



سیستم های داده کاوی توزیع شده چند عامله

فاطمه بهرامیان^۱، مریم خادمی^۲

۱- دانشجوی کارشناسی ارشد مهندسی کامپیوتر نرم افزار

۲- استادیار گروه ریاضی کاربردی

چکیده

تکنیک داده کاوی توزیع شده؛ یکی از شاخه های علمی جدید و رو به توسعه داده کاوی، دارای پتانسیل بسیار بالای برای پایه ریزی زیربنایی، تکنیکی و عملی است که براساس نیاز به کاوش در منابع غیر متتمرکز داده ای به وجود آمده است. همچنین می توان سیستم های چند عامله را انتخاب مناسبی برای حل مسائل توزیع شده دانست زیرا این سیستم ها اغلب با کاربردهای پیچیده مواجه هستند. در این مقاله به بررسی و مرور جدید ترین تحقیقات در خصوص یکپارچگی سیستم های چند عامله و سیستم های داده کاوی توزیع شده می پردازیم و سپس شاخص هایی نظیر میزان اهمیت، ساختار سیستم های موجود و روند تحقیقات در این زمینه را مورد بررسی قرار خواهیم داد.

کلید واژه: داده کاوی توزیع شده ، سیستم های چند عامله، داده کاوی چند عامله، KDD

A Survey on Multi Agent-based Distributed Data Mining

Fatemeh Bahramian, Maryam Khademi

ABSTRACT

Distributed Data Mining techniques is one of the new and promising field which has a great potential for groundbreaking work from foundational, technological and practical perspectives. It is originated from the need of mining over decentralized data sources. Totally, Multi-agent systems are an appropriate option for solving distributed problem because they often cope with complex applications.

^۱ دانشجوی کارشناسی ارشد کامپیوتر دانشگاه آزاد اسلامی واحد تهران جنوب، ۰۹۱۲۷۱۸۰۳۳۲

^۲ استادیار گروه ریاضی کاربردی دانشگاه آزاد اسلامی واحد تهران جنوب، ۰۹۱۲۲۱۹۲۵۵۴

In this article we survey the last current research related to the integration of multi agent system and distributed data mining based on significance, system overview, existing systems, and research trends.

KEYWORDS: Distributed data mining, Multi-Agent Systems, Multi Agent Data Mining, KDD

۱ - مقدمه

در دهه اخیر تکنولوژی داده‌کاوی برای یافتن قوانین و الگوهای از میان حجم زیاد داده از اهمیت زیادی برخوردار شده است. در این تکنیک معمولاً همه داده‌ها در یک سایت مرکزی به نام انبار داده جمع‌آوری شده و سپس الگوریتمی بر روی داده‌ها جهت پیش‌بینی و ارزیابی داشن، با استفاده از روش‌های یکپارچه‌سازی داده، اجرا می‌گردد. تکنیک‌های داده‌کاوی در محیط‌های پیچیده می‌باشد به صورت پویا و متغیر عمل کنند زیرا تغییرات موجود در این سیستم‌ها می‌تواند بر کارایی کلی آن تأثیرگذار باشد، لذا استفاده از یک تکنیک داده‌کاوی به تنها، برای هر دامنه و مجموعه داده‌ای مناسب نیست.

داده‌کاوی توزیع شده براساس نیاز به کاوش در منابع غیر مرکز داده‌ای ایجاد شده است. از آنجا که هدف از محاسبات عامل، به کارگیری در سیستم‌های پیچیده است، بنابراین با استفاده از آن‌ها می‌توان روش‌های مناسبی جهت ارتقاء سیستم‌های داده‌کاوی توزیع شده یافت. به طور کلی می‌توان گفت سیستم‌های چند عامله چون اغلب با کاربردهای پیچیده مواجه هستند، انتخاب مناسبی برای حل مسائل توزیع شده می‌باشد.

داده‌کاوی توزیع شده^۱ که منشأ آن کشف دانش از پایگاه داده^۲ یا همان داده‌کاوی^۳ است، منابع داده را صرف نظر از مکان فیزیکی آن‌ها مورد کاوش قرار می‌دهد. اهمیت این موضوع در این است که داده‌هایی که به طور محلی در هر سایت قرار گرفته‌اند به دلیل حجم بالا و مسائل امنیتی، معمولاً از طریق شبکه انتقال داده نمی‌شوند. اخیراً، تکنیک داده‌کاوی توزیع شده جزء بسیار مهمی از سیستم‌های مبتنی بر دانش شده است زیرا معماری غیر مرکز آن امکان دسترسی به هر شبکه تجاری را میسر می‌سازد (۱).

داده‌کاوی توزیع شده می‌باشد در هر سایت به صورت منفرد، تحلیل جزئی روی داده‌ها انجام دهد سپس نتایج تحلیل جزئی را به سایت‌های دیگر ارسال نماید؛ همچنین، در صورت لزوم می‌تواند نتایج سایت‌های دیگر را جمع‌آوری کرده و به یک نتیجه کلی برسد. در حقیقت برخی از راه حل‌های توزیع شده براساس تکنیک‌هایی مانند: قوانین انجمنی توزیع شده، خوشه‌بندی توزیع شده، یادگیری بیزین^۴، دسته‌بندی^۵ (رگرسیون^۶) و فشرده‌سازی شکل گرفته‌اند اما فقط تعداد کمی از آن‌ها قادر به استفاده از عوامل هوشمند هستند (۲).

علیرغم این‌که متخصصین تکنیک‌های توزیع شده گام مهمی در جهت ایجاد امنیت همانند داده‌کاوی مرکز برداشته‌اند ولیکن تحقیقات در این زمینه هنوز فاصله زیادی با حل مشکل امنیت در داده‌کاوی توزیع شده دارد (۲).

¹ Distributed data mining (DDM)

² Knowledge discovery from databases (KDD)

³ Data Mining (DM)

⁴ Bayesian learning

⁵ Classification

⁶ Regression

اشکال بعدی هنگام توسعه مجموعه داده‌های حجمی رخ می‌دهد که در تعداد زیادی از سایت‌های مختلف توزیع شده هستند. هرگونه کاوش برخط^۱ روی چنین مجموعه حجمی از داده‌های توزیع شده در یک انبار داده، از نظر ارتباطات و محاسبات، گران و هزینه برخواهد بود. به طور کلی، ویژگی‌های ذاتی نرم‌افزارهای هوشمند مبتنی بر عامل که عبارتند از: خودمختاری، قابلیت وقپذیری و خصایص مناسب دیگر، آنها را برای مقابله با مشکلات فوق و چالش‌های مطرح در داده‌کاوی توزیع شده، کاملاً متناسب می‌سازد. عوامل داده‌کاوی خودمختار، به عنوان نوع خاصی از عوامل اطلاعاتی می‌توانند انواع عملیات کاوش داده را از جانب کاربر یا با همکاری عوامل دیگر، اجرا کنند. به‌نظر می‌رسد سیستم‌هایی که مبتنی بر همکاری عوامل اطلاعاتی در انجام کارهای داده‌کاوی در محیط‌های توزیع شده، ناهمگون^۲ و داده‌های انبوه هستند در آینده نزدیک بیش‌تر مطرح شوند (۲).

در این مقاله، مروری داریم بر تعامل و یکپارچگی بین سیستم‌های چندعامله^۳ و داده‌کاوی توزیع شده که از آن تحت عنوان "سیستم‌های داده‌کاوی توزیع شده چندعامله" نام می‌بریم و در آن شاخص‌هایی نظیر میزان اهمیت، ساختار سیستم، سیستم‌های موجود و روند تحقیقات در این زمینه را مورد بررسی قرار خواهیم داد.

۲- آخرین دستاوردها

در این بخش، مهم‌ترین و شاخص‌ترین سیستم‌های توزیع شده مبتنی بر عامل که شامل PADMA، BODHI و Papyrus JAM است را بیان خواهیم کرد و براساس موارد زیر و زمان انجام عملیات داده‌کاوی که توسط سیستم دنبال می‌شود، بررسی می‌کنیم (۲ و ۳):

- نوع، روش و ابزار استفاده شده جهت برقراری امنیت داده پردازش شده
- تکنیک‌های داده‌کاوی استفاده شده، پیاده‌سازی سیستم و عامل‌ها
- معماری سیستم با توجه به هماهنگی و کنترل، اجرای پردازش داده و انتقال عامل‌ها، داده و مدل‌ها

BODHI چارچوبی است که جهت جمع‌آوری عملیات داده‌کاوی از سایت‌های داده ناهمگون طراحی شده است. این چارچوب صحت مدل‌های داده‌ای محلی و عمومی را به همراه سربار پایین ارتباط شبکه‌ای، تضمین می‌کند . BODHI به زبان جاوا پیاده‌سازی شده است. این چارچوب، از طریق تبادل پیام در محیط‌های اجرایی به اجرای عوامل سیار در هر سایت محلی کمک می‌کند. فرایند کاوش در حالی که هر کدام از عامل‌ها در حال حمل داده یا دانش هستند، بین ایستگاه‌های عامل محلی و عواملی که بر حسب تقاضا بین آنها - ایستگاه‌ها- در حال حرکت هستند، توزیع شده است. یک عامل تسهیل‌کننده^۴ مرکزی مسئول راهنمایی و هماهنگ کردن اعمال داده‌کاوی است که باید توسط عامل‌ها و ایستگاه‌های عامل، در سیستم انجام شود. او هم‌چنین مسئول برقراری ارتباطات و کنترل جریان بین عامل‌ها است.

JAM^۵ یک سیستم چندعامله مبتنی بر جاوا است که برای یادگیری در سطوح بالای^۱ داده‌کاوی توزیع شده، طراحی شده است. مدل‌های دسته‌بندی^۲ یادگیری متفاوت مانند: Ripper, CART, C4/5, ID3, Bayes و

¹ Online

² Heterogeneous

³ Multi-agent systems (MAS)

⁴ Facilitator

⁵ Java Agents for Meta-learning

WEPBLS می‌توانند برروی پایگاه داده‌های رابطه‌ای ناهمگن اجرا شوند. این کار توسط هر عامل JAM انجام می‌شود که یا در یک سایت قرار دارند و یا از سایت متناظر دیگری در سیستم، وارد شده‌اند. هر عامل در هر سایت، یک مدل دسته‌بندی ایجاد می‌کند درنتیجه، عوامل متفاوت با استفاده از تکنیک‌های مختلف، دسته‌بندهایی متفاوت می‌سازند. به علاوه، JAM مجموعه‌ای از عوامل یادگیر سطح بالا تولید می‌کند که از آن‌ها برای ترکیب چندین مدل که هر کدام در سایت‌های متفاوت آموزش دیده‌اند به یک مدل دسته‌بند سطح بالا استفاده می‌شود و از بسیاری جهات دقت کلی پیش‌بینی را بهبود می‌بخشد. هنگامی که دسته‌بندی‌های ترکیبی ایجاد شدند، سیستم JAM مرکزی، هماهنگ می‌کند که این بسته‌ها به‌طور همزمان و مستقل جهت دسته‌بندی مناسب مجموعه‌های داده در کلیه سایت‌ها، اجرا شوند. جزئیات بیشتر در مورد معماری و پیاده‌سازی سیستم JAM در^(۴) یافت می‌شود.

^۳ PADMA با مشکلات موجود در سایت‌های داده همگن در داده‌کاوی توزیع شده سر و کار دارد. ابتدا در سایت‌های متفاوت، مدل‌های خوش‌های داده همگن در داده‌کاوی ایجاد می‌شود. سپس کلیه مدل‌های محلی در یک سایت مرکزی جمع شده و یک الگوریتم خوش‌بندی در سطح بالاتر برای تولید یک مدل خوش‌های کلی، اجرا می‌شود. عوامل منفرد، خوش‌بندی به‌روش سلسله مراتبی را برای دسته‌بندی مستندات منتهی و اطلاعات تصویری مبتنی بر وب، اجرا می‌کنند. تمرکز^(۵) نشان دادن مزایای داده‌کاوی موازی مبتنی بر عامل است.

Papyrus یک سیستم مبتنی بر جاوا است که داده‌کاوی توزیع شده را به‌طور گسترده بر روی خوش‌های سایت‌های داده ناهمگن و خوش‌های سطح بالا اجرا می‌کند. این سیستم از انواع روش‌های مدل‌سازی و پیش‌بینی، نظیر C4/5 پشتیبانی می‌کند. عوامل داده‌کاوی سیار، نتایج میانی و مدل‌ها را برای اجرای محلی محاسبات و کاهش سربار شبکه، بین خوش‌های جایه‌جا می‌کنند و یا آن‌ها را از سایت‌های محلی به قسمت مرکزی می‌برند تا در آن‌جا نتایج نهایی تولید شود. هر خوش‌های دارای یک گره متمایز است. این گره طوری عمل می‌کند که خوش‌های مربوط به آن بتوانند به عامل‌ها دسترسی و روی آن‌ها کنترل داشته باشد. هماهنگی بین عملیات خوش‌بندی یا توسط سایت مرکزی انجام می‌شود یا بر روی شبکه‌ای از نقاط دسترسی‌پذیر خوش‌های توزیع می‌گردد. Papyrus، با استفاده از یک زبان خاص، از روش‌های متفاوت ترکیب و تبادل مدل‌های پیش‌بینی که از کاوش محلی به‌دست آمده، همچنین از داده‌های سطح بالای مورد نیاز جهت توصیف آن‌ها، پشتیبانی می‌کند.

آخرین نکته این که Papyrus سیستم خاصی است که برای خوش‌های طراحی شده در حالی که JAM و BODHI به‌منظور دسته‌بندی داده‌ها طراحی شده‌اند.

۳- عامل‌ها در داده‌کاوی توزیع شده

V. SREENIVASA RAO در^(۱) مهم‌ترین دلایل استفاده از عامل‌ها در داده‌کاوی توزیع شده را به شرح زیر بیان می‌کند:

^۱ Meta learning

^۲ Classifier

^۳ Parallel Data Mining Agents

- عامل‌ها می‌توانند طوری برنامه‌ریزی شوند که به جای انتقال داده‌های حجمی، عوامل سبک وزن در سرتاسر شبکه حرکت کنند؛ به عنوان مثال، فرض کنید عامل داده‌کاوی a_1 که در سایت s_1 قرار گرفته الگوریتم alg_1 را اجرا می‌کند. عمل داده‌کاوی t_1 که در سایت s_2 واقع شده می‌باشد داده‌کاوی را توسط alg_1 انجام دهد. در این مسأله، برای اجرای alg_1 ، به جای انتقال کلیه داده‌ها از s_2 به s_1 بهتر است عامل a_1 را به سایت s_2 انتقال دهیم.
- امنیت، به عنوان یک مسأله جدی در داده‌کاوی توزیع شده مبنی بر عامل مطرح است، لذا توصیه می‌شود برای افزایش امنیت، به جای انتقال داده، عامل‌ها از طریق سیستم شبکه حرکت کنند.
- توسعه‌پذیری نیز به عنوان یکی دیگر از مسائل قابل توجه در سیستم‌های توزیع-شده و به منظور آگاه‌کردن هر واحد از آخرین تغییرات پیکربندی سیستم، مطرح است. به طور مثال، وقتی یک سایت داده‌ای جدید وارد سیستم می‌شود، عوامل یادگیری همکار در مورد تغییرات پیکربندی سیستم با یکدیگر تعامل می‌کنند.

در ادامه این بخش به بحث در مورد مزایا و معایب استفاده از عوامل هوشمند در داده‌کاوی توزیع شده می‌پردازیم:

- استقلال منابع داده: یک عامل داده‌کاوی می‌تواند به عنوان یک توسعه پیمانه‌ای برای سیستم مدیریت داده در نظر گرفته شود تا بتواند دسترسی به منابع داده‌ای با اهمیت را با توجه به محدودیت‌های موجود در استقلال سیستم، داده و مدل اداره کند. این مورد کاملاً منطبق با مدل سیستم‌های اطلاعاتی اشتراکی است.
- داده‌کاوی توزیع شده تعاملی^۱: عوامل همکار می‌توانند میزان نظارت و دخالت کاربر انسان را هنگام اجرای فرایند داده‌کاوی محدود کنند. به عنوان مثال، عوامل داده‌کاوی می‌توانند محدودیت‌های فردی فضاهای جستجویی که قابلیت بزرگ شدن را دارند پیش‌بینی کنند. آن‌ها همچنین قادر به تخمین نتایج میانی مناسبی هستند که به طور خاص توسط اولویت‌های کاربران فردی و با درنظر گرفتن نوع خاص عمل داده‌کاوی که در حال انجام است، تعیین می‌شود.
- انتخاب پویای منابع و جمع‌آوری داده: یکی از چالش‌های موجود در عوامل داده‌کاوی هوشمند، انجام کار در محیط‌های توزیع شده داده‌ای باز^۲ است به طوری که به عنوان مثال برای بررسی امکان دسترسی به سایت‌های داده و محتویات آن‌ها که در هر لحظه می‌توانند تغییر کند، یکی از وظایف داده‌کاوی، کشف و انتخاب منابع مرتبط است. در چنین شرایطی عوامل داده‌کاوی می‌توانند برای انتخاب منابع داده سازگار با شرایط موجود مانند حجم، نوع و کیفیت مورد انتظار داده در منبع مورد نظر، شبکه حقیقی و بار سرویس‌دهنده داده‌کاوی، به کار گرفته شوند. چنین عوامل داده‌کاوی از جمله می‌توانند برای کنترل و مدیریت پویای فرایند جمع‌آوری داده در راستای پشتیبانی از انواع کاربردهای OLAP^۳ و انبار داده تجاری استفاده شوند.

¹ Interactive

² Open

³ Online analytical processing

- توسعه‌پذیری داده‌کاوی در برابر داده‌های توزیع شده حجمی: یک راه برای کاهش بار سرویس‌دهنده شبکه در کاربردهای داده‌کاوی این است که اجازه دهیم عوامل داده‌کاوی به هریک از سایت‌های داده محلی در سیستم داده‌کاوی توزیع شده مهاجرت کنند تا بتوانند عمل داده‌کاوی را به صورت محلی اجرا کنند و سپس به همراه داده‌های پیش‌فرض مرتبط برای انجام فرایندهای بعدی، به سرویس‌دهنده مبدأ بازگردند. بهتر است در زمینه استفاده از عوامل سیار فیلترکننده اطلاعات در محیط‌های توزیع شده، تحقیقات بیشتری انجام شود.
- داده‌کاوی توزیع شده چندرابربری: در کاربردهای پیچیده استفاده از ترکیب چندین تکنیک داده‌کاوی می‌تواند مفیدتر از به کار بردن تنها یک روش باشد. عوامل داده‌کاوی قادرند یاد بگیرند که کدام عمل خاص را در آینده انتخاب کنند و این کار با توجه به نوع داده بازیابی شده از سایت‌های متفاوت و نیز عملیات داده‌کاوی که باید انجام شود، صورت می‌گیرد.
- داده‌کاوی اشتراکی: عوامل داده‌کاوی می‌توانند به صورت مستقل روی داده‌هایی که در سایت‌های محلی جمع‌آوری کرده‌اند، عمل کنند و سپس مدل‌های خاص خود را با یکدیگر ترکیب نمایند و یا در مورد به اشتراک‌گذاری بخشی از دانشی که کشف کرده‌اند با یکدیگر به توافق بررسند و از این طریق از نظرات عوامل دیگر بهره‌مند شوند. تکنیک‌های یادگیری سطح بالا می‌توانند برای اجرای عمل داده‌کاوی روی داده‌های توزیع شده همگن استفاده شود. اگر سایت‌های داده، ناهمگن و متفاوت باشند و هر کدام داده را به صورت مجموعه‌ای از ویژگی‌های گوناگون و احتمالاً با برخی ویژگی‌های مشترک بین سایت‌ها، ذخیره کنند، در این صورت روش‌های بومی روی تحلیل داده‌های محلی ممکن است منجر به ایجاد مدل داده عمومی غیر واضح و یا ناصحیح شود. عوامل داده‌کاوی همکار با یکدیگر مذاکره می‌کنند و از طریق ارتباط با یکدیگر، راه حلی برای مشکل مطرح شده در بالا، ارائه می‌دهند. نیاز به همکاری و تعامل عوامل داده‌کاوی بسیار مهم است؛ به عنوان مثال در نظر بگیرید که کارت‌های اعتباری جعلی می‌باشد از طریق خواندن، تحلیل و جمع‌آوری جزئی رکوردهای داده توزیع شده روی منابع مستقل و متفاوت در سرتاسر جهان، کشف شود. کاربردهای دیگر که پتانسیل ارزش افزوده دارند شامل به خاطر سپردن توزیع جغرافیایی رکوردهای بیماران و کاوش در فضای داده برای کشف دانش پنهان و حمایت از روش‌های پیشرفته درمان بیماران بدون توجه به این‌که آن در گذشته در کدام و چند بیمارستان بستری بوده‌اند، است.
- امنیت و قابلیت اطمینان: در حقیقت، این مورد می‌تواند بحث علیه استفاده از عامل‌ها در داده‌کاوی توزیع شده باشد. البته هر سیستم داده‌کاوی مبتنی بر عامل باید با مشکل اطمینان از امنیت داده و حریم خصوصی، روبرو شود. اگرچه، هر خطایی در پیاده‌سازی باعث کاهش سود در یک منع داده می‌شود، که به معنای آن است که سایجکت‌های^۱ خوب با دسترسی کافی برای انجام وظایف‌شان، می‌توانند به هر عامل داده‌کاوی دسترسی تام به داده‌های حساس را بدهد. به علاوه، هر عمل داده‌کاوی که توسط عوامل یک سیستم داده‌کاوی توزیع شده اجرا می‌شود در صورتی که دارای ضعف در ساختار امنیتی باشد می‌تواند موجب استراق سمع، دسترسی غیرمجاز به داده و یا مورد حمله هکرها در شبکه قرار گیرد. از

¹ Subjects

این رو، کدگذاری عامل و یکپارچگی داده یک مسئله حیاتی در داده‌کاوی توزیع شده امن است. گمراه کردن یا در دست گرفتن یک عامل داده‌کاوی، در بخش مورد اعتمادی از نرم‌افزار اتفاق می‌افتد و به این ترتیب، هر داده حساس که توسط عامل مورد نظر انتقال یابد تحت کترول یک مزاحم خواهد بود. در مواردی که عوامل داده‌کاوی اجازه مهاجرت به محیط محاسباتی دیگر از سیستم توزیع شده را داشته باشند روش‌هایی برای اطمینان از قابلیت اطمینان و یکپارچگی یک عامل سیار باید به کار گرفته شود. با توجه به دسترس پذیر بودن عامل، راه قطعی برای جلوگیری از این‌که میزبان‌های نامهربان عواملی که به‌طور موقت در آن‌ها ساکن شده‌اند را بلاک یا خراب کنند، وجود ندارد؛ اما یک راه مفید، تکثیر انتخابی در یک معماری عامل داده‌کاوی توزیع شده تحمل‌پذیر خطا است. به علاوه، یکپارچگی یا جمع شدن داده در یک فرایند داده‌کاوی توزیع شده، می‌تواند موجب نگرانی بابت حملات جاسوسی به عنوان تهدیدی برای امنیت بهشمار آید (۲۰۳ و ۶).

۴- داده‌کاوی توزیع شده مبتنی بر عامل^۱

انواع مختلفی از کاربردهای داده‌کاوی توزیع شده وجود دارد که می‌توان به موارد زیر اشاره کرد: سیستم تشخیص کارت اعتباری جعلی، سیستم تشخیص نفوذ هکر، بیمه سلامت، کاربردهای مرتبط با امنیت، خوشبندی توزیع شده و غیره. این کاربردها می‌توانند در آینده توسعه عامل‌ها، افزایش یابند. سیستم‌های داده‌کاوی توزیع شده مبتنی بر عامل از داده‌کاوی به عنوان پایه استفاده می‌کنند و با استفاده از عامل‌ها گسترش می‌یابند. بنابراین، این تکنیک نوین داده‌کاوی، با به کارگیری تمام نقاط قوت عامل‌ها دارای ویژگی‌های مطلوب است.

به‌طور کلی ایجاد یک سیستم داده‌کاوی توزیع شده مبتنی بر عامل، دارای سه ویژگی اساسی زیر است (۷۱ و ۶):

- قابلیت همکاری
- پیکربندی پویای سیستم
- موارد مربوط به کارایی

قابلیت همکاری نه تنها شامل همکاری عامل‌ها در سیستم است بلکه تعامل آن‌ها با محیط خارج را نیز شامل می‌شود و به‌این ترتیب اجازه می‌دهد یک عامل به صورت یکپارچه از محیط بیرون وارد سیستم شود. معماری سیستم باید چنان انعطاف‌پذیر باشد که بتواند تعاملاتی شامل پروتوكل‌های ارتباطی، سیاست یکپارچگی و دایرکتوری سرویس را پشتیبانی کند.

پروتوكل ارتباطی، کدگذاری، رمزگذاری و انتقال پیام بین عامل‌ها را پوشش می‌دهد، با وجودی که این موارد توسط سازمان FIPA^۲ استاندارد شده‌اند و دسترسی عمومی به آن‌ها وجود دارد. اکثر سکوهای عامل مانند JACK3 و JADE2 مطابق با استاندارد FIPA هستند.

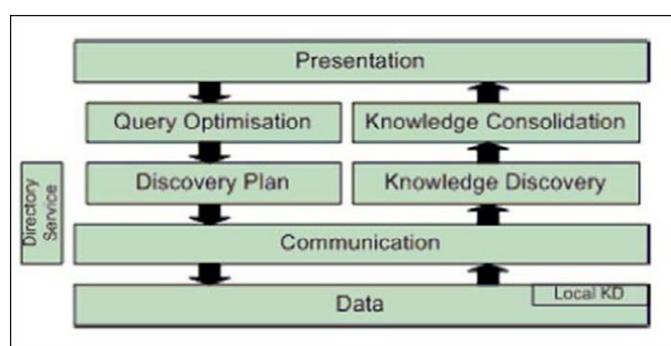
¹ Agent-Based Distributed Data Mining (ADDM)

² Foundation of Intelligent Physical Agents (FIPA)

سیاست یکپارچگی، رفتار یک سیستم را هنگامی که یک جزء خارجی مانند یک عامل یا سایت داده، وارد سیستم شود یا آنرا ترک کند، تعیین می‌کند.

پیکربندی پویای سیستم، اداره یک پیکربندی پویا برای سیستم را بر عهده دارد که در نهایت موجب بهبود کارایی سیستم می‌شود، چون توزیع داده خود از جمله مهم‌ترین محدودیت‌ها است.

اجزای اصلی یک سیستم داده‌کاوی توزیع شده مبتنی بر عامل مطابق با شکل (۱) است:



شکل ۱: اجزای اصلی سیستم داده‌کاوی توزیع شده مبتنی بر عامل

- داده: در محیط توزیع شده، داده به فرم‌های متفاوت نگهداری می‌شود مانند پایگاه داده رابطه‌ای برخط، جریان داده، صفحات وب و غیره و این بستگی به این دارد که داده به چه منظور استفاده شود.
- ارتباطات: سیستم، منابع مرتبط را از سرویس دایرکتوری انتخاب می‌کند که لیستی از منابع داده، الگوریتم‌های داده کاوی، طرح کلی داده، نوع داده و غیره در اختیار دارد. پروتوكول‌های ارتباطی با توجه به پیاده‌سازی سیستم می‌توانند متغیر باشد. مثلاً پروتوكول مشتری–سرویس دهنده^۱ یا نظریه به نظری^۲.
- نمایش: واسط کاربر^۳، برای دریافت و پاسخ به درخواست‌ها با کاربر تعامل دارد. این واسط، سیستم توزیع شده پیچیده را به ایزارهای کاربرپسند مانند دیاگرام‌های شبکه، ایزار گزارش تصویری و غیره تبدیل می‌کند.

از طرف دیگر، هنگامی که یک کاربر درخواست داده کاوی را از طریق واسط کاربر می‌دهد، اجزای زیر در گیر می‌شوند:

- بهینه‌سازی درخواست^۴: یک بهینه‌ساز درخواست، درخواست کاربر را تحلیل می‌کند و نوع عمل داده کاوی و انتخاب منابع مناسب برای پاسخ‌گویی به درخواست را تعیین می‌نماید. همچنین تعیین این که آیا می‌توان وظایف را موازی‌سازی کرد یا خیر نیز از وظایف این واحد است.
- کشف طرح^۱: یک طراح، وظایف کوچکتر را به منابع مربوطه اختصاص می‌دهد.

¹ Client-server

² Peer-to-peer

³ User interface (UI)

⁴ Query optimization

وقتی که یک عمل داده کاوی اجرا شود اجزای زیر وارد عمل می‌گردد:

- کشف دانش محلی^۲: فرایند داده کاوی در هر سایت داده، می‌تواند به صورت محلی انجام شود تا از این طریق عمل تبدیل داده به الگوهایی که دربردارنده خصوصیات داده هستند، انجام شود و سپس این الگوها در سرتاسر شبکه انتقال یابند.
- کشف دانش: می‌توان از آن تحت عنوان استخراج دانش نام برد. این واحد، برای به دست آوردن دانش از منابع داده تعیین شده الگوریتم را اجرا می‌کند.
- تثیت دانش: این واحد، دانش به دست آمده از انواع منابع را از طریق نرم‌السازی به نتایج داده کاوی معنی دار برای نمایش به کاربر، تبدیل می‌کند (۱۷).

۵ - تعامل و یکپارچگی

در این قسمت کارهای اساسی که بر روی سیستم‌های داده کاوی توزیع شده مبتنی بر عامل انجام شده را به طور مختصر شرح می‌دهیم. مروری بر تکنولوژی استفاده از عامل‌ها در داده کاوی در (۸) آورده شده است. (۹) یکی از اولین کارهایی است که توجه‌ها را به سوی سیستم‌های داده کاوی توزیع شده مبتنی بر عامل جلب کرد از طریق صحبت راجع به مزایای آن‌ها در کاوش حجم زیاد داده که در شبکه ذخیره شده و استفاده از قابلیت همکاری عوامل داده کاوی با یکدیگر. در مقاله اخیر، مطالعاتی بر انواع روش‌های مبتنی بر عامل در کشف دانش توزیع شده با استفاده از برنامه‌نویسی منطقی استنتاجی^۳ صورت گرفته که آن را در اختیار برخی از نتایج آزمایشی کاربردهای روش مبتنی بر عامل در داده کاوی توزیع شده، قرار می‌دهد.

در (۱۰) تمایل به حل مشکلات کاربردهایی دارد که اغلب نیازمند پردازش داده‌های حجمی و یا داده‌های جمع‌آوری شده در سایت‌های جغرافیایی متفاوت هستند، از این‌رو نمی‌توانند توسط سیستم‌های ترتیبی و مرکزی پردازش شوند. هم‌چنین در آن توانایی سیستم‌های چند عامله در تولید فرایندهای محاسباتی با دقت بالا، تحمل پذیری خطأ، توسعه‌پذیری و افزایش سرعت بررسی شده که با پیشرفت تکنولوژی کامپیوتر و شبکه و پشتیبانی از پیاده‌سازی فرایندهای اطلاعاتی توزیع شده و موازی، امکان‌پذیر است. براساس یادگیری چند عامله، سیستم نهایی می‌تواند طوری ساخته شود که کارایی خود را بهبود بخشد. سپس به طور خاص به مسئله انساب و ظایف توجه دارد که اساس بسیاری از مسائل مربوط به زمان‌بندی است. هدف، یافتن راه حل منطقی در زمان منطقی است. گره‌های مجموعه ترتیب جزئی کارهای، به عنوان ورودی‌ها یا عوامل فعال در نظر گرفته شده‌اند و کارهایی که به عنوان ورودی‌ها یا منابع غیر فعال تعیین شده‌اند، توسط عامل‌ها استفاده می‌شوند. عامل‌ها به منظور یافتن راه حلی برای رسیدن به برخی شرایط از پیش تعریف شده است. این راه حل در جستجوی رویه‌ای است که توسط عامل‌ها بر اساس یادگیری تقویتی توزیع شده پیاده‌سازی شده، و از راه حل اولیه‌ای است که از ارتباطات سطح پایین و همکاری بین عامل‌ها استفاده می‌کند، شروع می‌گردد. این مقاله به طور تجربی در جهت اثبات برخی مزایای توسعه مدل‌های مبتنی بر عامل است که برای پیاده‌سازی ماشین‌های یادگیری توزیع شده و موازی استفاده می‌شود.

¹ Discovery Plan

² Local Knowledge Discovery (KD)

³ Using Inductive Logic Programming (ILP)

در (۲) در مورد مزايا و ارزش افزوده استفاده از سистем‌های داده‌کاوي توزيع شده مبتنی بر عامل، مطالعه شده است. روش‌های مبتنی بر عامل برای داده‌کاوي توزيع شده، مرور و دسته‌بندی شده است. هم‌چنین پيشنهاد پياده‌سازی عامل محور برای يك سистем خوش‌بندی توزيع شده، داده شده است و در آن به طور واضح توسط فرمول اثبات شده که چرا روش مبتنی بر عامل برای داده‌کاوي توزيع شده مناسب است (استقلال، تعامل با محیط بیرون، قابلیت انتخاب پویای منبع داده در محیط تغیيرپذیر و غیره). شماي KDEC پيشنهاد شده، از تخمين چگالی محاسبات آماری و اطلاعات مثال‌های تئوري، برای کميته کردن ارتباطات بين سايت‌ها استفاده می‌کندکه بر اساس تكنولوژي عامل به عنوان سیستم خوش‌بندی توزيع شده پياده‌سازی شده‌اند. به علاوه، يكى از ویژگی‌های متماييز آن اين است که حریم خصوصی داده‌های محلی را حفظ می‌کند.

(۱۱) بر روابط دوجانبی بين تكنولوژي سیستم‌های چندعامله و داده‌کاوي توزيع شده تأکید دارد. تمرکز اصلی آن بر خوش‌بندی توزيع شده است که زمینه کاربردی آن رو به افزایش است؛ به عنوان مثال، در شبکه‌های حسگر که گره‌ها طوری قرار دارند که دسترسی به موقعیت آن‌ها مشکل است مانند میدان جنگ، که در آن حسگرها توسط لرزش، انعکاس، درجه حرارت و پیام صوتی، اندازه‌گیری می‌شوند. هم‌چنین شبکه حسگر برای نظارت منطقه، خانه‌های هوشمند و بسياري از زمینه‌های کاربردي ديگر مورد استفاده قرار می‌گيرد. در تحليل زمينه‌های کاربردی؛ به عنوان مثال، عمليات داده‌کاوي توزيع شده، به دليل محدوديت‌های زياد جزء مسائل مهم است؛ مانند پهنانی باند محدود در کانال‌های ارتباطي بسيم، روش ارتباطي نظری به نظر و الزام تعامل در شبکه غيرهمzman، حریم خصوصی داده، هماهنگی در کاربردهای توزيع شده، اهمیت عدم تركيب، تعداد هنگفت گره‌های شبکه، منابع کامپیوتروی محدود مثلاً به دليل محدوديت در توليد برق و غیره. نويسنده‌گان مقاله تصريح می‌کنند که چارچوب سنتي در تحليل داده متمرکز و الگوريتم‌های داده‌کاوي برای کاربردهای توزيع شده، مقیاس‌پذير نیستند. بر عکس، بهتر است مسائل توزيع شده توسط چارچوب چندعامله انجام شود زيرا اين روش از رفتار نيمه‌خودکار، همکاري و استدلال بين سیستم‌های چندعامله به خوبی پشتيبانی می‌کند. از ميان زمينه‌های تحقيقاتي داده‌کاوي توزيع شده، تمرکز اين مقاله بر روی خوش‌بندی در شبکه‌های حسگر است که با چالش‌های جدي روبرو است. در اين مقاله پيشنهاد شده که تكنيك‌های داده‌کاوي متمرکز سنتي ممکن است در زمينه‌های کاربردي فوق خوب کار نکنند، در حالی که الگوريتم‌های داده‌کاوي توزيع شده که با معماری سیستم‌های چندعامله يکپارچه شده‌اند می‌توانند پيشنهاد يك تكنولوژي اطلاعاتي تعاملی جدي را در اين زمينه‌های کاربردي بدهنند. اخيراً برخی تلاش‌ها برای توسعه مشكلات تكنولوژيکي داده‌کاوي چندعامله انجام شده که می‌توانند به عنوان نشانه‌ای از افزایش بلوغ و تکامل اين سیستم‌ها ارزیابی شود.

(۱۲) بيان می‌کند که مشکل اصلی داده‌کاوي توزيع شده و طراحی الگوريتم يادگیری ماشین، محدود به تكنيك‌های داده‌کاوي نیست. بلکه مشکل اساسی مربوط به توسعه زیر ساخت و پروتکل‌های حمایت‌کننده از عمليات همکاري و انسجام اجزاي نرم‌افزار توزيع شده (عامل‌ها) است که در حال اجرای يادگیري توزيع شده هستند. اين مقاله پيشنهاد يك معماری چندعامله از يك سیستم تركيبی اطلاعاتي را می‌دهد که دارای قabilites‌های داده‌کاوي توزيع شده و يادگیري ماشین است. هم‌چنین در آن يك تكنولوژي طراحی را پيشنهاد می‌دهد که هسته آن توسط تعدادی پروتکل محاوره‌ای با عامل خاص تعين شده که از عمليات عامل توزيع شده در انواع سناريوها به خصوص پروتکل داده‌کاوي توزيع شده، حمایت می‌کنند.

توسعه بعدی تكنولوژي طراحی سیستم داده‌کاوي توزيع شده در (۱۴) و (۱۵) بيان شده است.

(۸) پیشنهاد یک چارچوب (یک معماری انتزاعی) را برای یادگیری ماشین توزیع شده مبتنی بر عامل و داده کاوی می‌دهد. چارچوب پیشنهادی بر اساس مشاهدهای است که علیرغم استقلال و خود محور بودن یادگیری عامل‌ها، بسیاری از این سیستم‌ها بر اساس اهداف یادگیری یک همپوشانی مناسب دارند که در آن، عملیات درصورتی مؤنث است که یک مدل برای ایجاد تعامل انعطاف‌پذیر بین یادگیرهای خودمنختار موجود باشد و اجازه دهد عامل‌ها:

۱. از جنبه‌های مختلف مکانیزم یادگیری و در سطوح مختلف، تبادل اطلاعات داشته باشند بدون آن که مجبور باشند اطلاعات محترمانه را آشکار کنند.
۲. بتوانند در مورد میزان به اشتراک گذاری اطلاعات در طول فرایند یادگیری و استفاده از اطلاعات تولید شده توسط سایر عامل‌ها، تصمیم‌گیری کنند.
۳. بدانند چطور از این اطلاعات بهترین استفاده را در جهت ارتقاء کارایی یادگیری‌شان انجام دهند.

تئوری اساسی در چارچوب پیشنهاد شده این است که هر عامل می‌تواند اطلاعات محترمانه مربوط به سطوح بالای فرایند یادگیری خود را برای تبادل با عوامل دیگر به شکلی قابل قبول نگهداری کند و به صورت منطقی عمل کند یعنی به دنبال روشی باشد که باعث بهبود نتایج یادگیری خودش شود. نویسنده‌گان مقاله فوق تصریح می‌کنند که این امکان، برای عوامل یادگیر، فرضیه‌ای است که آن‌ها قصد دارند به صورت تجربی و با استفاده از چارچوب پیشنهادی، آنرا اثبات کنند و هم‌چنین نتایج تحقیقات گذشتگان را نشان می‌دهند و این‌که برای رسیدن به یک ارزیابی قابل قبول از فرضیات موجود، تلاش‌های زیادی انجام شده است.

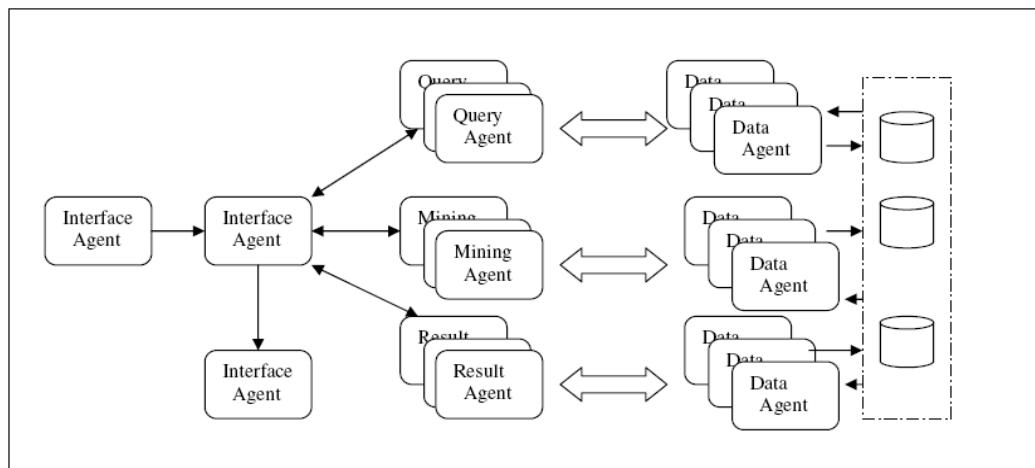
آخرین سیستمی که در این مقاله معرف می‌کنیم F-TRADE^۱ است (۱۶ و ۱۷). این سیستم از یک روش پویا استفاده می‌کند که اجازه یکپارچه شدن منابع داده خارجی با الگوریتم‌های داده کاوی را می‌دهد. سیستم، به تاجران و دلالان مالی، یک بستر آزمایشی نشان می‌دهد که روی مجموعه داده‌های واقعی قرار گرفته و می‌تواند به آنها در ارزیابی راهکارهای تجاری مطلوب‌شان، قبل از سرمایه‌گذاری پول در بازار واقعی، کمک کند. هدف سیستم F-TRADE تولید یک زیرساخت مکانیزه و انعطاف‌پذیر خاص برای تاجران و محققان حوزه مالی است.

۶- معماری کلی سیستم‌های داده کاوی چند عامله^۲

اغلب چارچوب‌های سیستم‌های داده کاوی چند عامله از معماری مشابه استفاده می‌کنند (شکل ۲) و اجزای ساختاری مشترک تولید می‌کنند (۱). آن‌ها از FIPA-A KQML^۳ یا A استفاده می‌کنند که یک زبان استاندارد برای ارتباط بین عامل‌ها است و محاوره بین عامل‌ها را تسهیل می‌نماید.

¹ Financial Trading Rules Automated Development and Evaluation

² Multi-agent Data mining (MADM)



شکل ۲: معماری کلی سیستم‌های داده‌کاوی چندعامله

در ادامه این بخش تعریفی برای عواملی که بیشتر در سیستم‌های داده‌کاوی چندعامله استفاده می‌شوند، ارائه می‌دهیم (۱):

- عامل واسط یا عامل کاربر^۱: این عامل با کاربر یا عامل کاربر تعامل دارد. او از کاربر می‌خواهد که آنچه نیاز دارد در اختیارش قرار دهد و در نهایت نتایج عمل داده‌کاوی (بعضًا به صورت گزارش‌های تصویری) در اختیار کاربر قرار می‌دهد.
- عامل تسهیل‌کننده یا عامل مدیر^۲: این عامل به جزئیات بیشتر نقشه کار می‌پردازد و مسئول اطمینان از اجرای صحیح نقشه کار است. این عامل شرح کلی وظایف را از عامل واسط دریافت می‌کند و می‌تواند در جستجوی سرویس یک گروه از عامل‌ها باشد. سپس نتایج به دست آمده را با یکدیگر ترکیب می‌کند و پس از ایجاد نتیجه نهایی آنرا به عامل واسط ارائه می‌دهد.
- عامل منع یا عامل داده^۳: این عامل مسئول بازیابی مجموعه داده‌هایی است که توسط عامل داده‌کاوی به عنوان پیش‌نیاز انجام عمل داده‌کاوی درخواست شده است. این عامل ناهمگونی در پایگاه‌های داده را در نظر گرفته و سعی در حل ناسازگاری‌های موجود در تعریف و نمایش داده دارد.
- عامل داده کاوی^۴: این عامل مسئول پیاده‌سازی تکنیک‌ها و الگوریتم‌های خاص داده‌کاوی است.
- عامل نتیجه^۵: این عامل ناظر بر اعمال عوامل داده‌کاوی است. نتایج را از آن‌ها دریافت می‌کند و با هماهنگی و تعامل با عامل تسهیل‌کننده، نتایج عمل داده‌کاوی را در اختیار او قرار می‌دهد تا به کاربر نمایش دهد.
- عامل دلال یا واسطه^۶: این عامل به صورت یک عامل تبلیغاتی کار می‌کند که تبلیغات را از تسهیل‌کننده‌های تدارکاتی^۷ و درخواست‌های توصیه را از تسهیل‌کننده‌های درخواست^۸ می‌گیرد. به طور کلی هر عامل جدید

¹ Interface Agent (or User Agent)

² Facilitator Agent (or Manager Agent)

³ Resource Agent (or Data Agent)

⁴ Mining Agent

⁵ Result Agent

⁶ Broker Agent (or Matchmaker Agent)

⁷ Supply facilitators

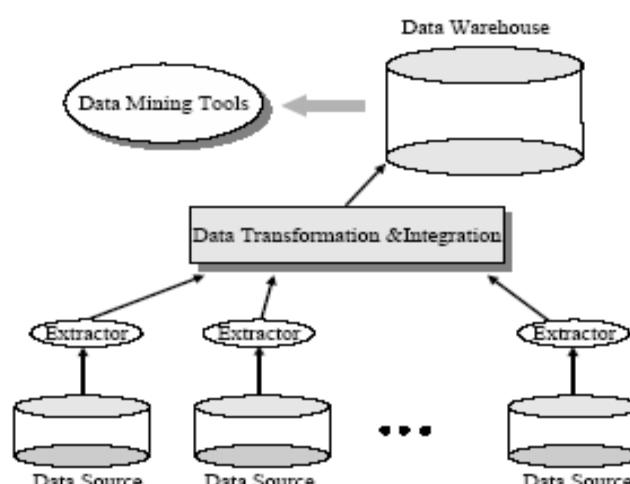
⁸ Request facilitators

که وارد سیستم می‌شود با استفاده از یک عامل واسطه باید توانایی‌های خود را ارائه کند تا بتواند به مجموعه عامل‌ها وارد شود.

- عامل درخواست^۱: این عامل با هر تقاضای کاربر ایجاد می‌شود.
- عامل سیار^۲: عامل سیار در سرتاسر شبکه حرکت می‌کند. در هر سایت، داده را پردازش می‌کند و نتایج را برای میزبان اصلی بازمی‌گرداند. این عمل به جای انتقال پرهزینه حجم بالای داده در شبکه صورت می‌گیرد.
- عامل وظیفه محلی^۳: در اغلب سیستم‌ها، عامل داده، یک عامل محلی است که در هر سایت محلی قرار دارد. این عامل می‌تواند اطلاعات را برای عامل تسهیل‌کننده بفرستد، همچنین قادر است به درخواست‌های داده‌کاوی مربوط به عوامل داده‌کاوی، پاسخ دهد.
- عوامل سیستم کشف دانش از پایگاه داده: برخی از سیستم‌های داده‌کاوی چندعامله شامل عوامل دیگری برای نگهداری همه فرایند کشف دانش از داده هستند، این فرایندها شامل آماده‌سازی و ارزیابی داده است. عامل‌ها عبارتند از:
 - عوامل پیش‌پردازش^۴: داده را برای عمل داده‌کاوی آماده می‌کنند (شامل فرآیند پاکسازی داده می‌شود)
 - عوامل پس از فرآیند داده کاوی^۵: که کارایی و دقت عوامل داده‌کاوی را ارزیابی می‌کند.

۷ - داده‌کاوی توزیع شده

داده‌کاوی و انبار داده دوش به دوش هم جلو می‌روند، به این صورت که در بیشتر کاربردها همه داده‌ها در یک سایت مرکزی، جمع شده، سپس یک الگوریتم داده‌کاوی بر روی آنها اجرا می‌شود (شکل ۳) (۱۸و۷).



شکل ۳: معماری انباره داده

¹ Query Agent

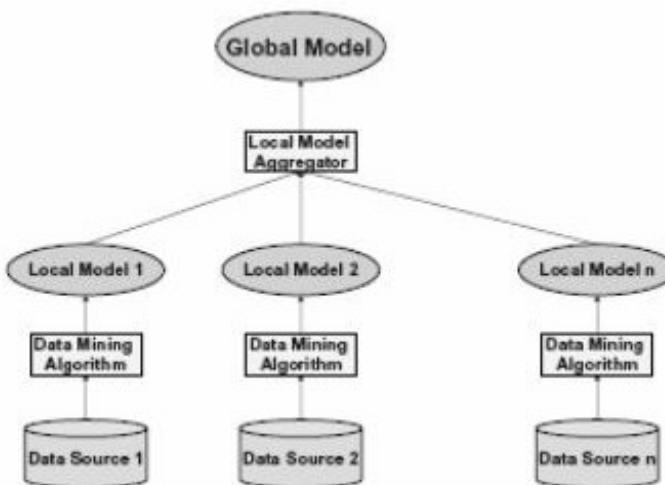
² Mobile Agent

³ Local Task Agent

⁴ Pre-processing Agent

⁵ Post data mining Agent

با این وجود تعدادی از کاربردها هستند که اجرای آنها تحت چنین روش‌شناسی غیرممکن است، زیرا نیاز به داده‌کاوی توزیع شده دارند. در حقیقت، در اغلب کاربردهایی که داده توزیع شده نسبت به زمان حساس است، لازم است که توجه کافی به منابع توزیع شده جهت انجام محاسبات، ذخیره‌سازی و هزینه ارتباطات، به عمل آید. به عنوان مثال، وب جهانی شامل داده‌ها و منابع محاسباتی توزیع شده است. تعداد رو به افزایشی از پایگاه‌های داده و جریان‌های داده در حال حاضر برخبط هستند و بسیاری از آنها مرتبأ در حال تغییر می‌باشند. می‌توان کاربردهای زیادی را یافت که نیازمند نظارت دقیق بر منابع داده متنوع و توزیع شده هستند. از این‌رو در این روش نیاز به عماری داده‌کاوی داریم که توجه کافی به توزیع داده، محاسبات و ارتباطات، به‌منظور دسترسی و استفاده بهینه از آن‌ها داشته باشد. همان‌طور که در شکل (۴) دیده می‌شود، هدف از داده‌کاوی توزیع شده اجرای عملیات داده‌کاوی براساس نوع و دسترسی‌پذیری منابع داده توزیع شده است.



شکل ۴: چارچوب کلی داده‌کاوی توزیع شده

۸ - چالش‌های موجود در زمینه داده‌کاوی توزیع شده

داده‌کاوی در زمان استقرار در محیط واقعی با مشکلات متعددی به‌خصوص در بحث مدیریت داده در کاربردهای پیچیده، روبرو است. در ادامه مهم‌ترین چالش‌های موجود در داده‌کاوی توزیع شده، شرح داده می‌شوند:

در (۷)، به بررسی مشکلات شبکه در یک محیط توزیع شده (مانند شبکه سنسور یا شبکه IP)، که دارای کاوش‌گرهای توزیع شده در موقعیت‌های حساس درون شبکه است می‌پردازد. مشکل اساسی این است که بتوان داده مشاهده شده در انواع کاوش‌گرهای را بهم مرتبط ساخت و در نهایت از کل داده‌های مشاهده شده در کاوش‌گرهای متفاوت، الگو استخراج کرد. مدل‌های متفاوتی از داده‌کاوی توزیع شده وجود دارد که یکی از آن‌ها NOC^۱ است که داده را از سایت‌های توزیع شده جمع‌آوری می‌نماید و همه سایتها را به صورت مساوی در نظر می‌گیرد. NOC‌ها مسئول نظارت بر شبکه هستند. آن‌ها برای هشدارها و نیز توجه در شرایط خاص استفاده می‌شوند تا از تأثیرات مخرب روی کارایی شبکه جلوگیری کنند. به عنوان مثال، در یک محیط ارتباط راه دور، NOC‌ها مسئول نظارت بر

¹ Network operations center (NOC)

قطع برق، هشدارهای خط ارتباط (مانند خطای بیتی، خطای چارچوب، خطای کد خط و اشکال در مدار) و سایر مشکلات شبکه هستند که می‌تواند بر کارایی شبکه تأثیرگذار باشد.

«داده‌کاوی بر روی منابع داده ناهمگون» سازمان‌های بزرگ نیاز به کاوش چندین پایگاه داده دارند که در شعبه‌های آن‌ها توزیع شده است. داده درون شرکت از نوع داده داخلی است درحالی که داده جمع‌شده از طریق اینترنت، داده خارجی درنظر گرفته می‌شود. چالش اصلی این است که: چطور می‌توان به‌طور کارا کیفیت دانش استخراج شده از چندین پایگاه داده را تعریف کرد.

«حریف داده کاوی» در زمینه‌های رو به رشدی مانند بدافزار ایمیل، عملیات ترویریستی، ایجاد اختلال در امنیت کامپیوتر، بدافزارهای موتور جستجو، نظارت، کشف جعل، اشتراک فایل و غیره، سیستم‌های داده‌کاوی با دشمنانی مواجه می‌شوند که تعمدًا داده را برای خرابکاری کنترل می‌کنند. در همه این زمینه‌های خرابکارانه، تکنیک‌های دسته‌بندی آماری نقش مهمی در تشخیص صحت و درستی از خرابی دارند. یک روش صحیح برای مبارزه با خرابی این است که اجازه دهیم دسته‌بند با عملیات خرابکاری سازگار شود، که می‌تواند به روش دستی یا مکانیزه انجام شود. مشکل این‌جا است که این، یک بازی بدون پایان بین دسته‌بند و خرابکار خواهد بود. Sreenivasa Rao و V. Sridharan در (۷) یک تئوری بازی پیشنهاد داده‌اند که هدف آن ایجاد یک راهکار یادگیری برای دسته‌بند برای مقابله با خرابکار نیست، بلکه آن‌ها می‌خواهند وضعیت پایانی بازی یعنی یک وضعیت توازن را پیش‌بینی کنند. همچنین در (۷)، مشکل به عنوان یک بازی دو نفره مدل‌سازی شده، به‌طوری که خرابکار سعی دارد بیشترین تأثیر را در سیستم بگذارد و داده‌کاو سعی بر کمینه کردن مقدار خرابی در دسته‌بندی دارد.

۹- چالش‌های موجود در زمینه عامل‌ها

در این بخش، به بررسی مشکلاتی می‌پردازیم که با استفاده از داده‌کاوی برطرف شده‌اند (۱۸):

- آگاهی عامل: آگاهی عامل مربوط به قابلیت یک عامل در مورد تشخیص تغییرات محیط داخلی و خارجی سیستم و تحلیل تغییرات است. عواملی با چنین قابلیتی باید خود تشخیص باشند یعنی قادر باشند تغییرات محیط را مقایسه و استدلال کنند.
- یادگیری عامل: در سازمان‌های چندعامله باز، تعامل به‌طور گسترده‌ای بین عامل و محیط و همچنین بین یک عامل و عوامل دیگر وجود دارد. انتظار می‌رود عامل‌ها از یکدیگر، از محیط، تعاملات و پویایی روابط، یاد بگیرند. به علاوه، عامل‌ها می‌توانند از کاربران و تعامل با بشر نیز یاد بگیرند. برای پرورش دادن چنین قابلیت یادگیری، لازم است عامل‌ها با الگوریتم‌های یادگیری و استدلال، آموخت داده شوند که می‌توانند از آن‌ها در کشف دلایل یا شبیه‌سازی اطلاعات موردنیازشان از داده‌های تعاملی و شرطی، پشتیبانی کنند.
- توانایی عمل عامل: این ویژگی اشاره دارد بر قابلیت یک عامل در انجام عملی که به نفعش باشد براساس دانش به‌دست آمده از طریق تحلیل دقیق، تشخیص و کشف. در پایان عامل‌ها نیاز دارند که سود و هزینه‌ها را متوازن نموده و آنچه باز می‌گردانند را بیشینه کنند و در عین حال مقدار ریسک را قبل از انجام یک عمل، به کمترین حد ممکن برسانند.
- پردازش توزیع شده عامل: در سیستم‌های چندعامله با مقیاس متوجه روحیه بزرگ، عامل‌ها با اعمال پردازشی توزیع شده سروکار دارند، مثلاً یادگرفتن از عامل‌ها در سازمان‌های چندگانه، هدایت همکاری‌های

غیرمت مرکز، همکاری و تعامل بین عامل‌ها بین منابع و پیاده‌سازی جمع‌آوری اطلاعات، ارسال و انتقال بین عوامل موجود در کاربردهای توزیع شده. برای انجام وظایف بالا در شرایط توزیع شده، عامل‌ها نیاز به تصمیم‌گیری پس از تحلیل و استفاده از اطلاعات مرتبط از چندین منبع دارند.

- سرویس‌های همه‌جانبه عامل^۱: معمولاً عامل‌ها برای تولید انواع سرویس‌ها طراحی شده‌اند؛ به عنوان مثال، سرویس‌های شبکه مانند سیستم‌های وب، عوامل سیار برای جستجو و انتقال اطلاعات و سرویس‌های کاربر نظری تعامل کاربر و مدل‌سازی کاربر و غیره. تولید سرویس‌های هوشمند با تکیه بر تحلیل همه جانبه سرویس داده‌ها و اطلاعات مرتبط سرویس، هم‌چنین داده‌های تاریخی سرویس و کارایی سرویس، به منظور درک عمیق داده‌های سرویس و انتخاب بهترین سرویس، صورت می‌گیرد.
- پردازش محدودیت عامل^۲: سیستم‌های عامل پیچیده باز معمولاً دارای انواع محدودیت‌ها از جنبه‌های مختلف هستند؛ به عنوان مثال، محدودیت‌های زمانی و فضایی^۳ یا محدودیت‌های اجرایی از جنبه‌های سازمانی، چنین محدودیت‌هایی شرایطی را ایجاد می‌کند که موجب پیشرفت قابلیت‌های عامل نظری یادگیری، سازگاری، توانایی عمل و سرویس‌دهی می‌شود (۱۸).

۱۰ - مشکلات مطرح در زمینه تعامل و یکپارچگی عامل و داده‌کاوی

تعامل و یکپارچگی موجود بین دو تکیک (عامل‌ها و داده‌کاوی) باعث ایجاد چالش‌های جدید شده است. با نظر گرفتن انواع ترکیبات می‌توان کلیدی برای افزایش سریع فرایند توسعه استفاده از این‌گونه سیستم‌ها یافت. در این بخش، به بررسی آن‌ها از جنبه‌های مختلف می‌پردازیم:

«چشم‌انداز تحقیقاتی» توزیع داده در کاربردهای واقعی روزمره به صورت همگن و ناهمگن است. داده‌ها می‌توانند به دو نوع افقی و عمودی افزایش شوند، به علاوه تقسیم داده ممکن است در سایت‌های داده امکان‌پذیر نباشد. به عنوان مثال، دو پایگاه داده مرتبط در ارتباط با مشتری می‌توانند بر هم تأثیری نداشته باشند، مثلاً یک مشتری ممکن است جزئیات تماس نداشته باشد در حالی که برخی محصولات را بخرد و از این طریق در یکی از پایگاه‌ها رکورد خرید ثبت شود. کاربردها، برای انجام مسائل توزیع شده، ارتباطات و انبار سیستم، به تکنولوژی داده‌کاوی نیاز خواهند داشت. راه دیگر برای توسعه داده‌کاوی توزیع شده مبتنی بر عامل، الهام گرفتن از طبیعت است که امروزه این روش در حال ارتقاء است. هوش دسته‌ای^۴ ارتباط نزدیک با عوامل هوشمند دارد. اخیراً، محققان متوجه امکان پیاده‌سازی سیستم‌های داده‌کاوی توزیع شده با هوش دسته‌ای، شده‌اند. مثالی از کاربرد هوش دسته‌ای در داده‌کاوی، دسته‌بندی‌های مبتنی بر قانون با استفاده از مورچگان، انتخاب ویژگی با بهینه‌سازی کلونی مورچگان^۵، داده‌کاوی و متن‌کاوی با خوشبندی سلسله مراتبی مورچگان و غیره (۲۰ و ۱۹) است.

«چشم‌انداز مهندسی نرم‌افزار»، داده‌کاوی توزیع شده مبتنی بر عامل به طور مرتب نیاز به تبادل داده و مدل‌های داده‌کاوی بین منابع داده، دارد. بنابراین، درک یکپارچه و شفاف از تکنولوژی داده‌کاوی توزیع شده نیازمند تعریف

¹ Agent in-depth services

² Agent constraint processing

³ Temporal and spatial

⁴ Swarm intelligence

⁵ Ant colony

شمای استاندارد برای نمایش و تبادل مدل‌ها است؛ از این‌رو، ابزار مهندسی نرم‌افزاری مطلوب است که از طراحی داده‌کاوی و پایگاه‌های توزیع شده، پشتیبانی کند. تاکنون، CRISP-DM¹ و تلاش‌های مرتبط دیگر در این زمینه مفید بوده است. اساس تمرکز این ابزارها پایگاه داده است. نه تنها پایگاه‌های داده با مقیاس بالا مانند پایگاه داده رابطه‌ای، بلکه پایگاه‌های داده سبک وزن که در کاربردهای محدود نظیر تلفن‌های همراه، استفاده می‌شوند، نیز می‌توانند در داده‌کاوی توزیع شده مبتنی بر عامل مورد استفاده قرار گیرند. عوامل سیار می‌توانند مهاجرت کنند و وظیفه‌شان را در سایت مورد نظر انجام دهند و برگردند درحالی‌که با خود یک مدل نمونه برای تحلیل‌های بعدی می‌آورند. عنصر دوم، ظهور معماری سرویس‌گرا² است. که کاربردهای مبتنی بر عامل را برای یکپارچگی بهتر قادر می‌سازد. معماری سرویس‌گرا یک معماری در حال پیشرفت است که قدرت سازگاری بالا در چندین کاربرد دارد. در حقیقت نمی‌توانیم این را انکار کنیم که امروزه کاربردهای مبتنی بر وب بیشتر و بیشتر محبوب می‌شوند؛ از این‌رو، اینترنت یک عنصر لازم برای سیستم‌های کامپیوتری شده است.

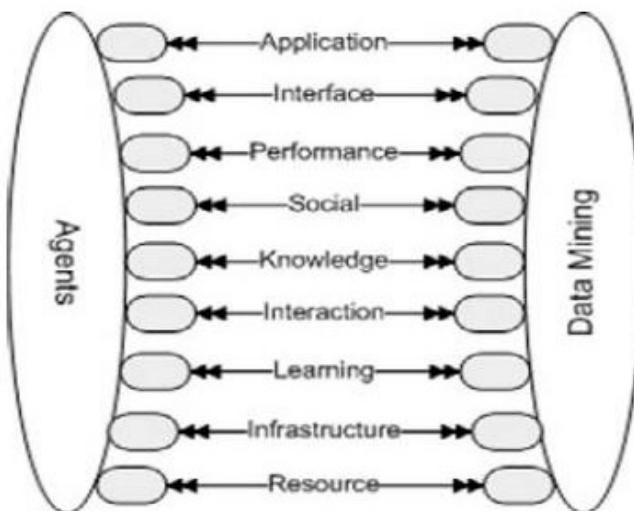
«چشم‌انداز سیستمی» یکی از وجوده جدیدی که در حوزه تعاملات بین عامل‌ها و داده‌کاوی کار تحقیقاتی کمتری در آن انجام شده، شبکه‌های سیار، فرآگیر و نظیر به نظیر است. یک ویژگی خاص از چنین سیستم‌های کامپیوتری این است که عنصر بعدی سیستم با مجموعه پویایی از منابع اطلاعاتی کار می‌کند. به عنوان مثال، وسایل سیار (مانند تلفن همراه) می‌توانند آزادانه حرکت کنند و به شبکه وارد و یا از آن خارج شوند بنابراین تغییر مجموعه گره‌های شبکه و توپولوژی ارتباطات، مجموعه سرویس‌های موجود را تغییر می‌دهد. مثال‌هایی از چنین زمینه‌های کاری فضای هوشمند و هوش‌فرآگیر هستند. در چنین محیط‌هایی، تصمیم‌ها بر اساس ترکیب اطلاعات دریافت شده از حسگرهای توزیع شده و وسایل سیار مقیم در محیط هستند. یکی از اهداف چنین کاربردی سازگاری با عادت‌های چندگانه بشر است. از طرف دیگر، در چنین کاربردهایی، روش چندعامله از اغلب چارچوب‌های طبیعی، معماری‌های مناسب و نیز تکنولوژی طراحی، پشتیبانی می‌کند؛ بنابراین، یکپارچگی عامل و داده‌کاوی در محیط‌های فرآگیر، مانند فضای هوشمند، هوش‌فرآگیر و غیره، می‌تواند چندجانبه و در حال پیشرفت باشد تا در سیستم‌هایی که به‌طور متناظر به کار گرفته می‌شوند، کارایی با کیفیت بالا تولید کند.

«چشم‌انداز کاربر» نهایتاً، مسئله تعامل کاربر و کامپیوتر در داده‌کاوی توزیع شده موجب برخی چالش‌های منحصر به‌فرد می‌شود. این مسئله نیازمند حمایت در سطح سیستم برای تعاملات گروهی، حل مشکلات همکاری، توسعه واسطه‌های متنوع (به‌خصوص برای وسایل سیار)، و مواجه بودن با مسئله امنیت، است.

«چارچوب تعاملی عامل - کاوی» تعامل و یکپارچگی بین عامل‌ها و داده‌کاوی، چندبعدی و دارای جنبه‌های گوناگون است، همان‌طور که در شکل (۵) نشان داده شده است.

¹ The Cross Industry Standard Process Model for Data mining

² Service oriented architecture (SOA)



شکل ۵: ارتباط چندبعدی عامل - کاوی

در (۱۸) به طور خلاصه لایه‌های زیر را برای «عامل - کاوی» به صورت زیر بیان می‌کند:

«لایه منبع» تعامل و یکپارچگی می‌تواند در لایه‌های داده و اطلاعات اتفاق بیفتد.

«لایه زیربنا» تعامل و یکپارچگی می‌تواند در بخش‌های زیربنایی، معماری و فرایند اتفاق بیفتد.

«لایه دانش» تعامل و یکپارچگی می‌تواند بر اساس دانش باشد که این شامل دامنه دانش، دانش خبره بشر، دانش سطح بالا، دانش بازیافت شده، استخراج یا کشف منابع است.

«لایه یادگیری» تعامل و یکپارچگی می‌تواند براساس روش‌های یادگیری، قابلیت‌های یادگیری با دورنمای بهبود کارایی باشد.

«لایه تعامل» تعامل و یکپارچگی می‌تواند در هماهنگی، همکاری، مذاکره و از منظر ارتباطی باشد.

«لایه واسط» تعامل و یکپارچگی می‌تواند واسط سیستم بشر، مدلسازی کاربر و طراحی واسط باشد.

«لایه اجتماعی» تعامل و یکپارچگی می‌تواند براساس معیارهای اجتماعی و سازمانی باشد؛ به عنوان مثال، نقش انسان.

«لایه کاربردی» تعامل و یکپارچگی می‌تواند در کاربردها و مسائل دامنه‌ای باشد.

«لایه کارایی» تعامل و یکپارچگی می‌تواند براساس افزایش کارایی در یکی از تکنولوژی‌ها و یا در هر دو سیستم باشد.

۱۱ - نتیجه‌گیری

امروزه، یکپارچگی و تعامل بین عامل و داده‌کاوی توزیع شده به عنوان یک حوزه تحقیقاتی مهم مطرح است. سیستم‌های چندعامله اساساً برای حل مشکلات به صورت گروهی در محیط‌های توزیع شده، طراحی شده است. بسیاری از این محیط‌های کاربردی به طور عملی با تحلیل و کاوش داده رویرو هستند؛ لذا تعامل و ادغام این دو تکنولوژی، نه تنها انواع مشکلات در هریک از این زمینه‌ها را حل می‌کند بلکه باعث ظهور تکنیک‌های خلاقانه نیز

می شود. در این مقاله سعی نمودیم که به طور اجمالی نگاهی بر مفاهیم اولیه سیستم های داده کاوی چند عامله و کاربردهای اساسی آن داشته باشیم.

پیشنهاد می شود در آینده تحقیقات و توسعه بیشتری روی مسئله عامل کاوی صورت بگیرد زیرا این موضوع ارزش صرف وقت و هزینه را دارد و می تواند موجب پیشرفت شایانی در علم داده کاوی گردد.

۱۲ - مراجع

- 1- Sreenivasa Rao V. 2010. Multi Agent-based Distributed Data Mining: an over view. (*IJRIC*) International journal of Reviews in Computing, 3: 83-92.
- 2- Klusch M., Lodi S. and Moro G. 2003. Agent-based distributed data mining: The KDEC scheme. *AgentLink*, Springer Lecture Notes in Computer Science, 2586.
- 3- Costa da Silva J., Klusch M., Lodi S. and Moro G. 2006. Privacy-preserving agent-based distributed data clustering, *Web Intelligence and Agent Systems*, 4(2): 221-238.
- 4- Leonidas Prodromidis A. 1999. Management of intelligent learning agents in distributed data mining systems. PhD thesis, Columbia University, New York, NY, USA.
- 5- Kargupta H., Hamzaoglu I. and Stafford. Scalable B. 1997. Distributed data mining using an agent based architecture. Proceedings the Third International Conference on the Knowledge Discovery and Data Mining, AAAI Press, Menlo Park, California, 211-214.
- 6- Klusch M., Lodi S. and Moro G. 2003. The Role of Agents in Distributed Data Mining Issues and Benefit. *Intelligent Agent Technology, IAT, IEEE/WIC International Conference*: 211 – 217.
- 7- Sreenivasa Rao V., Vidyavathi S. 2010. Distributed Data Mining and Mining Multi-Agent Data. (*IJCSE*) International Journal on Computer Science and Engineering, 2(4): 1237-1244
- 8- Panait L., Luke S. 2005. Cooperative Multi-Agent Learning: The State of the Art. *Autonomous Agents and Multi-Agent Systems*, 11(3):387–434.
- 9- Davies W. and Edwards P. 1995. Distributed Learning: An Agent-Based Approach to Data-Mining. Proceedings of Machine Learning 95 Workshop on Agents that Learn from Other Agents.
- 10- Weiss G. 1998. A multiagent perspective of parallel and distributed machine learning. *AGENTS '98 Proceedings of the second international conference on Autonomous agents*, 226–230.
- 11- Dasilva J., Giannella C., Bhargava R., Kargupta H. and Klusch M. 2005. Distributed data mining and agents. *Engineering Applications of Artificial Intelligence*, 18(7):791–807, 2005.
- 12- Gorodetsky V., Karsaev O. and Samoilov V. 2003. Multi-agent technology for distributed data mining and classification. *IAT, IEEE Computer Society*, 438–441.
- 13- Gorodetsky V., Karsaev O. and Samoilov V. 2005. Multi-Agent Data and Information Fusion. *Nato Science Series Sub Series Iii Computer And Systems Sciences*, 198:308.
- 14- Gorodetsky V., Karsaev O. and Samoilov V. 2006. Infrastructural Issues for Agent-Based Distributed Learning. *Proceedings of the 2006 IEEE/WIC/ACM international conference on Web Intelligence and Intelligent Agent Technology, IEEE Computer Society Washington, DC, USA*, 3–6.
- 15- Tozicka J., Rovatsos M. and Pechoucek M. 2007. A framework for agent-based distributed machine learning and data mining. *Proceedings of the 6th international joint conference on autonomous agents and multiagent systems. ACM New York, NY, USA*.
- 16- Cao L., Ni J., Wang J. and Zhang C. 2004. Agent Services-Driven Plug and Play in the F-TRADE. *The 17th Australian Joint Conference on Artificial Intelligence, Springer*, 3339:917–922.
- 17- Moemeng P., Cao L. and Zhang C. 2008. F-TRADE 3.0: An Agent-Based Integrated Framework for Data Mining Experiments. *Proceedings of IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology, Los Alamitos, CA, USA, IEEE Computer Society*, 3: 612–615.

- 18- Sreenivasa Rao V., Vidyavathi S. and Ramaswamy G. 2010. Distributed Data Mining and Agent Mining Interaction and Integration: a Novel Approach. (*IJRAS*) International Journal of Research and Reviews in Applied Sciences, 4(4): 388-398.
- 19- Abraham A., Grosan C. and Ramos V., editors. 2006. Swarm Intelligence in Data Mining. The Studies in Computational Intelligence, Springer, 34.
- 20- Van Dyke Parunak H., Brueckner Sven A. 2005. Swarming distributed pattern detection and classification. Environments for Multi-Agent Systems, Springer, 232–245.

This document was created with Win2PDF available at <http://www.daneprairie.com>.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.