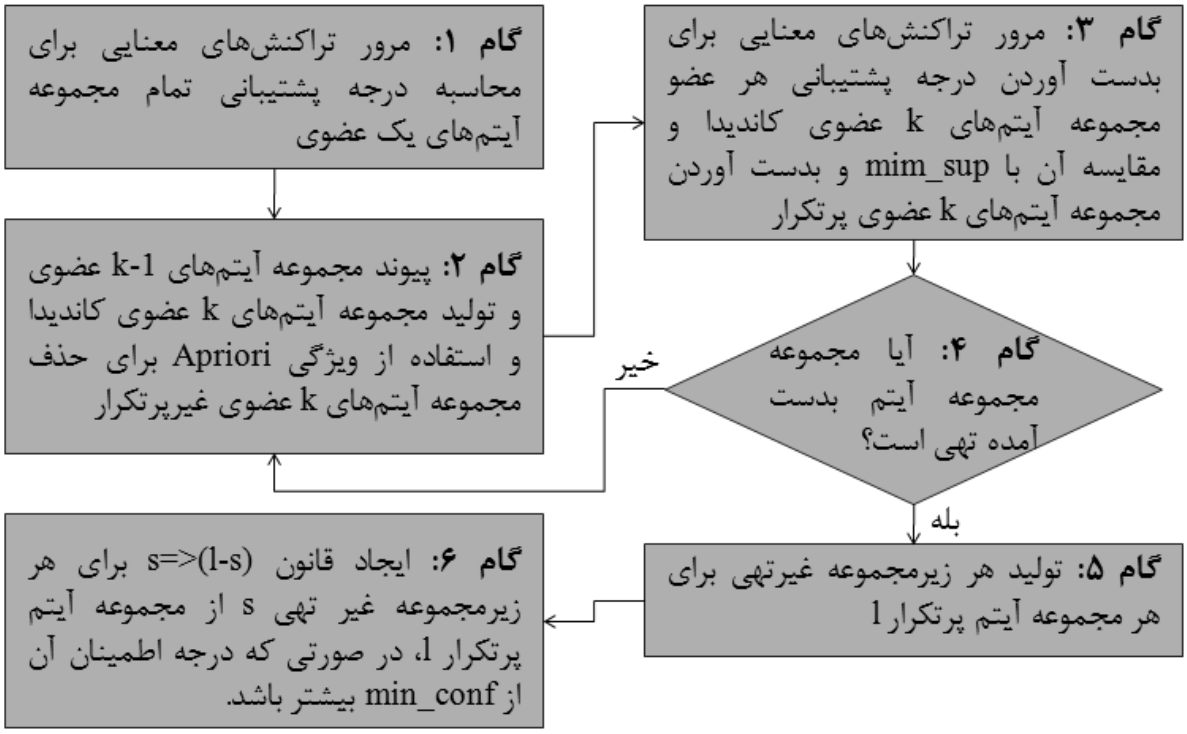
شکل ۳-۱: معماری سیستم پیشنهادی

کار عمده سیستم در دو ماژول کلی انجام می‌گیرد:

۱) ماژول ایجاد تراکنش:[1] این ماژول کار استخراج تراکنش‌های معنایی از جریان‌های داده معنایی را در دو سطح معنایی اولیه و ثانویه به انجام می‌رساند. در سطح اولیه با استفاده از مدل پنجره لغزنده که اندازه و نرخ حرکت آن با توجه به نمونه داده‌های کاربرد خاص مشخص گردیده است، استخراج تراکنش‌ها صورت می‌گیرد. بدین ترتیب که رویدادهای رخ داده در یک پنجره لغزنده در هر موقعیت زمانی، که به صورت سه‌تایی‌های زماندار در مخزن قرار دارند، آیتم‌های یک تراکنش را تشکیل می‌دهند. به علاوه هر تراکنش معنایی که در قالب آنتولوژی ARM در مخزن ذخیره می‌گردد، دارای یک شناسه تراکنش نیز می‌باشد.
در سطح ثانویه یعنی استخراج تراکنش‌های معنایی با سطح مفهومی بالاتر، پرس‌وجوهایی معنایی با توجه به مفاهیم و روابط موجود در آنتولوژی کاربرد تعریف می‌گردند. سپس این پرس‌وجوها بر روی تراکنش‌های معنایی سطح اول اعمال گردیده و تراکنش‌های معنایی سطح دوم را بوجود می‌آورند.

۲) ماژول کاوش قوانین انجمنی:[2] ورودی این ماژول هر دو سطح از تراکنش‌های معنایی ایجاد شده در ماژول قبلی است که در مخزن معنایی سیستم در قالب آنتولوژی ذخیره شده اند. در این بخش از سیستم الگوریتم Apriori برای کار بر روی این تراکنش‌های معنایی بهینه و پیاده‌سازی می‌شود. بنابراین همان طور که در توضیح الگوریتم Apriori بیان شد و در شکل ۳-۲ مشاهده می‌شود، مجموعه‌آیتم‌های پرتکرار با استفاده از ویژگی Apriori و با توجه با حداقل درجه پشتیبانی مشخص شده استخراج می‌گردند. سپس قوانین انجمنی در صورت برقراری شرط درجه اطمینان ایجاد می‌گردند. در انتها قوانین انجمنی معنایی ایجاد شده در قالب تعریف شده در آنتولوژی ARM ذخیره می‌شوند.

[1] Transaction Production Module
[2] Association Rule Mining Module



شکل ۳-۲: الگوریتم Apriori پیاده شده در روش پیشنهادی

۳-۳- مراحل عملیاتی کردن مدل پیشنهادی

فاز اول از مراحل عملیاتی کردن مدل پیشنهادی، جمع‌آوری منابع و بررسی نتایج کارهای انجام شده مرتبط و دستیابی به آگاهی نسبی از مرز دانش موضوع مورد نظر بوده است.

فاز دوم در سه گام عمده صورت گرفته است. ***گام اول از فاز دوم،*** فراهم‌آوری یک آنتولوژی یا چندین آنتولوژی برای کاربرد مورد نظر و توسعه آنتولوژی ARM با توجه به مفاهیم مورد نیاز در این تحقیق بوده است. از جمله کارهایی که در این گام صورت گرفته است، افزودن تعریف قالب و مفهوم تراکنش‌ها و آیتم‌ها، قوانین انجمنی نهایی و مقادیر و معیارهایی که در رابطه با کاوش قوانین انجمنی مطرحند به آنتولوژی مورد نظر می‌باشد. به علاوه قابلیت زماندار بودن و اصالت رویدادها نیز در نظر گرفته شده‌اند. همچنین آماده‌سازی مخزن داده‌های جریان‌داری که بر پایه این آنتولوژی فراهم می‌آیند نیز از فرآیندهای این گام بوده است. به علاوه در این گام نحوه پیش‌پردازش و کلی‌سازی داده‌ها برای تبدیل مقادیر عددی داده‌ها به مقادیر کیفی نیز مشخص گردیده است. پرس‌وجوهای مورد نیاز برای ایجاد تراکنش‌های ثانویه نیز با استفاده از مفاهیم و روابط موجود در آنتولوژی کاربرد آماده‌سازی شده‌اند. در این مرحله ابزارهای کار با آنتولوژی از قبیل Protégé و ابزار برنامه‌نویسی معنایی به خصوص کتابخانه Jena مورد استفاده قرار گرفته اند.

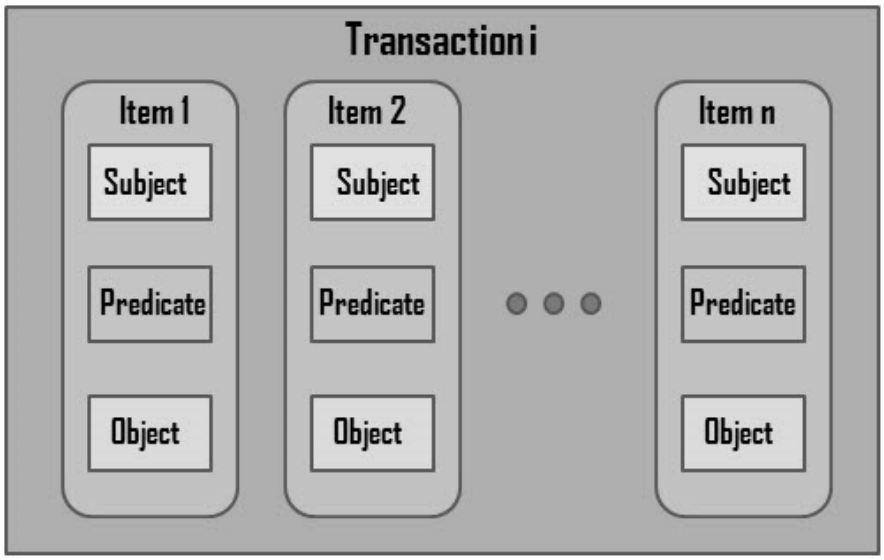
گام دوم از فاز دوم، طراحی الگوریتم مورد نظر با هدف کشف و ذخیره‌سازی قوانین انجمنی به صورت پیوسته از داده‌های معنایی جریان‌دار بوده است.

کارهای عمده‌ای که با اجرای این الگوریتم انجام گرفته‌اند در دو فرآیند اصلی خلاصه می‌شوند:

- تبدیل نمونه داده‌های RDF اولیه به قالبی معنایی به طوری که قابل قبول برای الگوریتم داده کاوی مورد نظر باشند.
- اعمال الگوریتم داده کاوی معنایی مناسب بر روی داده‌های خروجی مرحله قبل و تبدیل نتایج الگوریتم داده‌کاوی به قوانینی در قالب آنتولوژی و افزودن آنها به مخزن قوانین.

روند اجرای سیستم به این صورت است که در فرآیند اول داده‌های مورد نظر که در قالب RDF زماندار به سیستم وارد می‌شوند، در مخزن نمونه‌داده‌های معنایی ذخیره می‌شوند. سپس این داده‌ها با استفاده از پنجره لغزنده و با کمک ابزارهایی از قبیل زبان‌های پرس‌وجو به قالب مورد نظر درآمده و تراکنش‌ها و مجموعه آیتم‌ها استخراج می‌گردند تا از آنها به عنوان ورودی فرآیند دوم استفاده شود. به علاوه با اعمال پرس‌وجوهای طراحی شده در فاز قبلی تراکنش‌های سطح دوم نیز ایجاد می‌گردند.

قالب کلی هر تراکنش معنایی در سیستم پیشنهادی در شکل ۳-۳ نشان داده شده است.



شکل ۳-۳: تراکنش در سیستم پیشنهادی

در فرآیند دوم الگوریتم داده‌کاوی انتخاب شده، بر روی تراکنش‌های معنایی اجرا می‌گردد و طی مراحلی از قبیل کشف مجموعه آیتم‌های پرتکرار، قوانین انجمنی معنایی - که در سطوح مختلف از سطح نمونه‌داده تا سطح روابط و مفاهیم تعریف شده در آنتولوژی می‌باشند - استخراج می‌گردند. شرایط تولید قوانین با معیارها و مقادیری از قبیل درجه پشتیبانی و درجه اطمینان که در قالب آنتولوژی ذخیره شده‌اند، تعیین می‌گردد. با

بدست آمدن هر قانون در فرآیند دوم، قانون بدست آمده با استفاده از آنتولوژی به قالب RDF تعریف شده در آنتولوژی ARM تبدیل شده و در مخزن قوانین ذخیره می‌شود تا در تصمیم‌گیری هوشمند و همچنین جهت استخراج قوانین جدید مورد استفاده قرار گیرد.

با توجه به فرآیندهای ذکر شده، برخی مسائل که در فاز دوم تحقیق مورد توجه قرار گرفته‌اند، عبارتند از:

- تعیین نحوه استخراج تراکنش‌های معنایی: لازمه این کار آماده‌سازی آنتولوژی، داده‌ها، نحوه پیش‌پردازش، اندازه و نرخ حرکت پنجره لغزنده و پرس‌وجوهای معنایی، در گام دوم، جهت مشخص نمودن تراکنش‌ها بر اساس این موارد بوده است.

از آنجایی که در تراکنش‌های ورودیِ الگوریتم‌های کاوش قوانین انجمنی، رابطه‌ای که بین آیتم‌های یک تراکنش وجود دارد، رخداد همزمان آیتم‌ها می‌باشد، نحوه ساخت تراکنش‌ها وابسته به زمان رخداد هر آیتم و قرابت زمانی میان آیتم‌های هر تراکنش می‌باشد. بنابراین مکانیزم پنجره زمانی به صورتی انتخاب گردیده است که هر آیتم با تمامی آیتم‌هایی که با آن قرابت زمانی دارند حداقل در یک تراکنش به طور مشترک رخ دهند و به دلیل آفلاین بودن اجرای فرآیند کاوش، شرایط یکسانی برای همه آیتم‌ها در نظر گرفته شده است. بنابراین مدل پنجره لغزنده انتخاب شده است.

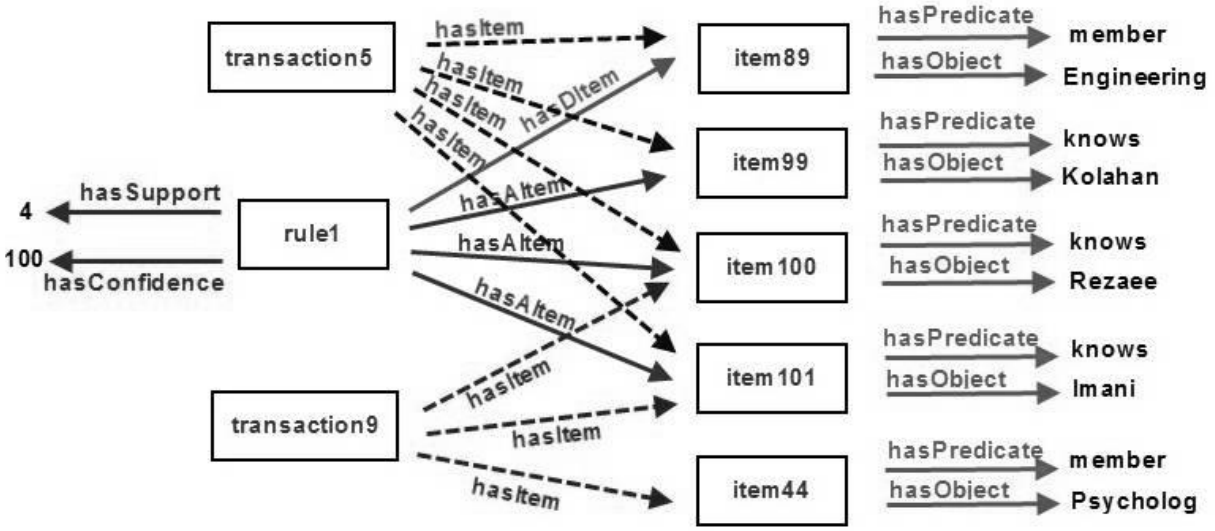
- نحوه تبدیل قوانین به قالب تعریف شده در آنتولوژی: قالب قوانین به گونه‌ای در آنتولوژی مشخص گردیده است که نزدیک به قالب قوانین انجمنی بوده و تبدیل قوانین انجمنی به آن آسان و کم هزینه باشد و برای استفاده در استنتاج‌های معنایی آینده نیز مفید واقع شوند.

گام سوم از فاز دوم، مرحله پیاده‌سازی الگوریتم طراحی شده در گام دوم بوده است، این پیاده‌سازی با استفاده از زبان برنامه‌نویسی جاوا صورت گرفته است و از کتابخانه‌های معنایی به خصوص Jena برای برنامه‌نویسی معنایی - به عنوان نمونه استفاده از زبان‌های پرس‌وجوی معنایی - استفاده شده است. علاوه بر پیاده‌سازی الگوریتم نحوه استخراج داده‌ها از ابر LOD و ورود این داده‌های جریان‌دار به سیستم مشخص شده و همچنین نحوه ذخیره‌سازی در بخش‌های مختلف سیستم از قبیل مخزن داده‌ها، مخزن قوانین و آنتولوژی مشخص گشته است.

پیاده‌سازی الگوریتم Apriori روی داده‌های معنایی ابتدا با داده‌های معنایی غیر جریان‌دار موجود به صورت موازی با آماده سازی داده‌های اصلی انجام گرفته است. داده‌های اولیه بخشی از داده‌های مربوط به برخی از اساتید دانشگاه فردوسی، منتشر شده در آدرس http://wtlab.um.ac.ir/linkeddata/profs بوده و مشخصه‌های انتخاب شده عبارتند از: http://xmlns.com/foaf/0.1/knows و http://xmlns.com/foaf/0.1/member. در برنامه اولیه از مباحثی از قبیل جریان‌دار بودن داده‌ها، پیش‌پردازش داده‌ها و سطوح معنایی مختلف برای قوانین

انجمنی با استفاده از پرس‌وجوهای معنایی صرف نظر شده و به نحوه اعمال الگوریتم Apriori بر روی داده‌های معنایی پرداخته شده است. در این برنامه با حذف مفعول آیتم‌ها دوتایی در نظر گرفته شده است تا در مجموعه داده اولیه داده‌های تکراری بوجود آید و بنابراین هر مفعول (یعنی هر استاد) معرف یک تراکنش می‌باشد. این مجموعه داده شامل ۵۰ تراکنش و ۵۶۸ آیتم مجزا می‌باشد.

در نتیجه برنامه پیاده سازی شده داده‌های مورد نظر را از مخزن معنایی با استفاده از فناوری معنایی جدا نموده و در قالب تراکنش‌های تعریف شده در آنتولوژی در می‌آورد. پس از فراهم‌آوری تراکنش‌های معنایی، الگوریتم Apriori بر روی آن پیاده‌سازی شده و قوانین معناداری استخراج شده است. در تمامی مراحل اجرای برنامه حاشیه نویسی معنایی داده‌ها حفظ شده و قوانین بدست آمده نیز معنایی می‌باشند. به عنوان مثال یک نمونه از قوانین بدست آمده عبارتست از: "کسانی که عضو دانشکده مهندسی هستند و آقایان کلاهان و رضایی پژند را می‌شناسند، آقای ایمانی را نیز می‌شناسند." که این افراد اساتید یک گروه هستند. این قوانین در قالب معنایی ذخیره شده و دارای قابلیت‌هایی از قبیل لینک شدن با داده‌های دیگر و استنتاج معنایی می‌باشند. یک نمونه از این قوانین معنایی در شکل زیر نشان داده شده است.



شکل ۳-۴: نمونه قانون معنایی بدست آمده از مجموعه داده غیرجریان‌دار اولیه

فاز سوم اجرا، تست و ارزیابی و همچنین رفع اشکالات احتمالی سیستم می‌باشد. جهت ارزیابی از معیارهای استاندارد سنجش الگوریتم‌های داده‌کاوی موجود که به استخراج قوانین انجمنی می‌پردازند استفاده شده است. همچنین جهت مقایسه و تعیین مزیت‌های حاصله از این تحقیق پیاده‌سازی‌های دیگری نیز در این مرحله انجام گرفته است.

در نهایت در ***فاز چهارم*** متن رساله کارشناسی ارشد بر اساس نتایج بدست آمده از تحقیق، تنظیم شده و آماده‌سازی مراحل دفاع از پایان‌نامه صورت گرفته است.

۴-۳- پاسخگویی روش پیشنهادی به چالش‌های موجود

در فصل قبل، کارهای مشابه که به کاوش قوانین انجمنی از داده‌های وب معنایی و داده‌های پیوندی می‌پردازند معرفی شده و تحلیل هر یک بیان گردید. تمامی کارهای معرفی شده در چندین ویژگی مشترک اند که باعث می‌شود برای کار با داده‌های معنایی و به خصوص جریان‌های داده معنایی مناسب نباشند. در این بخش مشکلات را بیان نموده و سپس توضیح داده می‌شود که روش معرفی شده چگونه به حل آنها می‌پردازد.

عدم وجود تعریف دقیق از تراکنش‌ها در داده‌های معنایی: بیان شد که داده‌های وب معنایی کاملا انعطاف پذیرند. بدین معنا که اطلاعاتِ موجودیت‌های مختلف را می‌توان در زمان‌های مختلف ثبت کرد و حتی به ازاء موجودیت‌های هم نوع، صفات مختلفی داشت؛ لذا پیدا کردن همبستگی و ارتباط بین موجودیت‌ها و داده‌ها بسیار سخت است. بر همین مبنا در داده‌های وب معنایی چیزی به معنای واقعی تراکنش وجود ندارد. اما تمام روش‌های معرفی شده در کاوش قوانین انجمنی، تمام محاسبات خود را بر اساس تراکنش‌ها انجام می دهند و لذا به نحوی به کمک کاربر اقدام به ساخت تراکنش از داده‌های موجود می‌کنند. لازمه این کار این است که دقیقا بدانیم دنبال چه نوع قوانین انجمنی هستیم و بر همان مبنا تراکنش‌ها ساخته شوند؛ اما یکی از هنرهای داده کاوی این است که دانش‌هایی را استخراج می‌کند که اصلا در مورد وجود آنها اطلاع نداریم. روش معرفی شده در این تحقیق با طراحی مفهوم تراکنش به طور کلی برای هرگونه رویداد زماندار معنایی، تراکنش‌های معنایی را ایجاد نموده و سپس قوانین انجمنی را بر اساس آنها تولید می‌نماید.

داده‌های معنایی ناهمگن: تمام کارهای معرفی شده از الگوریتم‌های سنتی به منظور کاوش قوانین انجمنی استفاده می‌کنند. اینگونه الگوریتم‌ها به منبع داده‌های همگن نیاز دارند و این لازمه وجود تراکنش است. ساده‌ترین حالت که آنالیز سبد خرید است را در نظر بگیرید؛ در این حالت هر تراکنش متشکل از چند عنصر ساده است. حالت پیچیده‌تر استفاده از پایگاه داده‌های رابطه ای است که پایگاه داده‌های رابطه ای نیز جزء منابع داده همگن به حساب می‌آیند. یعنی صفات هر موجودیت دقیقا مشخص است. اما داده‌های وب معنایی کاملاً ناهمگن هستند. بدین معنا که ممکن است دو موجودیت هم نوع، حتی یک نوع صفت مشترک نیز نداشته باشند که وجود این صفت‌ها و مقدار آنها در کاوش قوانین انجمنی موثر خواهد بود. الگوریتم معرفی شده برای تعامل با

این نوع داده‌ها، با استفاده از قالب تعریف شده در آنتولوژی کاوش قوانین انجمنی، صفات و مقادیر آنها را نیز در کاوش قوانین انجمنی در نظر می‌گیرد.

وجود رابطه بین موجودیت ها در داده‌های معنایی: همانطور که بیان شد در داده‌های وب معنایی، هر عنصر اطلاعاتی از سه بخش تشکیل شده است؛ به عبارت دیگر یک رابطه (گزاره) بین دو موجودیت (فاعل و مفعول). در الگوریتم های کاوش قوانین انجمنی سنتی مانند Apriori بین آیتم‌های یک تراکنش تنها یک رابطه وجود دارد و آن هم رابطه "باهم بودن" است؛ مثلا کالاهای باهم خریداری شده. اما به علت وجود روابط مختلف بین موجودیت ها در داده‌های وب معنایی، بایستی حتماً این روابط را در فرآیند کاوش در نظر گرفت. برای حل این مشکل، در الگوریتم معرفی شده هر آیتم برابر با یک سه‌تایی زماندار در نظر گرفته شده است و از مفاهیم و روابط ثابت موجود در آنتولوژی نیز با استفاده از استنتاج معنایی بر روی تراکنش‌ها بهره گرفته می‌شود.

درگیر نمودن کاربر نهایی در فرآیند کاوش: تمام روش‌های معرفی شده به منظور کاوش قوانین انجمنی نیازمند استخراج تراکنش هستند که این امر مستلزم دخالت دادن کاربر در تعریف تراکنش‌ها است. بدین منظور کاربر نهایی بایستی با ساختار داده‌ها و همچنین روش‌های تعیین الگو کاملا آگاه باشد که این امر باعث سختی فرآیند کاوش می شود. اما در الگوریتم معرفی شده کاربر نهایی درگیر فرآیند کاوش نخواهد شد.

کاوش جریان‌های داده معنایی: با وجود حجم عظیم جریان‌های داده معنایی در کاربردهای مختلف، نیاز مبرم به استخراج مفاهیم جدید از این داده‌ها و عدم توجه کارهای انجام شده در زمینه استخراج قوانین انجمنی از داده‌های معنایی، به جریان‌های داده معناییِ زماندار، در این روش به این گونه داده‌ها پرداخته شده و از مدل پنجره لغزنده جهت استخراج تراکنش‌های معنایی استفاده گردیده است.

۵-۳- نوآوری‌های روش پیشنهادی

همان طور که در فصل گذشته بیان شد، در سال‌های اخیر توجه بیشتری به ترکیب دو زمینه تحقیقاتی وب معنایی و داده کاوی صورت گرفته است. اما این ترکیب اغلب در قالب یادگیری آنتولوژی و کاوش وب بوده است و کار کمی در زمینه کاوش بر روی خود داده‌های معنایی صورت گرفته است. به علاوه در مقایسه با دیگر روش‌های داده کاوی بر روی داده‌های معنایی مانند خوشه‌بندی و کلاس‌بندیِ نمونه داده‌های معنایی، کارهای خیلی کمتری بر روی کاوش قوانین انجمنی از روی نمونه‌داده‌های معنایی انجام شده است. از آنجایی که در الگوریتم‌های کشف قوانین انجمنی تراکنش‌ها را داریم که مشاهده‌ای از رخداد مجموعه‌ای از آیتم‌ها می‌باشند، تکنیک‌های ارائه مورد نیاز در رابطه با قوانین انجمنی کاملاً با آنچه در روش‌های خوشه‌بندی و کلاس‌بندی مورد

نیاز است، متفاوت است. بنابراین در زمینه کاوش قوانین انجمنی از داده‌های معنایی کاملاً با حیطه جدیدی که کارهای معدودی را در بر می‌گیرد مواجه هستیم.

حال اگر داده‌های معنایی مورد نظر را جریان‌دار در نظر بگیریم که با توجه به کاربردهای مختلف امروزی بسیار پرمخاطب می‌باشد، مسأله کاوش این داده‌ها و استخراج قوانین انجمنی مفهومی از این داده‌های معنایی جریان‌دار مسأله ای کاملاً جدید بوده و در تحقیقاتی که تا کنون صورت گرفته است هیچ کار مشابهی با این تحقیق یافت نشده است.

ایده اصلی ما در این تحقیق استفاده از فناوری‌های معنایی جدید جهت رفع چالش‌های موجود در کاوش داده‌های معنایی جریان‌دار و ارتقای سیستم از لحاظ انسجام و سطح درک و استنتاج آن در مقابل رویدادهای ورودی به سیستم به کمک تعریف و بهره‌وری از آنتولوژی می‌باشد. برای پیاده کردن این ایده و مقابله با چالش‌های ذکر شده سعی شده است، در هر مرحله و به خصوص در تبدیل داده‌های معنایی جریان‌دار به آیتم‌های معنایی تراکنش‌ها از فناوری‌های معنایی، استفاده شود تا با استفاده از غنای معنایی داده‌ها بتوان به قوانین انجمنی در سطح مفهومی بالاتر جهت بهبود تصمیم‌گیری‌های معنایی دست یافت.

همچنین با یک شیوه تبدیل نوین علاوه بر در نظر داشتن محدودیت‌های زمانی و پیوستگی داده‌ها، بر چالش‌هایی که از پیچیدگی و ناهمگن بودن داده‌ها ناشی می‌گردد نیز غلبه خواهد شد. همچنین الگوریتم کشف الگوهای پرتکرار باید قابلیت کار با آیتم‌ها و تراکنش‌های معنایی را داشته باشند که این مسأله نیز در مقایسه با کارهای مشابه قابلیت جدیدی می‌باشد.

در نتیجه به طور خلاصه می‌توان گفت که کاوش داده‌های معنایی جریان‌دار و کشف قوانین انجمنی از آنها، مقابله با مشکلات مربوط به پیوستگی داده‌های معنایی مورد نظر، رفع چالش‌های مرتبط با پیچیدگی‌ها و ناهمگنی داده‌های معنایی و دستیابی به قوانین انجمنی با سطح مفهومی بالاتر از جنبه‌های نوآوری این تحقیق می‌باشد.

۶-۳- خلاصه فصل

در این فصل پس از مقدمه‌ای کلی در رابطه با ضرورت پرداختن به مساله مطرح در رابطه با روش پیشنهادی، ابتدا به معرفی روش پیشنهادی پرداخته شد. در این معرفی معماری کلی و اجزای سیستم پیشنهادی تشریح گردیده و ابزارها و الگوریتم‌های بکار رفته بیان شدند. سپس مراحل عملیاتی کردن مدل پیشنهادی و پیش‌نیازهای آن مطرح گردید.

در انتها چالش‌های موجود در کارهای مرتبط به طور مختصر گفته شده و نحوه پاسخگویی روش پیشنهادی به آنها بیان گردید و سپس نوآوری‌های روش پیشنهادی مورد بحث قرار گرفت.

فصل ۴. فصل۴- پیاده‌سازی مدل و ارزیابی

۱-۴- مقدمه

جهت ارزیابی راه‌کار معرفی شده در این تحقیق، با توجه به وابستگی نتایج به کاربرد خاص، جدید بودن موضوع و عدم وجود کارهای مشابه مناسب برای مقایسه، لازم است پس از مرحله پیاده‌سازی روش پیشنهادی، داده‌های معنایی جریان‌دار آزمایش، به داده‌های غیرمعنایی تبدیل شده و همان الگوریتم پایه Apriori برای کاوش قوانین انجمنی بدون استفاده از فناوری‌های معنایی از قبیل آنتولوژی، قوانین، استنتاج و پرس‌جوی معنایی پیاده‌سازی گردد. در این پیاده‌سازی هدف ایجاد یک بستر مناسب برای آزمودن کارایی روش ارائه شده در تحقیق با هدف نهایی امکان‌پذیر ساختن پردازش حجم وسیع داده‌های معنایی جریان‌دار موجود و کشف و ذخیره‌سازی قوانین انجمنی جدید با استفاده از مفاهیم موجود در آنتولوژی و قوانین اولیه می‌باشد.

پس از انجام این پیاده‌سازی‌ها می‌توان براساس هدف کلی سیستم، میزان کارایی راه‌کار پیشنهادی را از لحاظ معیارهای مختلف از قبیل سطح مفهومی قوانین، درجه پشتیبانی قوانین، درجه اطمینان قوانین و تعداد قوانین انجمنی استخراج شده در هر اجرا، نشان داد.

همچنین مزایا و برتری‌های راه‌کار پیشنهادی نسبت به کارهای مشابه با ارزیابی کیفی قابل بررسی و ارائه خواهد بود.

انتظار می‌رود با توجه به استفاده از فناوری‌های معنایی از قبیل آنتولوژی، استنتاج و پرس‌جوی معنایی، قوانین انجمنی در سطوح معنایی بالاتری حاصل شوند و بتوانند تصمیم‌گیری هوشمندی که براساس این قوانین انجام می‌گیرد را بهبود بخشند. همچنین از آنجایی که تمام بخش‌های سیستم بر اساس یک آنتولوژی واحد کار می‌کنند، یک انسجام و یکپارچگی معنایی در کل سیستم وجود خواهد داشت. این مزایا احتمالاً هزینه‌هایی از لحاظ زمان و حافظه برای سیستم در بر خواهد داشت که انتظار می‌رود با توجه به برتری‌های حاصله در نتایج قابل چشم‌پوشی باشند.

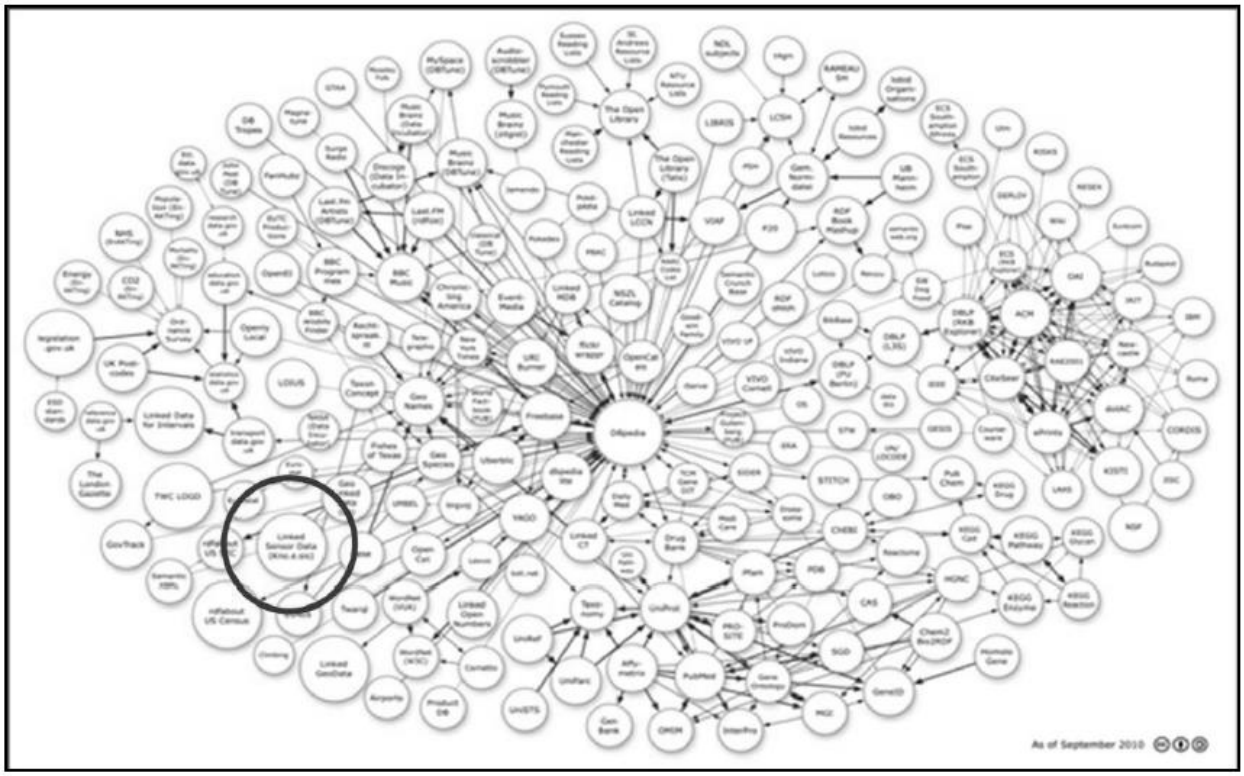
در این فصل به مراحل آماده‌سازی و پیاده‌سازی روش پیشنهادی از جمله آماده‌سازی مجموعه داده‌ها و آنتولوژی‌ها پرداخته می‌شود. سپس ورودی و خروجی ماژول‌های مختلف سیستم نشان داده می‌گردد. در انتها به ارزیابی کیفی و کمی روش پیشنهاد شده پرداخته می‌شود.

۴-۲- آماده‌سازی مجموعه داده

به منظور آماده‌سازی جریان‌های داده معنایی از مجموعه LinkedSensorData منتشر شده در LOD[1] حاوی رویدادهای هواشناسی در زیرمجموعه LinkedObservationData استفاده شده است. این مجموعه داده یک مجموعه داده در قالب RDF/n3 شامل توصیفاتی در رابطه با مشاهدات هواشناسی در ایالات متحده آمریکا می‌باشد. این مشاهدات در نتیجه اندازه‌گیری ویژگی‌هایی از قبیل دما، بارش، فشار هوا، سرعت باد، رطوبت هوا و غیره جمع‌آوری شده است. در حال حاضر این مجموعه داده حاوی بیش از یک بیلیون سه‌تایی می‌باشد [Pat11].

استخراج داده‌های مورد نظر با برنامه‌نویسی معنایی و به کمک کتابخانه jena [Pow03] صورت گرفته است. سپس با فیلترینگ مشخصه‌های مورد نظر رویدادهای هواشناسی جداسازی شده‌اند. مشخصه‌های انتخاب شده عبارتند از: AirTemperature، Precipitation، WindSpeed، WindDirection، RelativeHumidity، WindGust و DewPoint.

سپس در مرحله پیش‌پردازش تبدیل مقادیر عددی به مقادیر کیفی صورت می‌گیرد.



شکل ۴-۱: مجموعه داده Linked Sensor Data در LOD [Pat11]

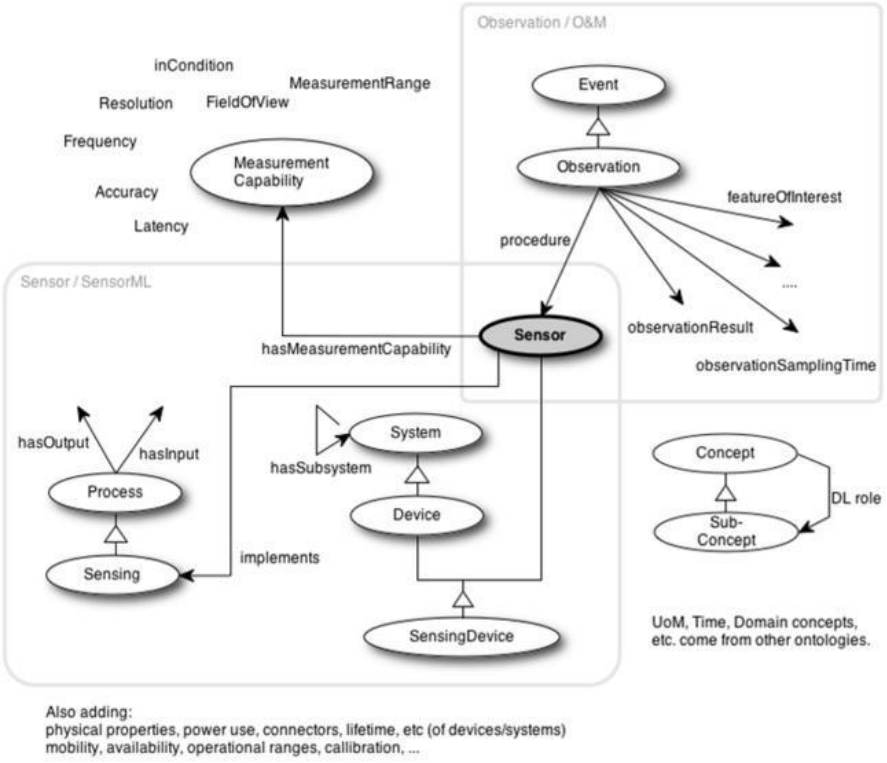
[1] Linked Open Data

۴-۳- آماده‌سازی آنتولوژی‌ها

همان‌طور که در فصل قبل گفته شد، جهت پیاده‌سازی روش پیشنهادی دو نوع آنتولوژی مورد نیاز است. آنتولوژی دسته اول آنتولوژی‌های مرتبط با حوزه کاربردی داده‌های مورد استفاده می‌باشند و آنتولوژی نوع دوم آنتولوژی مفاهیم مرتبط با استخراج قوانین انجمنی (ARM) است که به منظور استفاده در سیستم مورد نظر طراحی و پیاده‌سازی گردیده است. آنتولوژی طراحی شده شامل توصیف مفاهیمی از قبیل تراکنش، آیتم و قانون انجمنی به همراه مشخصه‌های مرتبط با آنها از قبیل درجه پشتیبانی و درجه اطمینان می‌باشد.

آنتولوژی ARM به کمک کتابخانه jena [Pow03] در محیط netbeans پیاده سازی شده است. jena یک کتابخانه جاوای متن‌باز است که برای پردازش داده‌های RDF در وب معنایی و داده‌های پیوندی بکار می‌رود. این کتابخانه علاوه بر امکان ایجاد مدل‌های مختلف آنتولوژی شامل RDFS و OWL، پردازش پرس‌وجوهای SPARQL و استنتاج مبتنی بر قانون را نیز حمایت می‌کند.

به علاوه استفاده از بخش‌هایی از آنتولوژی SSN[1] تعریف شده توسط گروه تحقیقاتی W3C و توسعه‌های آن به عنوان آنتولوژی‌های دامنه کاربردی استفاده شده است.



شکل ۴-۲: آنتولوژی SSN [Pat11]

[1] Semantic Sensor Network

آنتولوژی SSN بر اساس سه مفهوم اصلی سیستم‌ها، فرآیندها و مشاهدات طراحی شده است و توصیفات ساختار فیزیکی و پردازشی حسگرها را نیز پشتیبانی می‌کند. در این کار از مولفه مشاهدات[1]این آنتولوژی استفاده شده است.

۴-۴- ورودی و خروجی سیستم پیشنهادی

همان طور که در فصل قبل گفته شد سیستم پیشنهادی از دو ماژول اصلی تشکیل شده است. بنابراین می‌توان گفت داده‌ها از جریان‌های داده معنایی اولیه تا قوانین انجمنی معنایی نهایی، در سه سطح مورد انتظار می‌باشند.

جریان‌های داده معنایی اولیه داده‌هایی هستند که از ابر داده‌های پیوندی استخراج شده‌اند و در قالب RDF/n3 قرار دارند. نمونه‌ای از این داده‌ها در شکل زیر مشاهده می‌شود.

```
sens-obs:Observation_AirTemperature_KDAY_2011_02_28_10_56_00
    a      weather:TemperatureObservation ;
    om-owl:observedProperty
          weather:_AirTemperature ;
    om-owl:procedure sens-obs:System_KDAY ;
    om-owl:result sens-
obs:MeasureData_AirTemperature_KDAY_2011_02_28_10_56_00 ;
    om-owl:samplingTime sens-obs:Instant_2011_02_28_10_56_00 .

sens-obs:MeasureData_AirTemperature_KDAY_2011_02_28_10_56_00
    a      om-owl:MeasureData ;
    om-owl:floatValue "24.1" ;
    om-owl:uom weather:fahrenheit .

sens-obs:Instant_2011_02_28_10_56_00
    a      owl-time:Instant ;
    owl-time:inXSDDateTime
          "2011-02-28T10:56:00"
```

شکل ۴-۳: نمونه‌ای از جریان‌های داده معنایی اولیه

سپس بخشی از این داده‌ها انتخاب شده و به ماژول اول وارد می‌گردند. در ماژول ایجاد تراکنش‌های معنایی ابتدا سه‌تایی‌های مرتبط با رویدادهای هواشناسی که مشخصه آنها یکی از مشخصه‌های انتخاب شده در مرحله آماده‌سازی می‌باشد فیلتر می‌شوند و همچنین مقادیر عددی موجود در بخش مفعول سه‌تایی‌ها مطابق پیش‌پردازش مشخص شده به مقادیر کیفی تبدیل می‌گردند. پس از آن با حرکت پنجره لغزنده بر روی این سه‌تایی‌های حاصل از پیش‌پردازش، تراکنش‌ها که هر یک حاوی سه‌تایی‌های رخ داده در پنجره لغزنده در یک

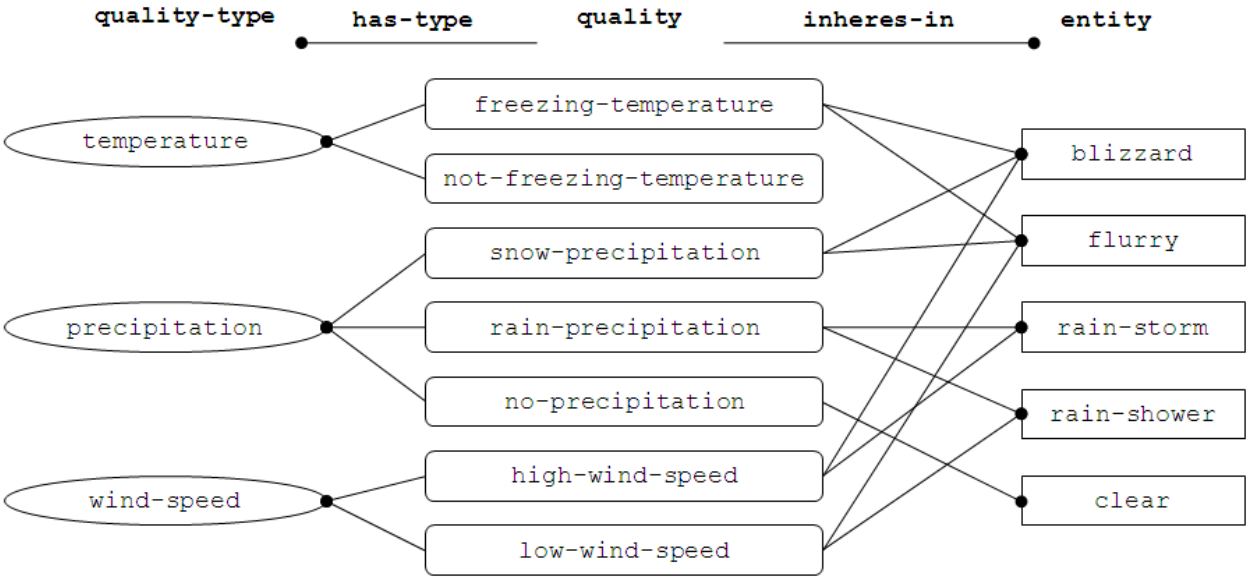
[1] Observation O&M

موقعیت زمانی می‌باشند، ایجاد می‌شوند. سپس تراکنش‌های ایجاد شده در قالب تعریف شده در آنتولوژی ARM ذخیره می‌گردند. نمونه‌ای از تراکنش‌های ایجاد شده در این مرحله در شکل زیر نشان داده شده است.

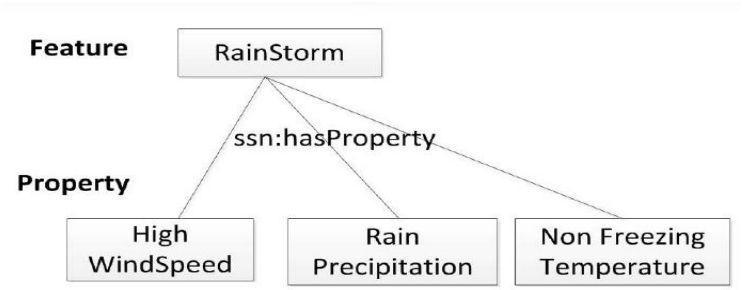
```
<rdf:Description rdf:about="http://wtlab.um.ac.ir/linkeddata/ARM#transaction1">
    <rdf:type rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#transaction"/>
    <j.0:hasItem rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item2"/>
    <j.0:hasItem rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item50"/>
    <j.0:hasItem rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item17"/>
    <j.0:hasItem rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item32"/>
    <j.0:hasItem rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item3"/>
    <j.0:hasItem rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item18"/>
    <j.0:hasItem rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item30"/>
    <j.0:hasItem rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item1"/>
 </rdf:Description>
 <rdf:Description rdf:about="http://wtlab.um.ac.ir/linkeddata/ARM#item1">
    <j.0:hasObject>low</j.0:hasObject>
    <j.0:hasPredicate rdf:resource="http://knoesis.wright.edu/ssw/ont/weather.owl#WindGustObservation"/>
    <j.0:hasSubject rdf:resource="http://knoesis.wright.edu/ssw/System_A01"/>
    <rdf:type rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item"/>
 </rdf:Description>
    <rdf:Description rdf:about="http://wtlab.um.ac.ir/linkeddata/ARM#item2">
    <j.0:hasObject>low</j.0:hasObject>
    <j.0:hasPredicate rdf:resource="http://knoesis.wright.edu/ssw/ont/weather.owl#WindSpeedObservation"/>
    <j.0:hasSubject rdf:resource="http://knoesis.wright.edu/ssw/System_A01"/>
    <rdf:type rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item"/>
 </rdf:Description>
```

شکل ۴-۴: نمونه تراکش‌های معنایی خروجی ماژول ایجاد تراکنش‌ها

همچنین پرس‌وجوهای معنایی طراحی شده بر روی این تراکنش‌ها اعمال شده و تراکنش‌های با سطح مفهومی بالاتر را نتیجه می‌دهد. مفاهیم و روابط موجود در آنتولوژی که پرس‌وجوهای معنایی از آنها نتیجه شده‌اند در شکل‌های ۴-۵ و ۴-۶ مشاهده می‌شوند.



شکل ۴-۵: مفاهیم و روابط موجود در آنتولوژی کاربرد مورد نظر جهت طراحی پرس‌وجوهای معنایی [Pat11]



شکل ۴-۶: نمونه پرس‌وجوی معنایی مرتبط با کاربرد مورد نظر در قالب آنتولوژی SSN [Pat11]

سپس خروجی ماژول اول وارد ماژول استخراج قوانین انجمنی می‌شود. در این ماژول ابتدا روش مبتنی بر الگوریتم Apriori بر روی تراکنش‌های معنایی اعمال می‌گردد و در نهایت قوانین تولید شده در قالب آنتولوژی ARM ذخیره می‌گردند. یک نمونه از قوانین انجمنی معنایی نهایی در شکل ۴-۷ نشان داده شده است.

```
<rdf:Description rdf:about="http://wtlab.um.ac.ir/linkeddata/ARM#rule2133">
  <j.0:hasConfidence>100</j.0:hasConfidence>
  <j.0:hasSupport>99</j.0:hasSupport>
  <j.0:hasDItem>http://wtlab.um.ac.ir/linkeddata/ARM#item100</j.0:hasDItem>
  <j.0:hasAItem>http://wtlab.um.ac.ir/linkeddata/ARM#item67</j.0:hasAItem>
  <j.0:hasAItem>http://wtlab.um.ac.ir/linkeddata/ARM#item125</j.0:hasAItem>
  <rdf:type>http://wtlab.um.ac.ir/linkeddata/ARM#rule</rdf:type>
</rdf:Description>

<rdf:Description rdf:about="http://wtlab.um.ac.ir/linkeddata/ARM#item100">
  <j.0:hasObject>very low</j.0:hasObject>
  <j.0:hasPredicate rdf:resource="http://knoesis.wright.edu/ssw/ont/weather.owl#RelativeHumidityObservation"/>
  <j.0:hasSubject rdf:resource="http://knoesis.wright.edu/ssw/System_A09"/>
<rdf:type rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item"/>

<rdf:Description rdf:about="http://wtlab.um.ac.ir/linkeddata/ARM#item67">
  <j.0:hasObject>hot</j.0:hasObject>
  <j.0:hasPredicate rdf:resource="http://knoesis.wright.edu/ssw/ont/weather.owl#AirTemperatureObservation"/>
  <j.0:hasSubject rdf:resource="http://knoesis.wright.edu/ssw/System_A05"/>
  <rdf:type rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item"/>
</rdf:Description>

  <rdf:Description rdf:about="http://wtlab.um.ac.ir/linkeddata/ARM#item125">
  <j.0:hasObject>low</j.0:hasObject>
  <j.0:hasPredicate rdf:resource="http://knoesis.wright.edu/ssw/ont/weather.owl#WindSpeedObservation"/>
  <j.0:hasSubject rdf:resource="http://knoesis.wright.edu/ssw/System_A12"/>
  <rdf:type rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item"/>
</rdf:Description>
```

شکل ۴-۷: نمونه قوانین انجمنی معنایی خروجی ماژول استخراج قوانین انجمنی

۴-۵- ارزیابی کیفی

در این بخش مدل پیشنهادی بر اساس روش‌های سنجش مدل، در سه زیربخش مورد ارزیابی قرار خواهد گرفت.

۱-۵-۴- ارزیابی مقایسه ای

در مقایسه با کارهای انجام شده در رابطه با استخراج قوانین انجمنی از داده‌های معنایی، روش پیشنهادی، یک ساختار معنایی برای داده‌کاوی ارائه می‌کند که بر مبنای مدل معرفی شده در [Neb12] است و جهت استخراج قوانین انجمنی از الگوریتم Apriori [Agr94] استفاده می‌گردد. در مدل معرفی شده لازم است ابتدا نمونه داده‌های جریان‌دار معنایی از ابر داده‌های پیوندی استخراج گردند، آنتولوژی‌های مورد نیاز جهت ایجاد یکپارچگی و ساخت‌یافتگی مستقر شوند و مراحل بعدی مبتنی بر این آنتولوژی‌ها انجام خواهند شد.

در ادامه با توجه به اینکه مدل پیشنهادی بر پایه روش ارائه شده در [Neb12] می‌باشد، به مقایسه مدل ارائه شده با [Neb12] پرداخته می‌شود. خلاصه این مقایسه در جدول ۴-۱ مشاهده می‌شود.

جدول ۴-۱: مقایسه مدل پیشنهادی با روش ارائه شده در [Neb12]

	روش ارائه شده در [Neb12]	مدل پیشنهادی
قابلیت استخراج دانش جدید از داده‌های معنایی	+	+
عدم ناهمگنی ساختاری	+ با حذف معانی	+ با تعریف آنتولوژی مفاهیم استخراج قوانین انجمنی
قابلیت کار با داده‌های جریان‌دار	-	+ با استفاده از مدل پنجره لغزنده
مفهوم‌سازی حوزه مدنظر	-	+ آنتولوژی کاوش قوانین انجمنی، مفاهیم را به صورت صریح و یکپارچه در می‌آورند.
فرمت استاندارد برای توصیف داده‌ها	-	+ استفاده از سه‌تایی‌های RDF در تمامی مراحل
وجود ابرداده برای تراکنش‌ها و قوانین انجمنی	-	+ با توجه به RDFS هر داده همراه با ابرداده ذخیره می‌شود.

دستیابی به قوانین با سطح مفهومی بالاتر	-	+ با استفاده از مفاهیم و روابط موجود در آنتولوژی
عدم دخالت کاربر	-	+

۴-۵-۲ ارزیابی مبتنی بر ویژگی

امروزه یک روند رو به رشد از تکنولوژی‌های وب معنایی در حال شکل‌گیری می باشد که در زمینه‌های مختلفی از علوم وارد شده است و در این تحقیق معنا به حوزه کاوش قوانین انجمنی اضافه شد که ویژگی‌های مثبتی به همراه دارد.

یکی از ویژگی‌های اصلی مدل، ایجاد سازگاری و همگنی داده‌ها در مراحل مختلف داده‌کاوی می‌باشد که دلیل اصلی آن مفهوم‌سازی حوزه مدنظر است و اینکه تمام مفاهیم مرتبط با استخراج قوانین انجمنی به صورت ساخت یافته در قالب آنتولوژی طراحی و پیاده سازی شده‌اند.

در این مدل ماهیت داده‌ها معنایی می‌باشد و پرس‌وجو بر روی این داده‌ها به کمک زبان پرس‌وجوی SPARQL انجام می‌گیرد. نکته ای که در این جا قابل ذکر می باشد، این است که کارایی این زبان پرس‌وجو برای مواقعی که پایگاه داده معنایی بسیار بزرگ باشد، پایین است و در مقایسه با SQL کندتر عمل می‌کند. اما با استفاده از این فناوری معنایی و با استفاده از مفاهیم و روابط موجود در آنتولوژی کاربرد، قوانینی با سطح معنایی بالاتر قابل دستیابی بوده‌اند.

۴-۵-۳ ارزیابی عملیاتی

با توجه به اینکه در این تحقیق برای اولین بار روشی برای کاوش جریان‌های داده معنایی و کشف قوانین انجمنی معنایی از آنها پیشنهاد می‌شود، بایستی عملیاتی بودن آن را نیز بررسی کرد و نشان داد که این روش در کنار فناوری‌های معنایی پاسخگو است و قابلیت عملیاتی شدن را دارد. از این رو با انتخاب حوزه هواشناسی نشان داده شده است که خروجی‌های مطلوب هر سطح به صورت معنایی، قابل استخراج است و روشی برای انجام هر ماژول با برنامه نویسی معنایی و مبتنی بر آنتولوژی وجود دارد.

۴-۶-ارزیابی کمی

در این بخش به ارزیابی کمی روش پیشنهادی بر اساس معیارهای ارزیابی متداول جهت ارزیابی قوانین انجمنی پرداخته شده است. معیارهای مورد استفاده در این بخش عبارتند از:

- سطح مفهومی قوانین: هر چه قوانین در سطح مفهومی بالاتری باشند، کمک بیشتری به تصمیم‌گیری هوشمند در سیستم استفاده کننده صورت می‌گیرد.
- درجه پشتیبانی قوانین: احتمال رخداد آیتم‌های موجود در قانون در یک موقعیت زمانی

$$Support\ (A->B) = P(A \cup B)$$

- درجه اطمینان قوانین: احتمال رخداد آیتم‌های تالی به شرط رخداد آیتم‌های مقدم

$$Confidence\ (A->B) = P(B \mid A)$$

در سناریوی اول داده‌های انتخابی که ۳۵۲۶۸ سه‌تایی زماندار را تشکیل می‌دهند، رویدادهای هواشناسی رخ داده در ۲۰ ایستگاه حسگری A01 تا A20 در بازه زمانی 2004.08.09 الی 2004.08.14 می‌باشند. اندازه پنجره لغزنده نیز ۶۰ دقیقه و اندازه گام آن ۳۰ دقیقه می‌باشد. در مرحله اول اجرای روش پیشنهادی ۲۷۸ تراکنش معنایی اولیه و ۲۵۰ آیتم منحصر بفرد استخراج شدند. نتیجه اجرای مرحله دوم سیستم پیشنهادی بر روی این داده‌ها با تغییر مقدار حداقل درجه پشتیبانی از ۸۰ تا ۱۰۰ درصد در اجراهای مختلف در جدول ۴-۲ ارائه شده است. در این جدول مشاهده می‌شود که می‌توان با بالا بردن حداقل درجه پشتیبانی به تعداد مناسبی قانون با معیارهای کیفیتی بالا دست یافت.

جدول ۴-۲: نتیجه اجرای روش پیشنهادی روی داده‌های سناریوی اول

حداقل درجه پشتیبانی	حداقل درجه اطمینان	تعداد قوانین انجمنی	میانگین درجه پشتیبانی	میانگین درجه اطمینان
۸۰	۹۰	۶۱۶۵	۸۹,۳۶	۹۸,۱۹
۸۵	۹۰	۳۰۸۳	۹۲,۰۸	۹۸,۱۲
۹۰	۹۰	۱۴۴۰	۹۵,۰۵	۹۸,۲۵
۹۵	۹۰	۳۸۷	۹۷,۰۱	۹۸,۸۵
۹۹	۹۰	۴۸	۹۹,۰۶	۹۹,۷۵
۱۰۰	۹۰	۳	۱۰۰	۱۰۰

در ادامه جهت بررسی سطح مفهومی قوانین انجمنی استخراج شده، پرس‌وجوهای معنایی نیز بر روی تراکنش‌های اولیه اعمال گردیده‌اند تا تراکنش‌های ثانویه با سطح معنایی بالاتر ایجاد گردند. در سناریوی دوم داده‌های انتخابی که ۱۵۷۶۶ سه‌تایی زماندار را تشکیل می‌دهند، رویدادهای هواشناسی رخ داده در ۱۰

ایستگاه حسگری دیگر در بازه زمانی 2004.08.09 الی 2004.08.14 می‌باشند. اندازه پنجره لغزنده نیز همچنان ۶۰ دقیقه و اندازه گام آن ۳۰ دقیقه می‌باشد. در اجرای ماژول اول ۵۷۶ تراکنش معنایی اولیه و ۱۵۵ آیتم منحصر بفرد استخراج شدند. پس از اعمال پرس‌وجوها نیز ۴ آیتم دیگر با سطح معنایی بالاتر به آیتم‌های اولیه افزوده می‌شود. در این اجرا مشاهده می‌شود که علاوه بر قوانین انجمنی در سطح داده‌های اولیه، قوانین انجمنی با سطح مفهومی بالاتر نیز قابل دستیابی می‌باشند.

در انتها جهت بررسی صحت، کامل بودن و عملیاتی بودن روش پیشنهادی، نتایج بدست آمده از اجرای روش پیشنهادی روی بخش کوچکتری از داده‌ها (داده‌های حسگر A02 در تاریخ 2004.08.09) با قوانین محاسبه شده با اعمال نیمه خودکار الگوریتم Apriori بر روی داده‌های یکسان مقایسه می‌گردد. اعمال الگوریتم Apriori بدین صورت بوده که ابتدا با توجه به پیش‌پردازش تعریف شده و پیش‌فرض‌هایی از قبیل اندازه پنجره، تراکنش‌ها استخراج می‌گردند. سپس با توجه به این مطلب که هر تراکنش دارای یک شناسه یکتا می‌باشد، این تراکنش‌ها در قالب ماتریس بولی درآمده و الگوریتم Apriori بر روی آنها اعمال می‌گردد. طبق جدول ۴-۳ مشاهده می‌شود که نتایج کاملاً مشابه می‌باشند.

جدول ۴-۳: مقایسه قوانین انجمنی بدست آمده از روش پیشنهادی با الگوریتم Apriori

قوانین انجمنی بدست آمده از اعمال نیمه خودکار الگوریتم Apriori	قوانین انجمنی بدست آمده از روش پیشنهادی
در صورتی که مقدار مشخصه دمای شبنم اندازه‌گیری شده در حسگر A02 پایین باشد و هوا گرم باشد، رطوبت هوای اندازه‌گیری شده در حسگر A02 بسیار پایین خواهد بود.	<rdf:Description rdf:about="http://wtlab.um.ac.ir/linkeddata/ARM#rule1"> <j.0:hasConfidence>100</j.0:hasConfidence> <j.0:hasSupport>31</j.0:hasSupport> <j.0:hasDItem>http://wtlab.um.ac.ir/linkeddata/ARM#item6</j.0:hasDItem> <j.0:hasAItem>http://wtlab.um.ac.ir/linkeddata/ARM#item4</j.0:hasAItem> <j.0:hasAItem>http://wtlab.um.ac.ir/linkeddata/ARM#item1</j.0:hasAItem> <rdf:type>http://wtlab.um.ac.ir/linkeddata/ARM#rule</rdf:type> </rdf:Description> <rdf:Description rdf:about="http://wtlab.um.ac.ir/linkeddata/ARM#item1"> <j.0:hasObject>low</j.0:hasObject> <j.0:hasPredicate rdf:resource="http://knoesis.wright.edu/ssw/ont/weather.owl#DewPointObservation"/> <j.0:hasSubject rdf:resource="http://knoesis.wright.edu/ssw/System_A02"/> <rdf:type rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item"/> </rdf:Description> <rdf:Description rdf:about="http://wtlab.um.ac.ir/linkeddata/ARM#item4">

	<j.0:hasObject>hot</j.0:hasObject> <j.0:hasPredicate rdf:resource="http://knoesis.wright.edu/ssw/ont/weather.owl#AirTemperatureObservation"/> <j.0:hasSubject rdf:resource="http://knoesis.wright.edu/ssw/System_A02"/> <rdf:type rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item"/> </rdf:Description> <rdf:Description rdf:about="http://wtlab.um.ac.ir/linkeddata/ARM#item6"> <j.0:hasObject>very low</j.0:hasObject> <j.0:hasPredicate rdf:resource="http://knoesis.wright.edu/ssw/ont/weather.owl#RelativeHumidityObservation"/> <j.0:hasSubject rdf:resource="http://knoesis.wright.edu/ssw/System_A02"/> <rdf:type rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item"/> </rdf:Description>
در صورتی که مقدار مشخصه دمای شبنم اندازه‌گیری شده در حسگر A02 پایین باشد و رطوبت هوا بسیار پایین باشد، دمای هوای اندازه‌گیری شده در حسگر A02 بسیار بالا خواهد بود.	<rdf:Description rdf:about="http://wtlab.um.ac.ir/linkeddata/ARM#rule2"> <j.0:hasConfidence>100</j.0:hasConfidence> <j.0:hasSupport>31</j.0:hasSupport> <j.0:hasDItem>http://wtlab.um.ac.ir/linkeddata/ARM#item1</j.0:hasDItem> <j.0:hasAItem>http://wtlab.um.ac.ir/linkeddata/ARM#item6</j.0:hasAItem> <j.0:hasAItem>http://wtlab.um.ac.ir/linkeddata/ARM#item4</j.0:hasAItem> <rdf:type>http://wtlab.um.ac.ir/linkeddata/ARM#rule</rdf:type> </rdf:Description> <rdf:Description rdf:about="http://wtlab.um.ac.ir/linkeddata/ARM#item1"> <j.0:hasObject>low</j.0:hasObject> <j.0:hasPredicate rdf:resource="http://knoesis.wright.edu/ssw/ont/weather.owl#DewPointObservation"/> <j.0:hasSubject rdf:resource="http://knoesis.wright.edu/ssw/System_A02"/> <rdf:type rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item"/> </rdf:Description> <rdf:Description rdf:about="http://wtlab.um.ac.ir/linkeddata/ARM#item4"> <j.0:hasObject>hot</j.0:hasObject> <j.0:hasPredicate rdf:resource="http://knoesis.wright.edu/ssw/ont/weather.owl#AirTemperatureObservation"/> <j.0:hasSubject rdf:resource="http://knoesis.wright.edu/ssw/System_A02"/> <rdf:type rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item"/> </rdf:Description> <rdf:Description rdf:about="http://wtlab.um.ac.ir/linkeddata/ARM#item6">

	<j.0:hasObject>very low</j.0:hasObject> <j.0:hasPredicate rdf:resource="http://knoesis.wright.edu/ssw/ont/weather.owl#RelativeHumidityObservation"/> <j.0:hasSubject rdf:resource="http://knoesis.wright.edu/ssw/System_A02"/> <rdf:type rdf:resource="http://wtlab.um.ac.ir/linkeddata/ARM#item"/> </rdf:Description>

۷-۴- خلاصه فصل

در این فصل به شرح پیاده‌سازی مدل پیشنهادی در حوزه داده‌های هواشناسی پرداخته شد و با اجراهای مختلف سیستم بر روی بخش‌های مختلف داده‌های معنایی جریان‌دار منتشر شده در این حوزه کاربردی، عملیاتی شدن مدل با برنامه نویسی معنایی نشان داده شد.

پس از توصیف مراحل آماده‌سازی از قبیل آماده‌سازی مجموعه داده‌ها و آماده سازی آنتولوژی کاربرد و آنتولوژی مفاهیم کاوش قوانین انجمنی و همچنین ارائه ورودی و خروجی‌های سیستم پیشنهادی، در نهایت به ارزیابی کیفی و کمی روش پیشنهادی با توجه به معیارهای استاندارد معرفی شده پرداخته شد.

فصل ۵. فصل۵- نتیجه گیری و کارهای آینده

۵-۱- مقدمه

روز به روز در کاربردهای بیشتری از قبیل شبکه‌های حسگر، کاربردهای نظامی، پردازش داده‌های آنلاین و غیره، تولید هر روزه حجم عظیمی از داده‌های جریان‌دار صورت می‌گیرد. کشف دانش بهینه از این جریان‌های داده، یک حوزه تحقیقاتی فعال در داده‌کاوی با کاربردهای متفاوت بوجود آورده است. برخلاف داده‌های ایستای موجود در پایگاه داده‌های سنتی، داده‌های جریان‌دار اغلب به صورت پیوسته با سرعت بالا و با حجم زیاد و همچنین با توزیع داده متغیر دریافت می‌شود. از طرفی مقادیر آنتولوژی‌ها و حاشیه‌نویسی‌های معنایی موجود بر روی داده‌ها نیز به طور پیوسته در حال افزایش می‌باشد. این نوع داده‌های پیچیده و ناهمگن نیز چالش‌های جدیدی را در حوزه تحقیقاتی داده‌کاوی بوجود آورده است. بنابراین در این تحقیق به رفع چالش‌های ذکر شده و امکان‌پذیر ساختن پردازش حجم وسیع داده‌های معنایی جریان‌دار موجود و کشف و ذخیره‌سازی قوانین انجمنی جدید در سطح معنایی بالاتر با استفاده از غنای معنایی مفاهیم موجود در آنتولوژی و بهره‌وری از دیگر فناوری‌های معنایی پرداخته شده است.

۵-۲- نتیجه‌گیری

در سال‌های اخیر توجه بیشتری به ترکیب دو زمینه تحقیقاتی وب معنایی و داده کاوی صورت گرفته است. با کمک استانداردسازی زبان‌های آنتولوژی RDF(/S) و OWL ، وب معنایی توسعه بیشتری یافته و حجم منابع داده‌ای که شامل حاشیه‌نویسی‌های معنایی هستند، دارای روندی رو به رشد می‌باشد. این ترکیب در اغلب کارهای انجام شده، در قالب یادگیری آنتولوژی و کاوش وب بوده است. اما کار کمی در زمینه کاوش و استخراج دانش از خود داده‌های معنایی صورت گرفته است. کاوش داده‌های معنایی فواید بسیاری را برای جوامع تحقیقاتی خاص دامنه‌ای که داده‌های آنها اغلب پیچیده و ناهمگن است و حجم زیادی دانش در قالب آنتولوژی و حاشیه‌نویسی‌های معنایی موجود است، به بار خواهد داشت.

از طرف دیگر روز به روز در کاربردهای بیشتری شاهد حجم عظیمی از داده‌های جریان‌دار می‌باشیم. کشف دانش بهینه از این جریان‌های داده، یک حوزه تحقیقاتی فعال در داده‌کاوی با کاربردهای متفاوت بوجود آورده است. برخلاف داده‌های ایستای موجود در پایگاه داده‌های سنتی، داده‌های جریان‌دار اغلب به صورت پیوسته با

سرعت بالا و با حجم زیاد و همچنین با توزیع داده متغیر دریافت می‌شوند. این ویژگی‌های داده‌های جریان‌دار مسائل جدیدی را بوجود می‌آورد که لازم است در توسعه تکنیک‌های کاوش قوانین انجمنی از داده‌های جریان‌دار مورد توجه قرار گیرند. به دلیل این ویژگی‌های منحصر به فرد داده‌های جریان‌دار، تکنیک‌های داده‌کاوی سنتی که نیاز به چندین بار مرور کل مجموعه داده دارند، نمی‌توانند به طور مستقیم برای کاوش داده‌های جریان‌دار مورد استفاده قرار گیرند زیرا این کاوش معمولاً باید با یک مرور و زمان پاسخ سریع صورت گیرد.

بنابراین می‌توان گفت مساله اصلی که این تحقیق مطرح می‌کند، پردازش و کاوش داده‌های جریان‌دار معنایی که حجم وسیعی از آنها در کاربردهای مختلف موجود است و استخراج قوانین انجمنی از آنها به گونه‌ای است که بتوان با پیش‌بینی برخی روابط به تصمیم‌گیری نهایی در سیستم‌های هوشمند کمک کرد. اهمیت این مساله از حجم عظیم این گونه داده‌ها در کاربردهای گوناگون، حساسیت این داده‌ها به زمان و استفاده از مفاهیم مستتر در داده‌های معنایی با وجود مشکلاتی که ناهمگنی این داده‌ها به آنها منجر می‌شود، ناشی می‌گردد. علاوه بر این با استفاده از غنای معنایی داده‌ها توانایی دست‌یابی به قوانین انجمنی مفهومی‌تری فراهم آمده است.

۳-۵- کارهای آینده

به عنوان کارهای آینده برای بهبود روش پیشنهادی، می‌توان ایده‌های زیر را مورد بررسی قرار داد:

۱. اعمال تمامی روابط موجود در آنتولوژی بر روی تراکنش‌های معنایی و استخراج قوانین سطح بالای جامع‌تر.
۲. بهینه‌سازی الگوریتم کاوش از لحاظ زمان و حافظه مصرفی جهت بالا بردن قابلیت توسعه و توزیع‌پذیری.
۳. استفاده از مدل‌های پیچیده‌تر کاوش جریان‌های داده به منظور کاوش جریان‌های داده معنایی به صورت آنلاین.
۴. توسعه واسط‌های یکپارچه برای مجموعه داده‌های معنایی منتشرشده جریان‌دار و همچنین غیرجریان‌دار به منظور کاوش آسان‌تر.
۵. توسعه روشی که با تعیین دامنه کاربردی خاص توسط کاربر، به طور خودکار پیمایش منابع داده معنایی و استخراج داده‌های مورد نظر را به انجام رساند.
۶. تعیین معیارهایی که روش پیشنهادی بر اساس آن بتواند قوانین مفید را از قوانین غیر مفید جدا نماید.

منابع

[Ant08] G. Antoniou, F. V. Harmelen, “Semantic web primer”, second edition, the MIT press, 2008.

[Agr94] R. Agrawal, R. Srikant, “Fast Algorithms for Mining Association Rules”, in Proceedings of the 20th Int. Conf. Very Large Data Bases, pp. 487-499, 1994.

[Agr95] R. Agrawal, R. Srikant, "Mining sequential patterns", in: Yu P, Chen A (Eds) Proceedings of the eleventh international conference on data engineering, Taipei, Taiwan, pp. 3-14, 1995.

[Baa03] F. Baader, D. Calvanese, D.L. McGuinness, D. Nardi, P.F. Patel-Schneider (Eds.), "The Description Logic Handbook: Theory, Implementation, and Applications", Cambridge University Press, 2003.

[Bab12] Y. J. Babu, G. J. Phani Bala, S. R. Krishna, “Extracting Spatial Association Rules From The Maximum Frequent Itemsets Based On Boolean Matrix”, International Journal Of Engineering Science & Advanced Technology, Vol. 2, Issue - 1, 79 – 84, ISSN: 2250–3676, Jan-Feb 2012.

[Bar10] D.F. Barbieri, D. Braga, S. Ceri, E.D. Valle, M. Grossniklaus, “Querying rdf streams with c-sparql”, SIGMOD Record 39, 2010.

[Bar10] D. F. Barbieri, D. Braga, S. Ceri, E. D. Valle, M. Grossniklaus, “Incremental Reasoning on Streams and Rich Background Knowledge”, in 7th Extended Semantic Web Conference (ESWC), 2010.

[Bec10] K. Becker, M. Vanzin, "O3R: Ontology-based mechanism for a human-centered environment targeted at the analysis of navigation patterns", Know.-Based Syst. 23, pp. 455–470, 2010.

[Ben12] N. G.-P. J. M. Benitez, F.Herrera, "Special issue on "New Trends in Data Mining" NTDM," Knowledge-Based Systems, pp. 1-2, 2012.

[Blo07] S. Bloehdorn, Y. Sure, "Kernel methods for mining instance data in ontologies", in: ISWC/ASWC, LNCS, Vol. 4825, pp. 58–71, 2007.

[Bod03] F. Bodon, “A Fast Apriori Implementation”, In B. Goethals and M. J. Zaki, editors, Proceedings of the IEEE ICDM Workshop on Frequent Itemset Mining Implementations, Vol. 90 of CEUR Workshop Proceedings, 2003.

[Bri04] D. Brickley, R.V. Guha, “RDF Vocabulary Description Language 1.0: RDF Schema”, W3C Recommendation, 2004.

[Bui05] P. Buitelaar, P. Cimiano, B. Magnini (Eds.), "Ontology Learning from Text: Methods, Evaluation and Applications", Frontiers in Artificial Intelligence and Applications, Vol. 123, IOS Press, Amsterdam, 2005.

[Cha03] J. H. Chang, W. S. Lee, "Finding recent frequent item sets adaptively over online data streams", in: Getoor L, Senator T, Domingos P, Faloutsos C (eds) Proceedings of the Ninth ACM SIGKDD international conference on knowledge discovery and data mining, Washington DC, pp 487–492 , 2003.

[Cha04] J. H. Chang, W. S. Lee, "A sliding window method for finding recently frequent item sets over online data streams". J INF Sci Eng, Vol. 20, No. 4, pp.753–762, 2004.

[Chi04] Y. Chi, H. Wang, P. Yu, R. Muntz, "Moment: maintaining closed frequent item sets over a stream sliding window", in: Proc. of the 4th IEEE international conference on data mining, Brighton, UK, pp. 59–66, 2004.

[Chi05] Y. Chi, R.R. Muntz, S. Nijssen, J.N. Kok, "Frequent subtree mining – an overview, Fundam". Inform., Vol. 66 No. 1–2, pp. 161–198, 2005.

[Chu08] C. J. Chu, V. S. Tseng, T. Liang, “An efficient algorithm for mining temporal high utility itemsets from data streams”, Journal of System Software, Vol. 81, No. 7, pp. 1105-1117, 2008.

[Don03] L. Dong, C. Tjortjis, “Experiences of Using a Quantitative Approach for Mining Association Rules”, in Lecture Notes Computer Science, Vol. 2690, pp. 693-700, 2003.

[Erw07] A. Erwin, R. P. Gopalan, N. R. Achuthan, “CTU-Mine: An Efficient High Utility Itemset Mining Algorithm Using the Pattern Growth Approach”, IEEE 7th International Conferences on Computer and Information Technology, pp. 71-76, 2007.

[Fan09] N. Fanizzi, C. d’Amato, F. Esposito, "Metric-based stochastic conceptual clustering for ontologies", in Inf. Syst., Vol. 34, No. 8, pp. 792–806, 2009.

[Gar08] A. García, R. Berlanga, R. Dánger, "A description clustering data mining technique for heterogeneous data", in Commun. Comput. Inform. Sci., Softw. Data Technol., Vol. 10, pp. 361–373, 2008.

[Gia04] C. Giannella, J. Han, J. Pei, X. Yan, P.S. Yu, “Mining Frequent Patterns in Data Streams at Multiple Time Granularities”, H. Kargupta, A. Joshi, K. Sivakumar, Y. Yesha (eds.), Next Generation Data Mining, 2004.

[Gia10] P. Giannikopoulos, I. Varlamis, M. Eirinaki, "Mining frequent generalized patterns for web personalization in the presence of taxonomies", in IJDWM, Vol. 6, No. 1, pp. 58–76, 2010.

[Gos05] J. Z. Gosta Grahne, "Fast Algorithms for Frequent Itemset Mining Using FP-Trees," Knowledge and Data Engineering, IEEE Transactions on, Vol. 17, pp. 1347- 1362, 2005.

[Gou01] K. Gouda, M. Zaki, "Efficiently mining maximal frequent item sets", in Cercone N, Lin TY, Wu X (eds) Proceedings of the 2001 IEEE international conference on data mining, San Jose, pp. 163–170, 2001.

[Gru93] T. R. Gruber, "A translation approach to portable ontology specifications", in Knowledge acquisition, Vol. 5, No. 2, pp. 199-220, 1993.

[Gru94] T. R. Gruber, G. R. Olsen, "An Ontology for Engineering Mathematics", in International Conference on Principles of Knowledge representation and Reasoning (KR), pp. 258–269, 1994.

[Guh01] S. Guha, N. Koudas, K. Shim, "Data Streams and Histograms", in ACM Symposium on Theory of Computing, p. 471-475, 2001.

[Gua98] N. Guarino, "Formal Ontology and Information Systems. In Proceedings of the 1st International Conference on Formal Ontologies in Information Systems (FOIS 1998), pp.3–15, 1998.

[Gut07] C. Gutierrez, C. A. Hurtado, A. Vaisman, "Introducing Time into RDF", in IEEE Trans. on Knowl. and Data Eng., Vol. 19, No. 2, pp. 207–218, 2007.

[Hor04] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, M. Dean, "SWRL: A Semantic Web Rule Language Combining OWL and RuleML", W3C Rec, 2004.

[Jai07] J. Han, M. Kamber, "Data Mining Concepts and Techniques", Second Edition , Morgan Kaufmann Publishers, 2007.

[Jia06] N. Jiang, L. Gruenwald, "Research Issues in Data Stream Association Rule Mining", SIGMOD Record, Vol. 35, No. 1, pp. 14-19, 2006.

[Jin05] R. Jin, G. Agrawal, "An Algorithm for in-core frequent itemset mining on streaming data", in Proceedings of the 5th IEEE International Conference on data mining, Houston, TX, USA, pp 210-217, 2005.

[Kie08] C. Kiefer, A. Bernstein, A. Locher, "Adding data mining support to SPARQL via statistical relational learning methods", in S. Bechhofer, M. Hauswirth, J. Hoffmann, M. Koubarakis (Eds.), ESWC, Lecture Notes in Computer Science, Vol. 5021, Springer, pp. 478–492, 2008.

[Koc07] K. Kochut, M. Janik, "SPARQLeR: extended SPARQL for semantic association discovery", in ESWC, LNCS, Vol. 4519, Springer, pp. 145–159, 2007.

[Kur01] M. Kuramochi, G. Karypis, "Frequent subgraph discovery", in N. Cercone, T.Y. Lin, X. Wu (Eds.), ICDM, IEEE Computer Society, pp. 313–320, 2001.

[Lee05] D. Lee, W. Lee, "Finding maximal frequent item sets over online data streams adaptively", in Proceedings of the 5th IEEE international conference on data mining, Houston, Texas, USA, pp. 266–273, 2005.

[Li04] H. F. Li, S. Y. Lee, M. K. Shan, "An Efficient Algorithm for Mining Frequent Itemsets over the Entire History of Data Streams", in Proceedings of the 1st Int'l. Workshop on Knowledge Discovery in Data Streams, pp. 20- 24, 2004.

[Li05] Y. C. Li, J. S. Yeh, C. C. Chang, "Efficient Algorithms for Mining Share-Frequent Itemsets", in Proceedings of the 11th World Congress of Intl. Fuzzy Systems Association, 2005.

[Li09] H. F. Li, S. Y. Lee, "Mining frequent item sets over data streams using efficient window sliding techniques" in Expert Systems with Applications, Vol. 36(2P1), pp. 1466–1477. 2009.

[Lis05] A. Lisi, F. Esposito, "Mining the semantic web: a logic-based methodology", in ISMIS, LNCS, Vol. 3488, Springer, pp. 102–111, 2005.

[Liu05] Y. Liu, W. K. Liao, A. Choudhary, "A Fast High Utility Itemsets Mining Algorithm", in Proc. UBDM'05, Chicago Illinois, 2005.

[Mal11] A. Mala, F. Ramesh Dhanaseelan, "DATA STREAM MINING ALGORITHMS – A REVIEW OF ISSUES AND EXISTING APPROACHES", in International Journal on Computer Science and Engineering (IJCSE), 2011.

[Man02] G. S. Manku, R. Motwani, "Approximate frequency counts over data streams" in Proceedings of the 28th international conference on very large data bases, Hong Kong, pp. 346–357, 2002.

[Man04] F. Manola, E. Miller, "RDF Primer", W3C Recommendation, 2004.

[Mao07] G. Mao, X. Wu, X. Zhu, G. Chen, "Mining Maximal Frequent Item sets from Data Streams", in J. Information Science, Vol. 33, No. 3, pp. 251-262, 2007.

[Mar10] C. Marinica, F. Guillet, "Knowledge-based interactive postmining of association rules using ontologies", in IEEE Trans. Knowledge Data Eng., Vol. 22, No. 6, pp. 784–797, 2010.

[Nap08] T. Näppilä, K. Järvelin, T. Niemi, A tool for data cube construction from structurally heterogeneous xml documents, in JASIST, Vol. 59, No. 3, pp. 435–449, 2008.

[Nar11] R. I. V.Narasimha, P. Kappara, R. Ichise, O.P.Vyas, "LiDDM: A Data Mining System for Linked Data", in Proceedings of the LDOW2011, Hyderabad, India, 2011.

[Neb10] V. Nebot, R. Berlanga, "Mining association rules from semantic web data", in Proceedings of IEA/AIE, pp. 0–10, 2010.

[Neb12] V. Nebot, R. Berlanga, "Finding association rules in semantic web data", in Knowledge-Based Systems, Vol. 25, pp. 51–62, 2012.

[Nie09] T. Niemi, T. Näppilä, K. Järvelin," A relational data harmonization approach to xml", in J. Inf. Sci., Vol. 35, No. 5, pp. 571–601, 2009.

[Par02] B. Park, H. Kargupta, "Distributed Data Mining: Algorithms, Systems, and Applications", in the Data Mining Handbook. Editor: N. Ye, 2002.

[Pat11] H. K. Patni, "REAL TIME SEMANTIC ANALYSIS OF STREAMING SENSOR DATA", A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science, Wright State University, 2011.

[Pas99] N. Pasquier, Y. Bastide, R. Taouil, L. Lakhal, "Discovering frequent closed item sets for association rules", in Beeri C, Buneman P (eds.) Proceedings of the 7th international conference on database theory, Jerusalem, Israel,,pp. 398–416, 1999.

[Pei07] J. Pei, J. Han, H. Lu, S. Nishio, S. Tang, D. Yang, "HMine: Fast and space-preserving frequent pattern mining in large databases", in IIE Transactions, Vol. 39; No. 6, pp. 593-605, 2007.

[Pin12] T. Pinniboyina, N. Dhulipala, R. R. Deevi, S. Nathani, "Mining Items From Large Database Using Coherent Rules", in International Journal Of Engineering Science & Advanced Technology, Vol. 2, Special Issue - 1, pp. 61–70, 2012.

[Pod07] V. Podpecan, N. Lavrac, I. Kononenko, "A Fast Algorithm for Mining Utility-Frequent Itemsets", in International Workshop on Constraint-based Mining and Learning at ECML/PKDD, 2007.

[Pow03] S. Powers, "jena: RDF in java ", Chapter 8 in book of Practical RDF, 1st edition (2003).

[Pug08] A. Pugliese, O. Udrea, V. S. Subrahmanian, "Scaling RDF with Time", in Proceedings of 17th WWW Conference, New York, USA, pp. 605–614, 2008.

[Raj12] R. Rajesh, J. Nidhi, "A Survey on Frequent ItemSet Mining Over Data Stream", in International Journal of Electronics Communication and Computer Engineering, Vol. 4, pp. 2278–4209, 2012.

[Ram12] R. Ramezani, "Finding Association Rules in Linked Data", A Thesis submitted in partial fulfilment of the requirements for the degree of Master of Science, 2012.

[Sha09] S. Shankar, T. Purusothaman, "Utility Sentient Frequent Itemset Mining and Association Rule Mining: A Literature Survey and Comparative Study", in International Journal of Soft Computing Applications, Issue 4 , pp. 81-95, 2009.

[Sin02] G. Singh Manku, R. Motwani, "Approximate Frequency Counts over Data Streams", in Int'l Conf. on Very Large Databases, pp. 346-357, 2002.

[Son06] M. Song, S. Rajasekaran, "A Transaction Mapping Algorithm for Frequent Itemsets Mining", in IEEE Transactions on Knowledge and Data Engineering, Vol. 18, No.4, pp. 472-481, 2006.

[Sul06] M. Sulaiman Khan, M. Muyeba, C. Tjortjis, F. Coenen, "An effective Fuzzy Healthy Association Rule Mining Algorithm (FHARM)", in Lecture Notes Computer Science, Vol. 4224, pp.1014-1022, 2006.

[Sul08] M. Sulaiman Khan, M. Muyeba, F. Coenen, "Fuzzy Weighted Association Rule Mining with Weighted Support and Confidence Framework", in ALSIP (PAKDD), pp. 52-64, 2008.

[Stu06] G. Stumme, A. Hotho, B. Berendt, "Semantic web mining: state of the art and future directions", in Web Semantics: Sci. Services Agents World Wide Web, Vol. 4, No. 2, pp. 124–143, 2006.

[Sur12] S. Suriya, S. P. Shantharajah, R. Deepalakshmi, "A Complete Survey on Association Rule Mining with Relevance to Different Domain", in INTERNATIONAL JOURNAL OF ADVANCED SCIENTIFIC RESEARCH AND TECHNOLOGY, Issue 2, Vol. 1, 2012.

[Tag10] A. Tagarelli, S. Greco, "Semantic clustering of xml documents", in ACM Trans. Inf. Syst., Vol. 28, No. 1, 2010.

[Tak07] Y. Takama, S. Hattori, "Mining Association Rules for Adaptive Search Engine Based on RDF Technology", in IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, Vol. 54, No. 2, pp 790, 2007.

[Tan02] S. Tanner, M. Alshayeb, E. Criswell, M. Iyer, A. McDowell, M. McEniry, K. Regner, "EVE: On-Board Process Planning and Execution", in Earth Science Technology Conference, Pasadena, CA, pp. 11-14, 2002.

[Val09] E. D. Valle, S. Ceri, F. V. Harmelen, D. Fensel, "It's a Streaming World! Reasoning upon Rapidly Changing Information", in IEEE Intelligent Systems, Vol. 24, No. 6, pp. 83-89, 2009.

[Viv10] V. T. Vivek Tiwari, S.Gupta, R.Tiwari, "Association Rule Mining: A Graph Based Approach for Mining Frequent Itemsets", in International Conference on Networking and Information Technology (ICNIT), 2010.

[Wan00] W. Wang, J. Yang, P. S. Yu, "Efficient Mining of Weighted Association Rules (WAR)", in Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 270 - 274, 2000.

[Wan04] C. Wang, C. Tjortjis, "PRICES: An efficient algorithm for mining association rules", in Lecture Notes Computer Science Vol. 3177, pp. 352-358, 2004.

[Wan09] C. Wang, W. Sun, T. Zhang, Y. Zhang, " Research on Transaction item association matrix mining algorithm in large scale transaction database", in IEEE sixth International Conference on FUZZY Systems and Knowledge Discovery, 2009.

[Xu05] Y. Xu, Y. Papakonstantinou, "Efficient keyword search for smallest LCAs in XML databases", in SIGMOD '05: Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data, ACM, New York, NY, USA, pp. 527– 538, 2005.

[Ye06] Y. Ye, C. C. Chiang, "A Parallel Apriori Algorithm for Frequent Itemsets Mining", in Fourth International Conference on Software Engineering Research, Management and Applications, pp. 87- 94, 2006.

[Yu04] J. Yu, Z. Chong, H. Lu, A. Zhou, "False positive or false negative: mining frequent item sets from high speed transactional data streams", in Nascimento et al. (Eds) Proceedings of the thirtieth international conference on very large data bases, Toronto, Canada, pp 204–215, 2004.

[Yu08] G. Yu, K. Li, S. Shao, "Mining High Utility Itemsets in Large High Dimensional Data", in International Workshop on Knowledge Discovery and Data Mining WKDD, pp. 17-20, 2008.

[Zha09] H. W. J. Zhang, Y. Sun, "Discovering Associations among Semantic Links", in International Conference on Web Information Systems and Mining (WISM), 2009.

[Zhu09] Q. X. Zhu, L. J. Guo, J. Liu, N. Xu, W. X. Li, "Research of Tax Inspection cases – Choice based on Association Rules in Data Mning", in Proceedings of the Eighth International IEEE conference on Machine Learning and Cybernetics, Baoding, 2009.

[Zia11] F. N. Ziawasch Abedjan, "Context and Target Configurations for Mining RDF Data", in the SMER '11 Proceedings of the 1st international workshop on Search and mining entity-relationship data, 2011.

واژه‌نامه

ابرداده	Metadata
استنتاج جریان‌دار	Stream reasoning
برچسب زمانی	Time stamp
برنامه نویسی معنایی	Semantic programming
پرس و جو	Query
پیش پردازش	Pre-processing
حاشیه نویسی	annotation
حدآستانه	threshold
داده جریان‌دار	Stream data
گزاره	predicate
ناهمگن	heterogeneous

Engineering Faculty
Department of Computer Engineering
Web Technology Lab

Association Rule Mining from Semantic Data Streams

A Thesis
Submitted in partial fulfillment of the requirements
for the degree of Master of Science

By:
Ashraf Sadat Heydari Yazdi

Supervisor:
Prof. Mohsen Kahani

September 2013

Abstract:

Day by day we face more and more applications such as traffic modelling, tracking sensor networks in military applications, online data processing, etc which produce large amounts of data streams every day. Unlike the static data which exists in traditional databases, data stream is often received with high speed, high volume and variable data distribution. Useful knowledge discovery from these kinds of data streams is an active research field among data mining applications. On the other hand, the amount of ontologies and semantic annotations for these data streams are constantly growing. This type of complex and heterogeneous semantic data stream has created new challenges in the area of data mining research. Therefore, in this research a new method is proposed to provide a way to address these challenges and enable processing large volumes of semantic data streams, perform association rule discovery and store these new higher-level semantic rules using semantic richness of the concepts of the defined ontology and other semantic technologies.

Keywords: Semantic Data Stream, Association Rule Mining, Ontology, Sliding Window.