جزوه درس:

سیستم‌های خبره
Expert systems

کتاب مرجع:
Expert Systems Design and Development

نوشته:
John Durkin

مانی عابدینی
تورج بنی‌رستم

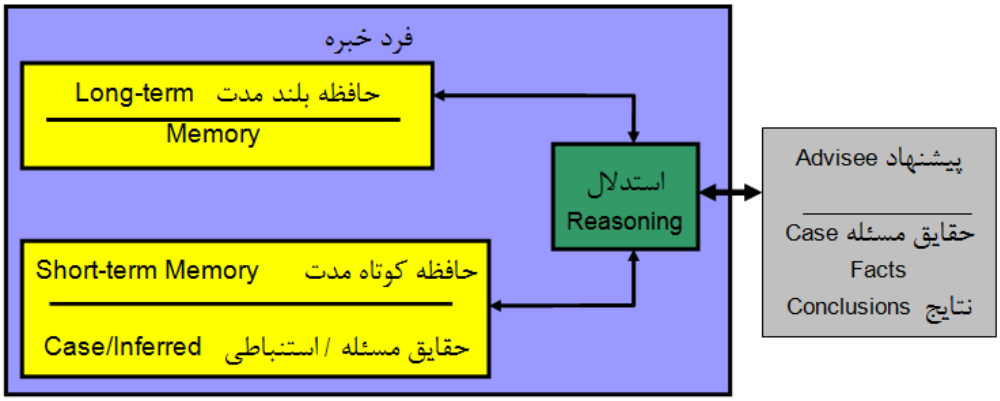
فصل دوم
ویژگی‌های سیستم‌های خبره

۱- ساختار سیستم خبره (Expert System Structure)

یک فرد را زمانی خبره به مسئله‌ای می‌گوییم که او دانش خاصی در مورد آن مسئله داشته باشد. در فیلد سیستم‌های خبره، به اینگونه از دانش Domain Knowledge (DK) یا دانش محیط کاربرد می‌گوییم. کلمه «محیط کاربرد» اشاره دارد که دانش مربوط به یک مسئله مشخص و محدود شده است.

یک فرد خبره دانش محیط کاربرد (DK) خود را در حافظه بلند مدت(Long-Term Memory (LTM خود نگهداری می‌کند. در زمان ارائه پیشنهاد به کسی، فرد خبره ابتدا حقایق (facts) مربوط به مسئله را جمع آوری می‌کند (Case Facts)، و در حافظه کوتاه مدت Short-Term Memory (STM) خود نگهداری می‌کند.

فرد خبره بر مورد مسئله یا مشکل با ترکیب حقایق درون STM و دانش درون LTM استدلال می‌کند. با بکارگیری این فرآیند، فرد خبره به استنباط‌های جدیدی و در نهایت به نتیجه (راه حل) مسئله خواهد رسید.



شکل (۱) نمایش روش استدلال در یک فرد خبره

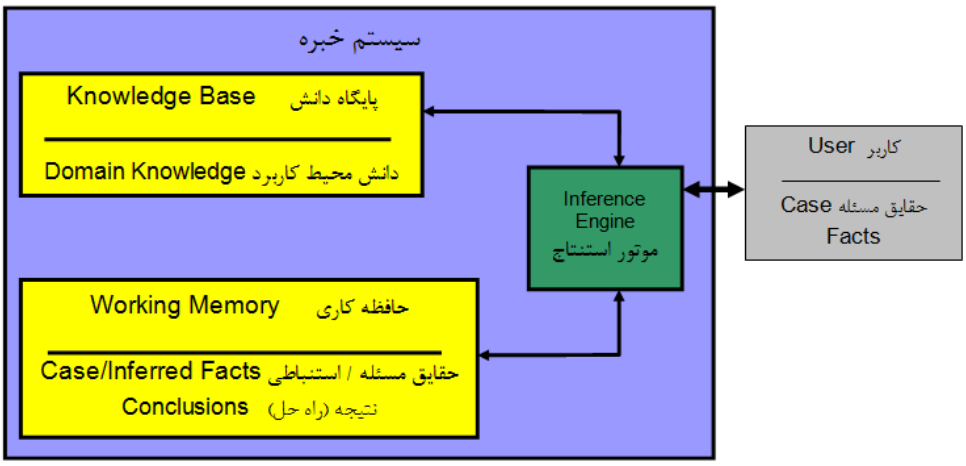
برای فهم بیشتر؛ فرض کنید در ماشین شما مشکلی رخ داده‌است. یک عمل منطقی آن است که وسیله را به یک مکانیک ماشین (فرد خبره) نشان دهیم.

در طی سالیان و تجربیات مختلف مجموعه دانش لازم جهت تشخیص عیب ماشین در حافظه بلند مدت LTM مکانیک نقش بسته است. فرض کنید به مکانیک بگویید که ماشین روشن نمی‌شود، مکانیک اطلاعاتی که شما می‌دهید را در حافظه کوتاه مدت STM خود نگهداری می‌کند و شروع به استدلال بر اساس آنها می‌کند.

با اطلاعات ارائه شده توسط شما و دانش محیط کاربرد DK، مکانیک ممکن است به این نتیجه برسد که مشکل در سیستم برق ماشین است و این یافته خود را در STM قرار داده و به استدلال در مورد مسئله ادامه دهد. در ادامه بررسی ممکن است تست باتری ماشین نشان دهد که مشکل به سبب خرابی باتری است.

در مثال مکانیک، نشان دادیم که نه تنها مکانیک مشکل را رفع می‌کند بلکه فرآیند استدلالی پشت آن را نیز نشان می‌دهد.

سیستم‌های خبره مسائل را با فرآیندی خیلی مشابه با روش فرد خبره حل می‌کنند و از ساختار زیر تشکیل شده‌اند.



شکل (۲) نمایش روش استدلال در یک سیستم خبره

۲- پایگاه دانش (Knowledge Base)

یک سیستم خبره دانش محیط کاربرد یک فرد خبره را در ماژولی بنام پایگاه دانش نگهداری می‌کند. پایگاه دانش همان LTM در مدل فرد خبره است.

تعریف: پایگاه دانش بخشی از یک سیستم خبره است که شامل دانش محیط کاربرد است.

وظیفه یک مهندس دانش Knowledge Engineer، آن است که دانش را از فرد خبره بدست آورده و در پایگاه دانش کد کند. تکنیکهای مختلفی برای این فرآیند وجود دارد که در بخش‌های بعدی اشاره می‌شود.

یکی از روشهایی‌که می‌توان دانش را در یک سیستم خبره ارائه کرد، با کمک قوانین (Rules) است. یک قانون از ساختار اگر/ آنگاه If/Then پیروی می‌کند و بطور منطقی اطلاعاتی که در قسمت اگر می‌آیند، با اطلاعاتی که در قسمت آنگاه ظاهر می‌شوند مرتبط هستند.

- برای مثال:

RULE ۱
IF The car will not start
THEN The problem may be in the electrical system

RULE ۲
IF The Problem may be in the electrical system
AND The Battery voltage is below ۱۰ volts
THEN The fault is a bad battery

۳- حافظه کاری (Working Memory)

حافظه کاری شامل حقایق در مورد مسئله است که در طی اجرا بدست آمده است.

تعریف: حافظه کاری، بخشی از یک سیستم خبره است که حقایق مسئله را که در طی اجرا بدست آمده است، در خود دارد .

سیستم خبره اطلاعات بدست آمده در طی اجرا را که در حافظه کاری ذخیره می‌شود با قوانین تطبیق می‌دهد تا حقایق جدید استنتاج شده و آنها را در حافظه کاری قرار دهد. ممکن است به نتایجی نیز برسد که قبلاً در حافظه کاری وجود داشته است. برخی سیستم‌های خبره توانایی بهره‌گیری از پایگاه‌داده‌های خارجی، صفحه‌های گسترده و حتی سنسورها را دارند. که آن اطلاعات به حافظه کاری منتقل شده و در جریان فرآیند استنتاج قرار می‌گیرد.

۴- موتور استنتاج (Inference Engine)

همانطور که گفته شد، سیستم خبره فرآیند استدلال انسانی را با ماژولی بنام موتور استنتاج مدل می‌کند.

تعریف: موتور استنتاج، پردازنده‌ای در یک سیستم خبره است که حقایق موجود در حافظه کاری را با دانش محیط کاربرد موجود در پایگاه دانش تطبیق می‌دهد تا در مورد مسئله به نتیجه برسد.

موتور استنتاج با حقایقی که در حافظه کاری و پایگاه دانش قرار دارد کار می‌کند تا به اطلاعاتی جدید برسد. بدین صورت که در مجموعه قوانین بدنبال قسمت اگر قوانینی می‌گردد که با اطلاعات موجود در حافظه کاری منطبق باشند. زمانیکه یک نمونه تطبیق شده از قوانین

پیدا شد، نتیجه قانون را به مجموعه حقایق موجود در حافظه کاری اضافه می‌کند «دانسته جدید»، تا جستجو برای یافتن قوانین تطبیق پذیر جدید ادامه پیدا کند.

برای مثال، دو قانون قبلی را در نظر گرفته و پرسش و پاسخ سیستم خبره و کاربر را ببینید:

STEP ۱

EXPERT SYSTEM: Does the car not start?
USER: TRUE

کاربر این واقعیت را به حافظه کاری اضافه می‌کند، از آنجا که این حقیقت قانون ۱ را پشتیبانی می‌کند، نتیجه آن به حافظه کاری اضافه می‌شود.

USER ASSERTS: The car Will not start.
SYSTEM ASSERTS: The problem may be in the electrical system

STEP ۲

EXPERT SYSTEM: Is the battery voltage below ۱۰ volts
USER: TRUE

حالا حافظه کاری شامل اطلاعات جدیدی است که قانون ۲ را پشتیبانی می‌کند، لذا سیستم خبره نتیجه آن قانون را به حافظه کاری اضافه می‌کند.

USER ASSERTS: The battery voltage is below ۱۰ volts
SYSTEM ASSERTS: The fault is a bad battery

در این مرحله کار اتمام می‌یابد زیرا قانون دیگری برای بررسی وجود ندارد.

۵- سهولت توضیح (Explanation Facility)

مشخصه بارز سیستم‌های خبره توانایی آنها در توضیح فرآیند استدلالی‌شان است. در سیستم خبره ماژولی قرار دارد که وظیفه‌اش توضیح روند استدلال است. با بکارگیری این ماژول، یک سیستم خبره می‌تواند برای کاربر توضیح دهد که *چرا Why* چنین سوالی از کاربر می‌کند و *چگونه How* به نتیجه نهایی‌رسیده‌است.

بکارگیری این ماژول «سهولت توضیح» برای توسعه دهنده سیستم این مزیت را دارد که خطاهای ممکن در دانش سیستم را تشخیص بدهد و برای کاربر نیز این مزیت را دارد که روش استدلالی سیستم را بطور شفاف ببیند.

۶- توضیح چگونگی

علاوه بر نتیجه نهایی، هم فرد خبره و هم سیستم خبره می‌توانند توضیح دهند که چگونه به نتیجه رسیده‌اند. این توانایی برای یک سیستم خبره بسیار مهم است.

بر خلاف یک برنامه متعارف که روی مسائل کاملاً تعریف شده کار می‌کند. یک سیستم خبره روی مسائلی کار می‌کند که ساختارمند نیستند. لذا دستاورد سیستم خبره زیر سوال خواهد رفت، به همین منظور باید ابزاری برای پشتیبانی نتایج بدست آمده ارائه شود و از آن نتیجه دفاع کند.

برای مثال:

EXPERT: The battery is bad.
USER: HOW
EXPERT: از آنجایی که ماشین روشن نمی‌شد، به این نتیجه رسیدم که در سیستم برق ماشین اشکالی وجود دارد، مشاهده شد که ولتاژ باطری از ۱۰ ولت کمتر است ، پس به این نتیجه رسیدم که باطری خراب است.

این گونه توضیح دادن، در واقع دنبال کردن فرآیند استدلالی فرد خبره است. سیستم‌های خبره نیز به دستور HOW بطریق مشابه دیده شده پاسخ می‌دهند. در حقیقت عقب گردی از روی قوانینی که منتج به نتیجه نهایی‌شده‌اند خواهد کرد. این مسیر طی شده از روی قوانین تا نتیجه نهایی، مسیر استدلالی سیستم را نشان می‌دهند. در نتیجه کاربر به نتیجه ارائه شده سیستم بیشتر اعتماد خواهد کرد زمانیکه منطق پشت توصیه سیستم خبره را ببیند.

۷- توضیح چرا

یک سیستم خبره همچنین می‌تواند توضیح دهد چرا چنین سوالی از کاربر می‌پرسد.
در شرایطی که با فرد خبره سوال و جواب می‌کنیم، ممکن است فرد از فرد خبره سوال کند که چرا چنین سوالی از او می‌کند، پاسخ فرد خبره باعث اعتماد بیشتر فرد به فرآیند استدلالی می‌شود و به این نتیجه می‌رسد که دانش فرد خبره کامل و قابل دفاع است.
برای مثال:

EXPERT: Will the car not start?
USER: WHY
EXPERT: اگر بدانم که ماشین روشن نمی‌شود، معمولاً به این نتیجه می‌رسم که در سیستم برق ماشین مشکلی وجود دارد.

۸- رابط کاربری (Interface)

تعامل بین یک سیستم خبره و یک کاربر باید به فرم خیلی طبیعی باشد، همانند گفتگوی بین انسانها. برای این منظور در مورد رابط کاربری سیستم خبره باید توجه لازم انجام گیرد. یکی از نیازهای اصلی در رابط کاربری نحوه سوال پرسیدن است. برای آنکه اطلاعات قابل اعتماد از کاربر بدست آید، باید به نحوه طراحی سوالات دقت کافی بشود. برای همین منظور ممکن است مجبور به بکارگیری منوها، گرافیک، نمودار و دیگر ابزار تعاملی با کاربر شویم.
حتی ممکن است ابزاری جهت تغییر و مشاهده اطلاعات درون حافظه کاری فراهم سازیم. زمانیکه کاربر بخواهد پاسخ سوالات قبلی را تغییر دهد این ابزار کمک خواهند کرد.

۹- ویژگیهای یک سیستم خبره:

ویژگی‌های یک سیستم خبره شامل:

۱- جدایی دانش از کنترل
۲- داشتن دانش خبره
۳- تخصص متمرکز
۴- استدلال با سمبلها
۵- استدلال بصورت ابتکاری
۶- توانایی ارائه استدلال نادقیق
۷- تنها به مسائل قابل حل محدود می‌شود
۸- با پیچیدگی معقول قابل توسعه است
۹- می‌تواند اشتباه کند

۹-۱- جدایی دانش از کنترل

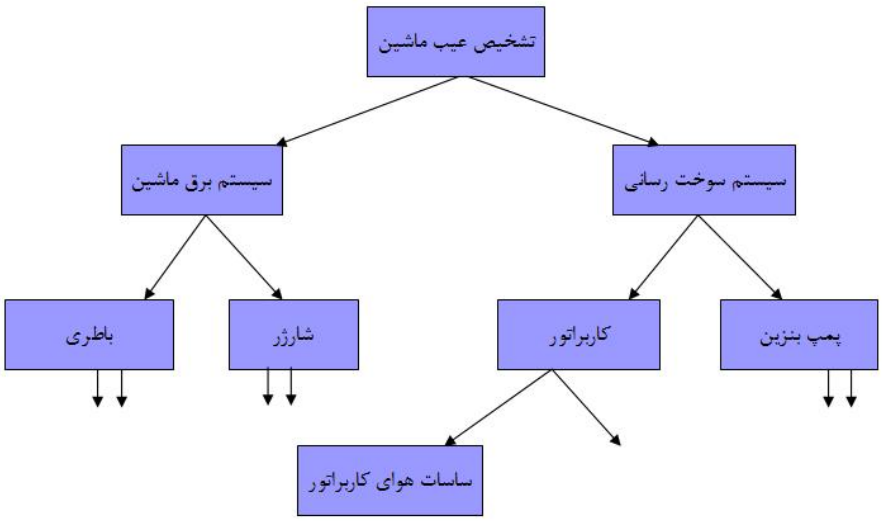
همانطور که در شکلهای قبلی نشان داده شد، پایگاه دانش از موتور استنتاج مجزا ست. این ویژگی وجه تمایز سیستم خبره از نرم افزارهای متعارف است و مزایایی به سیستم خبره می‌دهد. در برنامه‌های متعارف که دانش و کنترل باهم عجین هستند، تغییرات برنامه هم در دانش و هم در فرآیند کنترل برنامه تاثیر خواهد گذاشت. از طرفی بررسی برنامه و فهم دانش و روش استدلالی برنامه مشکل می‌شود.
جداسازی دانش از کنترل، نگهداشت و توسعه سیستم را آسانتر می‌سازد. می‌توان به راحتی قانونی را تغییر داد یا قانون جدیدی را اضافه کرد. اگر لازم باشد روش استدلالی تغییر کند کافی است الگوریتم استنتاج تغییر کند و به دانش کاری نخواهیم داشت.

۹-۲- داشتن دانش خبره

یکی از ویژگی‌های یک سیستم خبره آن است که تخصص یک فرد خبره را درونی می‌سازد. تخصص منبع با ارزشی است که تعدادی مشخص از افراد آن را دارند. افرادی که می‌توانند با موفقیت مسائلی را حل کنند که بقیه در حل آن عاجز هستند. بدست آوردن این سیستم خبره و کد کردن آن خبرگی بسیار ارزشمند خواهد بود.

۹-۳- تخصص متمرکز

اغلب افراد خبره در شاخه کوچکی از تخصصشان توانا هستند. اما در خارج آن محدوده توانایی کمتری دارند. همانند انسان یک سیستم خبره نیز در دامنه خاصی از مسئله تواناست. مثلاً از سیستمی‌که برای تشخیص عیب ماشین طراحی می‌شود، انتظار توانایی حل مسائل مالی نمی‌رود.
بهترین نتایج ارائه شده در سیستم‌های خبره‌ای بوده که برای دامنه مشخص و محدودی از مشکلات ایجاد شده‌اند. سیستم‌های خبره‌ای که دامنه وسیعی را هدف گرفته‌اند موفقیت چندانی نداشته‌اند.
مثال: برای ایجاد سیستم تشخیص عیب ماشین، یک سیستم جامع می‌توان ایجاد کرد اما مدیریت آن بسیار مشکل خواهد شد. لذا منطقاً سعی می‌کنیم که آن را به زیر سیستم‌های قابل مدیریت تقسیم کنیم.



شکل (۳) روند استدلال یک سیستم خبره

۹-۴- استدلال با سمبلها

سیستم‌های خبره دانش را به فرم سمبلیک ارائه می‌کنند. می‌توانیم با کمک سمبلها دامنه وسیعی از گونه‌های مختلف دانش را ارائه کنیم مانند حقایق، مفاهیم، قوانین. بخشی از هوش مصنوعی روی نحوه ارائه دانش متمرکز شده‌است. یکی از روشهای ارائه دانش با کمک سمبل‌هاست.

برای مثال:

علی تب دارد.

مردمی‌که تب دارند باید مقداری آسپرین بخورند.

Fever (Ali) ↔ علی تب دارد

با کمک ارائه دانش با زبان سمبلها، سیستم خبره می‌تواند مسایل را حل کند.

برای مثال:

ادعا	علی تب دارد	fever (Ali)
قانون	اگر مردم تب داشته باشند آنگاه باید مقداری آسپرین بخورند	X takes (aspirin) → fever(X)
نتیجه	علی باید مقداری آسپرین بخورد	Ali takes Asprin

منطق گزاره‌ای، از عملگر استدلال Implies یا → برای نمایش ساختار یک قانون کمک می‌گیرد.

از متغییر X برای آنکه عناصر مختلف بتوانند در قانون صدق کنند کمک می‌گیریم. وهمانطور که دیده می‌شود به جواب می‌رسیم.

۹-۵- استدلال به صورت ابتکاری

اغلب افراد خبره از خبرگیشان کمک می‌گیرند و مسائل را به صورت ابتکاری حل می‌کنند.

مثلاً ممکن است یک فرد خبره بگوید:

همیشه اول سیستم برق ماشین را چک می‌کنم.

مردم به ندرت در تابستان سرما می‌خورند.

اگر به سرطان مشکوک شوم، همیشه تاریخچه خانوادگی را بررسی می‌کنم.

افراد خبره از ابتکار برای میانبری به حل مسئله نگاه می‌کنند. با بکار گیری این خاصیت در سیستم خبره آن را از رویه‌های خشک برنامه‌های کاربردی متعارف متمایز می‌کند.

در برنامه‌های متعارف یک الگوریتم وجود دارد که اعمال خاصی را به ترتیب انجام می‌دهد. این الگوریتم همیشه عملیاتی را به ترتیب مشخص انجام می‌دهد. در حقیقت در برنامه‌های متعارف علاقه روی محاسبات عددی است.

در استدلال ابتکاری، بر اساس اطلاعات فراهم شده نتیجه گیری می‌کند، سمت و سوی مشخصی را دنبال نمی‌کند. بطور مثال برای تشخیص کاهش جریان در لوله‌ها می‌توان از روش ابتکاری زیر بهره برد:

ابتکار: لوله‌های قدیمی هنگام کاهش جریان عبوری می‌لرزند.

استدلال ابتکاری: اگر لوله‌ها می‌لرزند و لوله‌ها قدیمی‌هستند آنگاه به کاهش جریان عبوری باید مشکوک شد.

سیستم خبره می‌تواند از این قانون برای تشخیص کاهش جریان عبوری کمک بگیرد.
این قانون اصلاً تضمین نمی‌کند که حتماً فشار کاهش یافته‌است، بلکه جریان استنتاج را به سمت بررسی فشار جریان عبوری هدایت می‌کند. در حقیقت نقطه شروع بررسی مناسبی را برای سیستم خبره انتخاب می‌کند.
اگر این روش موفق نبود می‌توان از همان روشهای متعارف برای حل مسئله کمک گرفت.

۹-۶- توانایی ارائه استدلال نادقیق

سیستم‌های خبره در کاربردهایی که نیازمند استدلال نادقیق هستند بسیار خوب جواب داده‌اند. جایی که اطلاعات نامشخص، مبهم یا حتی در دسترس نباشد و یا دانش محیط کاربرد بصورت پایه‌ای شامل عوامل نادقیق باشد.
مثال، پزشک به سبب کمبود وقت در شرایط اورژانس و بدون انجام آزمایشات کافی تصمیم‌گیری مناسب می‌کند.
اطلاعات نادقیق:

ممکن است از حسن یک ساندویچ بخرم.
نتایج تست قلب را ندارم.
موتور ماشین داغ کرده‌است.

دانش نادقیق:

ساندویچ‌های حسن معمولاً خوب است.
اگر تست قلب نداشته باشیم، بیمار درد قفسه سینه داشته باشد، ممکن است ناراحتی قلبی داشته باشد.
مقداری روغن به موتوری که جوش می‌آورد اضافه کنید.

۹-۷- تنها به مسائل قابل حل محدود می‌شود

زمانیکه پروژه یک سیستم خبره آغاز می‌شود، باید مراقب باشید که آیا مسئله اصولاً قابل حل است یا خیر؟
این تصور اشتباه وجود دارد که هوش مصنوعی می‌تواند از پس حل تمام مسائل برآید. اگر فرد خبره‌ای وجود نداشته باشد تا مشکل را حل کند، امید بسیار کمی برای ایجاد سیستم خبره‌ای برای حل مشکل وجود دارد.
سیستم خبره برای مسائلی قابل ارائه‌است که فرد خبره‌ای برای حل آن مسئله وجود داشته باشد. مسائل نو یا مسائل با تغییرات زیاد کاندید خوبی نیستند.

۹-۸- با پیچیدگی معقول قابل توسعه است

مسئله باید پیچیدگی معقول داشته باشد، نه زیاد سخت باشد نه زیاد آسان. اگر مسئله خیلی آسان باشد، شاید صرف هزینه برای حل مسئله با کمک سیستم خبره مقرون به صرفه نباشد. البته برای برخی کاربردهای ساده، ممکن است اقدام به ارائه سیستم خبره بکنیم، چون حجم اطلاعات زیاد است یا حساسیت مسئله ایجاب می‌کند. مانند قیمت گذاری وسیله نقلیه.

مسئله نباید خیلی پیچده باشد، مسائلی که ساعت‌ها وقت کارشناس را می‌گیرد برای حل به کمک سیستم خبره مناسب نیستند. در صورت پیچیدگی باید به زیر مسائل ساده تر شکسته شود.

۹-۹- می‌تواند اشتباه کند

یک کارشناس یا فرد خبره از آنجایی که انسان است ممکن است اشتباه کند، این را می‌دانیم اما باز به فرد خبره اعتماد می‌کنیم. در یک سیستم خبره از آنجا که دانش فرد خبره است که استخراج می‌شود، ممکن است این دانش دچار اشکالاتی باشد. از این لحاظ برنامه‌های متعارف نسبت به سیستم‌های خبره برتری دارند، اما اگر در مقام جایگاه کاربرد در نظر بگیریم، می‌توان گفت این مقایسه نادرست است. برنامه‌های متعارف برای اطلاعات درست و دقیق بکار گرفته می‌شوند مانند پایگاه داده‌ها یا برنامه‌های حسابداری، اگر اطلاعات ناقص باشد، برنامه دچار خطا می‌شود. اما در یک سیستم خبره ممکن است با وجود کمبود اطلاعات نتیجه درست و مناسب توسط سیستم اتخاذ شود.

۱۰- مقایسه سیستم خبره با برنامه‌های کاربردی متعارف

سیستم‌های خبره	برنامه‌های متعارف
سمبلیک	عددی
ابتکاری	الگوریتم
دانش از کنترل مجزاست	اطلاعات و کنترل مجتمع است
اعمال تغییرات آسان است	اعمال تغییرات مشکل است
اطلاعات نا دقیق است	اطلاعات دقیق است
شامل گفتگو با کاربر به همراه توضیحات	رابط کاربری بصورت دستور است
توصیه و توضیحات لازمه ارائه می‌شود	نتیجه نهایی داده می‌شود
راه حل قابل قبول ارائه می‌شود.	راه حل بهینه ارائه می‌شود

۱۱- برنامه‌نویسی در مقابل مهندسی دانش

از زمان ایجاد کامپیوترها، افراد از آنها جهت توسعه برنامه‌هایی به منظور انجام سریعتر محاسبات، دستیابی به اطلاعات یا مدل کردن فرآیندهای پیچیده بهره برده‌اند. از تجربیات بدست آمده از ساخت خیلی از سامانه‌ها، مهندسان برنامه نویس به تکنیک‌ها و روشهای مناسبی جهت طراحی و پیاده‌سازی برنامه برای مسایل مختلف دست یافته‌اند.

اما سیستم‌های خبره هنوز نو هستند لذا به متدولوژی مناسبی جهت توسعه سیستم‌های خبره نرسیده‌ایم.

۱۱-۱- برنامه‌نویسی مرسوم

برنامه‌نویسی مرسوم معمولاً شامل سه بخش طراحی، پیاده‌سازی و رفع اشکال است. برنامه نهایی زمانی ارائه می‌شود که از تمام مراحل فوق با موفقیت گذشته باشد.

نقطه شروع برنامه‌نویسی مرسوم از نیازهای کابردی است (فاز طراحی). از همین جا برنامه‌نویس تصویری از برنامه نهایی‌را در ذهن خود دارد. در مرحله پیاده‌سازی برنامه نویس تنهاست و معمولاً تغییرات روی خصوصیاتی که در فاز طراحی در نظر گرفته شده پذیرفته نیست.
در مرحله رفع نقص، برنامه تست شده و بررسی می‌شود آیا به آن ویژگی‌های مورد انتظار رسیده‌است یا خیر.

۱۱-۲- مهندسی دانش (Knowledge Engineering)

در دنیای برنامه‌نویسی مرسوم «داده» عنصر اساسی است، تمرکز روی داده‌ها است و سعی داریم فرآیندی برای حل آنها پیدا کنیم. اما در سیستم‌های خبره تمرکز روی دانش است. آنها دانش را کسب، دسته‌بندی و بررسی می‌کنند تا به مسئله کاملاً فهمیده شود.
تعریف: مهندسی دانش، فرآیند ساخت یک سیستم خبره است.
برخلاف برنامه‌نویسی مرسوم، ایجاد سیستم خبره فرآیندی تعاملی است. طراحی بخشی از سیستم را ایجاد می‌کند تست کرده لذا در صورت لزوم دانش سیستم را اصلاح می‌کند. این فرآیند تا رسیدن به محصول نهایی ادامه دارد.

۱۲- مراحل ایجاد سیستم خبره

فاز اول- ارزیابی «سنجش» (Assessment) :

در این مرحله امکان پذیری انجام پروژه بررسی می‌شود. از مراحل مهم و اولیه است. در این مرحله نیازهای انجام پروژه مانند تعداد پرسنل و منابع مورد نیاز بدست می‌آید. منابع دانش مورد نیاز مانند افراد خبره و گزارشات لازم دیگر نیز مشخص می‌شود.

فاز دوم- اکتساب دانش (Knowledge Acquisition) :

هدف این فاز استخراج دانش مورد نیاز است. این دانش معمولاً از فرد خبره بدست می‌آید. این فاز شامل انجام جلسات با افراد خبره جهت مشخص شدن ابعاد مسئله، مفاهیم اصلی و روشهای حل مسئله توسط افراد خبره است. این فاز به عنوان گلوگاه ایجاد سیستم خبره است.

تعریف: اکتساب دانش فرآیند کسب، دسته‌بندی و بررسی دانش است.

فاز سوم- طراحی (Design) :

در این فاز ساختار کلی و سازمان دانش سیستم ایجاد می‌شود. ابزار نرم‌افزاری مناسب انتخاب می‌شود. در این فاز نمونه اولیه برنامه ایجاد می‌شود تا به درک مناسبی از صورت مسئله برسیم.

فاز چهارم- تست (Testing):

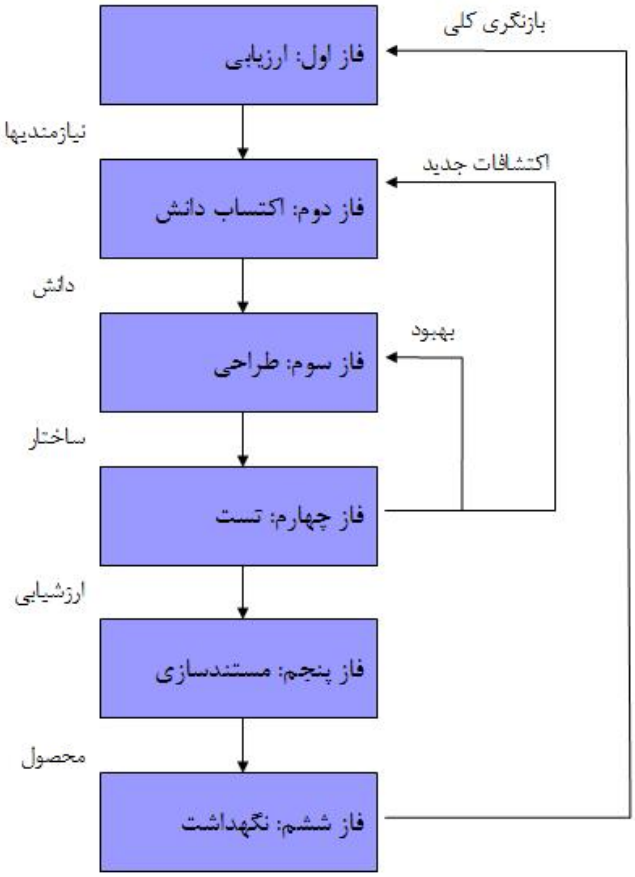
این فاز اگرچه جدا در نظر گرفته شده، اما تا پایان فرآیند تکاملی پروژه در حال انجام است. زمانیکه دانش جدیدی به سیستم اضافه می‌شود برنامه مورد تست جامع قرار می‌گیرد. در طی مرحله تست طراحی هم با فرد خبره جهت توسعه دانش سیستم و هم با کاربر جهت توسعه رابط کاربری همکاری نزدیک دارد.

فاز پنجم- مستندسازی (Documentation):

مستندسازی کارهای صورت گرفته در فرآیند تولید نرم‌افزار یکی از نیازهای اصلی پروژه نرم‌افزاری است. مستندات جهت آموزش، راه‌اندازی سیستم و بیان مشخصات کلی سیستم است.

فاز ششم- نگهداشت (Maintenance):

سیستم بعداز تحویل باید بصورت متناوب و دوره‌ای مورد بررسی قرار گرفته و پشتیبانی شود. همانند یک کودک سیستم خبره بزرگ می‌شود و می‌آموزد. دانش ثابت نیست، توسعه می‌یابد، رشد می‌کند و کامل می‌شود. حتی ممکن است تغییرات بنیادی باشد و ویژگی‌های سیستم را نیز دگرگون کند.



شکل (۴) فازهای پیاده‌سازی یک سیستم خبره

۱۳- برنامه‌نویسی در مقابل تولید سیستم خبره

تفاوتهای اساسی این دو فرآیند شامل موارد زیر است:

تولید سیستم خبره	برنامه‌نویسی مرسوم
تمرکز روی مسئله است	تمرکز روی راه‌حل است
کار گروهی است	برنامه‌نویس به تنهایی کار می‌کند
فرآیند تولید تکراری است.	فرآیند تولید پله‌ای است

۱۴- افراد درگیر در پروژه سیستم خبره

افراد موثر در پروژه تولید یک سیستم خبره شامل، کارشناس محیط کاربرد، مهندس دانش و کاربر نهایی است. که هرکدام در طی فرآیند نقش کلیدی خود را ایفا می‌کنند. مشخصات کلی آنها:

فرد خبره	- دانش کارشناسی دارد - توانایی کافی در حل مسائل را دارد - توانایی ارائه دانش را دارد - می‌تواند زمان بگذارد - خصومتی با انجام پروژه ندارد
مهندس دانش	- توانایی مهندسی دانش را دارد - تونایی برقراری ارتباط قوی را دارد - توانایی‌تطبیق مسئله را به نرم‌افزار را دارد - توانایی برنامه‌نویسی سیستم خبره‌ای دارد
کاربر نهایی	- می‌تواند در تعریف محیط کاربری موثر باشد - می‌تواند در کسب دانش کمک باشد - می‌تواند در توسعه سیستم کمک باشد.

۱۴-۱- کارشناس محیط کاربرد (Domain Expert)

تعریف: کارشناس محیط کاربرد فردی است که توانایی و دانش لازم جهت حل یک مسئله را بصورت بهتری نسبت به بقیه دارد.

وجود محیط کابرد بسیار مهم است، زیرا می‌خواهیم راجع به یک مسئله مشخص و محدود کار کنیم.

وجه تمایز یک فردخبره (کارشناس) با یک فرد ناخبره در دانش است. فرد خبره دانش حل مسئله را دارد. فرد خبره نسبت به فرد ناخبره این توانایی را دارد که اطلاعات مهمتر را طوری کنار هم قرار می‌دهد تا در حل مسئله بطور موثری بکار آیند.

افراد خبره تخصص‌شان را در حل مسائل در طی سالیان زیادی کسب کرده‌اند لذا می‌توانند بطور خلاصه رویه‌ای جهت حل موثر مشکلات ارائه دهند. افراد خبره باید بتوانند دانش خود را به راحتی بیان کنند تا به صورت کد در سیستم خبره درآید.

معمولاً افراد خبره افرادی ارزشمند هستند لذا باید زمان کافی برای همکاری در پروژه از این افراد را بدست آورد. زمان ایجاد یک سیستم خبره معمولاً طولانی و بصورت تکاملی است.

فرد خبره باید همکاری لازمه را در اجرای پروژه داشته باشد. اگر خصومتی در انجام کار نشان دهد پروژه مطمئناً با شکست مواجه خواهد شد. معمولاً افراد خبره نگاه خوبی نسبت به سیستم‌های هوشمند ندارند. لذا باید بسترسازی مناسب انجام گیرد.

۲-۱۴- مهندس دانش (Knowledge Engineer)

تعریف: مهندس دانش، فردی است که طراحی، ساخت و تست یک سیستم خبره را بر عهده دارد.
در نگاه اول، مهندس دانش و برنامه‌نویس نزدیک بهم هستند، چون هر دو کد می‌کنند. اما تفاوتهایی به شرح زیر دارند:
مهارت‌های مهندس دانش: مهندسی دانش، بخشی از ساخت یک سیستم خبره است. در حقیقت نوعی هنر است. زیرا فرآیندی پیچیده دارد و دستورالعمل‌های خیلی مشخصی برای انجام این فرآیند وجود ندارد. اما آنچه مشخص است آن است که مهندس دانش باید بتواند مشخص کند که حل مسئله با سیستم خبره ممکن هست یا خیر؟
ارزیابی امکان انجام پروژه شامل تحقیق روی مشخصات مسئله است. برخی مسائل برای حل به کمک سیستم خبره بسیار مناسب هستند در حالیکه برخی دیگر از مسایل با سیستم خبره به سختی حل می‌شوند. در حقیقت امکان سنجی برای پروژه سیستم خبره از رویه‌های سفت و سخت آنالیز هزینه‌ها و درآمدها که در مهندسی نرم‌افزار متعارف انجام می‌گیرد بهره نمی‌گیرند. معمولاً علاقه بکارگیری تکنولوژی جدید و نگهداشت دانش درون سازمان می‌تواند انگیزه‌ای قوی در جهت تایید انجام پروژه باشد.
در طی فرآیند اکتساب دانش، هدف اصلی مهندس دانش آن است که پرده از دانش فرد خبره بردارد. از طریق مصاحبه‌های مختلف، مهندس دانش بدنبال مفاهیم کلیدی مسئله و راه‌حلهایی که توسط افراد خبره در حل مسئله انجام می‌دهند، می‌گردد.
مهندس دانش باید از مهارت لازم جهت هدایت جلسه مصاحبه بر خوردار باشد تا در نهایت بتواند دانش فرد خبره را بطور موثر و در زمان کوتاه‌تری استخراج کند.
بعد از آن مهندس دانش، مجموعه دانش استخراجی را طوری دسته‌بندی می‌کند که به طور موثر در سیستم خبره قابل اعمال باشد.
مهندس دانش پکیج نرم‌افزاری مناسبی را انتخاب خواهد کرد که توانایی پذیرش دانش استخراجی باشد و شامل فرآیند استنتاج باشد.
مهندس دانش همچنین مسئول کد کردن، تست کردن و تجدید نظر کردن سامانه است تا زمانیکه سامانه به درجه‌ای از بلوغ رسیده باشد که کارایی یک فرد خبره را نشان بدهد. البته بعد از پایان پروژه مهندس دانش همچنان مسئول نگهداشت و توسعه سامانه خواهد بود.
مهارت‌های ارتباط موثر: بخش زیادی از موفقیت مهندس دانش در نحوه بیرون کشیدن دانسته‌های فرد خبره است. همان‌طور که گفته شد فرد خبره تمایل زیادی دارد که دانشش را طوری ارائه دهد که فرآیند حل سریع و کارا باشد پس آنها فرآیند استدلالی را عجیب خواهند کرد. با گرفتن مقادیری اطلاعات اولیه، به سرعت به نتیجه نهایی‌خواهند رسید. برای یک سیستم خبره که می‌خواهد در سطح یک فرد خبره نمایان شود، دانش پنهان در روند فرآیند استدلالی آشکار شود. این فرآیند مشکل را تجزیه دانش Knowledge Decomposition می‌نامیم. که خود مهارت خاص دارد و باید به مهندس دانش آموخته شود.

توانایی انطباق مسئله به نرم‌افزار: تعداد زیادی زبان برنامه‌سازی و ابزار توسعه سیستم‌های خبره (که به آنها پوسته یا Shell گفته می‌شود) وجود دارد تا با آنها بتوان سیستم خبره را ایجاد و توسعه داد. هر یک از آنها ویژگی‌های خاصی در حل دسته‌ای خاص از مسائل دارند. وظیفه مهندس دانش است که نرم‌افزار مناسب را انتخاب کند. مهندس دانش سعی می‌کند تا ارتباطی مناسب بین مسئله و نرم‌افزار برقرار سازد. افراد خبره گونه‌های مختلفی از دانش را بکار می‌گیرند، آنها را با روشهای مختلفی برای خودشان دسته‌بندی و مرتب می‌کنند و از روشهای استدلالی مختلفی بهره می‌برند تا مسئله را حل کنند. بهمین نسبت هم زبانهای مختلف و پوسته‌های مختلف می‌توانند دانش را به روشهای مختلفی ارائه، دسته‌بندی و پردازش کنند. بهمین دلیل برای مهندس دانش مهم است که همخوانی بین نیازمندیهای مسئله و توانایی‌های نرم‌افزار برقرار سازد.

مهارت برنامه‌نویسی مهندس دانش: مسئول اصلی کد کردن دانش استخراج شده از فرد خبره، مهندس دانش است. لذا ایشان باید در برنامه‌نویسی سیستم خبره مهارت لازم را داشته باشد. برای این منظور باید اول بدانیم که چگونه می‌توان دانش را ارائه کرد که در فصل بعد نگاهی به آن خواهیم انداخت. در فصل چهارم پردازش آنها را بررسی خواهیم کرد. در حقیقت یکی‌از اهداف اصلی این درس آن است که این مهارت را به شما ارائه دهد.

۱۴-۳- کاربر نهایی (End User)

کاربر نهایی فردی است که بطور مستقیم با سامانه کار می‌کند. پذیرش نهایی سیستم بقدار زیادی به این بستگی دارد که چقدر از نیازهای کاربر نهایی مرتفع می‌شود. تاریخچه سیستم‌های خبره پر از سیستم‌هایی است که به لحاظ تکنیکی درست ساخته شده بودند اما هرگز بکار گرفته نشدند، زیرا نیازهای کاربر نهایی اصلاً در نظر گرفته نشده بود.

تعریف ویژگی‌های رابط کاربری: سیستم خبره نهایتاً باید ویژگیهای رابط کاربری که کاربر نهایی انتظار دارد را داشته باشد. مشخصاتی مانند: دسترسی سیستم، ورود اطلاعات، توضیحات سیستم، فرم نتایج نهایی و ابزارهای کمکی مورد نیاز و ...

کاربر نهایی مشخص می‌کند که سیستم خبره چگونه راه‌اندازی شود، خواه از طریق منو یا از طریق یک برنامه دیگر، یا اصلاً با روشن شدن کامپیوتر فعال شود.

روشهای مختلفی برای ورود اطلاعات به یک سیستم خبره وجود دارد. باید مشخص شود که کاربر نهایی چه روشی را برای پاسخ به سوالات گزینه‌ای، سوالات چند جوابی یا حتی تایپی مناسب می‌داند و می‌پذیرد.

کاربر نهایی ممکن است به توضیحات کافی برای فرآیند استدلالی سیستم احتیاج داشته باشد، مخصوصلاً زمانیکه بخواهد پاسخ نهایی سیستم را ارزیابی یا چک کند. لذا کاربر نهایی می‌تواند نحوه ارایه توضیحات را مشخص کند.

کاربر نهایی ممکن است به نرم‌افزارهای کمکی احتیاج داشته باشد که برخی احتیاجات جانبی‌اش را رفع کرده و سامانه را پشتیبانی کنند. برای مثال ممکن است بخواهد برنامه قدرت آن را داشته باشد که روی پایگاه داده‌ای خارجی دست برده و اطلاعات آن را تغییر دهد یا فایل با فرمت صفحه گسترده ایجاد کند. طراحی رابط کاربری سیستم می‌تواند کاری پیچیده باشد، حتی نصف بودجه پروژه را مصرف کند. اما امری حیاتی است که موفقیت پروژه در گرو آن است و کاربر نهایی نقش تعیین کننده‌ای در آن دارد. بدون پشتیبانی کاربر نهایی، هرچقدر قدرتمند باشد، در تست آخر قبول نخواهد شد.

کاربر نهایی نقش اساسی در فرآیند آغازین پروژه دارد. اغلب مهندسین دانش تازه کار در ابتدای پروژه مبهوت جنبه‌های ریز مسئله می‌شوند زیرا آنها اول از فرد خبره شروع می‌کنند. در حالیکه فرد خبره همیشه برای توضیح مسئله وارد جزئیات زیاد و گاهی هم غیر ضروری می‌شود که تنها سر درگمی مهندس دانش را در پی دارد، در حالیکه ابتدا باید از کاربر نهایی شروع کرد زیرا او فهم گسترده‌ای از آنچه می‌خواهیم، را ارائه می‌دهد و در مرحله بعد فرد خبره جزئیات لازم را برای پر کردن فضاهای خالی صورت مسئله، ارائه می‌کند.