

UZMAN SİSTEMLER

Prof.Dr. Asuman DOĞAÇ^(*)

I. GİRİŞ

Bilgisayarların ortaya çıkması ile beraber, onların yalnızca hesap yapan makinalar olarak kalmayıp insan gibi düşünebilmelerini sağlamak için çalışmalar yapılmaya başlanmıştır. Bilgisayarların "akıllı" davranmalarını sağlayabilmek için yapılan çalışmalar "yapay zeka (artificial intelligence)" adı altında toplanır. Karar verme sistemleri, robotlar, ses ve görüntü tanımayaya yönelik çalışmalar bu konunun kapsamındadır. Yapay zeka ile ilgili çalışmaların çoğu henüz araştırma laboratuvarlarından çıkıp ticari, amaçla kullanılabilir hale gelmemiştir. Bu konudaki en önemli istisna "uzman sistemler (expert systems)"dir.

Günümüzde çok değişik alanlarda uzman sistemler geliştirilmektedir. Bunlar arasında yöneticilere karmaşık planlama problemlerinde yardımcı olan sistemleri, hastalık teşhisinde, maden yatakları bulmada ya da karmaşık bilgisayar donanım yapılarını verilen şartlara uygun olarak tasarlamada veya eğitimde kullanılan sistemleri sayabiliriz.

Stanford Üniversitesinden Prof. Feigenbaum uzman sistemleri şöyle tanımlamıştır.[6]:

"Uzman sistem, çözümü bir uzmanın bilgi ve yeteneğini gerektirecek zorluktaki problemleri, bilgi ve mantıksal çıkarım kullanarak çözebilen akıllı bir bilgisayar programıdır. Bu programın kullandığı bilgi ve mantıksal çıkarım mekanizmasının, problemi çözmede uzman kişi veya kişilerin bilgi ve mantıksal çıkarım mekanizmasını modellemesi gerekir."

Bildirinin ikinci bölümünde, çok popüler bir uzman sistem MYCIN tanıtılacaktır. Bildirinin üçüncü bölümünde diğer uzman sistemlerden kısaca bahsedilecektir. Dördüncü bölüm, uzman sistemlerin yapısına ayrılmıştır. Bildirinin son bölümünde ise uzman sistemlerin yararları ve kısıtları anlatılacaktır.

II. MYCIN

Uzman sistemler konusunda daha iyi bir fikir edinebilmek için, 1970'li yılların ortalarında Stanford Üniversitesinde geliştirilen MYCIN sistemine bir göz atalım, MYCIN geniş çaplı olarak geliştirilen uzman sistemlerin ilk örneğidir ve menenjit ve kandaki bakterilerin neden olduğu hastalıkların teşhisinde kullanılmaktadır.

Bilindiği gibi bu hastalıklar ölümcül olabilmekte ve genellikle nekahat dönemindeki hastalarda örneğin kalp ameliyatı sonrası iyileşme dönemindeki hastalarda kendini göstermektedir. Tedavinin erken başlaması, sonucu etkilemekte, bu nedenle doktorlar tahlil sonuçlarının tamamı ellerine geçmeden tedaviye başlamak zorunda kalmaktadırlar. Teşhis ve tedavi oldukça karmaşık olduğundan, hastanın doktoru bu konuda uzman başka bir doktora danışmak gerektiğini duymaktadır.

MYCIN, hastanın doktoruna menenjit ve kandaki bakterilerin neden olduğu hastalıklar konusunda, ancak bu konuda uzman bir doktorun verebileceği teşhis ve tedavi yöntemini bildiren bir bilgisayar programıdır. Stanford Üniversitesi hastanesinde MYCIN'i kullanmak isteyen doktor, MYCIN'in yüklenmiş olduğu DEC-20'ye bağlı bir terminale oturmakta ve program tarafından sorulara cevap vermektedir. Bu sorular, hastanın rahatsızlık belirttilerini, hasta hakkındaki genel bilgileri (örneğin yaşı ve cinsiyeti gibi) ve tahlil sonuçlarını içermektedir. Bu arada MYCIN, henüz sonucu elde edilememiş bir tahlil hakkında bilgi istediğinde doktor 'HENÜZ BİLİNİYOR' cevabını verebilmektedir. Çünkü MYCIN uzman bir kişi gibi eksik bilgi ile de karar vermeye devam edebilmektedir. Sonuç olarak MYCIN teşhisini ve önerdiği tedavi yöntemini doktora bildirmektedir.

MYCIN'in teşhis ve tedavide gösterdiği başarıyı bu konuda uzman doktorları ile karşılaştırmak üzere Stanford Üniversitesinde şöyle bir deney yapılmıştır: Sekiz uzman, dokuz hasta üzerinde, Stanford Üniversitesi Hastanesindeki on uzman doktorun teşhis ve tedavi

(*) ODTÜ Bilgisayar Mühendisliği Bölümü



'Program hiçbir şeyi sormayı unutmaz, teşhis çok açık bir şekilde belli olduğu durumlarda bile, sonucu vermeden önce bütün olasılıkları gözden geçirir.'

yöntemlerini MYCIN ile karşılaştırmışlardır.

Uzmanlar bu incelemeyi kesin tahlil sonuçları elde edilmeden ve hangi sonucun bilgisayar (MYCIN) hangisinin doktorlar tarafından verildiğini bilmeden yapmışlardır. İncelemede iki kriter kullanılmıştır: Birincisi, önerilen tedavi yönteminin, kesin tahlil sonuçları elde edildiğinde de geçerli olup olmadığıdır. Bunda MYCIN ve üç uzman doktor, dokuz hasta içinde geçerli tedavi yöntemini bulmakta başarı göstermişlerdir. İkinci kriter de, önerilen tedavi yönteminin aşırıya kaçmadan ortaya çıkabilecek diğer komplikasyonlarda da etkili olup olmadığıdır. Bu kriter kullanıldığında MYCIN'in bütün uzman doktorlardan daha başarılı olduğu gözlenmiştir. Bu konuda MYCIN'in önerileri %65 başarılı bulunmuş, uzman doktorların başarılarının ise %42.5 ile %62.5 arasında değiştiği gözlenmiştir.

MYCIN'in, uzman doktorlar karşısında elde ettiği bu başarının sebeplerini şöyle özetleyebiliriz:

1. MYCIN'in kullandığı bilgi, yine konusunda en iyi uzman olan kişilerden elde edildiğinden, bilgi açısından bir uzman kişi ile yarışabilmektedir.
2. MYCIN, bilgisayar programı olması nedeniyle, hiç bir ayrıntıyı gözden kaçırmaz ve hiç bir şeyi unutmaz.
3. Program hiç bir şeyi sormayı unutmaz, teşhis çok açık bir şekilde belli olduğu durumlarda bile, sonucu vermeden önce bütün olasılıkları gözden geçirir.
4. Konudaki bazı yeni buluşları ancak bu konudaki araştırmaları yakından takip eden doktorların bilmesine karşın, bu bilgiler devamlı olarak MYCIN'e verilmektedir.

Bu arada dikkat edilmesi gereken bir nokta da MYCIN'in 50-insan-senelik bir emek sonucu ortaya çıkmış olduğudur.

MYCIN sistemini geliştiren araştırmacılar bu sistemin aslında iki bölümden oluştuğunun bilincine vardılar. Bilgi tabanı (knowledge base) dediğimiz birinci bölümde, MYCIN'in uzmanlık alanına giren bilgiler bulunmaktadır. Bu bilgi üzerinde çalışan mantıksal çıkarım makinası (inference engine) dediğimiz bölüm ise genel amaçlıdır, yani bilgi tabanının bilgi içeriği değiştiğinde de çalışabilmektedir. Bu ayırım 'empty' MYCIN, ya da EMYCIN adı verilen, ve MYCIN benzeri uzman sistemlerin geliştirilmesinde kullanılan bir aracın ortaya çıkmasını sağla-

mıştır. Bu tip araçlara, Bilgi Mühendisliği araçları (Knowledge Engineering Tools) denmektedir.

EMYCIN kullanılarak geliştirilen uzman sistemlerden bazıları şunlardır:

PUFF: Nefes alma testleri sonucu elde edilen bilgiler ile ciğerlerin hava alma kapasitesinin, kana karışan oksijen miktarının ve kandan temizlenebilen karbondioksit miktarının tesbiti.

GRAVIDA: Gebelikle ilgili tavsiyelerde bulunmaktadır.

III. DİĞER UZMAN SİSTEMLER

Günümüze kadar pek çok uzman sistem geliştirilmiştir. Bunlardan bazıları ve uzmanlık alanları şöyledir:

-GUIDON- Bu sistem bakterilerin sebep olduğu kan hastalıklarının ve menenjitin teşhis ve tedavisi konusunda uzman doktor yetiştirmek üzere geliştirilmiştir. GUIDON, MYCIN'in o güne kadar teşhis ve tedavide bulunduğu bütün vakaları içermektedir. GUIDON'un bunlara ilave olarak ayrı bir mantıksal çıkarım mekanizması vardır.

GUIDON kullanmak isteyen bir tıp öğrencisi sisteme bir menenjit vakası görmek istediğini söyler. GUIDON belgesinde bulunan vakalardan herhangi biri hakkında daha önce MYCIN'e verilmiş olan ön bilgileri öğrenciye verir. Sonra öğrenciye bu vaka hakkında teşhisini ve tedavi yöntemini sorar. Dikkat edilecek olursa burada GUIDON'la kullanıcısı arasındaki ilişki, MYCIN'le kullanıcısı arasındaki ilişkinin tam tersidir.

GUIDON öğrenciye verdiği vakayı MYCIN kullanarak çözer ve keşfi mantıksal çıkarımının hangi safhalardan geçtiğini tesbit eder. Bu arada öğrenci GUIDON'a sorular sormaya başlar. Bu sorulardan GUIDON, öğrencinin mantıksal çıkarım safhalarının kendisinininkine uyup uymadığına bakar. Böylece öğrencinin sorduğu bazı soruları ilgisiz veya gereksiz bulabilir. O zaman öğrenciye bu soruların neden gereksiz olduğunu açıklar. Sonuçta öğrencinin bulunduğu teşhis ve tedavi MYCIN'inki ile aynı ise, GUIDON öğrencinin bu teşhisi şans eseri bulmuş olabileceğini hesaba katarak, öğrencinin teşhisi iyice anlamasını sağlayacak sorular sorar.

Eğer öğrenci henüz MYCIN teşhise ulaşmadan yani eksik bilgi ile bir teşhis yapmışsa, GUIDON öğrenciye, böyle bir teşhise ulaşmak için eksik olan bilgileri sıralar. Görüldüğü gibi GUIDON uzman sistemi eğitimde kullanılan bir sistemdir ve daha önce teşhis ve tedavi için geliştirilen MYCIN uzman sisteminin tamamı üzerine eklemelerle geliştirilmiştir. Bu da bize bir uzman sistemin uzmanlık öğreten bir sistem haline getirilebileceğini göstermektedir.

- DENDRAL- Uzman sistemi bilinmeyen bir molekülün spektroskopik analizini inceleyerek molekül yapısına karar vermektedir.
- MACSYA- Karmaşık matematik problemlerinin çözümünde kullanılmaktadır.
- INTERNIST- Hastalık teşhisinde kullanılmaktadır.

- PROSPECTOR- Jeologlara başlıca şu üç konuda yardım etmek üzere geliştirilmiştir:
 - 1) Bir madenin bulunabileceği bölgeler.
 - 2) Bir bölgede bulunabilecek madenler.
 - 3) Maden ocaklarının açılması için en uygun yerlerin seçimi.
- R1 (XCON) - Müşterinin gereksinimini karşılayacak VAX -II bilgisayar sisteminin yapısını tesbit etmektedir.
- MOLGEN - Verilen bir DNA molekülünün yapısını belirlemek üzere yapılması gereken deneyleri tasarlamaktadır.
- DELTA (CATS-1) - Bu «sistem General Electric şirketi tarafından, şirketin dizel-elektrik lokomotiflerindeki arızaların tesbiti ve giderilmesi için kullanılmaktadır.
- SHRINK- Ruh hastalıklarının teşhisinde kullanılmaktadır.
- HEARSAY- Konuşmaların anlaşılmasında kullanılmaktadır.

IV. UZMAN SİSTEMLERİN YAPILARI

Uzman sistemler daha önce de belirtildiği gibi bir bilgi tabanı (knowledge base) ve bu bilgi tabanından mantıksal çıkarım yapılmasını sağlayan bir mantıksal çıkarım ve kontrol mekanizmasından (inference and control mechanism) oluşur.

Bilgi tabanında bilgi saklamanın ve mantıksal çıkarım yapmanın çeşitli yolları vardır. Çok basit bir örnekle bilginin nasıl işlenebileceği hakkında bir fikir sahibi olabiliriz. Bu örnekte, okunacak materyalin gazete, kitap ve dergilerden oluştuğunu varsayalım. Okumak için bu materyali;

- satın almak veya kütüphaneden temin etmek gerektiğini,
- kitap ve dergilerin pahalı, gazetelerin ucuz olduğunu,
- kütüphanelerden okunacak materyalin ancak kütüp-

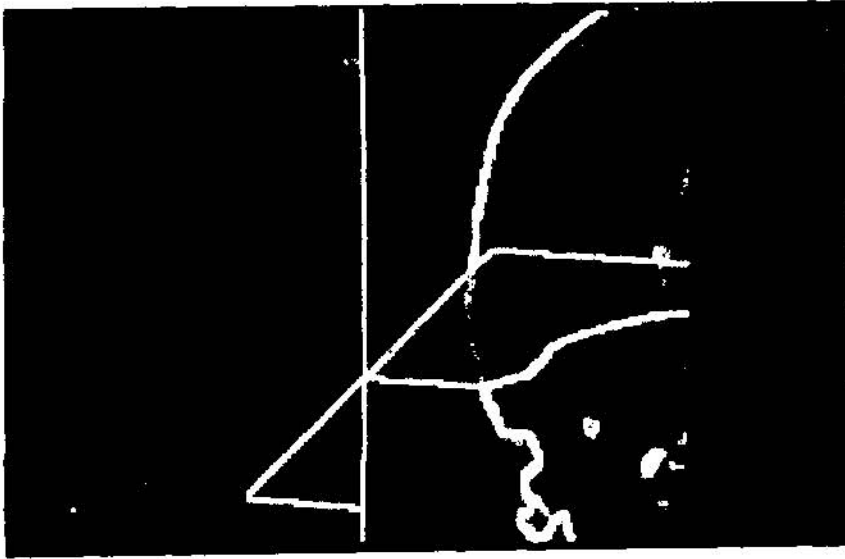
hane yakın ise temin edilebildiğini,

- yakın bir kütüphane bulabilmek için çok sayıda kütüphane olması gerektiğini,
- kütüphane sayısının az olduğunu, ve
- bir şeyin ancak ucuzsa satın alınabileceğini varsayalım. Bu bilgileri içeren bir bilgi tabanına daha sonra "Okunan materyal nedir?" diye sorduğumuzda yukarıda verilen bilgi üzerinde bir mantıksal çıkarım yaparak, okunan materyalin gazete olduğunu söyleyecektir.

Bilgi tabanında bilgi saklamanın yollarından biri de yüklem mantığının (predicate logic) Horn cümleleridir. Horn cümleleri kullanılarak yukarıdaki bilgi şöyle ifade edilir.

- 1) pahalıdır (kitap). /* kitap pahalıdır*/
- 2) pahalıdır (dergi). /* dergi pahalıdır 7
- 3) ucuzdur (gazete). /* gazete ucuzdur*/
- 4) az (kütüphane). /* kütüphane sayısı azdır*/
- 5) okunur (x): - satın-alınır (x).
/*bir şeyi okumak için satın almak gerekir*/
- 6) okunur (x): - ödünç- alınır (x, kütüphane).
/* bir şeyi okumak için kütüphaneden ödünç almak gerekir*/
- 7) satın-alınır(x): - not (pahalıdır (x)); ucuzdur (x).
/* bir şey ancak pahalı değilse veya ucuzsa satın alınır*/
- 8) ödünç-alınır (x, kütüphane): - yakın (kütüphane).
/* bir şey kütüphaneden ancak kütüphane yakınsa ödünç alınır*/
- 9) yakın (kütüphane): - not (az (kütüphane)).
/* yakın kütüphane bulunabilmesi için kütüphane sayısının az olmaması gerekir*/

Yukarıda bilgi tabanını ifade etmek için kullandığımız dil PROLOG dilidir [9]. PROLOG dili mantıksal çıkarım ve



"Uzman sistemler bir bilgi tabanı ve bu bilgi tabanından mantıksal çıkarım yapılmasını sağlayan bir mantıksal çıkarım ve kontrol mekanizmasından oluşur."



ttT
*Uzman
sistemleri orta
düzeydeki yönetici
robotlar olarak
görmek
mümkündür.***

kontrol mekanizmasını, çözme (resolution) ve geriye doğru zincirleme (backward chaining) yöntemleriyle sağlar.

Şimdi bu bilgi tabanına "Ne okunur?" diye sordüğümüzde:

? - okunur (Ne).

bize cevabını şöyle verecektir: 5 nolu kuraldan bir şeyin okunması için satın alınması gerektiğini, 7 nolu kuraldan bir şeyin satın alınması için pahalı olmaması veya ucuz olması gerektiğini, 3 nolu gerçekten gazetesinin ucuz olduğunu ortaya çıkarıp bize "gazete" cevabını verecektir.

Eğer "kitap okunur mu?" dersek:

? - okunur (kitap).

Önce, aynen yukarıdaki mantıkla 5 ve 7 nolu kurallarla 1 nolu gerçekten kitabın okunmadığına karar verecektir. Ama bu sonucu bize vermeden önce ödünç alınan şeylerin de okunabileceğini bildiğinden, bu sefer 6,8 ve 9 nolu kurallarla 4 nolu gerçeği kullanıp yeni bir çıkarım yapacaktır. Bu çıkarım da kitabın okunmadığı gerçeğidir.

- A. Bilgi Tabanında Bilgi Saklamanın Diğer Yolları
Bilgi tabanında bilgi saklamanın diğer yolları arasında
1. Anlamsal ağlar (Semantic Networks)
 2. Çerçeveler (Frames)
 3. Üretim kuralları (Production Rules) sayılabilir.

1. Anlamsal ağlarda en küçük yapı iki nokta ile arasındaki yönlü bir çizgiden oluşmaktadır. Her nokta bir kavramı, arasındaki ilişkiyi belirtmektedir, örneğin "Ali bankada çalışıyor" bilgisi anlamsal ağ ile

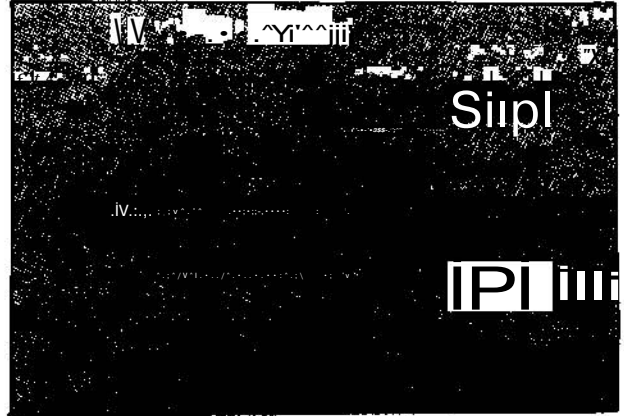
çalışıyor
Ali banka

şeklinde ifade edilir. Bir nokta bir sürü başka nokta ile bağlandığında bir ağ oluşur. Çizgilerin üzerindeki sözcükler yukarıda olduğu gibi bir fiili gösterebilir. Çizgilerin üzerinde "is-a", "is" veya "has" sözcüklerinin bulunması özel bir anlam içerir, "is-a" bir nesnenin başka bir

neşe grubunun üyesi olduğunu, "has" bir kavramın başka bir kavramın parçası olduğunu, "is" ise bir kavramın bir başkasının özelliği olduğunu gösterir. Şekil 1'de aşağıda verilen bilgilerin anlamsal ağ ile gösterimi verilmiştir.

Ali bankada çalışıyor.
Ali müdürdür.
AK kırk yaşındadır.
Ali'nin arabası vardır.

Anlamsal ağları LISP veya PROLOG dillerinde ifade etmek mümkündür [9].



Şekil 1. Anlamsal ağ

2. Nesnelerin ve olayların özelliklerinin prototipler haline düzenlenmesi sonucu çerçeveler ortaya çıkmıştır. Çerçevelerde nesnelerle ilgili bilgiler için boşluklar bulunur [4].

örneğin, bir "müdür" çerçevesi aşağıdaki şekilde görüldüğü gibi düzenlenebilir:

isim: MÜDÜR
isim:
yaş:
maaş:
çalıştığı yer:

Böyle bir çerçeve Ali için kullanıldığında boşluklar Ali ile ilgili bilgilerle doldurulur. Şekil 2'de MAAŞ başka bir çerçeveyi göstermektedir.



Şekil 2. 'Ali'ye ait çerçeve

Çerçeveleri de LISP veya PROLOG ile ifade etmek mümkündür [9].

3. Üretim kuralları bilgi tabanlarında bilgi saklamanın en çok kullanılan biçimidir. Üretim kuralları

"İF şart THEN eylem" yapısındadır [7].

Yani, belirli şart veya şartlar sağlandığında yapılması gereken işleri belirtirler. Bunlar arasında, kullanıcıya soru sorulmasını, bir program parçasının işlenmesini, ya da bir aletin çalıştırılmasını sayabiliriz. Üretim kuralları ile problem çözümüne bu kuralların uygulanması ile başlanır, elde edilen sonuçlar bilgi tabanında toplanır. Problemin sonucuna erişilene kadar kurallar, ara sonuçlara uygulanır.

B. Mantıksal Çıkarım ve Kontrol Mekanizması
Mantıksal çıkarım yöntemleri arasında "Modus Ponens" ve "Çözme (Resolution)" yöntemlerini sayabiliriz. "Modus Ponens", "A doğru ise B de doğrudur" şeklinde bir kural ile A'nın doğruluğu verildiğinde, B'nin doğru olduğunun çıkarımını sağlar.

Çözme yöntemi doğruluğu kabul edilen birtakım mantıksal cümlelerden çelişki yolu ile yeni bir gerçeğin çıkarılmasını sağlar.

Elimizde Eğer A doğru ise B doğrudur
Eğer B doğru ise C doğrudur
ve A doğrudur

bilgisinin bulunduğunu varsayalım. "Eğer A doğru ise B doğrudur" ifadesi "A doğru değildir veya B doğrudur" şeklinde yazılabilir.

Dolayısıyla elimizde

1. A doğru değildir veya B doğrudur
2. B doğru değildir veya C doğrudur
3. A doğrudur, cümleleri bulunmaktadır.

Çözme yönteminde önce C'nin doğru olmadığı kabul edilir. O zaman 2'den B'nin doğru olmadığına karar verilir, B'nin doğru olmadığından ve 1'den A'nın doğru olmadığı ortaya çıkar ki bu da "A doğrudur" ile çelişki yaratır. Bu durumda C'nin doğru olduğu sonucuna varılır.

Mantıksal çıkarım ve kontrol mekanizmasının kontrol kısmı, mantıksal çıkarımın nereden başlaması gerektiğine ve bu işlem sırasında hangi kuralların kullanılması gerektiğine karar verir.

V. UZMAN SİSTEMLERİN YARARLARI VE KISITLARI

- Uzman sistemler, yetişmiş uzmanın az olduğu alanlarda bu boşluğu doldurabilirler. Bir uzman sistem geliştirildikten sonra bunun binlerce kopyasını çıkarmanın çok az bir maliyeti vardır. Bu açıdan bakıldığında uzman sistemler yetişmiş uzmanın az bulunduğu az gelişmiş ülkeler için daha büyük önem taşımaktadır.

- Uzman sistemler yöneticilerin sağlıklı karar vermesini kolaylaştırmaktadır. Uzman sistemleri orta düzeydeki yönetici robotlar olarak görmek mümkündür.

- Uzman sistemler, yeni bilgilerin eklenebildiği, dolayısıyla sürekli gelişim gösterebilen sistemlerdir [8].

Uzman sistem geliştirmede göz önünde bulundurulması gereken noktalar şunlardır:

- Uzman sistem dar ve iyi tanımlanmış bir konuda geliştirilmelidir. Bir çok alanda birden uzmanlık verebilecek bir sistemi geliştirmek fevkalade zordur [1].

- Bir alan için bir uzman sistem geliştirmek yerine bir alanda belirli bir iş konusunda uzmanlık verebilecek bir sistem amaçlanmalıdır.

- Uzmanlık konusunun çok kolay (bir kaç dakikada çözülebilir) veya çok zor (bir kaç saatten fazla düşünme gerektiren) olmaması gerekir [1].

- Duyularla elde edilen bilgilere dayanan uzman sistem geliştirmek mümkün değildir. Örneğin tad veya koku hakkındaki bilgileri sözcüklere dökmek hem zor, hem gereklidir. Dolayısıyla çeşnicilik yapabilecek bir uzman sistem olamaz.

- Uzman sistem sonuç olarak uzman kişinin bilgisini içereceğinden, bu bilginin uzman sisteme aktarılması sırasında, uzman kişinin kendisini bu işe adanması gerekir ki bu da oldukça uzun bir süredir.

Son olarak, Orta Doğu Teknik Üniversitesi'nde bu konu ile ilgili olarak yapılan çalışmalar veri tabanı yönetim sistemleri için veri tabanı tasarımı yapan uzman sistemleri içermektedir [2,3]. Bu çalışmaların yenilik içeren kısımları veri tabanı tasarımı yapan bölümleri olup, bilgi tabanları, ticari sistemlerle kıyaslanamayacak kadar küçüktür.

KAYNAKLAR

1. L. Alty, M.J. Coombs, *Expert Systems: Concepts and Examples*, NCC Publications, 1984.
2. F. Civelek, A. Dogac, S. Spaccapietra, "Expert System Approach to View Definition and Integration", *Proceedings of the 7th International Conference on the Entity-Relationship Approach*, North Holland, November 1988.
3. A. Dogac, B. Yürüten, S. Spaccapietra, "A Generalized Expert System for Database Design: GESDD", *IEEE Trans. on Software Engineering*, Vol. 15, No. 4, pp. 479-491, April 1989.
4. R. Fikes, T. Kekler, "The Role of Frame-Based Representation in Reasoning", *CACM*, Vol. 28, No. 9, September 1985.
5. M.R. Geneserath, M.L. Ginsberg, "Logic Programming", *CACM*, Vol. 28, No. 9, September 1985.
6. P. Harman, D. King, *Expert Systems*, John Wiley & Sons, 1985.
7. F. Hayes-Roth, "Rule-Based Systems", *CACM*, Vol. 28, No. 9, September 1985.
8. F. Hayes-Roth, D.A. Waterman, D.B. Lenat, *Building Expert Systems*, Addison Wesley, 1983.
9. J. Malpas, *PROLOG: A Relational Language and Its Applications*, Prentice-Hall, 1987.