

ارایه یک بستر مبتنی بر عامل برای توسعه سیستم‌های خبره توزیع شده

علی رحیمی حسین آباد، محسن کاهانی و عبدالرضا سوادى

دانشگاه فردوسی مشهد، دانشکده مهندسی، گروه کامپیوتر

E-mail: al_ra45@stu-mail.um.ac.ir, {kahani, savadi}@um.ac.ir

چکیده - در این مقاله، یک بستر مبتنی بر عامل برای توسعه سیستم‌های خبره توزیع شده ارائه شده است. در این بستر، هر سیستم خبره می‌تواند تکنیک استدلال و نمایش دانش خاص خود را داشته باشد و با استفاده از یک عامل با بقیه سیستم ارتباط برقرار نماید. ارتباط، شناسایی و پیدا کردن عامل‌ها توسط یکدیگر، بر اساس استاندارد *FIPA* و سرویس‌هایی که محیط‌های مبتنی بر عامل ارائه می‌دهند، انجام می‌شود. نمایش دانش در این بستر با استفاده از *Ontology* انجام می‌گردد. پیاده‌سازی این سیستم در محیط *JADE* انجام شده و با یک سیستم خبره تشخیص پزشکی نیز تست شده است.

کلید واژه- سیستم توزیع شده، سیستم خبره، محیط‌های چند عامله، *FIPA*، *Ontology*

۱- مقدمه

خبره شوند. ولی همیشه مسایلی وجود دارند که برای حل آن، نیاز به ارتباط بین خبرگان چند حوزه مرتبط می‌باشد. سیستم‌های خبره نیز بدین‌گونه هستند. یعنی برای حل برخی مسائل، نیاز به برقراری ارتباط بین چند سیستم خبره که در حوزه‌های مرتبط فعالیت می‌کنند، می‌باشد.

بهترین روش برای وجود آوردن این ارتباط، سیستم‌های چند عامله می‌باشد [۱]. این سیستم‌ها از هوش مصنوعی توزیع شده سرچشمه می‌گیرند [۲]، [۳]. سیستم‌های چند عامله، ابزار، محیط و استانداردهای لازم برای برقراری ارتباط بین چند نرم‌افزار و سیستم را فراهم می‌کنند. استفاده از این سیستم‌ها باعث می‌شود که بتوان به مسایلی که توسط یک سیستم متمرکز قابل‌حل نیستند، پاسخ داد. همچنین با شکستن و تبدیل یک سیستم پیچیده به سیستم‌های کوچک‌تر، از پیچیدگی سیستم کاسته شده و نگهداری آن آسان‌تر می‌گردد. بویژه در زمانی که لازم است برای حل مسائل، بین چند سیستم که توسط شرکت‌های مختلف توسعه داده شده‌اند، ارتباط نزدیک برقرار نمود، سیستم‌های چند عامله می‌توانند بسیار مفید واقع شوند. از این‌رو، استفاده از سیستم‌های چند عامله، بهترین راه‌حل برای برقراری ارتباط بین سیستم‌های خبره می‌باشد.

در این مقاله، یک بستر مبتنی بر عامل برای توسعه سیستم‌های خبره توزیع شده، ارائه شده است. در این بستر،

سیستم‌های خبره در سه دهه گذشته رشد و پیشرفت بسزایی، چه از لحاظ تئوری و چه از لحاظ زبانها و ابزارهای پیاده‌سازی داشته‌اند و این مساله باعث شده است که از این سیستم‌ها در زمینه‌های بسیار متفاوتی استفاده شود. همه این سیستم‌های خبره از لحاظ معماری دارای یک پایگاه دانش و یک موتور استنتاج می‌باشند. هدف از ایجاد یک سیستم خبره، منتقل کردن دانش یک فرد خبره در حوزه‌ای خاص، به کامپیوتر می‌باشد تا بتوان از این دانش برای پاسخ به سؤالاتی که در آن حوزه مطرح می‌شود، استفاده کرد. اما با گذشت زمان، حوزه مربوطه وسعت پیدا کرده و باعث می‌شود حجم دانشی که باید وارد سیستم خبره شود، افزایش یابد و این کار باعث پیچیده شدن سیستم خبره شده و نگهداری و بروزرسانی آن را دشوار می‌سازد. از این‌رو، توسعه سیستم‌های خبره در حوزه‌هایی که دارای دانش وسیعی هستند، بسیار سخت و گاهی اوقات غیرممکن به نظر می‌رسد. لذا، سعی می‌گردد حوزه دانشی که برای سیستم خبره در نظر گرفته می‌شود، کوچک باشد و در عوض، به عمق دانش آن افزوده گردد. این مسئله در جوامع انسانی نیز به همین شکل اتفاق می‌افتد. یعنی با رشد و گستردگی دانش در یک حوزه خاص، زیرحوزه‌هایی از آن بوجود می‌آیند که افراد می‌توانند در آن زیرحوزه‌ها

هر سیستم خبره می‌تواند تکنیک استدلال و نمایش دانش خاص خود را داشته و با استفاده از یک عامل با بقیه سیستم ارتباط برقرار نماید. استفاده از این بستر، باعث می‌شود که بتوان بین چند سیستم خبره که در حوزه‌های مرتبط فعالیت می‌کنند، ارتباط برقرار نموده و از مجموع دانش آنها برای حل مسایل آن حوزه استفاده کرد. مبتنی بودن بستر ارایه شده بر استاندارد FIPA باعث می‌شود که بتوان با سیستم‌های مبتنی بر عامل دیگری که از این استاندارد طبیعت می‌کنند، ارتباط مناسبی برقرار کرد.

در بخش ۲ مرور کلی بر نمونه‌هایی از سیستم‌های خبره توزیع شده خواهیم داشت، در بخش ۳ استاندارد FIPA را بررسی خواهیم نمود، در بخش ۴ به ارایه بستر پیشنهادی می‌پردازیم، و در بخش‌های ۵ و ۶ نیز به پیاده سازی، تست و نتیجه‌گیری از کار خواهیم پرداخت.

۲- سیستم‌های خبره توزیع شده

در سال‌های اخیر نمونه‌های مختلفی از سیستم خبره توزیع شده، برای کاربردهای گوناگون، طراحی و پیاده‌سازی شده‌اند. برخی از این سیستم‌ها دارای یک پایگاه دانش متمرکز می‌باشند که از آن برای مدیریت و کنترل تعداد زیادی از وسایل و دستگاه‌ها که در یک محیط خاص توزیع شده‌اند، استفاده می‌شود. ورودی‌های این سیستم نیز بصورت توزیع شده در محیط قرار دارند. یک نمونه از این نوع سیستم‌های خبره، SINGRAR می‌باشد. SINGRAR [۴] یک سیستم خبره فازی توزیع شده است که برای کنترل و مدیریت کشتی‌های جنگی در زمان نبرد، توسط نیروی دریایی پرتقال طراحی و پیاده‌سازی شده است. برخی دیگر از سیستم‌های خبره توزیع شده، دارای پایگاه‌های دانش جداگانه‌ای هستند که مجموع دانش موجود در این سیستم‌ها قادر به حل مسایلی است که هر یک از پایگاه‌ها بصورت جداگانه قادر به پاسخگویی آنها نمی‌باشند. در ادامه اشاره کوتاهی به چند نمونه از این نوع سیستم‌ها خواهیم داشت.

یک نمونه از این نوع سیستم‌های خبره توزیع شده، توسط K. Shaalan و همکارانش طراحی و اجرا شده است [۱۴]. آنها سیستم خبره چند عامله‌ای ارایه کرده‌اند که بر روی سرورهای مختلف قرار دارد و گروه بزرگی از کاربران می‌توانند از طریق شبکه با آن ارتباط برقرار کنند. سیستم

آنها از دو نوع عامل تشکیل شده است. گروه اول، عامل‌های سیستم خبره هستند که بر روی سرورها قرار دارند. هر کدام از این عامل‌ها پایگاه دانش مستقلی دارند که در زمینه‌ای خاص، ولی مرتبط با عامل‌های دیگر فعالیت می‌کنند. برخی از آنها در زمینه تشخیص و بعضی دیگر در زمینه درمان فعالیت دارند. سیستم قادر است مسائلی را که نیاز به ترکیب دانش مجموعه عامل‌ها دارند، حل کند. گروه دوم، عامل‌های واسط کاربر و هماهنگ کننده هستند که وظیفه برقراری ارتباط کاربر با سیستم را بعهده دارند. آنها سیستم خود را در زبانهای Knowledge Representation (KROL Object Language) و JAVA و با استفاده از متدولوژی مهندسی دانش KADS و در بستر WWW پیاده سازی کرده و بر اساس یک سیستم خبره تشخیص و درمان بیماری‌های زراعی امتحان نموده‌اند.

Ex-W-Pert [۶] نیز یک معماری برای سیستم خبره توزیع شده تحت وب ارائه نموده است. در این سیستم، پایگاه‌های دانش توزیع شده و اشتراکی^۱ وجود دارند که باعث می‌شود بتوان از آن برای طراحی Groupware استفاده نمود. Ex-W-Pert از استدلال مبتنی بر مدل^۲ و قاب^۳ برای نمایش دانش استفاده می‌کند. در این سیستم، هر واحد، مانند یک سیستم خبره، دارای پایگاه دانش محلی و موتور استنتاج می‌باشد. همچنین آنها دارای یک موتور ارتباطی هستند که برای بدست آوردن دانش و داده از منابع دیگر از آن استفاده می‌کنند. ارتباطات این سیستم، مبتنی بر پروتکل‌های خاصی که از NFS و HTTP استفاده می‌کنند، می‌باشد.

¹ shared

² Model Based Reasoning

³ Frame

IEEE است که به ترویج و توسعه تکنولوژی‌های مبتنی بر عامل و استانداردسازی آن و قابلیت تعامل آن با تکنولوژی‌های دیگر می‌پردازد. استاندارد که FIPA ارائه کرده است، به بیان نحوه پیاده‌سازی سیستم‌های مبتنی بر عامل و معماری درونی عامل‌ها نمی‌پردازد. بلکه قواعدی را بیان می‌کند که طراحان سیستم‌های مبتنی بر عامل باید در زمان طراحی، آنها را ملاک قرار دهند تا بتوانند با سیستم‌های دیگر ارتباط برقرار کنند [۵]. این استاندارد از ۵ بخش تشکیل شده است:

۳-۱- معماری مجرد

مستندات معماری مجرد^۴ بیان‌کننده این هستند که در یک سطح انتزاعی، چگونه دو عامل می‌توانند با ثبت خودشان در سیستم، یکدیگر را پیدا نموده و با تبادل پیام با هم ارتباط برقرار کنند. همچنین، اجزای اصلی که برای برقراری این ارتباط در سیستم مورد نیاز می‌باشند و ارتباط بین این اجزا در این مستندات توصیف شده است.

۳-۲- انتقال پیام بین عامل‌ها

مستندات انتقال پیام بین عامل‌ها^۵، به انتقال و نمایش پیام‌ها بر روی پروتکل‌های ارتباطی مختلف شبکه، اعم از شبکه‌های باسیم و بی‌سیم می‌پردازد. در این استاندارد، ساختار پیام، ویژگی‌های اصلی سرویس انتقال پیام و Ontology مورد نیاز برای انتقال پیام ارائه گردیده است.

ExpertNet [۷] نیز یک سیستم خبره توزیع شده است که برای مدیریت شبکه کامپیوتر ملی اوکراین طراحی شده است. این سیستم چند عامله بوده و با استفاده از DEVICE (Data-driven & Event-driven rule Integration using) و CS-Prolog پیاده‌سازی گردیده و از ابزار NHMS+ برای بدست آوردن نقشه کاملی از شبکه استفاده می‌کند.

FIPA-DES [۸] یک سیستم خبره نامتمرکز مبتنی بر عامل است که مبتنی بر استاندارد FIPA می‌باشد. این سیستم از عامل‌های سیستم خبره و مشتری تشکیل شده است و با استفاده از پیام‌هایی که بین عامل‌ها مبادله می‌کند، Backward Chaining را بصورت توزیع شده در سیستم اجرا می‌نماید.

۳- استاندارد FIPA و محیط‌های مبتنی بر عامل

با رشد و گسترش شبکه در سطح دنیا، تکنولوژی‌ها و برنامه‌های کاربردی مربوط به آن نیز به شکل روزافزونی افزایش یافته‌اند. تنوع برنامه‌های کاربردی مرتبط با شبکه باعث شده است که تکنولوژی لازم، برای توسعه این برنامه‌ها نیز دائم در حال تغییر و پیشرفت باشد. یکی از این تکنولوژی‌ها که در سال‌های اخیر بسیار مورد توجه قرار گرفته است، تکنولوژی مبتنی بر عامل می‌باشد. این تکنولوژی بویژه در محیط‌های باز که سیستم‌های اطلاعاتی مختلف و متنوعی در آن قرار دارند، بسیار مورد استفاده قرار گرفته است. این تنوع سیستم‌های اطلاعاتی و برنامه‌های کاربردی که از تکنولوژی مبتنی بر عامل استفاده می‌کنند، ایجاب می‌کند که استاندارد مناسبی برای تعریف این تکنولوژی ارائه شود.

با اینکه مهم‌ترین مساله در تعریف استاندارد در تکنولوژی‌های مبتنی بر عامل، ارتباط بین عامل‌ها است، ولی برای اینکه بتوان از این تکنولوژی بطور کاربردی و گسترده استفاده کرد، نیاز به استاندارد کردن بخش‌های دیگری نیز وجود دارد. گروه‌های مختلفی مانند KSE (Knowledge Sharing Effort)، OMG (ObjectManagement Group) و FIPA (Foundation for Intelligent Physical Agents) در زمینه استانداردسازی این تکنولوژی فعالیت کرده‌اند.

FIPA یکی از واحدهای بخش استاندارد در جامعه کامپیوتر

⁴ Abstract Architecture

⁵ Agent Message Transport

۳-۳- مدیریت عامل‌ها

این بخش از مستندات، به کنترل و مدیریت عامل‌ها در داخل و بین سکوها‌ی عامل می‌پردازد. ایجاد، نام‌گذاری، ثبت، مکان‌یابی، ارتباط، انتقال و پایان دادن به کار عامل‌ها در این مستندات توضیح داده شده است.

۳-۴- ارتباط بین عامل‌ها

در این بخش، پیام‌های مبتنی بر زبان ارتباطی عامل‌ها، پروتکل‌های تبادل و انتقال پیام بین عامل‌ها، انواع پیام‌ها از نظر محتوا، و نمایش‌های مختلف زبان ارائه شده است.

۳-۵- برنامه‌های کاربردی عامل

مستنداتی که FIPA در این بخش ارائه کرده است، مربوط به برنامه‌های کاربردی می‌گردد که می‌توان از عامل‌های با استاندارد FIPA برای پیاده‌سازی آنها استفاده نمود. این مستندات شامل استاندارد مربوط به سرویس و Ontology در چند حوزه مختلف می‌باشد. همچنین Ontology مربوط به کیفیت سرویس در عامل‌ها نیز در این مجموعه آورده شده است.

۴- سیستم خبره توزیع شده مبتنی بر عامل

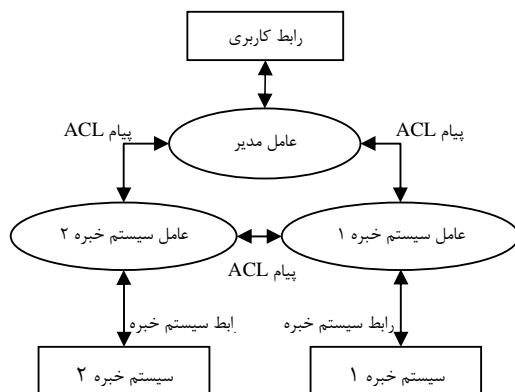
۴-۱- معماری سیستم

همان‌طور که گفته شد، هدف از ارائه این سیستم، ارائه بستری است که در آن سیستم‌های خبره بتوانند با یکدیگر ارتباط برقرار کنند و یا بتوان از مجموعه دانش چند سیستم خبره استفاده کرد. همان‌طور که در شکل ۱ نشان داده شده است، سیستم ارائه شده دارای دو نوع عامل می‌باشد. نوع اول، عامل‌های سیستم خبره می‌باشند که با سیستم خبره در ارتباط هستند و وظیفه تعامل با آن را به عهده دارند. کلیه ارتباطات با سیستم خبره، از طریق عامل مربوط به آن انجام می‌شود و بقیه بخش‌های سیستم نیازی به ارتباط مستقیم با سیستم خبره ندارند. نوع دوم عامل مدیر است که فقط یک نسخه از آن در سیستم وجود دارد. این عامل وظیفه پیدا کردن عامل‌های سیستم خبره که قبلاً ایجاد شده اند را به عهده دارد. عامل مدیر، پرسش‌ها را با استفاده از رابط کاربری از کاربر دریافت می‌کند و بر اساس حوزه پرسش، عامل‌های مورد نظر را پیدا کرده و برای رسیدن به پاسخ، تعاملات لازم را با آنها انجام می‌دهد. همچنین اگر

عامل‌های سیستم خبره نیز پرسشی داشته باشند، می‌توانند از عامل مدیر بپرسند تا وی از طریق عامل‌های دیگر، به پرسش آنها پاسخ دهد.

۴-۲- ارتباط عامل‌ها با سیستم خبره

سیستم‌های خبره را می‌توان با زبانهای برنامه‌نویسی مختلف و در بسترهای متفاوتی مانند JESS [۹] و CLIPS [۱۰] پیاده سازی کرد و این مساله، ارتباط بین سیستم‌های خبره را با مشکل مواجه می‌کند. بیشتر این بسترها رابطی با زبان‌ها و محیط‌های برنامه‌نویسی دیگر فراهم می‌کنند تا بتوان از خارج از سیستم خبره نیز با آن ارتباط برقرار کرد و در آن تغییرات داد یا از دانش آن استفاده کرد. عامل سیستم خبره که بخش اصلی بستر ارائه شده در این مقاله است نیز از همین رابط برای برقراری ارتباط با سیستم خبره استفاده می‌کند. در نتیجه، برای بسترهای خاص و سیستم‌های خبره مختلف، باید عامل‌های سیستم خبره متفاوتی توسعه داده شوند که قابلیت ارتباط با آن سیستم خبره را داشته باشند.



شکل ۱: معماری سیستم

۴-۳- نمایش دانش در سیستم

نمایش دانش نیز در بسترهای مختلف، به اشکال متفاوتی انجام شده است. مثلاً در JESS با استفاده از Template این کار انجام می‌شود. یکی دیگر از کارهایی که عامل سیستم خبره باید انجام دهد، نگاشت نمایش دانش در سیستم خبره به نمایش دانش در بستر ارائه شده که همان Ontology است، می‌باشد. در بستر ارائه شده، به منظور یکسان سازی نمایش دانش در کل سیستم از Ontology استفاده شده است. این Ontology بصورت JavaBeans در سیستم پیاده

سازی شده است. البته از ابزارهای تولید و توسعه Ontology مانند Protégé نیز می‌توان برای تولید Ontology استفاده کرد.

۴-۴- عامل مدیر

هر عامل سیستم خبره‌ای که در سیستم ایجاد می‌شود، خود می‌داند که در چه حوزه‌ای از دانش فعالیت می‌کند و چه سرویس‌هایی را ارائه می‌دهد. همچنین عامل سیستم خبره، مانند هر عامل دیگری، باید خود را به محیط معرفی کند. محیط‌های مبتنی بر عامل، سروری را ارائه می‌کنند تا عاملها بتوانند خود را به آن معرفی کنند و همچنین سرویس‌های قابل ارائه خود را در آن ثبت کنند. عامل سیستم خبره، غیر از معرفی خود، حوزه فعالیت خود را نیز (بر اساس Ontology) به سرور معرفی می‌کند تا عامل‌های دیگر بتوانند در صورت لزوم از سرویس‌های آن استفاده کنند.

در این بین، عامل مدیر وظیفه نگه‌داری و دسته‌بندی عامل‌ها و سرویس‌های قابل ارائه آنها و همچنین حوزه فعالیت آنها را بعهده دارد. همچنین عامل مدیر رابطی را فراهم می‌کند تا کاربران بتوانند سؤالات خود را با استفاده از آن رابط از سیستم بپرسند. این رابط نیز بر اساس Ontology تعریف شده در سیستم پیاده‌سازی شده و عمل می‌کند.

۴-۵- ارتباط بین عامل‌های سیستم خبره

پس از برقراری ارتباط بین عامل و سیستم خبره، حال نوبت به ارتباط بین عامل سیستم خبره و عامل مدیر می‌رسد. از آنجا که این بستر مبتنی بر FIPA می‌باشد، از پیام‌های ACL برای برقراری ارتباط بین عامل‌ها استفاده می‌شود. هر پیام ACL از ویژگی‌های مختلفی تشکیل شده است. یکی از این ویژگی‌ها Communicative Act نام دارد که بر اساس استاندارد FIPA می‌تواند مقادیر مختلفی به خود بگیرد. در بستر ارائه شده در این مقاله نیز از همین ویژگی برای بیان نوع پیام استفاده می‌شود. انواع پیام‌های مورد استفاده در این سیستم عبارتند از:

Query Ref: هنگامی که یک عامل سیستم خبره، سوالی از عامل دیگری داشته باشد که انتظار داشته باشد در جواب، داده‌ای دریافت کند، از این نوع پیام استفاده می‌کند.

Query If: چنانچه یک عامل سیستم خبره بخواهد درستی یا نادرستی گزاره‌ای را از عامل دیگری جویا شود، از این پیام استفاده می‌کند.

Confirm: زمانی که یک عامل سیستم خبره بخواهد عامل دیگری را از صحت گزاره‌ای مطلع کند، از این نوع پیام استفاده می‌کند. معمولاً این نوع پیام، در پاسخ به Query If داده می‌شود.

Disconfirm: زمانی که یک عامل سیستم خبره بخواهد عامل دیگری را از عدم صحت گزاره‌ای مطلع کند، از این نوع پیام استفاده می‌کند. معمولاً این نوع پیام در پاسخ به Query If داده می‌شود.

Inform Ref: اگر یک عامل بخواهد داده‌ای را به عامل دیگری اطلاع دهد، از این نوع پیام استفاده می‌کند. از این نوع پیام معمولاً در پاسخ به Query Ref استفاده می‌شود.

از دیگر ویژگی‌های یک پیام ACL، محتوای آن است که در موارد گفته شده در بالا، می‌تواند یک گزاره و یا داده باشد. همچنین هر عامل باید فرستنده و گیرنده پیام را مشخص کند. برای این منظور، عامل سیستم خبره ابتدا باید بداند که چه عامل‌هایی در سیستم وجود دارند که ممکن است پاسخ سوال وی را داشته باشند. دانستن اینکه چه عاملی دارای چه نوع اطلاعاتی است و در چه حوزه‌ای فعالیت می‌کند، بر عهده عامل مدیر می‌باشد که توضیح گفته شده است.

۵- پیاده سازی و تست

پیاده سازی بستر ارائه شده از چند بخش تشکیل می‌شود. بخش اول پیاده‌سازی عامل‌ها (عامل مدیر و عامل سیستم خبره) می‌باشد. از آنجا که این بستر باید بر اساس FIPA طراحی شود، در مراحل پیاده‌سازی نیز باید استانداردهای FIPA رعایت شوند. از این رو برای پیاده‌سازی عامل‌ها از JADE استفاده شده است. JADE (Java Agent Development Framework)، خود یک محیط مبتنی بر عامل است که امکان توسعه سیستم‌های مبتنی بر عامل مطابق با استاندارد FIPA را فراهم می‌کند [۱۱]، [۱۲]. این محیط با استفاده از زبان JAVA پیاده‌سازی شده است و کلاس‌ها و رابط‌های موردنیاز برای توسعه سیستم‌های مبتنی بر عامل را در اختیار برنامه‌نویسان قرار می‌دهد. ارتباط بین عامل‌ها نیز همانطور که پیش‌تر به آن اشاره شد،

- [1] M. Wooldridge, *An Introduction to MultiAgent Systems*, John Wiley and Sons Inc., 2002.
- [2] G. M. P. O'Hare, N. R. Jennings, *Foundations of Distributed Artificial Intelligence*, John Wiley & Sons Inc., 1996.
- [3] L. Gasser, M. N. Huhns, *Distributed Artificial Intelligence*, Morgan Kaufmann Publishers Inc., Vol(2).
- [4] M. S. Marques, F. J. Pires, "SINGRAR-A fuzzy distributed expert system to assist and command control activities in naval environment," *Journal of Operational Research*. v145. pp. 343-362, 2003.
- [5] K. Shaalan, M. El-Badry, A. Rafea, "A Multiagent Approach for Diagnostic Expert Systems via the Internet," *Expert System with Applications*, 27(1), pp. 1-10, 2004.
- [6] B. H. Far and Z. Koono, "Ex-w-pert system: A web-based distributed expert system for groupware design," in Proc. of the Third World Congress on Expert Systems, 1995.
- [7] I. Vlahavas, et al., "System Architecture of a Distributed Expert System for the Management of a National Data Network," *Proc. 8th Int'l Conf. Artificial Intelligence*, Springer, New York, 1998, pp. 438-451.
- [8] P. Skrzynski et. al., "FIPA Compliant Agent-based Decentralised Expert System", in *Proc. IIS' 2002 symposium*, Poland, 2002.
- [9] JESS- Java Expert System Shell. [Online]. Available: <http://www.jessrules.com>
- [10] CLIPS Basic Programming Guide, NASA JSC-25012
- [11] JADE – Java Agent DEvelopment Framework, [Online]. Available: <http://jade.tilab.com>
- [12] F. L. Belfemine, G. Caire, D. Greenwood, "Developing Multi-Agent Systems with JADE", John Wiley & Sons Inc., 2007.
- [13] The Protégé Ontology Editor and Knowledge Acquisition System. [Online]. Available: <http://protege.stanford.edu/>
- [14] FIPA, Foundation for Intelligent Physical Agents, [Online]. Available: <http://www.fipa.org>

با استفاده از پیام‌های ACL مبتنی بر FIPA که در JADE پیاده‌سازی شده‌اند، انجام می‌شود.

بخش دوم پیاده‌سازی مربوط به توسعه Ontology می‌باشد که بر اساس حوزه کار سیستم، باید توسعه داده شود. برای توسعه Ontology از ابزار Protégé استفاده شده است. Protégé یک ابزار ویرایش Ontology و مدیریت پایگاه دانش می‌باشد [۱۳]. Ontology مربوط به حوزه کاری سیستم با استفاده از این ابزار تعریف شده و در قالب JavaBean در سیستم مورد استفاده قرار گرفته است.

رابط کاربری نیز با استفاده از تکنولوژی JSF و Servlet در JAVA پیاده‌سازی شده که شمایی از Ontology تعریف شده در سیستم را در اختیار کاربر قرار می‌دهد تا بتواند پرسش‌های خود را به سیستم وارد کند و نتایج نیز در قالب همین Ontology در اختیار کاربر قرار می‌گیرد.

برای تست محیط از سیستم تشخیص پزشکی استفاده شده است. دو سیستم خبره تشخیص پزشکی ساده در زمینه های مشابه با دانش‌های مختلف در سیستم قرار داده شدند تا بتوانند تشخیص بیماری را با استفاده از مجموع دانش یکدیگر انجام دهند. بدین ترتیب، دو سیستم با استفاده از دانش یکدیگر، به تشخیص صحیح تری دست پیدا کردند.

۶- نتیجه گیری

حل بسیاری از مسایل توسط یک سیستم خبره، هر قدر هم که دانش زیادی داشته باشند، ممکن نیست. بویژه مسایلی که برای حل آنها نیاز به استفاده از دانش چند حوزه مرتبط باشد. از این رو، باید بتوانیم بین سیستم‌های خبره، ارتباط برقرار کنیم تا بتوانیم از مجموع دانش آنها برای حل مسایل استفاده کنیم. بستر ارایه شده در این مقاله، یک پیش‌نیاز در جهت برقراری این ارتباط است. برای تکمیل این بستر می‌توان یک رابط استاندارد برای ارتباط با سیستم خبره تعریف نمود که توسعه دهندگان سیستم‌های خبره ملزم به پیاده‌سازی این رابط باشند. همچنین می‌توان از تکنولوژی‌های دیگر تعریف Ontology مانند RDF و OWL استفاده نمود.

مراجع