استنتاج توزیع شده برروی آنتولوژی‌ها و قوانین در منطق order-sorted

ملیحه امینی[۱]، محمود نقیب زاده[۲]، سید هاشم محتشمی[۳]

۱- کارشناسی ارشد کامپیوتر، دانشگاه آزاد اسلامی- واحد مشهد- باشگاه پژوهشگران جوان- مشهد- ایران
m.amini@mshdiau.ac.ir
۲- گروه مهندسی کامپیوتر، دانشگاه فردوسی مشهد- مشهد- ایران
naghibzadeh@um.ac.ir
۳- دانشجوی کارشناسی ارشد کامپیوتر هوش مصنوعی، دانشگاه آزاد اسلامی واحد مشهد- مشهد- ایران
s.h.mohtashami@mshdiau.ac.ir

چکیده: عامل‌های هوشمند به عنوان وسایل برجسته‌ای در راستای تحقق بخشیدن به نگرش وب معنایی در نظر گرفته می‌شوند. این عاملها با یکپارچه‌سازی آنتولوژی‌ها و قوانین در وب معنایی، می‌توانند با یکدیگر تعامل و همکاری داشته باشند، اما مشکل اساسی، نبود امکان استنتاج توزیع‌شده کارا برای یکپارچه‌سازی آنتولوژی‌ها و قوانین در چندین پایگاه دانش است. روش‌های ارائه شده برای استنتاج توزیع‌شده، اغلب بسیار زمانبر و حافظه‌بر بوده، به استنتاج کامل و دقیقی منجر نمی‌شوند. در این مقاله ما برای رفع این مشکل سیستم استنتاج توزیع شده‌ای را پیشنهاد می‌کنیم که بر اساس منطق order-sorted به ارائه دانش می‌پردازد. طرح پیشنهادی برای استنتاج توزیع شده از انتشار صفات دائمی و غیردائمی معتبر بین پایگاه‌های دانش استفاده می‌کند. همچنین با در نظر گرفتن زمان و موقعیت صفات برای استنتاج، صفات غیردائمی معتبر بلا استفاده نمی‌مانند و از آنها در زمان و موقعیت معتبرشان برای استنتاج توزیع شده استفاده می‌شود. بنابراین، علاوه بر استنتاج کامل، دانش به دست آمده در پایگاه‌های دانش توزیع شده به طور کامل در نظر گرفته شده و استنتاج توزیع‌شده‌ای با کارایی بالا، دقیق و بدون از دست دادن دانش خواهیم داشت.

واژه‌های کلیدی: آنتولوژی‌ها و قوانین، استنتاج توزیع شده، دائمی بودن ویژگی، منطق Order Sorted.

۱- مقدمه

هدف وب معنایی گسترده نمودن شبکه ارتباطی جهانی وب، با معناست تا از این طریق انسان‌ها و ماشین‌ها بتوانند بهتر محتوای وب را درک کنند و به نیازمندی‌هایشان بهتر پاسخ داده شود. عامل‌های هوشمند به عنوان ابزار برجسته‌ای در راستای تحقق بخشیدن به نگرش وب معنایی در نظر گرفته می‌شوند. با به کارگیری عامل‌های هوشمند، برنامه‌ها قادر خواهند بود که بهتر، کاراتر و با حداقل مداخله بیشتر کارشان را انجام دهند. یکپارچه سازی تدریجی سیستم‌های چند عامله با تکنولوژی وب معنایی بر استفاده از وب در آینده تاثیر گذار خواهد بود ؛ نسل بعدی وب شامل گروه‌هایی از عامل‌های مرتبط با هم است که کارهای بسیار پیچیده‌ای را برای کاربرانشان انجام می‌دهند. عامل‌های هوشمند به علت قابلیت تعامل و همکاری با یکدیگر بسیار با تکنولوژی‌های معنایی سازگاری دارند. این عامل‌ها می‌توانند با یکپارچه‌سازی آنتولوژی‌ها و قوانین در وب معنایی، با یکدیگر همکاری و ارتباط داشته باشند زبان‌های نشانه‌گذاری مانند RDF [۱] ، OWL [۲] و RuleML [۳] برای مدل‌سازی آنتولوژی‌ها و قوانین در وب معنایی توسعه یافته‌اند.

[۱] تاریخ ارسال مقاله: ۱۳۹۰/۶/۲۸
تاریخ پذیرش مقاله: ۱۳۹۱/۲/۵
نام نویسنده مسئول: ملیحه امینی
نشانی نویسنده مسئول: ایران – مشهد – قاسم آباد
چهار راه استاد یوسفی– دانشکده مهندسی
دانشگاه آزاد اسلامی واحد مشهد

Archive of SID
www.SID.ir

منطق‌های توصیفی [۴] و Datalog مفاهیم رسمی و خدمات استنتاج تصمیم پذیری را برای OWL و RuleML فراهم کردند. در سال ۲۰۰۵ Horrocks و Petel-Schneider به طور خاص SWRL [۵] را که ترکیبی از OWL و RuleML بود، به منظور قادر ساختن عامل‌های نرم افزاری برای تعامل بین آنتولوژی‌ها و قوانین پیشنهاد کردند، اما به علت قدرت گویای SWRL، خدمات استنتاج آن تصمیم ناپذیر بودند و این یک مشکل بزرگ تلقی می‌شد [۶].

در سال ۲۰۰۳ برای اداره کردن آنتولوژی‌های توزیع شده، C-OWL استفاده شد [۷]. C-OWL آنتولوژی‌ها را محلی کرده، امکان نگاشت صریح بین دوآنتولوژی را می‌دهد، اما همان طور که می‌دانیم با اینکه عامل‌های نرم افزاری حقایق و قوانین خاص خود را در پایگاه‌های دانششان دارند، اما خدمات استنتاج توزیع شده برای یکپارچه سازی آنتولوژی‌ها و قوانین تاکنون توسعه نیافته است. برای برخورد با این مشکلات، مطالعه سرویس‌های استنتاج توزیع شده برای آنتولوژی‌ها و قوانین به منظور دستیابی به استنتاج تصمیم پذیر از طریق محاسبات موثر و کارا امری ضروری به نظر می‌رسد. به عبارت دیگر، با وجود توسعه استنتاج برای آنتولوژی‌ها و قوانین یکپارچه به استنتاج توزیع شده، تصمیم پذیری آن باید باقی بماند[۸]، در این تحقیق یک سیستم چند عاملی توزیع شده با قابلیت استنتاج توزیع شده با منطق order-sorted ارائه می‌دهیم که قابلیت کار بر روی آنتولوژی‌ها و قوانین را دارد و قادر به استنتاج توزیع شده بر روی آنهاست.

منطق order-sorted که بسطی از منطق مسندی مرتبه اول است، شامل چندین دسته و سلسله مراتبشان بوده که به آنها سلسله مراتب دسته گفته می‌شود. منطق order-sorted دارای سه مزیت اصلی است [۹]؛

۱. کاهش فضای جستجو با محدود کردن دامنه و محدوده‌های توابع، مسندها و متغیرها [۱۰]؛

۲. ارائه دانش ساختاری به وسیله دسته‌های مرتب جزئی [۱۱]؛

۳. تشخیص خطاهای دسته‌ها در فرمول‌های خوش ترتیب [۱۲].

در آنتولوژی رسمی، موجودیت‌ها اصولا به خواص، رویدادها، فرآیند ها، اشیا و قسمت‌ها تقسیم می‌شوند [۱۳]. روابط موجودیت‌ها نیز به صورت رسمی تعریف می‌شوند. آنتولوژی‌های سطح بالاتر که خواص را بر اساس دسته‌ای و دائمی بودن دسته بندی می‌کنند، توسط Guarino و Welty [۱۴] ارائه شده است.

در [۱۵] Serani، مشکل استنتاج با چندین آنتولوژی که با نگاشت مفهومی با هم ارتباط دارند، بررسی شده است. در این مقاله شیوه استنتاج توزیع شده‌ای ارائه شده است که استنتاج را از طریق ترکیب قطعات استنتاج محلی انجام می‌دهد، اما این روش استنتاج بسیار زمانبر و حافظه بر است. در [۱۶] Schlicht، یک شیوه تحلیل توزیع شده ارائه می‌کند که در آن مسأله را با تحلیل محلی و سپس انتشار تمام واقعیت‌ها بین استنتاج گرها حل می‌کند. این سیستم به هیچ وجه واقعیت‌ها را اعتبارسنجی نمی کند، لذا ممکن است داده‌ای که اعتبار خود را از دست داده نیز در استنتاج لحاظ شود و به استنتاج نادرستی منجر شود. در [۱۷] Kaneiwa، یک چهارچوب برای برنامه نویسی چندگانه منطق order-sorted ارائه کرده است که در آن یک سلسله-مراتب دسته در ارتباط با یک آنتولوژی است و از دسته بندی خواص برای تشخیص خواص دائمی از میان پایگاه‌های دانش استفاده شده است. در چندین پایگاه دانش هر پایگاه دانش می‌تواند اطلاعات خواص دائمی را از سایر پایگاه‌های دانش استخراج کند. این سیستم به علت نادیده گرفتن بسیاری از اطلاعات معتبر، استنتاج کامل و دقیقی ندارد.

در حوزه آنتولوژی مجرد در مقاله [۱۸] دسته‌بندی خواص بسیار با معنی ارائه شده است که در آن خواص موجودیت‌ها براساس فرموله‌سازی منطقی به دسته‌های دائمی/غیردائمی و دسته‌ای/غیردسته‌ای تقسیم می‌شوند. بر اساس تعریف فوق، یک مشخصه را دسته‌ای نامند اگر هر قسمت از موجودیت، آن مشخصه را نداشته باشد، بلکه کل موجودیت مشخصه فوق را دارا باشد. مانند مشخصه person(john) که در آن john به عنوان یک موجودیت دارای ویژگی person است، ولی هر قسمت از john این مشخصه را ندارد. و نیز یک مشخصه را غیر دسته‌ای نامند،

Archive of SID

www.SID.ir

اگر هر قسمت از موجودیت نیز آن مشخصه را داشته باشد، مانند مشخصه water(w_1) که در آن هر قسمت از water نیز مشخصه water را دارد. یک مشخصه را دائمی نامند اگر موجودیت آن برای همیشه آن مشخصه را داشته باشد و آن مشخصه برای تمام نمونه هایش صادق باشد. بر اساس این تعریف، مربوط به دسته‌ای و دائمی بودن، دسته بندی مشخصه‌های آنتولوژیک در شکل (۱) نشان داده شده است.

شکل (۱): دسته بندی مشخصه‌های آنتولوژیک [۱۸]

بر اساس شکل (۱) داریم:

- مشخصه ذاتی مشخصه‌ای است که دسته‌ای و دائمی باشد؛ یعنی موجودیت‌ها همیشه این مشخصه را دارند، مانند person(john).
- مشخصه غیرذاتی مشخصه‌ای است که دسته‌ای و غیردائمی باشد؛ یعنی بعضی از موجودیت‌ها ممکن است این مشخصه را در زمان‌ها یا موقعیت‌های مختلفی از دست بدهند به عبارتی دیگر، موجودیت‌ها این مشخصه‌ها را به عنوان نقش برای مدتی می‌گیرند، مانند student(john).
- مشخصه عمومی مشخصه‌ای است که غیر دسته‌ای و دائمی باشد؛ یعنی هر قسمت از موجودیت برای همیشه آن مشخصه را دارد، مانند water (w_1).
- مشخصه توصیفی مشخصه‌ای است که غیردسته‌ای وغیردائمی باشد؛ یعنی هر قسمت از موجودیت آن مشخصه را دارد، اما ممکن است این مشخصه را در زمان‌ها یا موقعیت‌های مختلفی از دست بدهد، مانند مشخصه red(r_1).

در این مقاله ما یک سیستم استنتاج چند عاملی توزیع شده ارائه می‌دهیم که بر منطق OS مبتنی بوده، قابلیت استنتاج کارا و دقیق را دارد. این سیستم استنتاج توزیع شده پیشنهادی، سیستمی متشکل از ماژول‌های مجزا (به نام عامل‌ها) و مجموعه‌ای از مسیرهای ارتباطی بین آنهاست. در این سیستم هر عامل به طور خود مختار به عنوان یک موجودیت مستقل حل مسأله عمل می‌کند. سیستم استنتاج توزیع شده پیشنهادی کاراست، چون هماهنگی مناسب بین عامل‌ها ایجاد نموده، ساختار ارتباطی مناسبی دارد و نیز تکنیک‌های استنتاج توزیع شده را بخوبی اجرا می‌نماید. در این صورت این سیستم استنتاج توزیع شده بسیار مؤثرتر و با کیفیت تر از سیستم متمرکز مشابه خود عمل استنتاج را انجام می‌دهد. در این سیستم پیشنهادی، اطلاعات و دانش موجود در پایگاه‌های دانش توزیع شده، در صورت معتبر بودن، به طور مؤثری در فرآیند استنتاج به کار می‌روند. در اینجا ما وابستگی‌های موقعیت و زمان مربوط به بعضی مشخصه‌های غیر دائمی را نیز در نظر گرفته، از این مشخصه‌ها تنها در بازه‌های زمانی و موقعیتی معتبرشان استفاده می‌کنیم. با این روش، بدون از دست دادن دانش مجاز و با حداکثر دقت به استنتاجی کامل و دقیقتر نسبت به سیستم‌های پیشین دست می‌یابیم.

این مقاله از بخش‌های زیر تشکیل شده است: در بخش ۲ دسته‌بندی مشخصه‌ها از لحاظ مفهومی و آنتولوژیک را بیان می‌کنیم و سپس مفاهیم دائمی بودن و وابستگی‌های زمان و موقعیت بیان می‌شوند. در بخش ۳ نحوه استنتاج در منطق OS با ویژگی‌های دائمی و غیر دائمی بیان می‌شود. در بخش ۴ نیز به ارائه الگوریتم پیشنهادی مان می‌پردازیم. بخش ۵ نیز به تعریف نحوه استخراج ویژگی‌های معتبر از پایگاه‌های دانش در الگوریتم پیشنهادی مان می‌پردازد. در بخش ۶ معماری سیستم و نحوه پیاده سازی آن به طور مختصر بیان می‌شود. درانتها نیز به نتیجه‌گیری و جمع‌بندی می‌پردازیم.

۲- دسته بندی مشخصه‌ها از لحاظ مفهومی و آنتولوژیک.

در این بخش ما از مفاهیم ویژه‌ای استفاده می‌کنیم تا وابستگی‌های مربوط به زمان، موقعیت یا موقعیت – زمان مربوط به مشخصه‌ها را براساس وجود مفهوم دائمی بودن [۱۴] مشخص کنیم.

در این تحقیق، ما بر رفتارهای وابسته به موقعیت و

Archive of SID

www.SID.ir

گذرای مشخصه‌ها تاکید می‌کنیم. وابستگی‌های مربوط به زمان، موقعیت یا موقعیت–زمان مربوط به مشخصه‌ها، از لحاظ مفهومی با محدود کردن یک کلاس با ساختار $\sum$ دسته‌ای $M = (W, \overrightarrow{w_0}, R_{Tim}, R_{Sit}, R, U, I)$ تعریف می‌شود. یعنی $M = (W, \overrightarrow{w_0}, R_{Tim}, R_{Sit}, R, U, I)$ شرایط آنتولوژیک به تفسیر مشخصه‌ها، تحت ضرب کارتزین $W = W_{Tim} \times W_{Sit}$ $W=W_{Tim}\times W_{Sit}$ اضافه می‌شود؛ به طور خاص، دو مجموعه W_{Tim} W_{Tim} و W_{Sit} از زمان‌ها و موقعیت‌ها و سه رابطه موجود R، R_{tim} و R_{sit} برروی $W\times W$ برای تعریف وجود دائمی بودن و وابستگی‌های زمان یا موقعیت مربوط به مشخصه‌ها به کار می‌رود. در مفاهیم ویژه‌ای، فرمول‌های وابسته به موقعیت–گذرا توسط سه نوع از عملگرهای کیفیتی با نمادهای ■ و ♦ (وابسته به موقعیت–گذرا) و □Tim و ◊Tim (گذرا) و □Sit و ◊Sit (وابسته به موقعیت) بخوبی توصیف می‌شوند. ما نیز برای بیان مفاهیم دائمی بودن و وابستگی به زمان و موقعیت از این عملگرهای وابسته به موقعیت، گذرا و وابسته به موقعیت– گذرا استفاده می‌کنیم تا نشان دهیم که فرمول F برای هر زمان، موقعیت یا جهان در دسترس و معتبر برقرار است. برای مثال، فرمول وابسته به موقعیت – گذرای زیر را در نظر بگیرید:

$\Box_{Tim} P_{male}(bob_{person})$

فرمول فوق بیان می‌کند که برای هر زمان در دسترس در جهان، Bob تا زمانی که زنده است، یک male person است.

۲-۱- دائمی بودن

از لحاظ مفهومی ما دائمی بودن مشخصه‌ها را که توسط دسته‌ها در ساختاردسته‌ای بیان می‌شود، تعریف می‌کنیم. فرض کنید یک نوع (یعنی دسته ذاتی) باشد، در این صورت دائمی بودن مربوط به انواع به صورت زیر تعریف می‌شود:

- برای همه جهان‌های ممکن $\overrightarrow{w_i}, \overrightarrow{w_j} \in W$ داریم: $I_{\overrightarrow{w_i}}(\tau) = I_{\overrightarrow{w_j}}(\tau)$.

این تعریف برای دائمی بودن مربوط به ثوابت و تابع هاست. حال فرض کنید c یک ثابت و f یک تابع باشد، لذا عبارت زیر برقرار است:

- برای هر جهان ممکن $\overrightarrow{w_i}, \overrightarrow{w_j} \in W$ داریم: $I_{\overrightarrow{w_i}}(c) = I_{\overrightarrow{w_j}}(c)$ و $I_{\overrightarrow{w_i}}(f) = I_{\overrightarrow{w_j}}(f)$

به هر حال، دامنه‌های ثابت واقع گرایانه نیست، چون ممکن است موجودیت‌های مختلفی در هر جهان ممکن وجود داشته باشند، اما وجودشان دائمی نیست، چون بالاخره روزی بر اثر عوامل طبیعی، مانند مرگ و ... از بین می‌روند. برای مثال: هر نمونه از مشخصه person روزی دیگر وجود نخواهد داشت، چون هر انسانی روزی خواهد مرد. به همین علت، ما از دامنه‌های متغیری استفاده می‌کنیم مانند $U_{\overrightarrow{w}}$ که مجموعه‌ای از موجودیت‌هاست که در یک جهان ممکن $\overrightarrow{w} =< tm, st >$ وجود دارند. دامنه‌های متغیر ما را قادر می‌سازند که حالتی را در نظر بگیریم که در آن $U_{\overrightarrow{w_i}}$ و $U_{\overrightarrow{w_j}}$ برای برخی جهان‌های ممکن $\overrightarrow{w_i}, \overrightarrow{w_j} \in W$ بر هم منطبق نشوند.

دائمی بودن مربوط به دسته ها، ثوابت و توابع با تامین شرایط وجود موجودیت‌ها مجدداً در زیر تعریف می‌شود:

تعریف: (وجود دائمی بودن)

فرض کنید $M = (W, \overrightarrow{w_0}, R_{Tim}, R_{Sit}, R, U, I)$ یک ساختار $\sum$ دسته‌ای باشد، در این صورت شرایط زیر برقرار است:

۱) برای هر ثابت c اگر $I_{\overrightarrow{w_i}}(c), I_{\overrightarrow{w_j}}(c) \in U_{\overrightarrow{w_i}} \cap U_{\overrightarrow{w_j}}$ برقرار باشد، در این صورت $I_{\overrightarrow{w_i}}(c)= I_{\overrightarrow{w_j}}(c)$

۲) به ازای هر تابع n تایی f، اگر داشته باشیم $d_1, \dots, d_n \in U_{\overrightarrow{w_i}} \cap U_{\overrightarrow{w_j}}$ و نیز اگر عبارت زیر درست باشد، $\left\{I_{\overrightarrow{w_i}}(f)(d_1, \dots, d_n), I_{\overrightarrow{w_j}}(f)(d_1, \dots, d_n)\right\} \subseteq U_{\overrightarrow{w_i}} \cap U_{\overrightarrow{w_j}}$

در این صورت داریم:

$I_{\overrightarrow{w_i}}(f)(d_1, \dots, d_n) = I_{\overrightarrow{w_j}}(f)(d_1, \dots, d_n)$.

۳) به ازای هر نوع τ و به ازای هر جهان $\overrightarrow{w}$، اگر داشته باشیم $d \in I_{\overrightarrow{w}}(\tau)$ و $< \overrightarrow{w}, \overrightarrow{w'} > \in R$، در این صورت خواهیم داشت که $d \in I_{\overrightarrow{w}}(\tau)$.

۴) به ازای هر دسته غیر دائمی σ و به ازای هر جهان $\overrightarrow{w}$، اگر $d \in I_{\overrightarrow{w}}(\sigma)$ باشد، در این صورت $\overrightarrow{w_i} \in W$ با شرایط $< \overrightarrow{w}, \overrightarrow{w_j} > \in R$ وجود خواهد داشت که در آن داریم: $d \notin I_{\overrightarrow{w_j}}(\sigma)$ و $d \in U_{\overrightarrow{w_j}}$.

Archive of SID

www.SID.ir

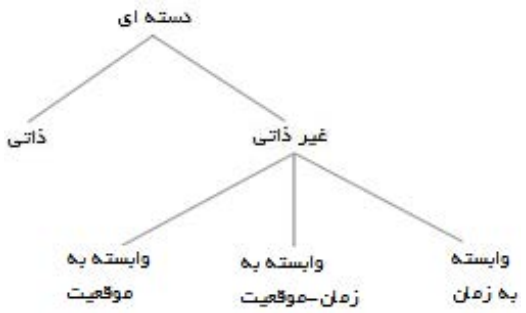
دقت شود ساختارهای دسته‌ای $\sum$ تضمین می‌کنند که هر $U_{\overrightarrow{W}}$ یک مجموعه‌ی غیر تهی از موجودیت‌هاست.

۲-۲- وابستگی‌های زمان و موقعیت

مفاهیم مربوط به وابستگی‌های زمان و موقعیت را به صورت زیر بیان می‌کنیم، در اینجا ما از کیفیت دو بعدی زمان و موقعیت برای تشخیص میان دسته‌های غیردائمی (به عنوان مشخصه‌های غیرذاتی) استفاده می‌کنیم. ما در اینجا چند نمونه از وابستگی‌های زمان، موقعیت و موقعیت- زمان را نشان می‌دهیم که در آن دسته‌های غیردائمی به صورت زیر دسته بندی می‌شوند:

- وابسته به زمان: elderly، adult، youth، child و baby
- وابسته به موقعیت: nurse، student و teacher
- وابسته به موقعیت - زمان: novice teacher

در شکل (۲) این مفاهیم به مشخصه‌های غیر ذاتی در دسته‌بندی آنتولوژیک مشخصه‌ها اضافه شده است. وابستگی به زمان بیان می‌کند که معنی یک مشخصه فقط بستگی به زمان آن دارد و یا ضرورتاً توسط زمان آن تعیین می‌شود. به عنوان مثال، مشخصه baby، وابسته به زمان است. لذا هر نمونه آن مشخصه مد نظر را فقط در زمان یا بازه خاصی دارد.



شکل (۲): وابستگی‌های زمان- موقعیت در دسته بندی آنتولوژیک مشخصه ها

وابستگی به موقعیت نشان می‌دهد که یک مشخصه برقرار می‌شود اگر یک موقعیت برقرار باشد، اما لزوما به برقراری زمان آن نیازی نیست.

در اینجا ما تعریف این وابستگی‌ها را از لحاظ مفهومی در جهان‌های ممکن بر روی ضرب کارتزین W_{Tim} و W_{Sit} بیان می‌کنیم. نکته اساسی در تفسیر وابستگی به زمان این است که برای مشخصه وابسته به زمان p و هر موجودیت $d \in U$، اگر $d \in I_{<tm,st>}(p)$ با شرایط $tm \in W_{Tim}$ درست باشد، در این صورت زمان دیگری به نام $tm_j \in W_{Tim}$ وجود دارد، به گونه‌ای که $d \notin I_{<tm_j,st>}(p)$.

این تعریف ساده بر مبنای دامنه‌های ثابت است که می‌توانند با در نظر گرفتن وجود موجودیت‌ها در هر جهان، تصحیح شود. در ادامه وابستگی‌های زمان، موقعیت، و موقعیت - زمان، موجودیت‌ها تحت رابطه $W \times W$ تعریف می‌شوند.

۲-۲-۱- وابستگی به زمان

تعریف: فرض کنید $M = (W, \overrightarrow{w_0}, R_{Tim}, R_{Sit}, R, U, I)$ یک ساختار $\sum$ دسته‌ای باشد. مسند یکانی p وابسته به زمان است اگر عبارت‌های زیر برقرار باشد:

۱) (وجود غیردائمی بودن گذرا)

به ازای همه مقادیر $\overrightarrow{w}, < st, tm >> \in R_{Tim}$ و همه مقادیر $d \in U_{<st,tm>}$ داریم:

اگر $d \in I_{<st,tm>}(p)$ باشد، لذا $tm_j \in W_{Tim}$ با شرایط $\ll st, tm >, < st, tm_j >> \in R_{Tim}$ وجود خواهد داشت به صورتی که $d \notin I_{<st,tm_j>}(p)$ با شرایط $d \in U_{<st,tm_j>}$.

۲) (وجود دائمی بودن وابسته به موقعیت در یک زمان)

به ازای همه مقادیر $\overrightarrow{w} \in W$ و همه مقادیر $tm \in W_{Tim}$ داریم:

اگر $d \in I_{<st,tm>}(p)$ با شرایط $< \overrightarrow{w}, < tm, st >> \in R_{Tim}$ باشد، در این صورت برای همه موقعیت‌ها $st' \in W_{Sit}$ با شرایط $\ll tm, st >, < tm, st' >> \in R_{Sit}$ و $d \in U_{<tm,st'>}$ بیانگر $d \in I_{<tm,st'>}(p)$ است.

وجود غیردائمی بودن گذرا بیان می‌کند که برای هر زمان tm در دسترس از جهان $\overrightarrow{w}$ اگر یک موجودیت d مشخصه p را در زمان tm داشته باشد، می‌توان زمان tm_j که از tm در دسترس است، یافت که موجودیت آن مشخصه را

Archive of SID

www.SID.ir

ندارد.

وجود دائمی بودن وابسته به موقعیت در یک زمان بیان می‌کند که بازای هر زمان tm در دسترس از جهان $\vec{w}$، اگر یک موجودیت d، مشخصه p را در زمان tm داشته باشد، در این صورت موجودیت این مشخصه را در هر موقعیت st که از زمان tm (در طول مدت حیات موجودیت) در دسترس است، داراست.

به صورت مشابه با وابستگی به زمان، وابستگی به موقعیت به شکل زیر تعریف می‌شود.

۲-۲-۲- وابستگی به موقعیت

تعریف

فرض کنید $M = (W, \overrightarrow{w_0}, R_{Tim}, R_{Sit}, R, U, I)$ یک ساختار $\sum$ دسته‌ای باشد. مسند یکانی p وابسته به موقعیت است اگر عبارت‌های زیر برقرار باشد:

۱) (وجود غیردائمی بودن وابسته به موقعیت)

به ازای همه مقادیر $< \vec{w}, < st, tm >> \in R_{Sit}$ و همه مقادیر $d \in U_{<st,tm>}$ داریم:

اگر $d \in I_{<st,tm>}(p)$ باشد، لذا $st_j \in W_{Sit}$ با شرایط $\ll st, tm >, < tm, st_j >> \in R_{Sit}$ وجود خواهد داشت بصورتیکه $d \notin I_{<tm,st_j>}(p)$ با شرایط $d \in U_{<tm,st_j>}$.

۲) (وجود دائمی بودن گذرا در یک موقعیت)

به ازای همه مقادیر $\vec{w} \in W$ و همه مقادیر $st \in W_{Sit}$ داریم:

- اگر $d \in I_{<tm,st>}(p)$ با شرایط $< \vec{w}, < tm, st >> \in R_{sit}$ باشد، در این صورت برای همه موقعیت‌ها $tm' \in W_{Tim}$ با شرایط $\ll tm, st >, < tm', st >> \in R_{Tim}$ و $d \in U_{<tm',st>}$ بیانگر $d \in I_{<tm',st>}(p)$ است.

علاوه بر آن، وابستگی‌های موقعیت - زمان به این صورت تعریف می‌شوند که معنا و مفهوم یک مشخصه به هر دوی زمان و موقعیت بستگی دارد. به عنوان مثال، مشخصه novice teacher یک مشخصه وابسته به موقعیت - زمان است لذا هر novice teacher پس از چند سال مشخص یک معلم کارآزموده خواهد شد لذا مشخصه novice teacher فقط در یک زمان مشخص و در موقعیت معینی برقرار است. به عبارت دیگر، وابستگی موقعیت - زمان بیانگر وابستگی زمان تحت یک شرایط و موقعیت خاصی است، در حالی که موقعیت آن هیچ ارتباطی با زمان آن ندارد. از لحاظ مفهومی با تغییرات دامنه ها، وابستگی موقعیت - زمان می‌تواند به عنوان یک وابستگی پیچیده به صورت زیر تعریف می‌شود.

۲-۲-۳- وابستگی به موقعیت - زمان

تعریف

$$M = (W, \overrightarrow{w_0}, R_{Tim}, R_{Sit}, R, U, I)$$

فرض کنید یک ساختار $\sum$ دسته‌ای باشد. مسند یکانی p وابسته به موقعیت - زمان است اگر عبارت‌های زیر برقرار باشد:

۱) (وجود غیردائمی بودن وابسته به موقعیت)

به ازای همه مقادیر $< \vec{w}, < st, tm >> \in R_{Sit}$ و همه مقادیر $d \in U_{<st,tm>}$ داریم:

اگر $d \in I_{<st,tm>}(p)$ باشد، لذا $st_j \in W_{Sit}$ با شرایط $\ll st, tm >, < tm, st_j >> \in R_{Sit}$ وجود خواهد داشت بصورتیکه $d \notin I_{<tm,st_j>}(p)$ با شرایط $d \in U_{<tm,st_j>}$.

۲) (وجود غیردائمی بودن گذرا در یک موقعیت)

به ازای همه مقادیر $\vec{w} \in W$ و همه مقادیر $st \in W_{Sit}$ داریم:

اگر $d \in I_{<tm,st>}(p)$ با شرایط $< \vec{w}, < tm, st >> \in R_{sit}$ باشد، در این صورت برخی از مقادیر $tm_i, tm_j \in W_{Tim}$ با شرایط $\ll tm, st >, < tm_i, st >, \ll tm, st >, < tm_j, st > \in R_{Tim}$ وجود دارند به قسمی که $d \in U_{<tm_i,st>}(p)$ و نیز $d \notin I_{<tm_j,st>}(p)$ با شرایط $d \in U_{<tm_i,st>} \cap U_{<tm_j,st>}$.

علاوه بر غیر دائمی بودن وابسته به موقعیت، غیر دائمی بودن گذرا در یک موقعیت بیان می‌کند که به ازای هر موقعیت st در دسترس از یک جهان $\vec{w}$ اگر یک موجودیت d مشخصه p را در موقعیت st داشته باشد، در این صورت زمان‌های tm_j و tm_i در دسترس از st وجود دارد؛ به

Archive of SID

www.SID.ir

گونه‌ای که موجودیت، مشخصه p را در زمان tm_i دارد، اما در زمان tm_j این مشخصه را ندارد. وابستگی موقعیت- زمان در ارتباط با واقعیت هایی است که وابستگی زمان و وابستگی موقعیت هیچ یک نمی توانند مشخصه‌های با این وابستگی را توصیف کنند (مانند novice teacher)ه به همین دلیل است که وابستگی موقعیت- زمان با هر دو وابستگی زمان و وابستگی موقعیت متفاوت است.

۳- استنتاج در منطق OS با ویژگی‌های دائمی و غیر دائمی

برای در نظرگرفتن عبارات دسته‌ای غیردائمی و دائمی در یک سیستم استنتاج با منطق OS، از قوانین جانشینی دسته‌ای و جانشینی غیردسته‌ای به صورت زیر استفاده می‌کنیم [۱۷]:

جانشینی دسته ای: جانشینی دسته‌ای یک تابع جزئی $\theta: V \to \Gamma$ است به قسمی که داشته باشیم $\theta(x: s) \in \Gamma_s - \{x: s\}$ و نیز $Dom(\theta)$ متناهی است.

مثال: عبارت $p(x: animal)\{x: animal/$ $john: person\}$ با شرایط $person \leq_+ animal$ درست است.

جانشینی غیردسته ای: جانشینی غیردسته‌ای یک تابع جزئی $\theta^u: V \to \Gamma$ است به قسمی که داشته باشیم $\theta^u(x: s) \in \Gamma_s - \{x: s\}$ و نیز $Dom(\theta^u)$ متناهی است.

مثال: عبارت $p(x: student)\{x: student/$ $john: person\}$ با شرایط $person \not\leq_+ student$ درست است.

۳-۱- حساب بند هورن دسته ای

ما در این تحقیق از حساب بند هورن دسته‌ای که برای مسندهای دسته‌ای و جانشینی‌های غیردسته‌ای توسعه یافته است [۱۷] استفاده می‌کنیم.

در حساب بند هورن دسته‌ای، بندهای هورن را با نماد C_i و واقعیت‌ها را به صورت $L \leftarrow := p(\vec{t})$ و قوانین را به صورت $L \leftarrow G := p(\vec{t}) \leftarrow \{p_1(\vec{t_1}), \dots, p_n(\vec{t_n})\}$ نشان می‌دهند. پایگاه‌های دانش نیز مجموعه‌ای معین از بندهای هورن بصورت $K = \{C_1, \dots, C_n\}$ است.

قوانین استنتاج دسته‌ای استاندارد به صورت زیر است:

- قانون جانشینی دسته ای: اگر $L \leftarrow G \in K$ باشد، در این صورت داریم: $K \vdash (L \leftarrow G)\theta$
- قانون برش: اگر $K \vdash L \leftarrow G$ و $K \vdash L' \leftarrow G' \cup \{L\}$ باشد، دراین صورت داریم $K \vdash L' \leftarrow G \cup G'$

قوانین استنتاج دسته‌ای برای زیردسته‌ها و مسندهای دسته‌ای به صورت زیر است:

- قانون زیردسته: اگر $K \vdash s(t) \leftarrow G$ و $s <_+ s'$ باشد، در این صورت داریم $K \vdash s'(t) \leftarrow G$
- قانون مسند نوعی: اگر $sort(t) \leq_+ \tau$ باشد، دراینصورت داریم: $K \vdash \tau(t) \leftarrow$. که در آن $sort(t)$ بیانگر دسته t است.

مثال: پایگاه دانش $K = \{canary\ (peter: animal) \leftarrow\}$ با شرایط $canary \leq_+ bird$ ، $bird \leq_+ animal$، $person \leq_+ animal$ ، $peter: \to animal$ و $john: \to person \in \Omega$ را در نظر بگیرید. در این صورت استنتاج‌های زیر طبق قوانین فوق درست است.

$$\frac{canary(peter: animal) \leftarrow}{bird(peter: animal) \leftarrow}\ (subsort)$$

$$\frac{bird(peter: animal) \leftarrow}{animal(john: person) \leftarrow}\ (type\ predicate)$$

قوانین استنتاج غیر دسته‌ای نیز به صورت زیر است:

- قانون مسند نوعی غیردسته ای: اگر داشته باشیم $sort(t) \leq_+ \tau$ و نیز عبارت زیر برقرار باشد:

$K \vdash s_1(t_1) \leftarrow G_1, \dots, K \vdash s_n(t_n) \leftarrow G_n$ ، در این صورت داریم:

$$K \vdash \tau(t)\{x_1: s_1/t_1, \dots, x_n: s_n/t_n\} \leftarrow G_1 \cup \dots \cup G_n$$

که در آن $EVar(t) = \{x_1: s_1, \dots, x_n: s_n\}$ برقرار است.

- قانون جانشینی غیردسته‌ای: اگر داشته باشیم $L \leftarrow G \in K$ و $K \vdash s_1(t_1) \leftarrow G_1, \dots, K \vdash s_n(t_n) \leftarrow G_n$

لذا داریم:

$K \vdash (L \leftarrow G \cup G_1 \cup \dots \cup G_n)\{x_1: s_1/t_1, \dots, x_n: s_n/t_n\}\theta$

که در آن داریم:

Archive of SID

www.SID.ir

$\{x_1 : s_1, \dots, x_n : s_n\} \subseteq EVar(L \leftarrow G)$.

مثال: پایگاه دانش

$K = \{student(john:person) \leftarrow, getting-a-scholarship(x:student) \leftarrow \{excellent(x:student)\}\}$

که در آن داریم:

$person \not\leq_+ student \ ، \ student \in N \ ، \ person \in T$

و

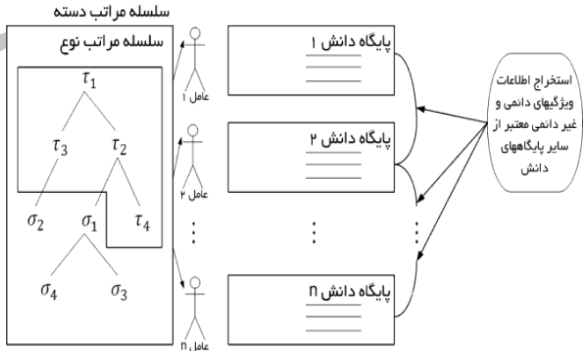
$getting-a-scholarship: student \in \Omega$ ، را در نظر بگیرید. استنتاج زیر برای این پایگاه دانش برقرار است:

$$\frac{student(john:person) \leftarrow, getting-a-scholarship(x:student) \leftarrow \{excellent(x:student)\}}{getting-a-scholarship(john:person) \leftarrow \{excellent(john:person)\}}$$

که با جانشینی غیر دسته‌ای $\theta^u = \{x: student/john: person\}$ می‌باشد.

۴- الگوریتم پیشنهادی

استنتاج چند عامله‌ای با چندین پایگاه دانش را که به چندین عامل نسبت داده شده‌اند، مطابق شکل (۳) در نظر بگیرید.



شکل (۳): نمای گرافیکی الگوریتم پیشنهادی

در این حالت هر عامل پایگاه دانش مخصوص به خود را دارد که با منطق OS بیان می‌شود. همچنین، همه عامل‌ها به یک پایگاه دانش طبقه‌بندی شده اشتراکی که در حقیقت سلسله مراتبی مفهومی از دسته‌هاست، دسترسی دارد؛ یعنی کلمات، واقعیت‌ها و قوانین در هر پایگاه دانش بیانگر دانش اضافی در هر عامل هستند. به عبارت دیگر، یک سلسله مراتب دسته بیان کننده دانش سلسله مراتبی است که اغلب در وضعیت‌های مختلف به کار می‌رود. در سیستم استنتاج چند عاملی پیشنهادی ما ابتدا عامل‌ها به طور موازی بر روی پرس و جوی مورد نظر شروع به استنتاج بر اساس قوانین استنتاج معرفی شده در بخش قبل، می‌کنند وهر عامل بر روی پرس و جوی مورد نظر با استفاده از دانش موجود در پایگاه دانش خود و نیز دانش سلسله مراتبی که در سلسله مراتب دسته وجود دارد، عمل استنتاج را انجام داده، به واقعیت‌های جدیدی دست می‌یابد. واقعیت‌های به دست آمده پس از بررسی در صورت مجاز بودن از لحاظ زمان و موقعیت به سایر عامل‌ها منتشر می‌شود. این روال استنتاج تا زمانی ادامه می‌یابد که سیستم استنتاج توزیع شده به پرس و جوی مورد نظر پاسخ گوید ویا اینکه تعداد مشخصی تکرار صورت گیرد، ولی برای پرس وجوی مورد نظر جوابی پیدا نگردد، در حالت دوم گفته می‌شود که پرس و جوی مورد نظر جوابی ندارد.

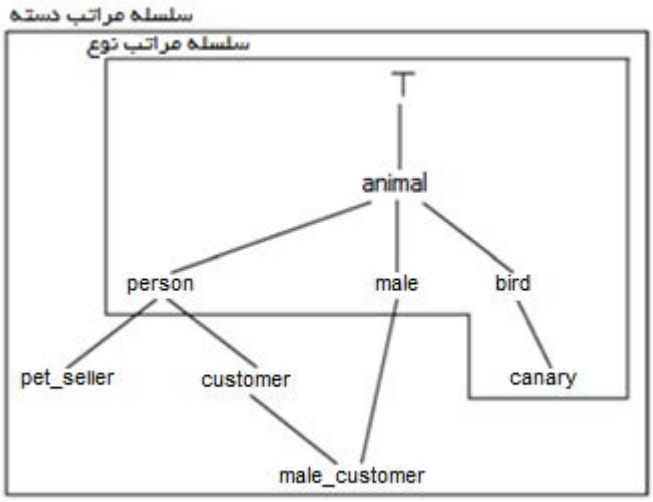
همان طور که در شکل (۳) مشاهده می‌شود، در الگوریتم پیشنهادی ما هر عامل پس از هر مرحله استنتاج، اطلاعات و واقعیات به دست آمده‌اش را بررسی نموده، اطلاعات ویژگی‌های معتبر را برای سایر پایگاه‌های دانش منتشر می‌کند تا در مرحله بعدی استنتاج مورد استفاده آنها قرار گیرد؛ یعنی عامل‌ها علاوه بر اطلاعات ویژگی‌های دائمی، اطلاعات ویژگی‌های غیردائمی و وابسته به زمان، موقعیت و حتی وابسته به موقعیت- زمان را بر اساس مفاهیم و اصولی که در بخش قبل ذکر گردید، اعتبارسنجی می‌کنند و در صورت معتبر و مجاز بودن، آنها را نیز برای سایر پایگاه‌های دانش ارسال می‌کنند. شایان ذکر است که این مراحل اعتبارسنجی در سمت سایر عامل‌ها هنگام دریافت اطلاعات نیز صورت می‌گیرد و تنها در صورتی که هنوز اعتبار لازم را داشته باشند، در پایگاه دانششان ذخیره می‌گردند و در استنتاج توزیع شده استفاده می‌شود.

نحوه عملکرد سیستم پیشنهادی مان را بر روی یک مثال نشان داده، نحوه استنتاج آن را شرح می‌دهیم.

سیستم چند عامله‌ای با سلسله مراتب دسته و نوع مطابق با شکل (۴) درنظر بگیرید.

Archive of SID

www.SID.ir



شکل (۴): سلسله مراتب دسته و نوع

این یک سیستم استنتاج توزیع شده با چهار عامل است که ارتباط عامل‌ها با هم و با سلسله مراتب دسته همانند شکل (۳) مربوط به الگوریتم پیشنهادی است. پایگاه‌های دانش مرتبط با عامل‌ها در این سیستم استنتاج توزیع شده به صورت جدول (۱): تعریف شده‌اند.

جدول (۱): پایگاه‌های دانش مرتبط با عامل‌ها

Knowledge base 1	(1a) male customer(john: person),
	(1b) excellent(john: person),
	(1c) (∀x: customer)(excellent(x: customer) ⇒ obtaining a discount(x: customer))
Knowledge base 2:	(2a) pet seller(mary: person),
	(2b) (∀x: customer)(cares about(mary: person, x: customer)),
	(2c) (∀x: bird)(cares about(mary: person, x: bird))
Knowledge base 3:	(3a) canary(peter: animal)
	(3b) (∀x: animal)(bird(x: animal) ⇒ canfly(x: animal))
Knowledge base 4:	(4a) father (tony: animal, peter: animal),
	(4b) (∀y : animal) (∀x: animal) (father (y : animal, x: animal) ∧ bird(x: animal) ⇒ bird(y : animal))

واقعیت (3a) فقط در پایگاه دانش ۳ درست است و با فرض درست بودن canary<bird که در سلسله مراتب دسته موجود است، می‌توان عبارت bird(peter: animal) را در پایگاه دانش ۳ استخراج کرد. براساس مفهوم دائمی بودن که قبلا شرح داده شد، این عبارت در هر پایگاه دانش دیگری نیز درست است، چون اعتبارسنجی نوع bird بستگی به موقعیت خاصی ندارد و اصولاً یک مشخصه دائمی است. به عبارت دیگر، عناصر نوع bird می‌توانند به سایر پایگاه‌های دانش منتشر شوند (peter در هر صورت یک bird است). از این رو، ما استنتاج می‌کنیم که واقعیت bird(peter:animal) در پایگاه‌های دانش ۱ و ۲ و ۴ نیز درست است. بر مبنای این اطلاعات اضافی، پایگاه دانش ۲ یک واقعیت جدید cares-about(mary:person,peter:animal) را از (2c) استنتاج می‌کند که این واقعیت بدون انتشار اطلاعات مشخصه‌های دائمی به دست نمی‌آمد. علاوه بر آن، از واقعیت father(tony:animal,peter:animal) در پایگاه دانش ۴ نمی‌توان به صورت نرمال و عادی استنتاج کرد که tony یک bird است. با ارسال واقعیت bird(peter:animal) می‌توان با استفاده از پایگاه‌دانش ۴ واقعیت bird(tony:animal) (یعنی tony یک bird است) را از واقعیت (4a) و قانون (4b) استنتاج کرد. بنابراین، واقعیت

Archive of SID

www.SID.ir

دائمی bird(tony:animal) به صورت بازگشتی در canfly(tony,animal) از پایگاه دانش ۳ و واقعیت cares-about(mary:person,tony:animal) از پایگاه دانش ۲ اثر می‌گذارد.

بر اساس سیستم استنتاج توزیع شده معرفی شده در مقاله [۱۷] داریم: زمانی که واقعیت (1a) و استلزام male- customer < customer درست باشد، customer(john: person) فقط در پایگاه دانش ۱ درست است، چون در این مقاله فرض شده که فقط اطلاعات ویژگی‌های دائمی بین پایگاه‌های دانش منتشر می‌شوند و اطلاعات ویژگی‌های غیر دائمی، حتی آنهایی که معتبر هستند، نادیده گرفته می‌شوند. از آنجایی که customer از دسته غیردائمی است، لذا واقعیت (2a) در پایگاه دانش ۲ به واقعیت cares-about (mary: person,john: customer) در پایگاه دانش ۳ منجر نمی‌شود، اما این یک استنتاج کامل و دقیق نیست، زیرا واقعیت customer ممکن است در زمان فعلی درست باشد، و نادیده گرفتن این واقعیت مهم به استنتاج ناکامل منجر می‌گردد.

برای رفع این مشکل بزرگ یعنی ناکامل و غیر دقیق بودن نتایج استنتاج توزیع شده در سیستم‌های پیشین، الگوریتم فوق برای استنتاج توزیع شده درمیان چندین پایگاه دانش ارائه گردید. آزمایش‌ها و نتایج تجربی به دست آمده نشان می‌دهند که سیستم پیشنهادی ما، بسیار دقیقتر و کاملتر از سیستم‌های مشابه پیشین عمل می‌کند. حال مثال فوق را با سیستم پیشنهادی استنتاج می‌کنیم. در سیستم پیشنهادی زمانی که واقعیت (1a) و استلزام male- customer < customer درست باشد، لذا واقعیت customer(john: person) در پایگاه دانش ۱ از عمل استنتاج محلی در عامل ۱ بدست می‌آید، حال چون این واقعیت (customer بودن john:person) یک ویژگی غیردائمی و وابسته به زمان است، عامل ۱ بازه زمانی معتبر برای این ویژگی که در اطلاعات آن موجود است را بررسی نموده، در صورت معتبر و مجاز بودن، آن را نیز به پایگاه‌های دانش ۱و ۳ و۴ می‌فرستد. در پایگاه دانش ۲ نیز به محض دریافت این واقعیت، ابتدا آن را اعتبارسنجی کرده، سپس در صورت معتبر بودن آن را به پایگاه دانشش می‌افزاید. اگر واقعیت ارسالی معتبر باشد، در پایگاه دانش ۲ به همراه واقعیت (2b) به واقعیت cares-about(mary:person,john:customer) در پایگاه دانش ۳ منجر می‌شود، حال آنکه این واقعیت بر اساس سیستم‌های استنتاج قبلی بدست نمی آمد.

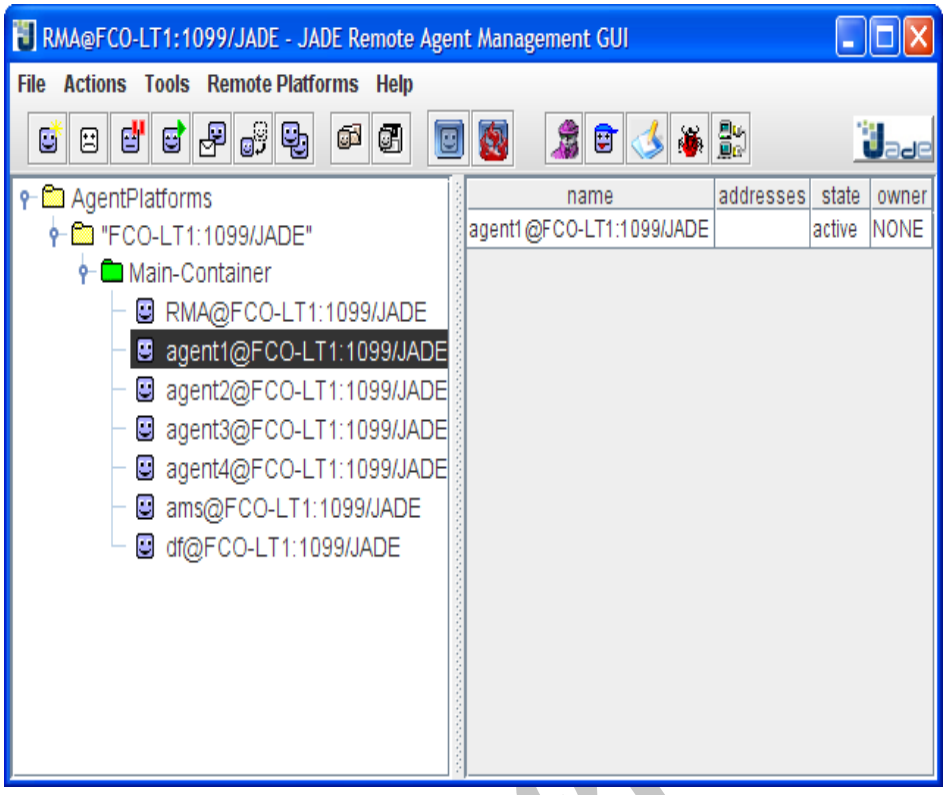
۵- معماری و پیاده سازی سیستم پیشنهادی

هر چند JADE [۱۱ و ۱۲] یک ابزار توسعه سیستم‌های توزیع‌شده چندعاملی است که با Java [۲۱] پیاده‌سازی شده و منطبق با مشخصات FIPA است، اما توانایی داشتن رفتار هوشمند برای هر یک از عامل‌هایش را ندارد. موتور استنتاج JESS [۲۲] که یک محیط برنامه‌نویسی مبتنی بر قوانین و مبتنی بر Java است، ابزاری قدرتمند برای توسعه سیستم‌هایی با قابلیت استنتاج هوشمند دارد. در این تحقیق با یکپارچه سازی JADE و JESS بر مبنای پلتفرم Java ، سیستم توزیع‌شده چندعاملی هوشمندی را که مبتنی بر منطق Order-sorted است، طراحی و پیاده‌سازی کرده‌ایم.

عامل‌های طراحی شده برای این سیستم توزیع شده در شکل (۵) به همراه اجزاء و عامل‌های اصلی JADE در واسط کاربری گرافیکی RMA نشان داده شده است [۱۹]. این عامل‌ها هر یک دارای پایگاه دانش مخصوص به خود هستند. ارتباطات بین این عامل‌ها بر اساس ویژگی‌های FIPA از طریق پیام ACL صورت می‌گیرد [۲۰].

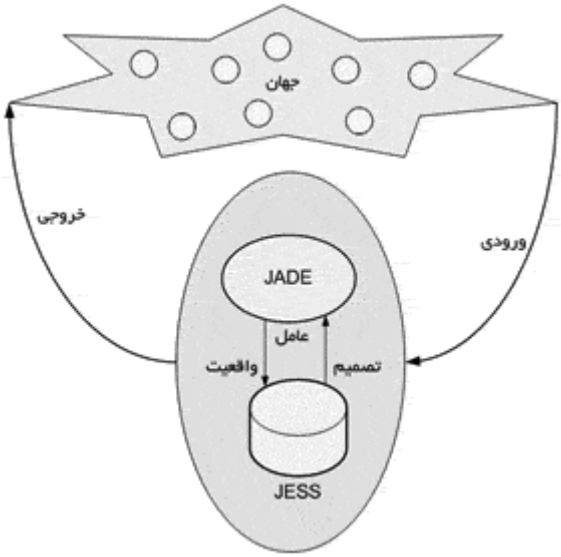
Archive of SID

www.SID.ir



شکل (۵): واسط کاربری گرافیکی JADE برای چهار عامل طراحی شده

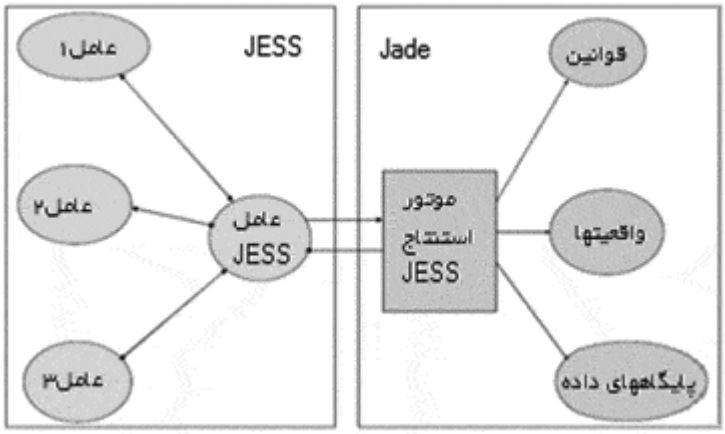
شکل (۶) اجرای قوانین عامل JADE را با به کارگیری JESS نشان می‌دهد. همان طور که در شکل مشاهده می‌شود، ورودی‌های سیستم که همان پرس و جوی مورد نظر است، از جهان خارج توسط عامل JADE دریافت می‌شود. این عامل از طریق موتور استنتاج JESS تصمیم‌گیری نموده، سپس خروجی به جهان خارج ارسال می‌گردد.



شکل (۶): اجرای قوانین عامل JADE با به کارگیری JESS

Archive of SID

www.SID.ir



شکل (۷): نحوه ارتباط عامل JADE با موتور استنتاج JESS

در شکل (۷) نحوه ارتباط عامل JADE را با موتور استنتاج JESS نشان می‌دهد. همان طور که مشاهده می‌شود، تمامی عامل‌ها با یک JESS Agent در ارتباط هستند که به موتور استنتاج JESS متصل بوده، از این طریق فرآیند تصمیم گیری صورت می‌گیرد.

نتیجه گیری

در این مقاله ما یک سیستم استنتاج توزیع شده ارائه دادیم که بر اساس منطق order-sorted به ارائه دانش می‌پرداخت. این منطق قادر به توصیف سلسله مراتب گزاره‌ها و ارثبری عباراتی است که در زبان طبیعی وجود دارند. این منطق مزایای زیادی از جمله کاهش فضای جستجو را دارد. سیستم استنتاج توزیع شده پیشنهادی بسیار مقیاس پذیر بوده، علاوه بر داشتن امکان استفاده از الگوریتم‌های استنتاج و زبان‌های گوناگون، امکان مخفی سازی و خود مختاری را نیز به عامل‌ها می‌دهد. طرح پیشنهادی برای استنتاج توزیع شده از انتشار صفات دائمی و غیردائمی معتبر بین پایگاه‌های دانش استفاده می‌کرد. همچنین، با در نظر گرفتن زمان و موقعیت صفات برای استنتاج، صفات غیردائمی معتبر بلااستفاده نمی‌ماند و از آنها در زمان و موقعیت معتبرشان برای استنتاج توزیع شده استفاده می‌شد. با این روش، علاوه بر استنتاج کامل، دانش به دست آمده در پایگاه‌های دانش توزیع شده به طور کامل در نظر گرفته می‌شد و استنتاج توزیع شده‌ای با کارایی بالا، دقیق و بدون از دست دادن دانش خواهیم داشت.

نتایج آزمایش‌های تجربی نشان داد که سیستم پیشنهادی مان به بیش از ۷۵٪ از پرس و جوهایی که در سیستم‌های پیشین از جمله سیستم ارائه شده در [۱۷] پاسخ نمی دادند، پاسخ می‌دهد. آزمایشات مربوط به صحت عمل استنتاج نیز بر روی این سیستم صورت گرفت و در تمامی موارد سیستم به طور دقیق و بدون خطایی استنتاج را انجام داد. تنها عیوبی که در سیستم پیشنهادی مشاهده شد، این بود که اولا زمان اجرای این سیستم نسبت به سیستم قبلی [۱۷] به طور متوسط ۲ ثانیه بیشتر بود که البته این مسأله نسبت به دقت و کارایی دقیق آن قابل اغماض است و ثانیا در سیستم پیشنهادی ما و نیز سیستم‌های پیشین مانند [۱۷] هیچ گونه بررسی برای حالات تعارض احتمالی بین استنتاج پایگاه‌های دانش مختلف صورت نگرفته است.

در کارهای آینده تلاش می‌کنیم تا زمان اجرای سیستم را به حداقل رسانده، حالات تعارض احتمالی بین پایگاه‌های دانش را مورد بررسی کرده و نیز سیستم پیشنهادی مان را بر روی یک کاربرد پیاده‌سازی نماییم.

مراجع

[1] Lassila, O. and Swick, R., "Resource description framework (rdf) model and syntax specification", 1998.

[2] Patel-Schneider, P.F., Hayes, P. and Horrocks, I. "OWL Web Ontology Language Semantics and Abstract Syntax",

Archive of SID

www.SID.ir

W3C Recommendation.http://www.w3.org/tr/2004/rec-owl- semantics-20040210/.
[3] Boley, H. and Tabet, S., “The Rule Markup Initiative” http://www.ruleml.org/.
[4] Baader, F., Calvanese, D., McGuinness, D., Nardi, D. and Patel-Schneider P.F. (Eds.) "The Description Logic Handbook: Theory, Implementation, and Applications", Cambridge, 2nd ed. 2007.
[5] Horrocks, I., Patel-Schneider, P.F., Boley, H., Tabet, S., Grosof, B. and Dean, M. "SWRL: A SemanticWeb Rule Language Combining OWL and Rule ML, W3C Recommendation" .http://www.w3.org/submission/swrl/
[6] Horrocks, I. and Patel-Schneider, P.F. "A proposal for an owl rules language", Proceedings of the Thirteenth International World Wide Web Conference (WWW 2004), ACM, 2004, pp. 723–731.
[7] Bouquet, P., Giunchiglia, F., Harmelen, F., Serafini, L. and Stuckenschmidt, H. "C-WL: contextualizing ontologies", Proceedings of the 2nd International Semantic Web Conference (ISWC 2003), 2003, pp. 164-179.
[8] Kaneiwa, K. "The completeness of logic programming with sort predicates", Systems and Computers in Japan, 35: 37-46. DOI: 10.1002/scj.10409. 2004.
[9] Beierle, C., Hedtsuck, U., Pletat, U., Schmitt, P.H., Siekmann, J. "An order-sorted logic for knowledge representation systems", Artificial Intelligence, 55:149–191, 1992.
[10] Walther, C. "A mechanical solution of Schubert’s steamroller by many-sorted resolution", Artificial Intelligence 26 (2) 217–224, 1985.
[11] Cohn, A.G., "Taxonomic reasoning with many sorted logics", Artificial Intelligence Review 3, 89–128, 1989.
[12] Oberschelp, A., "Order sorted predicate logic", Workshop on Sorts and Types in Artificial Intelligence, 1989.
[13] Smith, B., "The Basic Tools of Formal Ontology ", Formal Ontology in Information Systems, 19-28, 1998.
[14] Guarino, N. and Welty, C. "Ontological analysis of taxonomic relationships", Proceedings of ER-2000: The Conference on Conceptual Modeling, 210-224, 2000.
[15] Sera_ni, L. and Tamilin, A., "DRAGO: Distributed Reasoning Architecture for the Semantic Web", Proceedings of the European Semantic Web Conference (ESWC), 361-376. 2005.
[16] Schlicht, A. and Stuckenschmidt, H., "Towards Distributed Ontology Reasoning for the Web", Web Intelligence and Intelligent Agent Technology, vol 1, 536-539, 2007.
[17] Kaneiwa, K. and Riichiro, M., "Distributed reasoning with ontologies and rules in order-sorted logic programming", Web Semantics: Science, Services and Agents on the World Wide Web 7, 252–270, 2009.
[18] Guarino, N., Carrara, M. and Giaretta, P., "An ontology of meta-level categories", Proc. of the 4th Int. Conf. on the Principles of Knowledge Representation and Reasoning, 270–280, 1994.
[19] Bellifemine, F., Caire, G. and Greenwood, D., "Developing multi-agent systems with JADE" John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester , 2007.
[20] Bellifemine, F., Poggi, A. and Rimassa, G., "Developing Multi-agent Systems with JADE", Intelligent Agents VII, LNAI 1986, 89–103, 2001.
[21] Horstmann, S. and Cornell, G., "Core Java", Volume 1, Fundumental, eight edition, 2007.
[22] FRIEDMAN-HILL, E., "Jess in Action", Rule-Based Systems in Java, MANNING, 2003.

Archive of SID

www.SID.ir