T.C
TRAKYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

DENETLEMELİ KELİME ANLAMI BELİRGİNLEŞTİRMEDE KULLANILAN ÖZELLİKLERİN AYIRDEDİCİLİĞİNİN BİÇİMSEL KAVRAM ANALİZİ YARDIMI İLE DEĞERLENDİRİLMESİ

Mehmet Ali Aksoy TÜYSÜZ

Doktora Tezi

Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Yılmaz KILIÇASLAN

EDİRNE-2010

T.C.
TRAKYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

Denetlemeli Kelime Anlamı Belirginleştirmede Kullanılan Özelliklerin Ayırtediciliğinin Biçimsel Kavram Analizi Yardımı İle Değerlendirilmesi

Mehmet Ali Aksoy TÜYSÜZ

Doktora Tezi

Bilgisayar Mühendisliği Anabilim Dalı

Bu tez 27 / 09 / 2010 tarihinde aşağıdaki jüri tarafından kabul edilmiştir.

Jüri

Doç.Dr. Yılmaz KILIÇASLAN

Danışman
Jüri Başkanı

Yrd.Doç.Dr. Rafet AKDENİZ
Üye

Doç.Dr. Yılmaz ÇAN
Üye

Yrd.Doç.Dr. Erdem UÇAR
Üye

Yrd.Doç.Dr. Özlem UÇAR
Üye

T.C
TRAKYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENTSTİTÜSÜ

DENETLEMELİ KELİME ANLAMI BELİRGİNLEŞTİRMEDE KULLANILAN ÖZELLİKLERİN AYIRDEDİCİLİĞİNİN BİÇİMSEL KAVRAM ANALİZİ YARDIMI İLE DEĞERLENDİRİLMESİ

Mehmet Ali Aksoy TÜYSÜZ

Doktora Tezi
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Doç. Dr. Yılmaz KILIÇASLAN

EDİRNE-2010

Doktora Tezi
Trakya Üniversitesi Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Bölümü

ÖZET

Kelime Anlamı Belirginleştirme (KAB) alanında Denetimli Makine Öğrenmesi (DMÖ) teknikleri yoğun olarak kullanılmaktadır. Makine öğrenmesi alanındaki en önemli problemlerden biri kullanılacak özelliklerin seçimidir. Çünkü öğrenme algoritmalarının başarımı ve çalışma zamanı buna bağlıdır. Bu çalışmada, özellik seçimi için Biçimsel Kavram Analizi (BKA) tabanlı bir filtrenin kullanılabileceği geliştirilen bir uygulama aracılığı ile gösterilmiştir.

Birinci bölümde, tezin amacı ana hatlarıyla verilmiştir.

İkinci bölümde, Kelime Anlamı Belirginleştirme alanına ilişkin kapsamlı bir bilgi verilmiştir.

Üçüncü bölümde, Denetimli Makine Öğrnemesi konusu KAB alanından basit bir örnek yardımı ile açıklanmıştır.

Dördüncü bölümde, Biçimsel Kavram Analizine ilişkin temel kavramlar ve yazılımlar matematiksel ayrıntılara girilmeden verilmiştir.

Beşinci bölümde, KAB alanında sınırlı bir veri seti aracılığı ile denetimli makine öğrenmesi uygulandığında ortaya çıkabilecek durumlar ve bu durumların BKA tabanlı bir filtre yardımı ile görselleştirilmesi sonucu elde edilecek bilgiler uygulama yardımı ile verilmiştir.

Sonuç olarak, bir uygulama aracılığı ile BKA tabanlı bir filtrenin denetimli makine öğrenmesinden teknikleri kullanan KAB yöntemlerinin başarımına nasıl katkı sağlayacağı açıklanmıştır.

Anahtar Sözcükler: Kelime Anlamı Belirginleştirme, Denetimli Makine Öğrenmesi, Biçimsel Kavram Analizi, latis

Yıl: 2010
Sayfa: 116

Doctorate Thesis
Trakya University Graduate School of
Natural and Applied Sciences
Department of Computer Engineering

ABSTRACT

Supervised Machine Learning techniques are frequently used in the field of Word Sense Disambiguation (WSD). One of the most important problems of machine learning is the selection of features which will be used for learning process. Performance and time requirements of learning algorithms are affected by this selection. In this study, it is shown that a Formal Concept Analysis (FCA) based filter can be used for the selection of features.

Chapter 1 gives an overview of the aim of the thesis.

Chapter 2 provides some background information about Word Sense Dismabiguation.

Chapter 3 explains Supervised Machine Learning with the help of a simple WSD application.

Chapter 4 introduces FCA related basic concepts and software without formal/mathematical definitions.

Chapter 5 presents the details of application with a limited data set to show that an FCA-based filter usage.

The evaluation of filter and work to be done in the future are discussed in Chapter 6.

To sum up, as a result of this study, an FCA-based filter has been developed to be used with WSD techniques which benefit from supervised machine learning methods and has been explained in details.

Keywords: Word Sense Disambiguation, Supervised Machine Learning, Formal Concept Analysis, Lattice

Year: 2010

Page: 116

TEŞEKKÜR

Hiçbir zaman desteğini eksik etmemesi, bilgisini paylaşmakta ve yol göstermede takınmış olduğu üslubu, gerçek bir bilim insanının sahip olması gereken bütün özellikleri üzerinde taşıması ile bir öğrencinin sahip olabileceği en iyi tez hocalarından biri belki de en iyisi olan Doç. Dr. Yılmaz KILIÇASLAN'a herşey için en içten biçimde teşekkürü bir borç bilirim. Kendisine tüm yaptıkları için minnettarım.

Araş. Gör. Fatma BÜYÜKSARAÇOĞLU SAKALLI olmasa idi bu çalışmanın bürokratik işlemler yüzünden sona ulaşması mümkün olmazdı. Hem sıcacık yardımları hem de ablalığı için kendisine teşekkürler. Aynı biçimde Yrd. Doç. Dr. Tolga SAKALLI'ya da ağabeyliği için teşekkürler. Ayrıca Trakya Üniversitesi Bilgisayar Mühendisliği Bölümününün tüm asistan ve öğretim girevlilerine de teşekkürü borç bilirim. Beni Edirne'de hiç yalnız bırakmadılar ve hep kendilerinden biri olarak kabul ettiler.

Son olarak, yoğun çalışma dönemlerimde benden desteğini hiç eksik etmeyen aileme, sonsuz anlayışı ile beni hep şaşırtan ve destekleyen nişanlım Ebru DEMİRBAŞ'a, beni desteklemek için kendince yöntemler geliştiren biricik anneanneme, en kötü zamanlarımdaki sohbetleri için kardeşim Yrd. Doç. Dr. Fatih TÜYSÜZ'e, beni sıkıntılı gördüğünde Kanada'dan bana moral vermeye çalışan en küçük kardeşim Oğuzhan TÜYSÜZ'e, aynı kübikte çalıştığımız için bütün somurtmalarıma katlanan dostum Erkan ERSAN'a, çalışmalarım konusunda desteğini hiç eksik etmeyen proje yöneticim Merdan METİN'e ve adını sayamadığım tüm sevdiklerime göstermiş oldukları sonsuz anlayış ve destek için teşekkür ederim.

Mehmet Ali Aksoy TÜYSÜZ

1 . GİRİŞ

1.1 Tezimizin Amacı

Tezimizin amacı, Kelime Anlamı Belirginleştirme (KAB) alanında sıklıkla faydalanılan denetimli makine öğrenmesi tekniklerinde kullanılan özelliklerin ayırdediciliklerinin biçimsel kavram analizi tabanlı bir filtre yardımı ile değerlendirilmesidir.

Kelime Anlamı Belirginleştirme problemi, bir kelimenin kullanıldığı bağlamdaki anlamını hesaplamalı olarak belirleme çalışması şeklinde tanımlanabilir ve bu hali ile KAB'ın kendisi de makine öğrenmesinin konusu olan bir sınıflandırma problemi olarak düşünülebilir. Belirtilen benzerlik sebebi ile makine öğrenmesi teknikleri KAB alanında sıklıkla kullanılmaktadır. Tezimizde sadece denetimli makine öğrenmesi teknikleri ile ilgilenilmektedir.

Makine öğrenmesi alanında kullanılan özelliklerin değerlendirilmesi ve seçimi önemli bir araştırma alanıdır. Çünkü, başarımı arttırmanın en pratik çözümü olarak mümkün olan bütün özelliklerin kullanılması düşünülse de bu durum gerçek hayatta beklenen sonuçları vermemektedir. Bununla birlikte özellik sayısının artması, karmaşıklığın artmasına, çalışma zamanı olarak maliyetin artmasına ve hatta bazı durumlarda kullanılan ilgisiz (irrelevant) özelliklerin faydalı olanları engellemesi ile performansta düşüşe sebep olmaktadır. Tüm bu sebeplerle, kullanılabilecek olan bütün özellikler arasından amaca en uygun olanların seçilmesi ve kullanılması hem maliyet hem de performans açısından önemlidir.

Biçimsel Kavram Analizi (BKA) latis tabanlı bir matematiksel metodoloji olarak tanımlanabilir. BKA yardımı ile kullanılan özelliklerin matematiksel bir şekilde değerlendirilmesi ve yorumlanması bildiğimiz kadarı ile özellikle KAB alanı için yeni bir uygulamadır. Daha önce on katlamalı çapraz doğrulama (ten-fold cross validation) vb. istatistiksel yöntemlerle KAB için kullanılan özelliklerin değerlendirilmesi ve gerekiyorsa özellik vektörünün boyutlarının değiştirilmesi türünde uygulamalar yapılmıştır. Ancak belirtildiği gibi BKA'nın bu alana uygulaması yenidir ve çıkarılan

yorumlar açısından da özelliklerin ayırdediciliğinin değerlendirilmesinin ötesine geçebilmektedir.

1.2 Tez Organizasyonu

Belirtilen amaçlar doğrultusunda öncelikli olarak Kelime Anlamı Belirginleştirme, denetimli makine öğrenmesi ve Biçimsel Kavram Analizi alanlarındaki çalışmaların incelenmesi gerekmiştir. Yapılan araştırmanın ardından da bir uygulama geliştirilmiş, uygulama aracılığı ile elde edilen sonuçlar değerlendirilmiştir. Tezin organizasyonu da belirtilen duruma uygun olarak ortaya çıkmıştır.

İlk bölümde, KAB alanı için bir literatür taraması verilmiştir. Bu bölümde KAB alanında kullanılan teknikler, özellikler, elde edilen başarımlar hakkında kapsamlı bir bilgi sunulmaktadır.

İkinci olarak makine öğrenmesi ve özellikle Denetimli Makine Öğrenmesi (DMÖ) konusunda genel bilgilerin verildiği bir bölüm mevcuttur. Bu bölümde alan hakkında genel bilgi verilmesinin yanında tezin ana konusu olan özellik seçimi problemi de tanımlanacak ve konu ile alakalı bilgi verilecekir.

Üçüncü bölümde, Biçimsel Kavram Analizi ile alakalı genel bilgiler verilecektir. Bu bölümde metodolojinin matematiksel ayrıntılarına girmek yerine uygulamaya yönelik yanları ağırlıklı olarak verilecektir.

Dördüncü olarak tezin temelini oluşturan ugulama, uygulama aracılığı ile yapılan çıkarımlar verilecektir.

Son bölümde de yapılan uygulama ile elde edilen sonuçlar toparlanarak sunulmakta ve gelecekte yapılabilecek çalışmalar konusunda bilgi verilerek tez son bulmaktadır.

2 . KELİME ANLAMI BELİRGİNLEŞTİRME (KAB)

2.1 Tanım

Kelime anlamı belirginleştirme (KAB) için farklı kaynaklarda yapılan tanımlamaların bazıları aşağıdaki gibidir :

"KAB, bir kelimenin belli bir bağlamda kullanılmasıyla hangi anlamının aktif hale getirildiğinin hesaplamalı (computationally) olarak belirlenmesi problemi olarak tanımlanır. KAB temel olarak bir sınıflandırma problemidir : kelime anlamları sınıflardır, bağlam kanıt sunar ve kanıtlara dayanarak bir kelimenin her kullanımı, kendisine ait bir veya daha fazla sınıfa atanır. Bu KAB'ın, sabit sayılı kelime anlamı envanteri ile açık bir belirginleştirme süreci olarak gören geleneksel ve ortak tanımlamasıdır/karakterizasyonudur. Kelimelerin, sözlükten, sözlüksel (lexical) veri tabanından veya bir ontolojiden (...) sonlu sayıda ve ayrık anlamlarının olduğu varsayılmaktadır." (Agirre ve Edmonds, 2006)

"Hesaplamalı dilbilimde, KAB bir kelimenin taşıdığı anlamı verilen bir bağlamda (örneğin bir cümle veya internet aramasındaki bir sorguda vb.) otomatik olarak belirleme problemidir." (Chen, 2007)

"Anlam ayrımı, ... ara bir iştir, fakat birçok doğal dil işleme (DDİ) işini tamamlayabilmek için bir seviyede ya da diğerinde gereklidir." (Ide ve Véronis, 1998)

"Çok anlamlı kelimelerin farklı anlamları[1] "sense" olarak bilinir ve belli bir bağlamda hangisinin kullanıldığının belirlenmesi sürecine de 'kelime anlamı belirginleştirme' denir." (Stevenson, 2003)

Verilen tanımlamalardan da anlaşıldığı üzere KAB bir kelimenin içinde bulunduğu bağlamda kullanıldığı anlamının hesaplamalı dilbilim yöntemleri ile belirlenmesi işlemidir ve tez boyunca kabul edilen tanımlama bu olacaktır.

[1] Kaynaktan çeviri yapılırken iki defa Türkçe "anlam" kelimesi kullanmak yerine İngilizce "sense" kelimesi kullanılmıştır. Bu istisnai durum dışında tezimiz boyunca "sense" yerine "anlam" kelimesi tercih edilmiştir.

Kelimelerin farklı anlamlarına örnek olarak İngilizce “bank” kelimesi düşünülebilir. Türkçe karşılık olarak verilebilecek olan “banka”, “kıyı” kelimeleri verilen örneğin İngilizcedeki farklı anlamlarını göstermektedir. Benzer şekilde Türkçe’deki “kara”, “burun”, “kahve” gibi kelimelerin de farklı anlamları bulunmaktadır. Belirtilen kelimelerin kullanıldıkları bağlamdaki/cümledeki anlamlarını bilgisayar yardımı ile belirlemek bir kelime anlamı belirginleştirme işidir.

Kelime anlamı belirginleştirmeye ihtiyaç duyulması için bir kelimenin birden fazla anlamının olması gerekmektedir. Bu tip kelimelere çok analamlı (polysemous) kelimeler denir. Dilde bulunan çok anlamlı kelimelerin sayısı ile ilgili olarak, Zipf tarafından 1945 yılında (İngilizce için) yapılan bir analiz sonucu ortaya şöyle bir olgu çıkmıştır : Sıkça kullanılan sözcükler daha az kullanılanlara göre daha yüksek miktarda çok anlamlılığa sahiptirler. Bu durum literatürde Zipf fenomeni olarak bilinmektedir. Dolayısıyla çok anlamlılık dillerin kaçınılmaz birer parçasıdır diyebiliriz. Ayrıca, Zipf fenomeninin İngiliz Ulusal Derlemindeki (British National Corpus) varlığı (Edmonds, 2005)’te onaylanmıştır.

2.2 KAB Nerelerde Kullanılabilir?

Farklı kaynaklarda daha değişik kullanım alanları verilmişse de (Ide ve Veronis, 1998)’de verilen aşağıdaki liste temel kullanımlar konusunda fikir vericidir. Dolayısıyla, bu bölümde belirtilen kaynaktaki liste ile yetinilecektir. KAB’ın örnek kullanım alanları olarak;

- Makine çevirisi : Farklı anlamlarına göre farklı çevirileri olan kelimelerin çevirisi konusunda KAB gereklidir. Örneğin Türkçe “kara” kelimesi İngilizce çeviride “land”, “black” ya da “dark” gibi karşılıklara sahip olabilir. Hangisinin en uygun olduğu ancak Kab ile belirlenebilir. Çeviri programları doğrudan ya da dolaylı olarak KAB modülleri barındırabilirler.
- Bilgi çekme (Information Retrieval - IR) ve hipermetin (hypertext) dolaşımı : Bazı sorgulamalar için KAB gereklidir. Örneğin İngilizcedeki “depression” kelimesi

hastalık, hava durumu ve ekonomik bir terim olarak kullanılabilmektedir. IR'da genel olarak kullanıcının anlam belirginleştirme için yeterli miktarda kelime/bağlam sağlayacağı varsayılır. Ancak KAB modülü ile desteklenerek de belirtilen işlem yapılabilir.

- İçerik ve tematik analiz : İçerik ve tematik analizinde yaygın bir yaklaşım (verilen bir kavramın, fikrin vb. belirteci olan) kelimelerin önceden tanımlanmış kategorilerinin dağılımlarını metin boyunca analiz etmektir
- Gramatikal analizler : KAB, kelime türü işaretleme (part-of-speech tagging) için yararlı olmaktadır
- Ses/konuşma (speech) işleme : Anlam belirginleştirme, ses sentezinde, makine çevirisinde de olduğu gibi, doğru seslendirme için gereklidir.
- Metin işleme : Yazım düzeltme için bir KAB modülünden faydalanılabilir.

verilebilir.

2.3 KAB'ın Karmaşıklık Derecesi

KAB problemi "AI-complete" olarak tanımlanmaktadır, yani ilk olarak yapay zekadaki bütün zor problemler çözüldükten sonra çözülebilecek bir problemdir. (Ide ve Veronis, 1998) Dolayısıyla, zor bir problemdir. Ayrıca, dilbilim alanında makine çevirisi ile birlikte eskiden beri uğraşılan bir konudur.

2.4 KAB İçin Kullanılan Yöntemler

Aşağıda verilecek olan sınıflandırmadaki maddeler için (Ide ve Veronis, 1998) ile birlikte (Agirre ve Edmonds, 2006) kaynağından faydalanılmıştır.

1 – Yapay zeka tabanlı metotlar

a – Sembolik metotlar

b – Bağlantısal (connectionist) metotlar

2 – Bilgi Tabanlı Metotlar (Knowledge-Based Methods)

3 – Derlem Tabanlı Metotlar

a – Denetimli (supervised) derlem tabanlı metotlar

b – Denetimsiz (unsupervised) derlem tabanlı metotlar

İlerleyen bölümlerde yukarıda maddeler halinde verilen yöntemler için kısa açıklamalar verilecektir. Ancak yapay zeka tabanlı yöntemler için (Ide ve Veronis, 1998)'de "1970 ve 80'lerin yapay zeka tabanlı çalışmaları teorik olarak ilgi çekici olsa da dil anlama için son derece sınırlı alanlar hariç hiç pratik değildir" biçiminde bir belirleme yapıldığından ayrıca açıklanmayacak ve yukarıda verilen ana başlıklar ile yetinilecektir.

2.4.1 Bilgi tabanlı yöntemler

Bilgi tabanlı yöntemler de kendi içlerinde (Mihalcea, 2006) da belirtildiği şekli ile aşağıda verilen alt gruplara ayrılabilir:

1. Sözlük tanımlamaları göz önüne alarak bağlamsal örtüşmeyi kullanan metotlar : Lesk Algoritması, Lesk varyasyonu olan Simulated Annealing, Simplified Lesk Algoritması gibi algoritmalar bu gruba aittir.

2. Anlambilimsel ağlar (semantic networks) üzerinde hesaplanan benzerlik tabanlı metotlar.
3. Seçimsel tercihleri (selectional preferences) verilen bir bağlamdaki kelimenin anlamlarını kısıtlamak için araç olarak kullanan metotlar.
4. En sık kullanılan anlam, konuşma (discourse) başına bir anlam ve her eşdizimlilik (collocation) için bir anlamı gibi durumları da içeren, insan dilinin özelliklerine güvenen sezgisel (heuristic-based) metotlar.

Bilgi tabanlı yöntemler makine tarafından okunabilen sözlükler, eşanlamlılar sözlüğü, hesaplamalı sözlükler (bunlar da kendi içlerinde sayımlamalı (enumerative) ve üretici (generative) sözlükler diye ikiye ayrılmaktadır) gibi kaynaklardan faydalanmaktadırlar. Dolayısıyla sadece KAB işlemine sokulmak istenen kelimenin bulunduğu bağlamı değil harici kaynakları da kullanmaktadırlar.

Sözlük yayıncıları, hazırladıkları sözlükler için kullandıkları teypleri kullanıma açtıklarında makineler tarafından okunabilen sözlükler elde edilmiştir. Ancak bu yapıda verilen bilgiler doğrudan faydalanılacak biçimde değildir. O sebeple sözlük içinde bulunan örtülü bilgilerin yine makineler kullanılarak elde edilmesiyle ortaya çıkan kaynaklara da **sözlüksel bilgi tabanı** denmektedir. Adı geçen her iki tür kaynak da bilgi tabanlı yöntemler tarafından harici kaynak olarak kullanılmaktadır.

2.4.2 Derlem tabanlı yöntemler

Bu gruptaki yöntemler işlemlerini yapmak için bir derleme ihtiyaç duymaktadırlar. Kendi aralarında denetimsiz ve denetimli olarak iki ana başlığa ayrılırlar. Denetimsiz metotlar, kelimelere anlam etiketleri atamadan üzerinde işaretleme yapılmamış derlemlerdeki bilgilerden faydalanarak anlam ayrımı yapmaya çalışırlar. Bunu yaparken işaretlenmemiş derlemler kullanıyor olsalar da bu derlemler farklı diller

için hazırlanmış paralel derlemler olabilir. Denetimli derlem tabanlı yöntemler, denetimli makine öğrenmesi tekniklerinden ve işaretleme yapılmış derlemlerden faydalanmaktadırlar. Sonraki iki bölümde sırasıyla bu iki yöntem açıklanacaktır.

2.4.2.1 Denetimsiz Derlem Tabanlı Yöntemler

Bilgi tabanlı yöntemler önceki bölümde anlatıldığı gibi harici kaynakları kullanarak işlem yapmaktadır. Fakat her zaman harici kaynakların istenen biçimde elde edilmesi mümkün olmamaktadır. Harici kaynak kullanmadan işlem yapan (Pedersen, 2006)'da belirtildiği gibi dağılımsal (distributional) yaklaşımlar ve çevrimsel denklik (translational equivalence) yaklaşımlar bulunmaktadır.

Dağılımsal yaklaşımlar, benzer bağlamlarda kullanılan kelimelerin benzer anlamları olacağı varsayımı üzerinden kelime anlamlarında ayrım yapmaktadır. Çevrimsel denklik yaklaşımları, paralel derlemleri kullanarak işlem yapmaktadırlar. Her iki yaklaşım da bilgi açısından zayıf olarak değerlendirilmektedir, çünkü işratelenmemiş bir derlem ve kelime hizalaması yapılmış paralel metin dışında kaynak kullanmamaktadırlar.

Dağılımsal yaklaşımların anahtar özelliği, kelimeleri önceden varolan bir anlam deposuna göre ayırmamalarıdır. Bunun yerine kelimeleri derlemde gözlemlenen bağlamlarına göre gruplamaktadır. Ayrıca dağılımsal yaklaşımlar kelimeye anlam atamamakta, onun yerine herbir grubun kelimenin belli bir anlamda kullanılışını gösterdiği benzer bağlamların gruplarını belirleyerek kelimenin anlamları arasında ayrım yapabilmemizi sağlar.

Dağılımsal yaklaşımlar kavram-tabanlı ve kelime-tabanlı olmak üzere iki ana gruba ayrılmaktadır. Latent Semantic Analysis (LSA), Hyperspace Analogue to Language (HAL) ve Clustering by Committee (CBC) kavramsal tabanlı, Context Group Discrimination ve McQuitty's Benzerlik Analizi ise kelime tabanlı dağılımsal algoritmalara örnektir.

Çevrimsel denkliğe dayalı metotlarda ise kaynak dildeki bir kelimenin farklı manalarının hedef dilde tamamen farklı kelimelere çevrileceğine güvenmektedir.

2.4.2.2 Denetimli Derlem Tabanlı Yöntemler

Denetimli derlem tabanlı yöntemler için (Marquez vd. 2006)'da aşağıdaki gibi bir açıklama bulunmaktadır.

"Deneysel ve istatistiki yaklaşımlar DDİ üzerindeki etkilerini büyük biçimde arttırdılar. Bunların arasında, makine öğrenmesi topluluğundan gelen algoritmalar ve teknikler, çok çeşitli DDİ alanlarına dikkate değer bir başarı ile uygulandı ve artan bir ilginin odağı haline geldi. ... İstatistiki teknikler ve makine öğrenmesi teknikleri tarafından ilk olarak uygulanan problem türü dildeki belirsizlik çözümü olmuştur, ... Bunlar makine öğrenmesi topluluğu tarafından geniş biçimde üzerinde çalışılan, sınıflandırma problemleri olarak görülebilecekleri için özellikle makine öğrenmesi için uygun alanlardır."

KAB işlemini de bir sınıflandırma olarak ele alacak olursak denetimli derlem yöntemleri olarak makine öğrenmesi alanından algoritmaların kullanılması ve belli ölçüde başarı kazanmları normal görülecektir.

Denetimli derlem tabanlı yöntemlerde öğrenme işleminin gerçekleşebilmesi için üzerinde işaretleme yapılmış örneklere, anlam depolarına ihtiyaç vardır. Yani harici kaynaklar kullanılmaktadır.

Denetimli KAB metotlarının ana grupları olarak, olasılıksal yöntemler, örneklerin benzerliğine dayanan yöntemler, ayırt edici kurallara dayanan yöntemler, kural kombinasyonuna dayanan yöntemler, doğrusal sınıflandırıcılar ve çekirdek tabanlı yaklaşımlar, Yarowsky'nin Bootstrapping Algoritması gibi söylev özelliklerinden faydalanan yöntemler verilebilir.

Denetimli KAB algoritmalarının en önemli problemi uygulanabilmeleri için gerekli formattaki verinin elde edilmesidir. Bu probleme kısaca bilgi kazanımı

darboğazı denilmektedir. Özellikle üzerinde gerekli işaretlemelerin yapıldığı derlemlerin elde edilmesi ya da üretilmesi kolay olmamaktadır ya da istenilen her alanda bu tip bir kaynak bulunamamaktadır. Bu sebeplerle öğrenme örneklerinin otomatik elde edilmesi, aktif öğrenme, farklı kelimelerden öğrenme örneklerinin elde edilmesi, paralel derlemlerden faydalanılması ve hem etiketleme yapılmış hem de yapılmamış örneklerden öğrenme gibi yöntemlerle problem aşılmaya çalışılmaktadır.

2.5 KAB İçin Kullanılan Kaynaklar

Bu bölümde bilgisayarlı dilbilim uygulamalarında kullanılmak üzere hazırlanan ya da bulunan kaynaklar verilecek ve açıklanacaktır. Değerlendirme İngilizce ve Türkçe kaynaklar olmak üzere iki ana başlık halinde verilecektir. Bu şekilde hareket edilmesinin bir sebebi de Türkçe için kaynak sayısının ne kadar kısıtlı olduğunun gösterilmesidir. Türkçe KAB için yapılan bir çalışma olan (Aydın vd., 2007)'de de vurgu yapılan kaynak azlığı bu şekilde göz önüne serilmiş olacaktır.

2.5.1 İngilizce KAB için kullanılan ana kaynaklar

İngilizce kaynakların incelenmesinde öncelikle bilgisayar ortamında kullanılabilecek şekilde hazırlanmış sözlüklere değinilecektir. Dilbilim uygulamaları açısından en temel gereksinimlerden olan sözlüklerin farklı yapıları ve sağladıkları faydalar her başlıkta ayrıntılı şekilde açıklanacaktır.

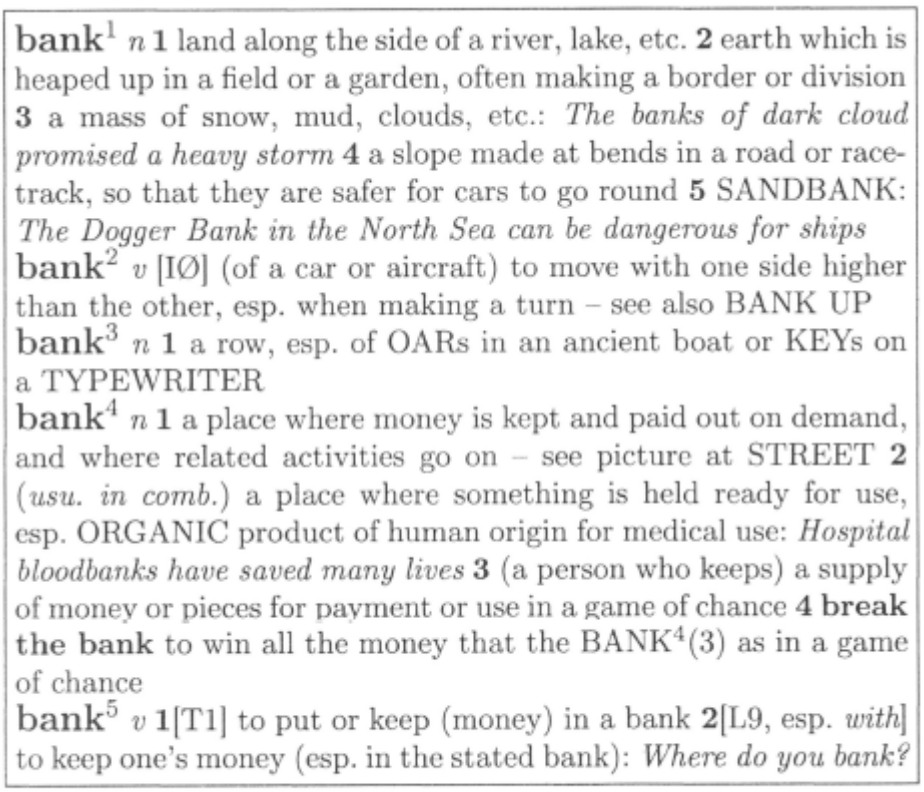
2.5.1.1 Longman'ın çağdaş İngilizce sözlüğü

İngilizcesi ile Longman Dictionary of Contemporary English olan ve LDOCE diye kısaltılan bir sözlüktür. En temel özelliği yaklaşık 2000 kelimeden oluşan temel bir kelime kümesini kullanarak bütün diğer kelimeleri açıklamaya çalışmasıdır. Ancak bir açıklamada temel kümede olmyan bir kelime de kullanılabilmektedir ve kullanılan bu kelime tamamen büyük harflerle yazılmaktadır. Böylece kelime sözlük içinde bulunarak anlamı öğrenilebilir ve içinde geçtiği tanımlama daha anlaşılır hale gelebilir.

LDOCE üç seviyeli anlam ayrımı uygulamaktadır. Bunlardan ilki eşyazımlılar seviyesinde olan kaba bir ayrımdır. İkincisi anlam seviyesinde olan daha ince bir ayrımdır. Üçüncüsü ise altanlam diye adlandırılan ve seçimlik olan bir alandır. Her eşyazımlının yanında sözdizimsel türü bilgisi yer almaktadır. Ayrıca köşeli parantezler içinde geçişlilik vb. biçiminde dilbilgisi kodları (bütün girişler için olmasa da) verilebilmektedir.

LDOCE yaygın kullanımı 1980'lerde başlamıştır ve makineler tarafından okunabilen sözlük formatında verilen teypte fazladan iki bilgi daha bulunmaktadır. Bunlardan ilki pragmatik ya da konu (subject) kodudur. Dört harften oluşan bu bilginin ilk iki harfi birincil/ana kullanım alanını sonraki iki harflik bilgi ise ikincil/alt kullanım alanını vermektedir. Sözlükteki tüm girişler için bu bilgi mevcut değildir. Bazı mevcut girişlerde de alt kullanım alanı bilgisi yoktur. İkinci fazladan bilgi ise seçimlik tercihler hakkında bilgi veren on boyutlu karakter dizisi biçimindeki gibi kutu kodudur (box code). Kutu kodları özne, nesne ve dolaylı nesne biçiminde üç ana parça ile tanımlanabilirler. Bir fiil geçişlilik dercesine göre her üç alana da sahip olabilir. İsim, sıfat ve zarflar ise sadece özne koduna sahiptirler. Bu kodların değerleri 36 anlambilimsel türden gelmektedir. Örneğin insan için H (human), sıvı için L (liquid), cansızlar için W gibi. Sıfat ve fiiller için özne kodu argümanları için tercih ettikleri anlambilimsel türü gösterirken, isimler için kendi anlambilimsel türlerini gösterirler.

Anlatılan durum (Stevenson, 2003)'ten alıntılanan aşağıdaki şekil ile daha net şekilde görülebilir :



bank[1] *n* **1** land along the side of a river, lake, etc. **2** earth which is heaped up in a field or a garden, often making a border or division **3** a mass of snow, mud, clouds, etc.: *The banks of dark cloud promised a heavy storm* **4** a slope made at bends in a road or race-track, so that they are safer for cars to go round **5** SANDBANK: *The Dogger Bank in the North Sea can be dangerous for ships*
bank[2] *v* [IØ] (of a car or aircraft) to move with one side higher than the other, esp. when making a turn – see also BANK UP
bank[3] *n* **1** a row, esp. of OARs in an ancient boat or KEYs on a TYPEWRITER
bank[4] *n* **1** a place where money is kept and paid out on demand, and where related activities go on – see picture at STREET **2** (*usu. in comb.*) a place where something is held ready for use, esp. ORGANIC product of human origin for medical use: *Hospital bloodbanks have saved many lives* **3** (a person who keeps) a supply of money or pieces for payment or use in a game of chance **4 break the bank** to win all the money that the BANK[4](3) as in a game of chance
bank[5] *v* **1**[T1] to put or keep (money) in a bank **2**[L9, esp. *with*] to keep one's money (esp. in the stated bank): *Where do you bank?*

FIGURE 2 The entry for "bank" in LDOCE

Şekil 2.1. Longman'in çağdaş İngilizce sözlüğündeki "bank" girişi

2.5.1.2 Roget'in eşanlamlılar sözlüğü

En bilinen eşanlamlılar sözlüğüdür. KAB çalışmalarında da kullanılan 1977 versiyonu 15 üst sınıf ve bunların alt alanlarını gösteren kategorilerden oluşmaktadır. Her kategoride sözdizimsel türe göre sıralanmış paragraflar bulunmaktadır. Sıralama isim, fiil, sıfat, zarf, edat (preposition), bağlaç, ünlem (interjection) biçimindedir. Bazı türler için hiç paragraf yokken bazıları için birden fazla paragraf bulunabilmektedir. Paragraflar sözlüğün çekirdeğini oluşturmaktadır ve birbiriyle yakından ilgili kelimeleri ve öbekleri içermektedir. Paragraflar noktalı virgüllerle ayrılmış olan daha küçük tam eşanlamlılar kümelerine ayrılmıştır. Yabancı dillerdeki deyimlerle (phrases) yaygın özel

isimler de bir çok paragrafta bulunmaktadır. Roget'in eşanlamlılar sözlüğünün çevrimiçi bir versiyonunu sunan http://thesaurus.reference.com/roget/ adresinden kelime sorgulaması yapılarak sözlük girişleri hakkında bilgiler edinilebilmektedir.

Anlatılan durum (Stevenson, 2003)'ten alıntılanan aşağıdaki şekil ile daha net şekilde görülebilir :

833 PREVIOUSNESS

NOUNS **1 previousness, earliness** 844, **antecedence** *or* antecedency, priority, **anteriority, precedence** *or* precedency 8134, precession; *status quo ante* ⟨L⟩, previous *or* prior state, earlier state; preexistence; **anticipation**, predating, antedating; antedate; **past time** 836

2 antecedent, precedent, premise; forerunner, **precursor** 815, ancestor

VERBS **3 be prior**, be before *or* early *or* earlier, come on the scene *or* appear earlier, **precede, antecede, forerun**, come *or* go before, set a precident; **herald**, usher in, proclaim, announce; **anticipate**, antedate, predate; **preexist**

ADJS **4 previous, prior, early** 844.7, **earlier**, *ci-devant* or *ci-dessus* ⟨Fr⟩, **former**, fore, prime, first, **preceding** 165.3, foregoing, above, anterior, **anticipatory**, antecedent; **preexistent**; older, elder, senior

FIGURE 3 The start of the entry for Roget Category 833; Previousness

Şekil 2.2. Roget'in eşanlamlılar sözlüğündeki "previousness" girişi

2.5.1.3 WordNet

WordNet projesi psikolojik prensiplere dyanarak büyük boyutlu bir sözlüksel (lexical) veritabanı oluşturmak içn tasarlanmıştır. İnsanın zihinsel sözlüğü üzerine araştırmalar yapan bilişsel psikolog Miller tarafından başlatılmıştır. Deneyimlerini zihinsel sözlüğün yapısını mümkün olduğunca yakın bir biçimde yansıtan bir kaynak oluşturmak için kullanmıştır. Ancak bazı kaynaklarda anlam ayrımlarının normal bir

insan zihnindekinden daha ince olduğu şeklinde yorumlar da bulunmaktadır. Ayrıca Jorgensen 1990 yılında yaptığı bir araştırmadan sonra şu kanıya varmıştır : Sözlükler, zihinsel sözlüktekilerden çok daha hassas ayrımlara sahiptirler. Belirtilen durum WordNet için sıklıkla dile getirilmektedir.

WordNet'in temel yapıtaşları eşanlamlı setleri olarak adlandırılabilecek İngilizce SYNonymSET teriminin kısaltması olan SYNSET'lerdir. Bunlar yakın anlamlı kelimelerin gruplarıdır. Bu setlerin belli bir kısmı kısa açıklamalara sahiptir. WordNet klasik sözlüklere benzese de en ilgi çekici yanı kendisini bir eşanlamlılar sözlüğü haline getiren, SYNSET'lerin bir hiyerarşiye sokulmuş olması durumudur. WordNet SYNSET'leri diğer SYNSET'lere bazı anlambilimsel ilişkilerle bağlıdır. Bu ilişkiler sözcük türüne göre değişmektedir.

Anlatılan durum (Stevenson, 2003)'ten alıntılanan aşağıdaki şekil ile daha net şekilde görülebilir :

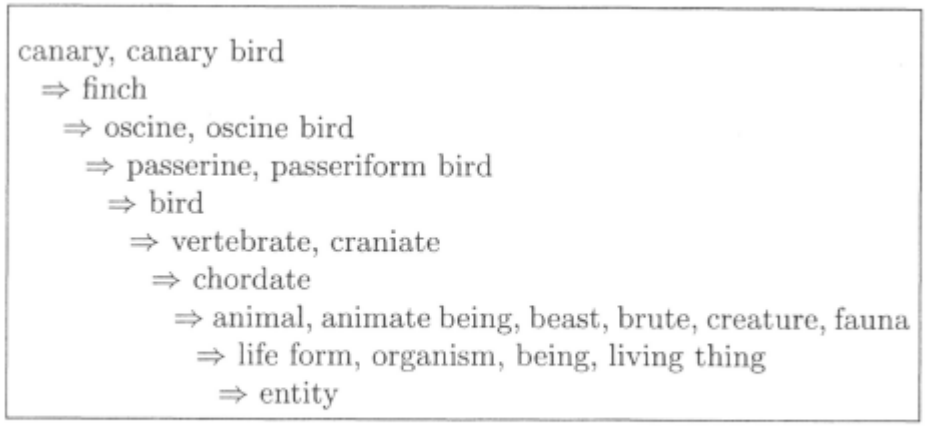


FIGURE 4 Hyponym chain for "canary" in WordNet

Şekil 2.3. WordNet'teki "canary" girişi

2.5.2 İngilizce İçin Kullanılan Diğer Kaynaklar ve Sınıflandırmaları

Bu bölümde İngilizce bilgisayarlı dilbilim alanında kullanılan derlemler ve önceki bölümde verilen sözlükler kadar popüler olmayan diğer sözlük ve sözlüksel bilgi tabanları konusunda bilgi verilecektir. Derlemler üzerinde işaretleme yapılmış ve yapılmamış olanlar olarak iki ayrı başlıkta incelenecektir.

2.5.2.1 Üzerinde İşaretleme Yapılmamış Derlemler

Brown Derlemi, 1961 yılında Amerika'da yayınlanmış olan metinlerin kolleksiyonu olan bir milyon kelimelik bir derlemdir. Toplam 15 kategoride yaklaşık 2000 kelimelik 500 dökümandan oluşur. Araştırma amaçlı olarak ücretsiz edinilebilir.

İngiliz Ulusal Derlemi (The British National Corpus – BNC), sözlük yayımcıları ve akademik araştırma merkezlerinin ortak çalışmasının ürünüdür. Bir ücret karşılığı kullanılabilmektedir.

Wall Street Yayınları Derlemi, doğrudan işaretsiz olarak erişilebilir değildir. Ancak Penn Ağaç Bankası aracılığı ile erişilebilmektedir. DSO Derlemi, Penn Ağaç Bankası ve PropBank için temel teşkil etmiştir. Bir ücret karşılığı elde edilebilir.

New York Times Derlemi, doğrudan kullanılabilir değildir. English Gigaword Derleminin bir parçasıdır. Bir ücret karşılığı elde edilebilir.

Reuters Haberleri Derlemi, ücretsiz olarak edinilebilmektedir ve elle işaretleme yapılan derlemler elde etmek için kullanılmaktadır.

2.5.2.2 Anlam İşaretlemesi Yapılmış Derlemler

DSO Derlemi, Singapur Savunma Bilimi Organizasyonu'ndan (Defence Science Organization) bir ekip tarafından oluşturulmuştur. Brown ve Wall Street derlemlerinden metinler içerir. WordNet 1.5 synset'leri kullanılarak elle işaretleme yapılmıştır. Bir ücret karşılığı elde edilebilmektedir.

Semcor, Princeton Üniversitesi'nde WordNet'i oluşturan aynı ekip tarafından hazırlanmıştır. Ücretsiz olarak kullanılabilecek en büyük anlam işaretlemesi yapılmış derlemdir.

Open Mind Word Expert Derlemi, WordNet 1.7 sysnset'leri kullanılarak internet kullanıcıları tarafından işaretleme yapılmış bir derlemdir ve ücretsiz olarak elde edilebilmektedir.

Senseval test ortamları, çeşitli diller için Senseval yarışmalarında oluşturulmuş olan test verisidir. Ücretsiz olarak elde edilebilir.

MultiSemCor, İngilizce ve İtalyanca için paralel bir derlemdir. Araştırma amaçlı olarak ücretsiz elde edilebilir.

Line-Hard-Serve Derlemi, İngilizce “line” (isim), “hard” (sıfat) ve “serve” (fiil) kelimeleri için oluşturulmuş yaklaşık 4000 anlam işaretlemesi yapılmış örnek içeren bir derlemdir. Ücretsiz olarak elde edilebilir.

Interest Derlemi, 2396 adet İngilizce “interest” kelimesini içeren Wall Street derleminden elde edilmiş anlam işaretlemesi yapılmış örnek içerir. Ücretsiz olarak elde edilebilir.

Ulusal İlaç Kütüphanesi KAB Test Koleksiyonu, tıbbi makalelerde görülen 50 adet belirsiz (ambiguous) kelime için elle işaretleme yapılmış bir derlemdir. Ücretsiz olarak elde edilebilir.

Alana özel Sussex Derlemi, BNC ve Reuters derlemlerinden elde edilmiştir.

Orwell'in 1984 Test Verisi, Bulgarca, Çekce, Yunanca, Romanca, Sırpça ve Türkçe çevirileri ve orjinal İngilizce kelime hizalı versiyonlarından oluşan bir derlemdir. Dan Tufis ile irtibata geçilerek kullanılabilir.

PropBank, Penn Ağaç Bankasının Wall Street parçasının bağlılık yapıları ile birlikte işaretlemesinin yapıldığı bir derlemdir. VerbNet kullanılarak anlam etiketleri oluşturulmuştur. Bir ücret karşılığı kullanılabilmektedir.

FrameNet örnekleri, FrameNet'in üzerinde işaretleme yaptığı örneklerdir. Araştırma amaçlı olarak ücretsiz elde edilebilir.

SenseCorpus, WordNet 1.6'dan alınan örneklerle oluşturulmuştur. Ücretsiz olarak elde edilebilir.

2.5.2.3 Sözlükler ve Sözlüksel Bilgi Tabanları

Hector, Senseval-1'de anlam deposu olarak kullanılan ve BNC için temel oluşturan bir sözlüktür.

Sözlüklerde içerilen dolaylı bilginin çıkarılması/elde edilmesi işleminin otomatize edilmesi elde edilen iyileştirilmiş veritabanlarına sözlüksel bilgi tabanı denir. Örnek olarak EuroWordNet, WordNet Domains, FrameNet, UMLS verilebilir.

2.5.3 Türkçe İçin Kaynaklar

Bu bölümde Türkçe için kullanılabilecek kaynaklar ve özellikleri verilecektir. Ancak İngilizce için verilen kaynaklar ile karşılaştırma yapılması gerekirse Türkçe kaynakların hem tür hem sayı hem de büyüklük olarak yetersiz oldukları görülecektir.

2.5.3.1 Güncel Sözlük

Türk Dil Kurumu (TDK) tarafından hazırlanan güncel bir Türkçe sözlük bulunmaktadır. Sözlüğe internet üzerinden erişim ve sorgulama imkanı mevcuttur. Yakın zamanda yoğun disk versiyonu da hazırlanmıştır. Ancak sözlüğün yapısı daha önce açıklanan ve İngilizce için kullanılan makineler tarafından okunabilen sözlük yapısında değildir. Ayrıca dilbilimsel olarak güçlendirilmiş değildir. Açıklaması yapılan sözcüklerin türü, eğer belli bir bilim dalında kullanılıyorsa bu bilim dalının adı ve sözcüğün anlamları verilmektedir. Ayrıca o kelimeye ilişkin atasözleri, deyimler ve birleşik yapıdaki kelimeler için de ayrı bölümleri bulunmaktadır. http://www.tdk.gov.tr adresinden kelime sorgulamaları yapılabilmektedir.

Anlaşıldığı gibi dilbilimsel çalışmalarda kullanılmak üzere tasarlanmadığı ve bir veritabanı olarak sunulmadığı için dilbilim çalışmaları açısından çok da uygun bir yapıda değildir. Dilbilimsel çalışmalarda kullanılabilmesi için bir takım işlemlerden geçirilmesi ya da sadece başvuru kaynağı olarak kullanılması gerekmektedir. Duruma (Aydın, Tüysüz, Kılıçaslan, 2007)'de de değinilmiştir.

2.5.3.2 Türkçe Derlemler

Orta Doğu Teknik Üniversitesi (ODTÜ) tarafından hazırlanan iki milyon kelimelik ODTÜ Türkçe derlemi 10 farklı türde kaynaktan toplanmış 2000'er kelimelik parçalardan oluşmaktadır. İçerik olarak 1990 yılı sonrası Türkçesi cümleleri içeren bir derlemdir. Derlem Kodlama Standardına (Corpus Encoding Standard – CES) göre uygun etiketler kullanılarak hazırlanmıştır. Ancak (Aydın, Tüysüz, Kılıçaslan, 2007)'de de belirtildiği gibi derlemin işaretlemesi sırasında bazı hatalı girişler yapılmış ve bu sebeple de ayrıştırılması zor bir hal almıştır. Ayrıntılar için adı geçen kaynağa başvurulabilir. Kaynağa http://www.ii.metu.edu.tr/~corpus/corpus.html adresinden

erişelebilir.

ODTÜ derleminin bir parçası kullanılarak Sabancı Üniversitesi ile birlikte hazırlanmış olan ODTÜ-Sabancı Ağaç Yapılı Derlemi de Türkçe için bir diğer kaynaktır. Burada işaretleme takılar (morphological) ve sözdizimsel olarak yapılmıştır. Toplamda 7262 cümle içermektedir. XML tabanlı bir yapısı bulunmaktadır. Kaynağa http://www.ii.metu.edu.tr/~corpus/treebank.html adresinden erişmek mümkündür.

Trakya Üniversitesi Bilişsel Bilimler Topluluğu tarafından hazırlanan ve üzerinde hiçbir işaretleme yapılmamış olan Trakya Derlemi de mevcuttur. http://tbbt.trakya.edu.tr/download/corpus.htm

2.5.3.3 Türkçe Takı Analizi Yazılımı

Türkçe için takı analizi yapan Zemberek isimli Java programlama dili tabanlı yazılım bulunmaktadır. Kütüphane olarak bazı açık kaynak kodlu ve özgür yazılım projelerinde de kullanılan bir yazılımdır. Daha fazla bilgi için projenin ana sayfasına ve dökümanlarına bakılabilir.

2.6. KAB Alanında Yapılan Değerlendirmeler İçin Temel Kavramlar

Bu bölümde (Palmer vd., 2006)'da verilen ve KAB sistemlerinin değerlendirirken kullanılan bazı kavramlar açıklanacaktır. KAB sistemlerinin katıldığı ve karşılaştırmalarının yapıldığı Senseval ve Semeval gibi yarışmalarında sistemlerin değerlendirilmesi için aşağıda verilen kavramlar kullanılmaktadır. Dolayısıyla adı geçen

değerlendirmelerin ve KAB literatürünün rahat anlaşılabilmesi için aşağıdaki kavramlar bilinmelidir.

2.6.1 Altın standart

KAB işleminde, aynı test verilerini iki farklı kişinin işaretlemesi ve sonunda ortaya çıkan anlaşmazlıkların çözülmesi ile elde edilen verinin durumu için kullanılan bir deyimdir.

2.6.2 Anlam deposu

KAB işlemindeki belki de en önemli seçimdir. Her kelimenin tanımlamasını anlamlara ayıran hesaplamalı sözlük (computational lexicon) veya makineler tarafından okunabilen sözlükler anlam deposu olarak kullanılmaktadır.

2.6.3. Görev tanımı

KAB sistemlerini değerlendirmede iki yaklaşım kullanılabilir. Bunlardan ilki in vitro diye bilinen ve KAB işlemini tek başınja yani herhangi bir başka uygulama ile

birlikte ele almayan yaklaşımdır. İkincisi ise in vivo diye bilinen ve KAB işleminin belli bir DDİ uygulamasının performansına katkısını ele alan yaklaşımdır. Genel olarak KAB değerlendirmeleri daha kolay olduğu için tek başına bir uygulama olarak kabul edilen ilk yaklaşımı kullanmaktadır.

Tek başına KAB işlemi de iki farklı alt göreve sahiptir. Bunlar sözlüksel örnek görevi (lexical sample task) ve bütün kelimeler (all-words) görevidir. Bütün kelimeler görevinde KAB sistemleri bütün (isim, sıfat, fiil gibi) içerik kelimelerini etiketlemek zorundadırlar. Kelime türü işaretlemeye (part-of-speech tagging) benzermiş gibi görünse de tamamen farklı bir anlam işaretleme etiketi seti gerektirdiğinden oldukça farklı bir iştir. Sözlüksel örnek görevinde ise örnek kelimeler her kelime için derlem örnekleri ile beraber sözlükten özenle seçilirler. Sistemler de seçilen bu kelimeleri kısa metinler için de doğru biçimde etiketlemeye çalışırlar.

2.6.4. Derlem

Sözlüksel örnek görevi için veri, hedef kelimeyi içeren ve anlam deposundaki anlamına göre bir işaretçiye sahip cümle örnekleridir. İşaretlenmiş verinin bir kısmı denetimli makine öğrenme sistemleri için eğitim verisi olarak kullanılır, kalan kısmı ise test amaçlı olarak kullanılır. Öğrenme tekniklerinin başarısı veri miktarına göre artış gösterebilmektedir.

2.6.5. Puanlama

Burada kullanılabilecek en basit kriter tam doğruluk (exact match) kriteridir. Ancak bir sistem belli bir kelime için birden fazla anlam ataması yapmak isterse bu durumda basit bir olasılık hesabı da yapılabilir.

Anlam deposunun organizasyonuna göre üç seviyeli bir atama hesabı da yapılabilir. Bunlar sırası ile iyi, kaba ve karışık seviyeli hesaplardır. İyi seviyesinde sadece benzer anlam etiketleri doğru sayılır. Kaba seviyeli hesaplamada ise hiyerarşik bir anlam yapısı olduğu varsayılır ve atanan anlam ile doğru anlam en üst seviyede aynı kökten geliyorlarsa doğru kabul edilirler. Karışık yapıda da yine anlam deposunda bir hiyerarşiye sahiptir. Burada da doğru anlamın çocuklarının ya da atasının seçilmesine göre olasılık hesabı yapılarak puanlama yapılmaktadır.

Ayrıca bütün kelimelere anlam ataması yapacak olan sistemler için bazı değerlendirme kriterleri daha kullanılabilmektedir. Bunlardan ilki kapsama (coverage) sistemin değerlendime kümesindeki kelimelerden ne kadarı için tahminde bulunduğu bilgisidir. Hassasiyet (precision), sistemin tahminde bulunduklarından ne kadarının doğru olduğu bilgisidir. Hatırlama (recall) ise doğru tahmin ettiklerinin toplamda (tahmin etmesi gerekenlere) oranıdır. Anlam işaretleme görevi için doğruluk (accuracy) hatırlama olarak değerlendirilmektedir.

2.6.6. Alt sınır

Değerlendirmeler için ortaya konulması gereken bir alt sınır bulunmaktadır. Bunun için en basit algoritma kelime için en sık kullanılan anlamı almaktır (Gale vd., 1992). Bunun dışında Lesk algoritması gibi basit algoritmalar da alt sınır olarak kullanılabilmektedir.

2.6.7. Üst sınır

Otomatik KAB sistemleri için kavramsal üst sınır, aynı veya karşılaştırılabilir veri üzerinde insan etiketleyicilerin seviyesinde doğruluğa sahip olmaktır (Gale vd., 1992). Çünkü sistemlerin tutarlılıklarının insanların tuttarlılığını geçmesi beklenmemektedir.

2.6.8. İşaretleme Yapanlar Arası Uyuşum (Inter-Annotator/Tagger Agreement – ITA)

El ile anlam işaretlemesi yapanların arasında da her zaman %100 bir uyuşum elde edilemeyebilir. Senseval-2'de “train” kelimesi için %28, “find” kelimesi için %44.3, “serve” kelimesi için %90.8, “dress” kelimesi için %86.5 uyum sağlanmıştır. Bu durumun dört ana nedeni olduğu düşünülmektedir: farklı anlamların bir anlam altında toplanması (sense subsumption), sözlüklerde olmayan veya yetersiz girişler, belirsiz kullanımlar/bağlamlar ve dünyaya ilişkin bilgilerdeki eksiklik ve farklılıklar. Hassas anlam belirlemeleri yapmak yerine gruplama yaparak daha kaba manada anlam ayrımı yapmak ITA'yı da arttırmaktadır. Ancak gruplama yolu ile kaba ayrımlara gitmek de ne kadar önemli olurlarsa olsun önemli ayrımların kaybedilmesine/gözden kaçmasına sebep olabilmektedir.

2.7. KAB Sistemlerinin Değerlendirilmesi ve Senseval

Farklı KAB sistemlerinin değerlendirilmesi konusunda (Stevenson, 2003)'de şu şekilde bilgi verilmektedir.

Bazen, araştırmacılar farklı kelime anlamı ayrımları, eğitim ve test verileri kullandıkları ve algoritmalarını farklı kelimeler üzerinde test ettikleri için KAB sistemlerinin karşılaştırmalı değerlendirmesini yapmak zordur. Örneğin Yarowsky'nin herbiri ikili anlam ayrımı içeren 12 kelime için değerlendirilen algoritmasını bir sözlükten anlam ayrımları kullanan ve bir metindeki bütün içerik kelimeleri için test edilen simulated annealing metodu ile karşılaştırmak zordur.

Bu problem SENSEVAL değerlendirme çatısı (framework) altında çözülmeye başlanmıştır. SENSEVAL Resnik ve Yarowsky'nin önerilerini kaynak alarak, ARPA tarafından deteklenen MUC ve TREC konferansları stilinde organize edilmektedir. Katılımcılara derlem verilerini içeren eğitim verileri ve doğru etiketler sağlanıp kısa bir süre içinde kendi sistemlerini hazırlamaları sağlanmaktadır. Değerlendirme ise, doğru etiketlerle etiketlenmemiş test verisinin dağıtılması ve katılımcılara kısa bir süre verilerek kendilerine verilen veri için işaretlemelerinin alınması biçiminde olmaktadır.

SENSEVAL-1'de elde edilen en iyi sonuçlar şu şekilde olmuştur: İyi seviyede ayrım için %77.1 ve daha kaba seviyede ayrım için %81.4. SENSEVAL-1 ana kaynak olarak HECTOR derlemini kullanıyordu. Bu kısıtlı bir kaynak olduğu için SENSEVAL-2 ve devamında WordNet sysnset'lerinin ve daha farklı derlemlerin kullanılmasına geçilmiştir. Ancak bu sefer de ortaya konan görevler zorlaşmış ve başarı yüzdeleri aşağıdaki şekilde verildiği gibi düşmüştür. Ancak belirtildiği gibi bu düşüş KAB işleminin zorlaşmasıyla da alakalıdır. Aşağıda SENSEVAL-2 ve SENSEVAL-3'te elde edilen başarım yüzdeleri verilmektedir.

Table 1.2. Performance of WSD systems in the Senseval-2 evaluation (Edmonds and Kilgarriff 2002).

Language	Task[a]	Systems	Lemmas	Instances	ITA[b]	Baseline[d]	Best score
English	AW	21	1,082	2,473	75%	57%/–[e]	69%/55%
Estonian	AW	2	4,608	11,504	72	85	67
Basque	LS	3	40	5,284	75	65	76
English	LS	26	73	12,939	86[c]	48/16	64/40
Italian	LS	2	83	3,900	21	–	39
Japanese	LS	7	100	10,000	86	72	78
Korean	LS	2	11	1,733	–	71	74
Spanish	LS	12	39	6,705	64	48	65
Swedish	LS	8	40	10,241	95	–	70
Japanese	TM	9	40	1,200	81	37	79

[a] *AW* all-words, *LS* lexical sample, *TM* translation memory.

[b] ITA is inter-tagger agreement, which is deemed as upper bound for the task.

[c] The ITA for English nouns and adjectives is reported. Verbs had an ITA of 71%.

[d] The baseline is most-frequent sense.

[e] Scores separated by a slash are supervised/unsupervised methods; supervised when there is no slash.

Şekil 2.4. Senseval-2'de elde edilen başarımlar

Table 1.3. Performance of WSD systems in the Senseval-3 evaluation (Mihalcea and Edmonds 2004).

Language	Task[a]	Systems	Lemmas	Instances	ITA[b]	Baseline[c]	Best score
English	AW	26	–	2,081	62%	62%/–[d]	65%/58%
Basque	LS	8	40	7,362	78	59	70
Catalan	LS	7	27	6,721	93	66	85
English	LS	47	57	–	67	55/–	73/66
Italian	LS	6	45	7,584	89	18	53
Romanian	LS	7	39	11,532	–	58	73
Spanish	LS	9	46	12,625	83–90	67	84
Hindi	TM	8	41	11,984	–	56	67
English	GL	10	–	42,491	–	–	68

[a] *AW* all words, *LS* lexical sample, *TM* translation memory, *GL* gloss task.

[b] ITA is inter-tagger agreement.

[c] The baseline is most-frequent sense.

[d] Scores separated by a slash are supervised/unsupervised methods; supervised when there is no slash.

Şekil 2.5. Senseval-3'de elde edilen başarımlar

2.8. KAB İçin Kullanılan Bilgi Kaynakları

Bu kısımda (Agirre ve Stevenson, 2006)'da listelenerek verilen ve KAB işlemlerinde kullanılan farklı bilgi türleri sıralanacaktır. Bu bölümde verilenler ana başlıkları teşkil etmektedir. Listelenen kaynakların uygulamada nasıl kullanıldığı sınıflandırmalarıyla birlikte bir sonraki başlıkta verilmektedir.

Sözdizimsel kaynaklar, kelime türü (part-of-speech), takı, eşdizimlilikler (collocations), alt öğeleme (subcategorization) şeklindedir.

Anlambilimsel kaynaklar, anlamların sıklığı (frequency of senses), anlambilimsel kelime ilişkileri, hypernymy ve meronymy gibi kelimelerin anlamları arası ilişkiler (paradigmatic), sözdizimsel bağımlılık ilişkileri (syntagmatic), seçimsel

tercihler (selectional preferences), anlambilimsel roller (tematik roller) olarak sıralanabilir.

Pragmatik/konusal kaynaklar, alan (domain) bilgisi, konusal kelime ilişkileri, pragmatik olarak verilebilir.

2.9. KAB İçin Kullanılan Özellikler

Yukarıda sıralanan bütün kaynaklar KAB işleminde kullanılmaktadırlar. Ancak kullanılabilmeleri (uygulanabilmeleri) için özellikler (features) olarak kodlanmaları gerekir. Bu özellikler de derlem, makineler tarafından okunabilen sözlükler ya da sözlüksel bilgi tabanları gibi kaynaklardan elde edilir.

Bağlamın büyüklüğüne göre özellikler üç gruba ayrılırlar. Bunlar sırası ile hedef kelimeye özgü özellikler, yerel özellikler ve global özelliklerdir.

Hedef kelimeye özgü özellikler, hedef kelimenin biçimi, hedef kelimenin türü, hedef kelimenin anlam dağılımı alt başlıklarında toplanmaktadır. Kelimenin biçimi dile bağlı olarak kelimenin türünü ve takılarını kodlar. Kelimenin türü bilgisi doğrudan kodlanır. Anlam dağılımı ise anlamların sıklığını kodlar. Prensip olarak bu bilgi, üzerinde işaretleme yapılmış bir derlem analiz edilerek elde edilir.

Yerel özellikler yerel kalıplar (local patterns), alt öğeleme, sözdizimsel bağımlılıklar ve seçimlik tercihlerdir. Yerel kalıplar KAB sistemleri tarafından en sık kullanılan öğelerdir. Bunlar eşdizimlilikler, alt öğeleme ve sözdizimsel bağımlılık ilişkileridir. Ayrıca n-gram kullanımı da bu gruba girmektedir.

Global özellikler ise kelimeler topluluğu (bag-of-words), bağlamdaki kelimeler ile ilişki, bağlamdaki kelimelere benzerlik, alan kodları şeklinde alt öğelere sahiptir. Kelimeler topluluğu, alan kodlarının bilgisinin yanında anlambilimsel ve konusal kelime ilişkilerini kodlar. Pencereleme yöntemi kelimelerin listesinin çıkarılması ve

incelenmesi ile uygulanır. Özellikler metnin incelenmesi ile çıkartılır. Başka bir dilbilimsel işleme gerek duyulmaz. Bağlamdaki kelimelerin ilişkisi, kelimeler topluluğu ile aynı bilgileri kodlar ancak bu bilgileri sözlük tanımlarından elde eder. Bağlamdaki kelimelere benzerlik, taksonomik bilgi içeren WordNet gibi kaynaklardan elde edilebilir. Alan kodları, alan bilgisini kodlar. Bu bilgi LDOCE gibi bazı kaynaklarda verilmektedir.

Verilen kaynakların geliştirilen uygulamada özellik olarak kullanımımı konusunda gerekli bilgiler ilgili bölümde verilmiştir. Ancak geliştirilen uygulamada, burada verilen listenin KAB alanında en sık kullanılan ve yararlılıkları defalarca ispatlanmış olan bir alt kümesi kullanılmıştır.

3. DENETİMLİ MAKİNE ÖĞRENMESİ

3.1 Giriş

Tez boyunca özellikle denetimli (supervised) makine öğrenmesi tekniklerine yoğunlaşılacaktır. Dolayısıyla bu noktadan sonra anlatımda sadece makine öğrenmesi tanımı kullanılsa bile kastedilen denetimli makine öğrenmesidir.

Kavram için farklı kaynaklarda verilen bazı tanımlamalar aşağıdaki gibidir :

"Denetimli makine öğrenmesi gelecekte karşılaşılacak örnekler hakkında tahmin yapmakta kullanılmak üzere genel hipotezler üretmek için harici olarak sağlanan örneklerden çıkarımda bulunan algoritmaların aranmasıdır" (Kotsiantis, 2007)

"Denetimli makine öğrenmesinde amaç, girdiden doğru değerleri bir uzman tarafından sağlanan çıktı arasında bir eşleşmeyi(mapping) öğrenmektir." (Alpaydın, 2004).

Özellikle son tanımlamada da belirtildiği gibi öğrenme verisinde hangi girdi için hangi çıktının elde edileceği belirtilmiştir.

Bu noktada makine öğrenmesi yapılırken geçilmesi gereken adımlar kabaca şu şekilde sıralanabilir :

1- Girdileri göstermek için kullanılacak özelliklerin ve sınıf etiketlerinin belirlenmesi.
2- Girdi olarak kullanılacak örneklemelerin seçilmesi
3- Kullanılacak algoritmanın seçilmesi
4- Öğrenme işleminin gerçekleştirilmesi
5- Sınıflayıcının elde edilmesi

Yukarıda verilen sıralama farklı kaynaklarda (örneklemelerin seçimi ile gösterim için kullanılacak özelliklerin seçimi gibi maddelerin yerinin değiştirilmesi şeklinde) farklı sıra ile ya da farklı başlıklarla verilebilmektedir. Bunun dışında (Kotsiantis,

2007)'de de olduğu gibi test kümesi ile değerlendirme ve parametre düzenleme gibi adımlar da ilave edilebilmektedir.

Makine öğrenmesinde genel olarak özellik kümesi bir defa kararlaştırılır ve veri setindeki her örnek aynı özellik kümesi kullanılarak gösterilir. Aslında öğrenme algoritmasının girdisi sadece (basit) örneklemeler değil o örneklemelere ait özelliklerin vektörleridir. Kullanılan özellikler için işaretlemeler ayrıştırıcı vb. kullanarak yazılım aracılığı ile yapılabileceği gibi insan gücü ile de yapılabilir. Ancak burada önemli olan nokta özellik vektörlerine karşılık gelen sonuç değerlerinin bir uzman tarafından sağlanmış olmasıdır. Bu şekilde hangi özellik vektörüne karşılık hangi değerin elde edileceği doğru şekilde belirlenmiş olur. Bu anlatımdan faydalanarak makine öğrenmesi türleri konusundaki denetimli-denetimsiz ayrımı için şu tanımlama verilebilir:

Eğer örnekler hangi sınıfa ait olduklarına dair etiketler ile veriliyorlarsa öğrenme "denetimli", eğer sınıf etiketi verilmiyorsa "denetimsiz" olarak isimlendirilir.

Durumu şekilsel olarak göstermek gerekirse (Kotsiantis, 2007)'de verilen tablodan faydalanılabilir:

Tablo 3.1. Doğru çıktıları bilinen ya da bilinen etiketlerle verilen örnekler

Standart formattaki veri					
Örnek	**Özellik 1**	**Özellik 2**	**…**	**Özellik N**	**Sınıf**
1	a	b		C	Iyi
2	a	b		C	Iyi
3	x	y		Z	Kötü
…					

Yukarıda verilen adımlara uyarak KAB ile alakalı olabildiğince basit bir örnek ile işleyişi açıklamaya çalışılırsa adımlar aşağıdaki gibi olacaktır :

1 – Kullanılacak özelliklerin seçilmesi : Basit bir KAB işlemi için sadece kelime türü özelliğinin kullanaldığı varsayılsın. Bu özelliğin değerleri olarak da "isim", "sıfat", "fiil" değerlerinin verilecek olsun.

2- Sınıf etiketlerinin belirlenmesi : KAB işlemi için "yüz" kelimesi kullanılsın. Farklı anlamları olarak da sadece insan yüzü ve suda yüzme seçilsin.

3 – Örneklemelerin seçilmesi : Kullanılacak örneklemeler olarak

i-) Çocuğun yüzü sararmıştı.

ii-) Çocuk havuzda yüzüyordu.

cümlelerini kullanılacak olsun.

Bu noktada makine öğrenmesi yapılabilmesi için verilen örneklemelerin seçilen özelliklere göre özellik vektörlerine çevrilmesi gerekmektedir. Verilen örnek için bir tek özellik (kelime türü) ve alabileceği üç farklı değere (isim, sıfat, fiil) karşılık alınabilecek iki farklı etiket/anlam (insan organı, suda yüzmek) bulunmaktadır. İster bir kelime türü işaretleyici (part-of-speech tagger) ve ayrıştırıcı (parser) yardımı ile ister elle hedef kelime olan "yüz" için işaretleme yapılsın örneklemeler için sonuç aşağıdaki gibi olacaktır :

Tablo 3.2. Örneklemeler için yapılan işaretlemeler

Örnekleme	**Özellik**	**Sınıf**
Cümle	**("yüz" için) Kelime Türü**	**("yüz" için) Anlam**
Çocuğun yüzü sararmıştı	isim	insan organı
Çocuk havuzda yüzüyordu	Fiil	suda yüzmek

Verilen örnek için kullanılan bir tek özellik olduğu için özellik vektörü de tek elemanlıdır. Bu noktadan sonra seçilecek bir öğrenme algoritmasına elde edilen özellik vektörleri verilerek makine öğrenmesi gerçekleştirilebilir ve ardından daha önce görülmemiş örneklerin özellik vektörlerine bakılarak sınıfları tahminlenmeye çalışılabilir.

Kelime Anlamı Belirginleştirme ile alakalı bölümde kullanılabilecek özelliklerin bir listesi ve kısa açıklamaları verilmişti. O sebeple burada tekrarlanmayacaktır. Klasik olarak denetimli makine öğrenmesi kullanarak yapılan KAB uygulamalarında da (yukarıda belirtildiği gibi) ilk olarak yapılması gereken, kullanılacak olan özelliklere karar verilmesidir. Ardından kullanılacak algoritma öğrenme verisi üzerinde çalıştırılarak özellik vektörlerine bağlı olarak sınıflayıcı elde edilir. Son olarak da çıkartılan özellik vektörlerinden faydalanılarak yeni örneklemelerin sınıfları belirlenir.

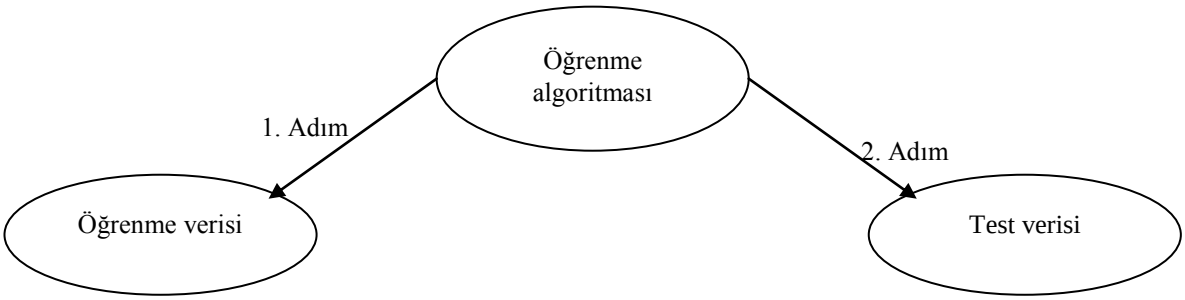
3.2 İlgilenilecek Olan Problem

Önceki başlıkta verilen denetimli makine öğrenmesinin eldeki veriler üzerinde çalışma adımlarını aşağıdaki gibi sıralı iki madde haline getirmek mümkündür :

1 – Üzerinde işaretleme yapılmış öğrenme verisi (training data) üzerinde öğrenme işlemi gerçekleştirilir ve özellik vektörlerine bağlı olarak sınıflayıcı elde edilir.

2 – Öğrenilen özellik vektörleri yardımı ile test verisi üzerinde daha önce rastlanmamış örneklemeler için sınıflandırma yapılır.

Şekilsel gösterim aşağıdaki gibidir :



Şekil 3.1. Bir makine öğrenmesi uygulamasının adımları

Test verisi üzerinde makine öğrenmesi yapılması konusunda iki ana problem bulunmaktadır : Kullanılan özelliklerin ayırdediciliği en başta bilinemez ve test verisinin yapısından kaynaklanan problemler olabilir. Problemlerden ilki literatürde özellik alt kümesi seçimi başlığı altında ele alınmaktadır. İkincisi ise veri seti dengeleme olarak ele alınmaktadır. Tez boyunca ilk probleme odaklanılacak ve özelliklerin ayırdediciliklerinin tespiti ile özellik seçimi konusunda bir filtre sunulacaktır. (Geliştirilen yöntem için filtre tanımlamasının kullanılmasının sebebi, "Özellik Seçimi ile Alakalı Metotların Sınıflandırılması" başlığı altında verilen açıklamalardan anlaşılabilir.) Sunulacak filtre, şekilde gösterilen ilk adım gerçekleştikten sonra ancak ikinci adımdan hemen önce devreye girecek ve kullanılan özellikleri değerlendirerek ayırdedici özelliklerin olup olmadığını kontrol edecektir.

Tez boyunca algoritma seçimi konusuna girilmeyecektir. Ancak belirtmek gerekir ki hangi öğrenme algoritması kullanılarak işlem yapılacağına karar vermek de kritik bir adımdır. Öğrenme algoritmasına bağlı olarak elde edilen sınıflayıcının değerlendirilmesi ve gerekiyorsa değiştirilmesine karar verilmesi de üzerinde çalışılan bir konudur. Bu konuda çoğunlukla kullanılan yöntem tahminleme doğruluğu değeridir. Tahminleme doğruluğu değeri, doğru tahminlenenlerin toplam tahminlere bölünmesi ile elde edilen değerdir. Bu konuda kullanılan üç ana teknik şunlardır :

1- Öğrenme verisini üçe bölüp iki tanesinde öğrenme gerçekleştirilip bir tanesinde de performans testi yapmak,
2- Çapraz doğrulama (cross-validation) : Öğrenme verisi birbirini ayrık (karşılıklı olarak birbirini dışlayan) ve eşit boyutlu alt kümelere ayrılır. Öğrenme algoritması bir küme üzerinde çalıştırılmadan önce diğer bütün alt kümeler kullanılarak eğitilir.
3- Birini dışarıda bırak doğrulama (Leave-one-out validation) : Çapraz doğrulamanın özel bir halidir. Her test alt kümesi yalnızca bir elemandan oluşur. Bu doğrulama yöntemi maliyeti yüksek ancak ürettiği değer açısından en doğru yöntemdir.

Sunulacak olan filtre Şekil 3.1'de verilen ilk adım sonrası, ikinci adım öncesi oluşturulan özellik vektörlerini inceleyerek ayırt edici özelliklerin olup olmadığını belirleyecek ve buna göre ikinci adıma geçmeye ya da geriye dönerek bazı kontroller yapılması gerektiğine karar verecektir. Her ne kadar algoritma seçimi konusu ile ilgilenilmese de kullanılan filtre öğrenme algoritmasının değiştirilmesine kadar giden çıkarımlarda bulunulmasına da sebep olabilir.

Sonraki bölümde tez boynca yoğunlaşılacak olan ve makine öğrenmesi konusundaki en önemli problemlerden özellik seçimi konusunda bilgi verilecektir.

3.3 Özellik Seçimi (Problemi)

Özellik seçimi konusunun makine öğrenmesi alanında son derece önemli bir problem olduğu belirtilmişti. Bu bölümde durum literatürdeki diğer araştırmalardan faydalanılarak ayrıntılandırılacaktır.

Makine öğrenmesi yapılırken kullanılmak üzere seçilen özellikler ve bunların aldıkları değerlerle oluşan özellik vektörleri elde edilen sınıflayıcıyı belirlemektedir. Durum (Koller ve Sahami, 1996)'de aşağıdaki gibi dile getirilmektedir.

"Bir veri örneği tipik olarak sisteme özellikler kümesine atanan değerler olarak açıklanır. … Bir sınıflayıcı bir veri örneğini girdi olarak alan mümkün olan ve sınıflardan birine ait olarak sınıflandıran bir prosedürdür. Sınıflayıcı kararını bir örnekleme ile ilişkilendirilen, atanmış değerler üzerinden verir. Optimistik bir yaklaşımla, uygun sınıflandırmayı tamamıyla özellik vektörü belirler."

Belirleyici olanın özellik vektörü ve dolayısıyla kullanılan özellikler olması sebebiyle bazı karakteristikleri olmalıdır. Durum (Kononenko, 1994)'te aşağıdaki gibi açıklanmaktadır.

"… iyi özellik farklı sınıflardan örnekleri birbirinden ayırabilmeli ve aynı sınıftaki örnekler için aynı değere sahip olmalıdır."

Belirtilen karakteristiklere sahip özelliklerin kullanımı sırasında çalışma zamanı ve performans da ayrıca kriterler olarak kullanılmalı, gerekiyorsa optimal bir nokta bulunmaya çalışılmalıdır. Konu ile alakalı olarak (Vafaie ve Imam, 1994)'da aşağıdaki açıklama mevcuttur.

"Özellik seçimi bir çok alanda ve özellikle de yapay zekada ele alınması gereken bir problemdir. Özellik seçme teknikleri geliştirmedeki ana konular, kabul edilebilir derecede yüksek tanıma oranına erişmeye ek olarak verilen sistemin maliyet ve çalışma zamanını düşürebilmek için küçük bir özellik kümesi seçmektir. Bu durum mümkün olan özelliklerin oluşturduğu daha büyük kümelerden optimal bir alt kümenin seçilmesi için çeşitli tekniklerin geliştirilmesine yol açmıştır. Bu özellik seçme teknikleri iki ana kategoriye ayrılmaktadır. İlk yaklaşımda özelliklerin sayısını idare edilebilir büyüklüğe indirgeyen belli bir alana ait probleme özel yaklaşımlar geliştirildi (Dom 89) İkinci yaklaşım, alana özel bilgi mevcut değilse ya da yararlanmanın maliyeti yüksekse

kullanıldı. Bu durumda, m tane kullanılabilir özellik arasından d tanelik alt küme seçmek için genel sezgisel yaklaşımlar, temelde açgözlü (greedy) algoritmalar uygulandılar,(Kittler 78)"

Dolayısıyla özellik seçme işlemi sadece en iyi olduğu düşünülen özelliklerin seçilmesi değil, aynı zamanda çalışma zamanı gibi kriterler ya da amaçlar doğrultusunda iyi kabul edilen özellik kümesinin belirlenmesi işlemidir denilebilir. Verilen tanıma uygun bir anlatım (John vd., 1994)'de aşağıdaki gibi verilmiştir.

"Özellik alt kümesi seçme problemi bazı amaç fonksiyonları altında iyi özelliklerin kümesini bulmayı içerir. Genel amaç fonksiyonları kestirim doğruluğu, yapı boyutu ve giriş vektörlerinin minimal kullanımıdır (özelliklerin kendileri ile ilişkili bir maliyetleri olduğu zaman)"

Kullanılan özellik sayısının çok fazla olması sebebi ile optimal seçim için sadece en iyi özellikleri belirleyip hepsini kullanmak yeterli olmamaktadır. En iyi kabul edilen özelliklerin de belli bir kısmı kullanılarak hem başarılı hem de performanslı çalışmalar yapmak mümkün olabilmektedir ve bazen de gerekmektedir. Özellik seçimim konusu ile alakalı araştırmaların en yoğun yapıldığı alanlardan biri de metin öğrenmedir ve durum (Mladenic, 1998)'de aşağıdaki gibi dile getirilmektedir.

"Metin öğrenme için yapılan özellik alt kümesi seçimi deneylerinin sonuçları %2 ile %5 arası en iyi özelliklerin kullanılmasını önermektedir."

En iyi sonucu verecek özelliklerin seçimi konusundaki en önemli problem uygun özellik kümesinin boyutunun bilinememesidir. Belirtilen durum (Kira ve Rendell, 1992)'de aşağıdaki ifade ile verilmektedir.

"Çünkü uygun hedef özellik alt kümesinin boyutu genellikle bilinmemektedir. … Yapay zeka alanındaki araştırmalar özellik seçiminin ayrı bir problem olarak görmek yerine tümevarımın örtülü bir parçası olarak ele almaktadır"

Uygun özellik kümesinin boyutu bilinmediği için (performans ve çalışma zamanı gibi kriterler göz önüne alınmadan) uygulanabilecek en basit yöntem olası bütün özellikleri kullanmak olarak dursa da bu yaklaşım beklenen sonucu vermemektedir. Konu ile alakalı olarak önceden yapılan çalışmaların anlatıldığı (Almuallim ve Dietterich, 1991) ve (Koller ve Sahami, 1996)'de aşağıdaki anlatımlar bulunmaktadır.

"Örneğin bir çok pratik uygulamada hangi özelliklerin ilgili olduğu ya da nasıl gösterileceği pek bilinmez. Kullanıcıların bu duruma doğal tepkisi, ilgili olabileceğini

düşündükleri tüm özellikleri kullanmak ve öğrenme algoritmasının hangi özelliklerin gerçekten değerli olduğunu belirlemesidir. Diğer bir durum da, bir çok farklı ikili fonksiyonları öğrenmek için aynı öğrenme verisinin kullanılması ve bu durumda bir çok ilgisiz özelliğin de bulunabilmesidir. Bu gibi durumlarda, verinin içinde bulunan özelliklerin bütün hedef fonksiyonları öğrenebilmek için yeterli olduğu garanti edilmelidir. Bununla birlikte, herbir fonksiyonu öğrenirken özelliklerin küçük bir alt setinin yeterli olması muhtemeldir."

"Klasik denetimli öğrenme işleminde bir sınıflandırma modeli ortaya koymak için etiketlenmiş sabit uzunluklu özellik vektörleri ya da örneklerinin kümesi verilir. Bu model daha sonra, önceden görülmemiş örneklerin kümesi için sınıf etiketi tahminlemek için kullanılır. Böylece, özelliklerin içinde varolan sınıf hakkındaki bilgi, modelin doğruluk derecesini belirler. Teorik olarak, daha çok özelliğe sahip olmak bize daha çok ayırdedici güç sağlamalıdır. Bununla birlikte, gerçek dünya bize bunun neden genellikle böyle olmadığına dair bir çok sebep sunar."

Alıntılardan da anlaşılacağı üzere daha çok özellik daha fazla ayırdetme gücü sağlamamaktadır. Hatta bazı durumlarda çok özellik kullanmak işleyişi yavaşlatmaktan da öteye geçerek öğrenme algoritmasını yanıltıp performansı daha da kötüye götürebildiği (Yu ve Liu, 2004)'te aşağıdaki gibi dile getirilmektedir.

"Klasik denetimli makine öğrenmesinde, etiketlenmiş sabit uzunluklu vektörler kümesi (örnekler) verilir. Bir örnekleme tipik olarak özellikler kümesine ve sınıf etiketine atanmış değerler olarak tarif edilir. Yapılması gereken iş yeni karşılaşılan örneklemelerin etiketlerini doğru şekilde tahminleyecek olan hipotezi (sınıflandırıcıyı) ortaya koymaktır. Sınıflayıcının öğrenilmesi özelliklerin aldıkları değerler tarafından belirlenir. Teoride, daha fazla özellik daha fazla ayırdetme gücü sağlamalıdır, fakat pratikte, sınırlı miktarda öğrenme verisi ile, fazla sayıdaki özellik sadece öğrenme sürecini yavaşlatmakla kalmayıp … ilgisiz ya da gereksiz veriler öğrenme algoritmasını yanıltabilmektedirler."

Benzer bir bilgi daha önceki bir çalışmada olan (Caruana ve Freitag, 1994)'te aşağıdaki gibi de dile getirilmiştir.

"Bir zorluk da geniş aday özellik kümesinden öğrenme için kullanılmak üzere en iyi özelliklerin seçilmesidir. İdeal olarak, bir öğrenme algoritmasının genelleme

performansı, ek özellikler tarafından sağlanan bilgi kendisine verildiğinde iyileşir. Malesef, sıklıkla tersi olur : ek özellikler diğer daha faydalı özellikleri engelleyebilir."

Anlatılanlardan da anlaşıldığı üzere özellik seçimi problemi iyi bir sınıflayıcı elde edebilme, çalışma zamanını uygun bir aralıkta tutma gibi bazı amaçlar göz önünde bulundurulursa, yapılması kaçınılmaz olan bir işlemdir. Metin sınıflandırma gibi bir alandaki en büyük zorluk olarak dile getirildiği (Yang ve Pedersen, 1997)'de belirtildiği gibi tek başına bir alandaki en büyük problem de olabilmektedir.

"Metin sınıflandırmanın ana karakteristiği ya da zorluğu özellik uzayının çok boyutluluğudur" (Yang ve Pedersen, 1997)

Tüm alıntılardan ve anlatımdan da anlaşıldığı üzere yeterli miktarda özellik kullanarak iyi bir sınıflayıcı elde etmek makine öğrenmesi ve diğer alanlarda üzerinde çalışılan bir konudur. Özellik alt kümesi seçimi olarak da adlandırılan bu işlem ile ;

- Çok boyutluluk probleminin hafifletilmesi,
- Genelleştirme kapasitesinin iyileştirilmesi,
- Öğrenme sürecinin hızlandırılması,
- Öğrenme algoritmasının (bazı gereksiz özelliklerin algoritmayı yanıltmasını engelleyerek) performansının artması,

şeklinde faydalar sağlanabilmektedir.

Öğrenme algoritmalarının hepsi ilgisiz ya da gereksiz özelliklere karşı dayanıklı ya da seçici değillerdir. Örneğin (Mitchell, 1997)'de belirtildiği gibi karar ağaçları bazı özelliklerin örnekleri ne kadar iyi ayırdettiğine bakarak düğümler seçmektedir. Ancak (Domingos, 1996) ve (Mihalcea, 2002)'de belirtildiği gibi örnek tabanlı öğrenme algoritmaları verideki gürültüye karşı dayanıklı olmakla birlikte ilgisiz özelliklerden olumsuz şekilde etkilenmektedirler. Dolayısıyla ilgisiz ya da gereksiz özelliklerin ayıklanarak bu tip algoritmaların kullanılması başarım açısından önemlidir. Bundan sonraki kısımda özellik seçimi ile alakalı olarak KAB ve zamir çözümlemesi alanından olmak üzere iki farklı uygulamadan örnek verilerek durum açıklanacaktır.

Geliştirilen uygulama ile doğrudan alakalı olan KAB uygulaması için (Mihalcea, 2002)'de yapılan uygulama örnek olarak verilebilir. Belirtilen makalede KAB alanındaki eforun büyük bir çoğunluğunun denetimli makine öğrenmesi algoritmalarına harcandığı ve bu algoritmaların da genellikle en iyi performansı verdikleri belirtilmektedir. Uygulama için örnekleme tabanlı öğrenme algoritması kullanılmakla

birlikte bu algoritmanın ilgisiz özellikler tarafından yanlış yönlendirilebildiği açıkca belirtilmektedir. Hatta örnekleme tabanlı öğrenme algoritmasının mensubu olduğu tembel öğreniciler (lazy learners) grubundaki tüm algoritmaların bu durumdan muzdarip oldukları açık şekilde belirtilmektedir. Ayrıca her kelime için ayrı bir özellik uzayının tanımlanmasının da daha iyi sonuçlar vereceği yani dinamik özellik vektörleri kullanımının da KAB için olumlu etkisi olduğu yine bu çalışma ile verilmektedir. Yine aynı kaynakta, özellik seçimi yerine bilgi kazanımı, kazanç oranı, ki-kare veya diğer bilgi içeriği ölçümleri kullanılarak özellik ağırlıklandırma yöntemi yararlı olsa bile bazen çok küçük bir ağırlığa sahip olan bir özelliği kullanmak yerine dışarıda bırakmanın/kullanmamanın daha iyi olduğu belirtilmektedir. Son olarak da özellik seçimi için kullanılan on-katlı çapraz doğrulama yöntemine dayanan algoritma verilmektedir. Bu noktadan sonra da uygulama ile elde edilen sonuçların son derece başarılı olduğu belirtilmektedir.

Zamir çözümleme konusunda (Kılıçaslan vd., 2009) çalışmasında iyileştirme süreci konusunda birkaç durumun dile getirildiği bölümde ilişkili özelliklerin üzerinde ayarlama/iyileştirme yapılmasının sınıflandırma performansında ciddi bir etkiye sahip olduğu belirtilmektedir. On-katlı çapraz doğrulama ve bilgi kazanımı hesaplama yöntemleri ile özelliklerin ne kadar değerli olduklarının değerlendirildiği, bunun sonucunda da kullanılan özelliklerden birinin bilgi verici olmadığı sonucuna ulaştıkları belirtilmektedir. Ayrıca en bilgi verici özellikler olarak hesaplanan özelliklerin de diğer kaynaklarda aynı şekilde belirtildiklerine vurgu yapılmıştır.

Kısacası, makine öğrenmesi uygulanan alanlarda kullanılabilecek olan tüm özelliklerden faydalanmak pratikte pek fayda sağlamamakta hatta performansı olumsuz yönde etkileyebilmektedir. Bazı öğrenici sınıfları ilgisiz kabul ettikleri özellikleri eleme yeteneğine sahip olsalar bile örnekleme tabanlı öğrenme algortimasında olduğu gibi bu yeteneğe sahip olmayanlar da mevcuttur. Daha da kötüsü tembel öğreniciler sınıfındaki algoritmaların hepsinde olduğu gibi öğrenme performansı ilgisiz özelliklerden olumsuz şekilde etkilenebilmektedir. Dolayısıyla, özellik seçimi makine öğrenmesi konusunda son derece önemli bir problemdir ve konu ile alakalı bazı metotlar mevcuttur. Bu metotların sınıflandırması sonraki bölümde verilmiştir. Bizim tezimiz ile bu metotlara yeni bir tanesi daha eklenmektedir.

3.4 Özellik Seçimi ile Alakalı Metotların Sınıflandırılması

Tez boyunca ilgilenilen alan olan KAB için makine öğrenmesi teknikleri kullanılırken de özellik seçimi son derece önemli bir hale gelmektedir. Ayırt edici özellikler olmadan test verisi üzerinde çalışma yapılırsa bir kelimeye ait farklı anlamlar ayırt edilemeyecektir. İşte tam da bu noktada önerilen filtre kullanılabilir. Önerilen metot literatürde filtre olarak tanımlanan sınıfa girmektedir.

Özellik seçme ile alakalı olarak geliştirilen algoritmalar aşağıdaki gibi üç ana gruba ayrılmaktadır.

1 - Sarma (wrapper) metotlar : Mümkün olan özellikler uzayını bir arama algoritması ile arayan ve her bir alt kümeyi üzerinde bir model çalıştırarak değerlendiren algoritmalardır. Bilgisayarlı hesaplama açısından maliyetlidirler.

2 - Filtreler : Bir filtre aracılığı ile kullanılması düşünülen özelliklerin değerlendirmesi yapılır.

3 - Gömülü (embedded) algoritmalar : Modele özel ve model içinde gömülü algoritmalardır.

Yukarıdaki sınıflandırma göz önüne alındığında sunulacak olan yöntem bir filtreleme algoritmasıdır.

Uygulamanın açıklandığı bölümde, bu kısımda ortaya konan probleme (özellikle) KAB alanı ile alakalı olarak ve BKA kullanımı ile görselleştirmeler yapılarak nasıl çözüm üretilebileceği ifade edilecektir.

4. BİÇİMSEL KAVRAM ANALİZİ[2]

4.1. Giriş

Biçimsel Kavram Analizi (BKA) için (Wormuth ve Becker, 2004)'te maddeler halinde sıralanan kısa açıklamalardan faydalanarak

- kavramın felsefi anlamının matematik hale getirilmesi,

- veriyi analiz etme ve yapısal hale getirmenin insan merkezli bir metodu,

- veriyi ve içinde barındırdığı yapıları, dolaylı bilgileri ve bağımlılıkları görselleştiren bir metot

şeklinde tanımlamalar yapılabilir. Rudolf Wille tarafından 1982 yılında (Wille, 1982) duyurulmuş ve psikoloji, sosyoloji, antropoloji, tıp, biyoloji, dilbilim, bilgisayar bilimleri, matematik ve endüstri mühendisliği gibi bir çok farklı alanda uygulanmıştır (Wolff., 1993)

Bu bölümdeki anlatımda (Priss, 1996)'daki yaklaşım kullanılarak, (Ganter ve Wille, 1998)'de matematiksel olarak anlatılan biçimsel kavram analizine ilişkin kavramlar matematiksel detaylara girilmeden verilecektir. Ayrıca içiçe çizgi diyagramları (nested line diagrams) gibi ileri seviye kabul edilebilecek konulara da girilmeyecektir. Referanslar bölümünden faydalanılarak ileri seviye konular ve matematik altyapı ile alakalı bilgi edinilebilir.

2 Ingilizcesi ile “Formal Concept Analysis” olan analiz için Türkçe olarak “Biçimsel Kavram Analizi” tanımlaması kullanılmıştır.

4.2 Temel Kavramlar

(Ganter ve Wille, 1997)'de aşağıdaki gibi bir paragraf bulunmaktadır.

“Biçimsel kavram analizinin sofistike isminin açıklanmaya ihtiyacı vardır. Metot esas olarak veriyi analiz etmek için kullanılmıştır yani açıkca verilen bilgiyi incelemek ve işlemek için. Bu veri yapısallaştırılarak, anlamlı ve anlaşılabilir, yorumlamaya izin veren, insan düşüncesine ait olan kavramın biçimsel soyutlaması olan birimler haline getirilir. Biçimsel önekini, bu biçimsel kavramların matematiksel varlıklar oluşunu ve zihnin kavramları şeklinde tanımlanmaması gerektiğini vurgulamak için kullanıyoruz. Aynı önek temel veri biçiminin, yani biçimsel bağlamın, genellikle bağlam olarak nitelendirilen yapının sadece küçük bir parçasının biçimsel bir hale getirildiğini göstermektedir.

Uygulamalar için gerekli matematiğin büyük çoğunluğu doğrudan latis teorisinden alınmıştır.”

Dolayısıyla “biçimsel” önekinin, kavram için kullanıldığında insan zihnindeki kavramlardan farklı olarak matematiksel yapıları gösterdiğini, bağlam için kullanıldığında da gerçek bağlama oranla küçük bir parçasını gösterdiğini söyleyebiliriz. Yine belirtildiği gibi matematiksel olarak latis teorisinden faydalanılarak geliştirilmiştir. Tez boyunca latis teorisinin matematiksel ayrıntılarına girilmeyecektir. (İstenildiği takdirde konu ile alakalı olarak referanslar kısmında verilen kaynaklardan faydalanılabilir.)

Benzer şekilde (Wolff, 1993) içinde de temel kavramlara ilişkin aşağıdaki kısım bulunmaktadır.

“Felsefi bir bakış açısıyla kavram, extension ve intension şeklinde iki parçadan oluşan bir düşünce birimidir. Extension bu kavrama ait bütün nesneleri kapsar ve intension bütün bu nesneler için geçerli olan bütün özelliklerden oluşur (Wagner 73). Bu sebeple nesneler ve özellikler bazı ilişkilerle birlikte çok önemli bir rol oynamaktadırlar ...”

Yukarıdaki açıklamalardan da anlaşıldığı gibi biçimsel kavram analizinin temel öğeleri, matematiksel bir şekle dönüştürülmüş olan felsefi manasıyla “kavram”, bu

kavramın iki ayrı öğesi olan “biçimsel nesne” ve “biçimsel özellik”tir. Ayrıca (Priss, 1996)’da geçen aşağıdaki kısımlar nelerin nesne ve özellik olarak kabul edilebileceği hakkında bilgi vermektedir.

“... Fakat 'nesne' ve 'özellik' kelimelerinin kullanımı bir belirticidir. Çünkü birçok uygulamada nesneye benzer unsurların biçimsel nesne olarak ve bu nesnelerin özelliklerini ya da karakteristiklerini biçimsel özellikler olarak seçmek yararlı olabilir. Bir bilgi çekme uygulamasında, dökümanlar nesneye benzer ve terimler de özellik benzeri olarak düşünülebilir. Biçimsel nesneler ve özelliklere diğer örnekler ... kelimeler ve anlamları, vb.'dir.

Biçimsel nesnelerin ve özelliklerin birbirlerine olan ilişkileri ile birlikte kümeleri bir tablo olarak gösterilebilecek olan 'biçimsel bağlam'ı meydana getirir. ”

Yine aynı kaynaktan alınan ünlü hayvanlara ait biçimsel bağlam örneği aşağıdaki gibidir.

Tablo 4.1. Ünlü hayvanlara ait biçimsel bağlam

	Cartoon	Real	Tortoise	Dog	Cat	Mammal
Garfield	X				X	X
Snoopy	X			X		X
Socks		X			X	X
Greyfiar's Bobby		X		X		X
Harriet		X	X			

Burada en sol kolondaki elemanlar biçimsel nesneler, en üst satırda yer alan elemanlar biçimsel özelliklerdir ve aralarındaki ilişkiler çarpı işaretleri ile gösterilmiştir.

Herhangi bir biçimsel nesneler kümesi ile başlanırsa ortak olan bütün biçimsel özellikler belirlenebilir. Örneğin Harriet ve Bobby ile başlanmışsa, ortak özellikleri (yani real) belirlenir ve sonra diğer “real” özelliğine sahip nesneler belirlenirse ortaya “Harriet, Bobby ve Socks” şeklinde bir nesne kümesi çıkar. Bu noktada ilişki “kapalıdır”. Çünkü ne nesne kümesi ne de özellik kümesi genişletilemez. Bu şekilde kapalılık özelliği gösteren biçimsel nesneler ve özellikler kümlerinin ikililerine

“biçimsel kavram” denir. Biçimsel bir kavramın biçimsel nesnelerine “extension”, biçimsel özelliklerine de “intension” denir. Verilen biçimsel bir bağlam için biçimsel kavramlar, “extension”ları ve “intension”ları net bir biçimde tanımlanır ve belirlenir. (Priss, 1996)

Yukarıda anlatılanları şu şekilde ifade etmek de mümkündür : Önce bir nesne ya da nesneler seçilir. Sonra bu nesnelere ait özelliklerin hepsi seçilir. Ardındanda tüm bu özellikleri gösteren nesneler seçilir. Bu şekilde son adımda elde edilen nesneler ile bu nesnelerin sahip oldukları bütün özellikler bir kavram oluştururlar.

Anlatılanları bir örnek ile açıklamak gerekirse “araba” kavramı için “tekerlekli olmak”, “motorlu olmak” gibi özellikler belirlenebilir. Bu özelliklere sahip olan farklı markalara ait otomobiller araba olarak tanımlanabilirler. Küme kavramı ile tanımlama yapmak gerekirse nesneler kümesi ve özellikler kümesi ile aralarındaki ilişki bir kavram meydana getirir diyebiliriz.

Eğer A nesneler kümesi B ise özellikler kümesi olarak tanımlanırsa;

- Bir kavrama ait bütün nesneler B ile gösterilen bütün özelliklere sahiptir.
- Bir kavrama ait bütün özellikler A ile gösterilen bütün nesneler tarafından paylaşılır/ortaktır.
- A kavramın extension’ı ve B de kavramın intension’ı olarak adlandırılır.

Bütün bu kümeler ve özellikleri bütün çıkarımlarımız için taban oluşturacaktır – tüm kavramlarımız ve çıkardığımız dolaylı bilgiler elimizdeki bağlama göredir/bağlam tabanlıdır. Bağlamı değiştirmek kavramları ve yapılarını, dolayısıyla da çıkarımlarımızı, değiştirecektir.

4.3 Biçimsel Bağlamın Matematiksel Gösterimi

G ve M şeklinde iki kümeden ve G ile M arasındaki bir ilişkiden oluşan bir biçimsel bağlam,

K := (G, M, I)

şeklinde gösterilmektedir. G'nin elemanlarına bağlamın nesneleri, M'in elemanlarına ise bağlamın özellikleri denmektedir. g ile gösterilen br nesnenin m ile gösterilen bir özellikle I ilişkisi içinde olduğunu göstermek için gIm ya da (g, m) Є I kullanılır ve "nesne olan g, m ile gösterilen özelliğe sahiptir" biçiminde okunur.

Biçimsel Kavram Analizi ile ilgili kaynaklarda matematiksel gösterimler de sıklıkla kullanılmaktadır. Ancak yukarıdaki açıklamanın, kullanılan matematiksel gösterimleri anlamak için yeterli olduğu düşünülerek daha ileri seviyede açıklama yapılmayacaktır.

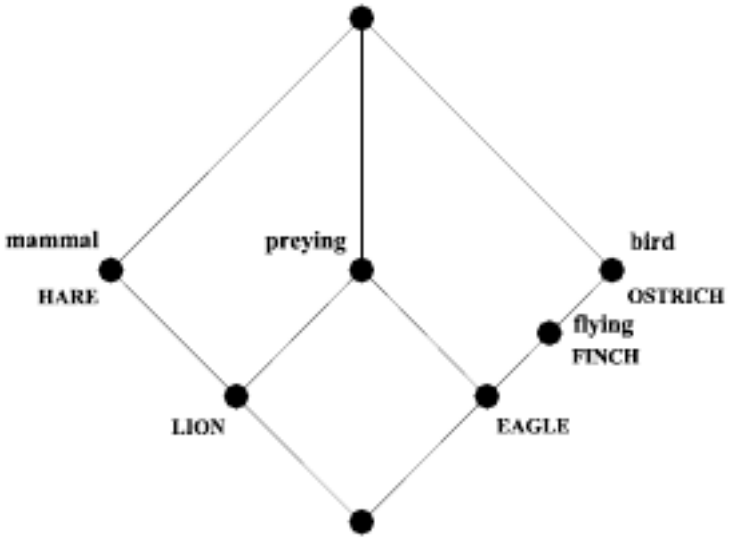
4.4 Örnek Bir Bağlam ve Bu Bağlama Ait Latis

Aşağıda (Wolff, 1993)'ten alınan bir biçimsel bağlam ve bu bağlama ilişkin kavram latisi görülmektedir.

ANIMALS	preying	flying	bird	mammal
LION	×			×
FINCH		×	×	
EAGLE	×	×	×	
HARE				×
OSTRICH			×	

Şekil 4.1. Hayvanların nesne olarak kullanıldığı bir biçimsel bağlam örneği

Yukarıdaki biçimsel bağlama ilişkin kavram latisi aşağıdaki gibidir.



Şekil 4.2. Yukarıda Şekil 4.1 ile verilan bağlamın kavram latisi

4.5 Diyagramlar/Latisler Nasıl Okunmalıdır?

Yukarıdaki kavram latisinde daireler kavramları göstermektedir. Dairelerin alt kısımlarındakiler “extent”leri yani nesneleri, üst kısmındakiler ise “intent”leri yani özellikleri göstermektedir. Ayrıca g ile gösterilen bir nesne m ile gösterilen bir özelliğe sadece ve sadece g ile gösterilen daireden m ile gösterilene, yukarı doğru bir yol varsa sahiptir.

4.6 Dolaylı Olarak Bulunan Bilgi (Implication)

Tablo 4.1 ile verilen örnekte biçimsel özellik olan “mammal”, “dog” ve “cat” biçimindeki düğümlerin üzerinde bulunmaktadır. Bu ise “dog” ve “cat” biçimsel özelliklerinin dolaylı olarak “mammal” özelliğini işaret etmesidir. Durum biyolojik bir gerçeği yansıtmaktadır ve latiste doğrudan değil, ancak inceleme ile erişilebilecek şekilde yani dolaylı olarak bulunmaktadır. Verilen örnekte gerçekte geçerli olmayan ve sadece bu örneğe özgü olarak varolduğunu söyleyebileceğimiz dolaylı bilgiler de bulunmaktadır. Örneğin “cartoon” özelliği “mammal” özelliğini, “tortoise” ise “real” özelliğini dolaylı olarak işaret etmektedir. Belirtildiği gibi bu bilgiler de dolaylı olarak elde edilmektedir ancak gerçek dünyada geçerli değildir. Sahip olunan bağlam sebebiyle ortaya çıkmıştır. Dolayısıyla, sahip olunan bağlamın çıkartılan bilgileri fazlası ile etkilediği söylenebilir. Ayrıca dolaylı olarak bulunan bilgiler özelliklerin kombinasyonu biçiminde de olabilir.

Dolaylı olarak bulunan bilgiler üzerine çalışılmıştır. Özellik keşfi olarak adlandırılan, adım adım bilgisayar tabanlı kavramsal bilgi oluşturma için kullanılmaktadırlar. Özellik keşfi ConImp ve ConExp yazılımları tarafından sunulmaktadır. Yazılım bir başlangıç bağlamından faydalanarak özellikler arasındaki ilişkiler hakında kullanıcıya bir takım sorular sormaktadır. Örnekteki bağlam için kullanıcıya bütün çizgi hayvanların memeli olup olmadığı sorulabilir. Bu durumda kullanıcı ya bu durumun her zaman doğru olduğunu kabul etmeli ya da çizgi hayvan olup memeli olmayan bir karşı örnek vermelidir. Eğer verilmişse bu karşı örnek bağlama eklenir. Süreç, mümkün olan bütün dolaylı bilgi kontrol edildikten sonra durur. Özellik keşfetme daha ileri seviyede de çalışılmış ve ortaya “kavram keşfetme” çıkmıştır. Ancak bilindiği kadarı ile bu sadece teorik bir ileri uygulama olarak mevcuttur ve bu işlemi gerçekleştiren bir yazılım bulunmamaktadır.

4.7 En alt kavram ve en üst kavramın özellikleri

Ortaya çıkan latis gösterimlerindeki en alt ve en üst kavramın özel bir önemi vardır. (Priss, 1993)'te konu ile alakalı olarak aşağıdaki bilgiler verilmiştir:

“Bir kavram latisindeki en üst ve en alt kavramlar özeldir. En üstteki kavram extension’ı olarak bütün biçimsel nesnelere sahiptir. Intension'ı sıklıkla boştur/yoktur ancak böyle olmak zorunda değildir. ... En alttaki kavram ise bütün biçimsel özelliklere intension’ı olarak sahiptir. Eğer biçimsel özelliklerden bazıları karşılıklı olarak birbirini dışlıyor ise (örneğin köpek ve kedi) en alttaki kavramın extension'ı boş olmak zorundadır (Çünkü hiçbir biçimsel nesne hem köpek hem de kedi olamaz)”

Bu iki özel durumdan faydalanılarak gerçek dünyada olamayacak bir durumun bağlamda kodlanıp kodlanmadığı rahatlıkla görülebilir.

4.8 Altkavram, üstkavram ve miras

Ortaya çıkan latislerde “altkavram” ve “üstkavram” ilişkileri yani bir hiyerarşi de oluşmaktadır. Oluşan hiyerarşiye bağlı olarak bir “miras” da söz konusudur. Latis örneğine dönülecek olursa İngilizcesi ile “preying flying birds” diye geçen uçan yırtıcı kuşlar yine İngilizcesi ile “flying birds” olarak geçen “uçan kuşlar” için bir alt kavramdır. Uçan kuşlar burada üst kavram olmaktadır. Alt kavram olan yırtıcı uçan kuşlar, uçan kuşlara ait bütün özellikleri miras yoluyla almaktadır.

4.9 Ölçeklendirme

Bol miktarda nesne ve özellikten oluşturulan bir latis, kompleks bir hal alabilmektedir. Bu tip latisler kullanışlı değillerdir. Çünkü görsel bir hale getirilseler bile anlaşılması çok güç olur. Bu tip kompleks latisler için bir kullanım (Lindig ve Sneltig, 1997)'de önerilmiştir. Durum, eğer bir kodun bağımlılıkları için kompleks bir latis çıkıyorsa o koda tekrar mühendislik yapılmaması[3] gerektiğini gösterir bir durum olarak değerlendirilmiştir.

Eğer elde büyük bir bağlam varsa ve buna bağlı olarak kompleks bir latis ortaya çıkacaksa ilgili özellikler gruplanarak ortaya çıkacak latis farklı bileşenlere/gruplara bölünebilir. Bu gruplar daha sonra latis olarak görselleştirilebilir. Bu duruma (Ganter ve Wille, 1989)'da kavramsal ölçekler denilmiştir. İsimsel (nominal), sırasal (ordinal) ve aralıksal (interval) ölçeklendirme şeklinde türleri mevcuttur. Bu konuda daha fazla bilgi ve örnekler için (Priss, 1996), (Wolff, 1993) ve (Wormuth ve Becker, 2004) kaynaklarına bakılabilir. Ancak anlatılanları aşağıdaki şekilde de ifade etmek mümkündür;

- Çok değerli (multi-valued) verilerden tek değerli bağlamlar oluşturma sürecine **kavramsal ölçeklendirme** denir.
- Kavramsal ölçeklendirme standart hale getirilebilir olsa da çoğunlukla insanın yorumlamasına bağlıdır.

İlk maddede bahsi geçen çok-değerli ve tek değerli bağlam kavramları tez boyunca sık sık kullanılacaktır. Dolayısıyla fazladan bir açıklama gerektirmektedirler.

Tek değerli bağlam ile kastedilen durum, kullanılan özelliklerin mümkün olan değerlerinin, o özelliğe sahip olma ve olmama biçiminde olmasıdır. Önceki başlıklarda verilen örneklere bakılacak olursa, nesne listesinde verilen hayvanlar için uçma özelliği ya vardır ya da yoktur. Benzer şekilde nesne listesinde verilen isimler ya bir çizgi film karakteridir ya da değildir. Dolayısıyla belirtilen özelliğe sahip nesneler için "X" değeri kullanılmış, özelliğe sahip olmayanlar içinse hiçbir belirleme yapılmamıştır. Dikkat

3 Burada İngilizce "re-engineering" kelimesinin tam Türkçe karşılığı olarak "tekrar mühendislik yapılması" karşılığı kullanılmıştır.

edilecek olursa, tek değerli şeklinde ifade edilmiş olsa bile bu tip özelliklerin olma ya da olmama durumları göz önüne alındığında aslında iki değere sahip oldukları düşünülebilir. Bu durumda da ikili (binary) değerlere sahip oldukları söylenebilir ve bağlamlar da ikili bağlamlar olarak adlandırılabilir. Tez boyunca tek değerli bağlam ve ikili bağlam terimleri, aynı durumu ifade ettikleri düşünülerek, birbiri ile yer değişebilir isimler olarak kullanılmıştır.

Çok-değerli bağlam kavramı ise bağlamdaki özelliklerin olma ya da olmama dışında değerlere sahip oldukları durumları göstermektedir. Örneğin, Türkçe için ismin halleri özelliğinin, yalın hali, -e hali, -i hali, -de hali ve –den hali biçiminde beş farklı değer alması söz konusudur. Benzer şekilde kelime türü bilgisi için isim, sıfat ve fiil gibi değerler kullanıldığı düşünülürse bu da çok değer alabilen bir özellik olmaktadır. Bu tip özelliklerden oluşan bağlamlara çok-değerli bağlam denmektedir.

Ölçeklendirme ile yapılan işlem, ilk maddede de belirtildiği gibi çok-değerli bağlamlardan tek değerli/ikili bağlamların elde edilmesidir. Bunun için akla gelebilecek en kolay yöntem, çok-değerli bağlamdaki bir özelliğin her değerinin ayrı bir özellik olarak tanımlanmasıdır. Örneğin kelime türü özelliği için isim, sıfat ve fiil biçiminde üç farklı değer varsa “isim olma”, “sıfat olma” ve “fiil olma” şeklinde üç ayrı özellik haline dönüştürüldüğünde nesneler için bu özelliklere sahip olma ve olmama bilgisi işaretlenebilir. Dolayısıyla bağlam çok-değerli halden tek değerli/ikili hale getirilebilir. Yapılan işleme de ölçeklendirme denir.

Tez boyunca gereken yerlerde ölçeklendirme yapılmıştır. Örneğin, gerçekleştirilen uygulama esnasında, kullanılan özellikler sebebi ile ölçeklendirmeye ihtiyaç duyulmuştur. Ayrıca BKA yazılımlarının incelemesi sırasında kullanılan basit bağlam için de ölçeklenmiş hali ilgili bölümde verilmiştir.

Sonraki bölümde verilen örnek ile ölçeklendirme konusu açıklanmaya devam edilecektir.

4.9.1 Basit bir ölçeklendirme örneği

Aşağıda (Wolff, 1993)'ten alınan basit bir ölçeklendirme örneği adım adım verilmektedir. İlk olarak çok değerli bağlam verilmektedir.

Tablo 4.2. Ölçeklendirme yapılması gereken çok değerli bir bağlam

K0	Sex	Age
ADAM	M	21
BETTY	F	50
CHRIS	/	66
DORA	F	88
EVA	F	17
FRED	M	/
GEORGE	M	90
HARRY	M	50

Yukarıdaki tabloda bilinmeyen değerler yerine “/” işareti konmuştur. Yukarıdaki tablonun biçimsel bir bağlama dönüştürülmüş hali aşağıdaki gibidir.

Tablo 4.3. İkili değerlere dönüştürülmüş biçimsel bağlam

	Sex		Age				
K0	m	f	< 18	< 40	<= 65	> 65	>= 80
ADAM	X			X	X		
BETTY		X			X		
CHRIS						X	
DORA		X				X	X
EVA		X	X	X	X		
FRED	X						
GEORGE	X					X	X
HARRY	X				X		

Bu tablodan elde edilen ölçeklendirmeler ise aşağıdaki gibidir.

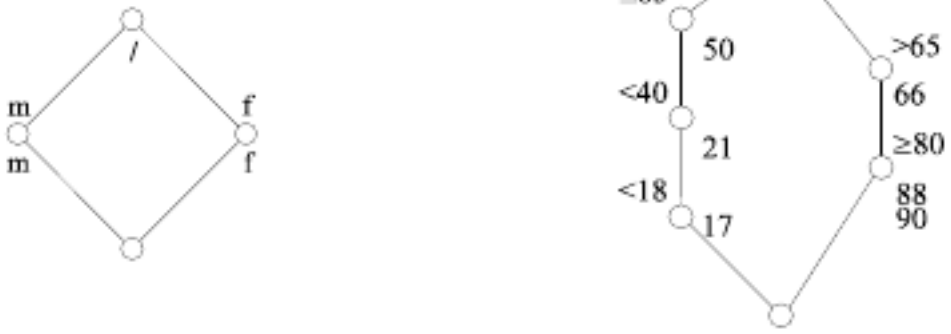
Tablo 4.4. Ölçeklendirmeye ilişkin ilk biçimsel bağlam

S1	m	f
M	X	
F		X
/		

Tablo 4.5. Ölçeklendirmeye ilişkin ikinci biçimsel bağlam

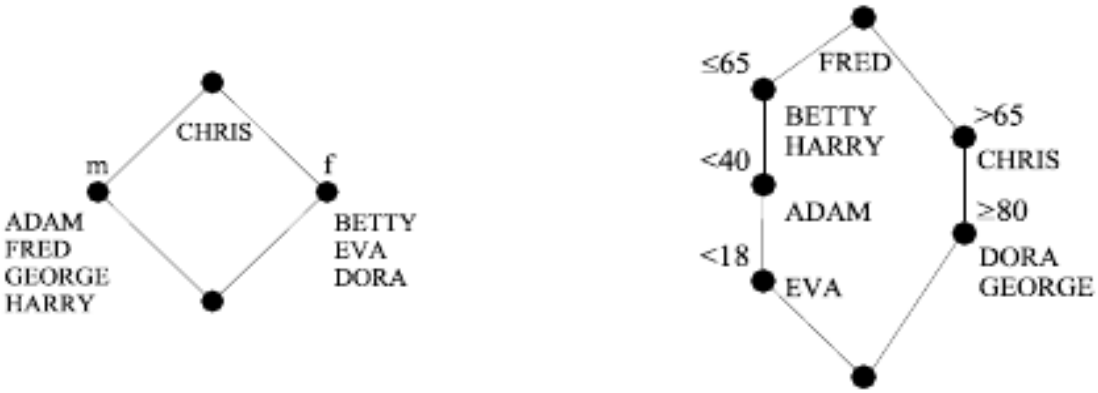
S2	< 18	< 40	<= 65	> 65	>= 80
17	X	X	X		
21		X	X		
50			X		
66				X	
88				X	X
90				X	X
/					

Bu noktada özelliklerin görselleştirilmiş hali aşağıdaki gibi olmaktadır.



Şekil 4.3. Ölçeklendirmelere ilişkin nesnelerin olmadığı latisler

Her iki durumda da “/” herhangi bir özelliğe sahip olmadığından en üst kavram olarak gösterilmiştir. Nesnelerin de yerleştirilmesi ile elde edilen son durum aşağıdaki gibidir.



Şekil 4.4. Ölçeklendirmelere ilişkin nesnelerin yerleştirildiği latisler

4.10 Biçimsel Kavram Analizi İçin Kullanılan Yazılımlar

BKA için kullanılabilecek yazılımların incelemesine geçmeden önce bu inceleme için kullanılan temel yapıları ve varsayımları vermek faydalı olacaktır.

İlk olarak, tüm yazılımları değerlendirirken kullanılacak veri aynı olacaktır. Bu amaçla aşağıda verilen çok-değerli (multi-valued) bağlam kullanılacaktır. Böylelikle yazılımların ölçeklendirme yapabilme ve yapamama durumları net şekilde görülecektir. Örneğin basit tutulması ve uygulamadaki durumu da yansıtması amacıyla özellik olarak sadece Türkçe'deki ismin hal ekleri seçilmiştir. Dolayısıyla beş farklı değer alabilen bir tek özellik ve toplam beş farklı nesne kullanılmıştır.

Tablo 4.6. BKA yazılımlarının değerlendirilmesinde kullanılacak bağlam

	Özellik
Nesne_1	yalın_hal
Nesne_2	e_hali
Nesne_3	i_hali
Nesne_4	de_hali
Nesne_5	den_hali

Bazı yazılımlar ölçeklendirme yapamamaktadır. Dolayısıyla Tablo 4.6 ile verilen bağlam, mevcut hali ile bu yazılımlar kullanılarak görselleştirilemez. Bu noktada (Andrews, 2009)'da anlatılan yöntem kullanılarak her bir değer için ayrı bir özellik tanımlanıp bağlamın ikili hale dönüştürülmesi sağlanabilir. İkili hale dönüştürülmüş bağlam aşağıdaki gibidir.

Tablo 4.7. Tablo 4.6'nın ölçeklendirilmesi ile elde edilen bağlam

	öz_yalın_hal	öz_e_hali	öz_i_hali	öz_de_hali	öz_den_hali
Nesne_1	X				
Nesne_2		X			
Nesne_3			X		
Nesne_4				X	
Nesne_5					X

Tablo 4.6 ve 4.7 ile verilen iki bağlamın da kodladığı bilgi aynıdır. Sadece tablo biçimindeki gösterim değişmiştir. Tablo 4.7'de ölçeklendirme yapılırken, özellik adı ile özelliğin aldığı değer bilgisi birleştirilerek yeni özellik adları elde edilmiştir. Zaten Tablo 4.6 ile verilen gösterimi kullanabilen yazılımların da ölçeklendirme ile veriyi önce Tablo 4.7'deki hale getirdiğini ve ardından görselleştirmeyi yaptığı görülmektedir/görülecektir.

Farklı yazılımlar tarafından değişik şekillerde çizilen latisler elde

edilebilmektedir. Konu ile alakalı olarak (Priss, 2008b)'de “grafik benzerlik” terimi tanımlanmakta ve bizim de örnek bağlam ile kullanacağımız yazılımları http://www.upriss.org.uk/fca/examples.html adresinde verilen örnekler için değerlendirmektedir. Elde edilen en ilginç sonuç ise, BKA'nın teorisini ortaya koyanlardan Wille'nin kullandığı çizim yöntemine en yakın gösterimlerin, BKA için tasarlanmamış, genel amaçlı bir grafik çizim/görselleştirme yazılımı olan Graphviz ile elde edilenler olmasıdır. (Adı geçen yazılım bu bölümde incelenmeyecektir.) Priss’in makalesinde yukarıda verilen web adresindeki örnekler kullanılmış ve bu kısımda ele alınan yazılımların görselleştirme örnekleri verilmiştir. Ancak kullanılan bağlamların büyük ve karmaşık yapıları olduğundan burada onlardan faydalanmak yerine yukarıda verilen basit örnek üzerinden hareket edilmiştir. Ayrıca Priss sadece elde edilen görselleştirmeleri değerlendirdiği ve yazılımların diğer özelliklerine değinmediği için tek başına yeterli bir kaynak oluşturmamaktadır.

İncelenecek yazılımlar için (Tilley, 2004)’te ve (Priss, 2008b)’de verilen listelerden ve bilgilerden faydalanılmıştır. Bu şekilde elde edilen ve görselleştirme için kullanılabilecek yazılımların isimleri sırasıyla ToscanaJ yazılım takımı[4], ConExp ve Galicia’dır. Yaygın olarak kullanılan bu yazılımların haricinde veri dönüşümü için FCA Stone uygulamasından bahsedilecektir. Ayrıntılı olarak bahsedilecek olan yazılımların dışında FCA Bedrock ve In-Close gibi başka yazılımlar da mevcuttur. FCA Bedrock, bağlam oluşturma için kullanılabilen bir yazılımdır. Ancak sadece Microsoft Windows platformunda çalışmaktadır. In-Close ise CXT formatlı bağlam dosyalarını inceleyerek kavram sayısı gibi bazı bilgiler sunan konsol tabanlı bir uygulamadır. Bu yazılımlar hakkında daha fazla bilgi verilmeyecektir. Sadece uygulama ve sonuç elde etme sürecinde aktif olarak kullanılmış olan yazılımlar ilerleyen başlıklarda ayrıntılandırılacaktır.

[4] İngilizce kaynaklarda geçen “ToscanaJ suite” terimi yerine Türkçe “ToscanaJ yazılım takımı” kelimesi kullanılmıştır.

4.10.1 ToscanaJ yazılım takımı

ToscanaJ yazılım takımının ayrıntılı açıklaması ve TOSCANA sisteminden nasıl ToscanaJ haline geldiği (Becker ve Correira, 2005)'te ayrıntılı şekilde anlatılmaktadır. Bu bölümde yazılımın tarihi gibi ayrıntılara girilmeyecektir. Ayrıca (Tilley, 2004)'te de yazılımlar ile ilgili bilgiler bulunmaktadır.

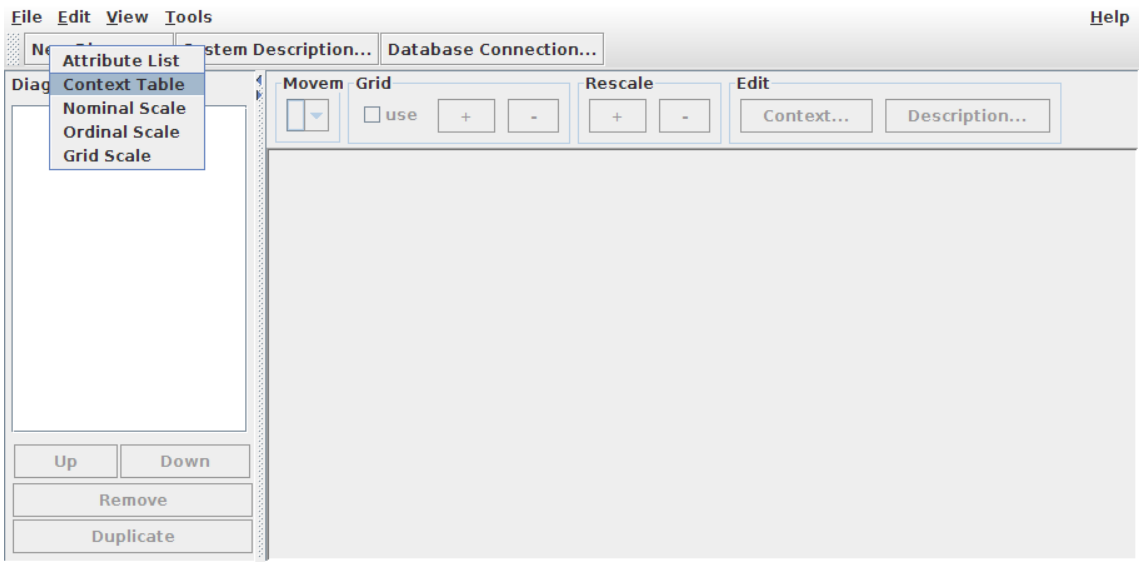
ToscanaJ yazılımı değil de yazılım takımı olarak nitelendirilmesinin sebebi, ToscanaJ'nin bir tek yazılım olmayıp üç ayrı yazılımdan oluşmasıdır. Bunlar sırası ile Elba, Siena ve ToscanaJ'dir. Elba, veritabanı bağlantısı olan ve verinin veritabanında olduğu durumlarda kullanılmak üzere bulunan kavramsal sistem editörüdür (conceptual system editor). Siena, veritabanı bağlantısının olmadığı durumlarda kullanılmak üzere hazırlanmış bir kavramsal sistem editörüdür. Burada kavramsal sistem editörü ile kastedilen BKA analizi için bağlam oluşturma, değiştirme gibi işlemlerin yapılabildiği ve elde edilen bilgilerin kavramsal sistem şeması (conceptual system schema) adı verilen bir yapıda (bu yapı csx uzantılı bir dosya şeklindedir) saklayan yazılımlardır. ToscanaJ yazılımı ise editörler aracılığı ile oluşturulan kavramsal sistem şemasını okuyup görüntüleme işlemi yapan kavramsal sistem görüntüleyicidir[5]. ToscanaJ yazılımının diğer iki editör gibi bağlam oluşturma ve değişiklik yapma özelliği yoktur, sadece diğer iki editörle oluşturulan ve kavramsal sistem planı olarak kaydedilen bilgiyi görselleştirebilmektedir. Dolayısıyla sadece bir görüntüleyicidir. ToscanaJ ile görüntülenen kavramsal şemalar/latisler png, jpg gibi farklı resim formatlarında kayıt edilebilmektedir. Belirtilen işlem editörler ile yapılamamaktadır. ToscanaJ, Java programalama dili ile geliştirilmiş açık kaynak kodlu bir yazılım takımıdır.

Geliştirilen uygulama ve aşağıda verilen örnek için veritabanı dosyaları hazırlamaya gerek duyulmamıştır. Dolayısıyla, ilk anda sadece Siena ile bağlam girişi yapılarak latislerin çizdirilebileceği düşünülmüştür. Ancak Windows XP Service Pack 3 yüklü bir makinede Java 1.6, 1.5, 1.4 ve hatta 1.3 ile çalıştırıldığı halde basit bir bağlam girişi yapılamamıştır. Yazılım Java 1.6 ile hiç açılmamaktadır. Diğer Java sürümleri ile açılmakta, Burmeister formatı olarak bilinen cxt uzantılı dosyaları okuyup

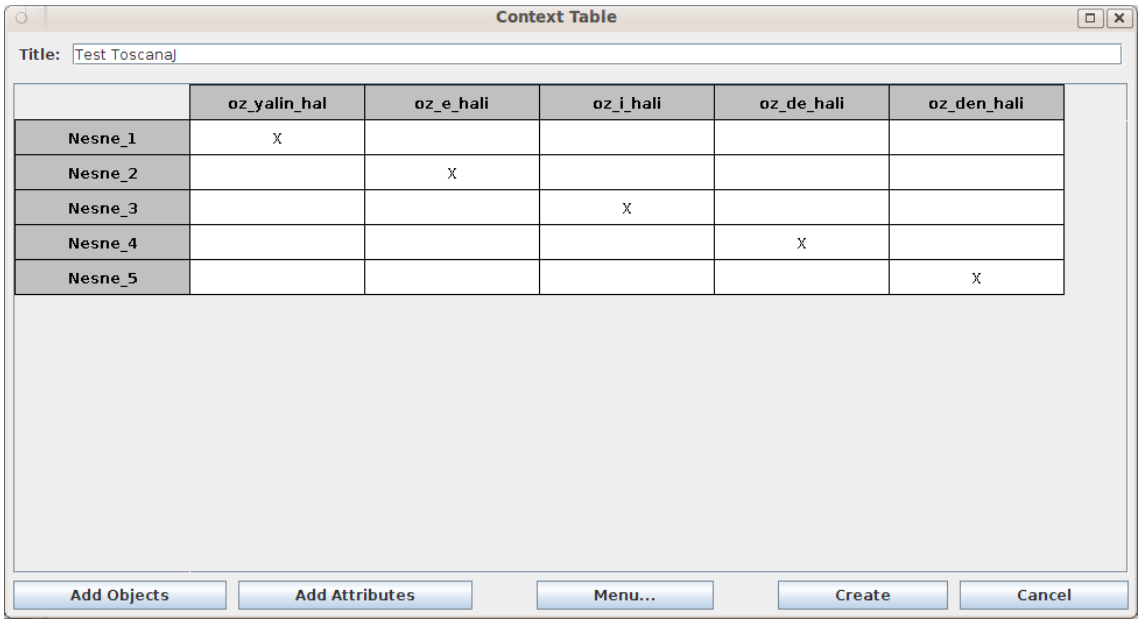
[5] İngilizce "browser" yerine Türkçe görüntüleyici kelimesi kullanılmıştır.

görüntüleyebilmekte ancak basit bir bağlam oluşturmaya izin vermemektedir. Durum Ubuntu 9.10 yüklü Linux sisteminde de Java 1.6 ile hiç açılmama şeklinde kendini tekrarlamıştır. Bu sebeplerle Elba yazılımı veritabanı bağlantısı olmadan kullanılarak görseller elde edilmiştir. Elba ile veritabanı oluşturmadan bağlam girişi yapılabilmesi için grafik arayüzünün sol üst köşesindeki "New Diagram" düğmesine basılıp gelen listeden "Context Table" seçeneği seçilmiştir. Bu şekilde tablo biçiminde nesne ve özellik listeleri girilip ardından nesnelerin sahip oldukları özellikler fare yardımı ile çift tıklanarak işaretlenmişlerdir. Belirtilen biçimde oluşturulan bağlam kaydedilmeye çalışıldığında veritabanı bağlantısı olmadığına dair bir uyarı penceresi gelmektedir. Bu pencerede bulunan "Drop database information" düğmesine basıldığında oluşturulan bağlam sorunsuz şekilde kavramsal şema (csx uzantılı dosya) olarak kaydedilip kullanılabilmektedir.

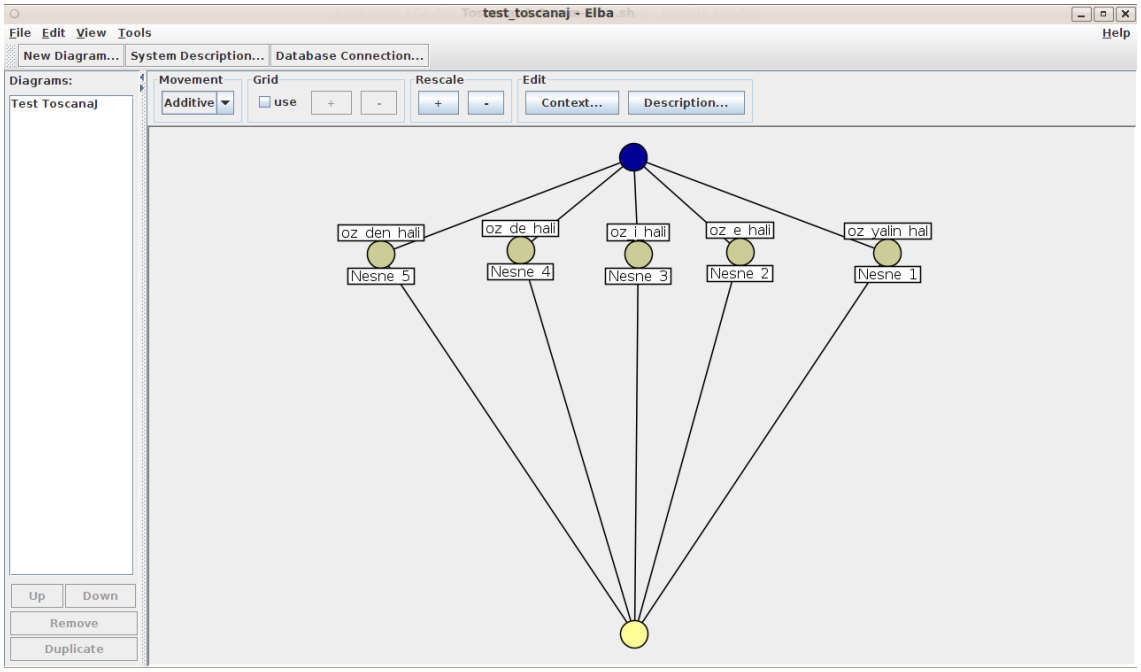
Elba ile mevcut veri giriş yöntemi kullanılarak ölçeklendirme işlemi yapılamamıştır. Bu sebeple Tablo 4.7 ile verilen bağlam kullanılarak işlemler gerçekleştirilmiştir. Örnek bağlamın Elba ile girişi, elde edilen latis görselleştirmesi, ToscanaJ ile kavramsal şemanın açılıp farklı formatlarda kayıt edilebilmesi durumları aşağıda verilmiştir.



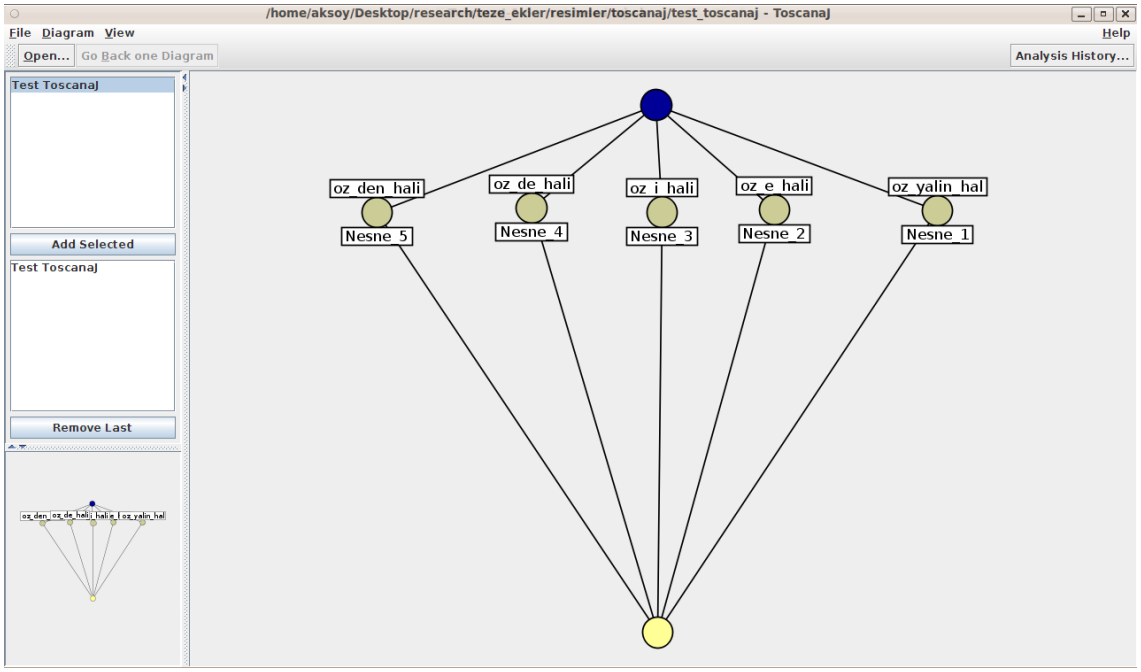
Şekil 4.5. Elba ile bağlam girişi yapılmasını sağlayan menü seçeneği



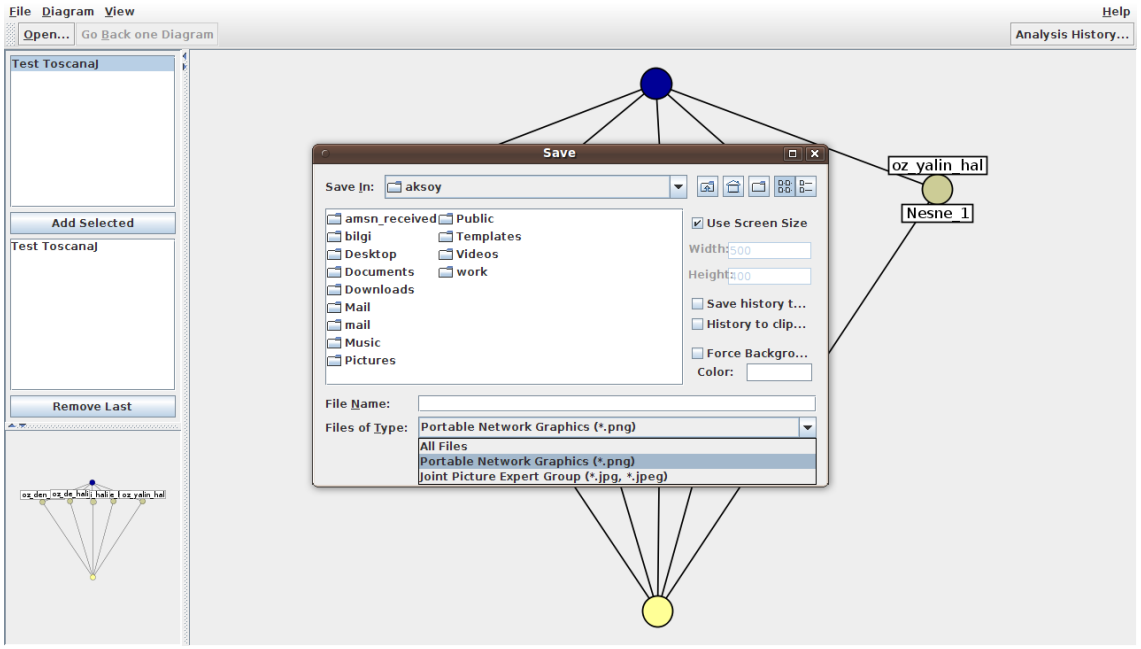
Şekil 4.6. Elba ile ikili bağlam girişi



Şekil 4.7. Elba ile latis gösteriminin elde edilmesi



Şekil 4.8. Elba ile oluşturulan bağlamın/latisin ToscanaJ ile görüntülenmesi



Şekil 4.9. ToscanaJ ile görüntülenen latisin farklı formatta ve özellikte kaydedilmesi

4.10.2 ConExp yazılımı

ConExp ismi İngilizce **Con**cept **Exp**lorer kelimelerinin ilk kısımlarının birleştirilmesi ile elde edilmiştir. Java programlama dili ile geliştirilen ve açık kaynak kodlu bir yazılımdır. Bağlam oluşturma ve görselleştirme için ayrı ayrı yazılımları yoktur, tek bir araç olarak her iki işi de yapmaktadır. ToscanaJ sistemindeki Elba gibi veritabanı bağlantısı yapamamaktadır. Bağlamlar, Burmesiter formatı olarak bilinen cxt uzantılı dosyalar halinde ConExp'e aktarılabilmekte ve ConExp'ten alınabilmektedir. Ayrıca virgülle ayrılmış değerler şeklindeki CSV dosyalarını da alıp kullanabilmektedir. Yazılım Duquenne-Guigues-tabanlı dolaylı bilgileri hesaplayabilmekte ve kullanıcı ile etkileşimli olarak özellik keşfi yapabilmektedir. Duquenne-Guigues-tabanlı dolaylı bilgilerinin en temel özelliği, bağlamdan çıkarılması mümkün olan tüm dolaylı bilgiler içinden mümkün olan en az sayıdaki dolaylı bilgiyi çıkartmasıdır. Elde edilen dolaylı bilgiler aşağıdaki formatta görüntülenmektedir:

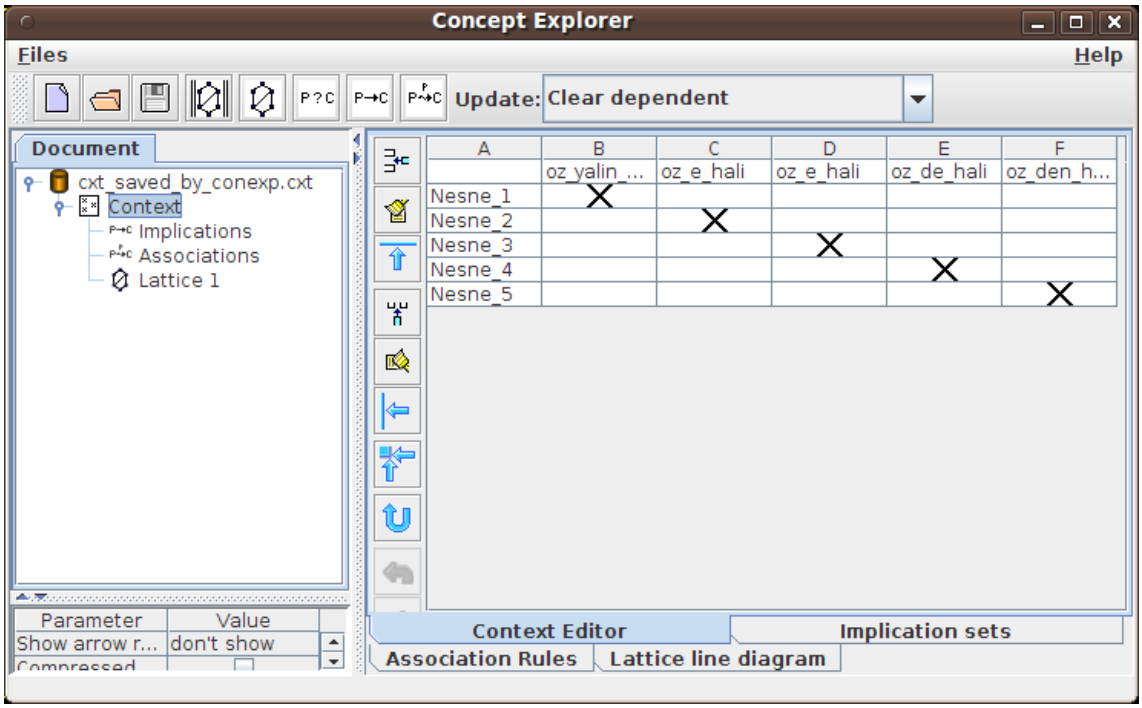
No <Nesne sayısı> Önerme ==> Sonuç

No ile kastedilen sıra numarasıdır. Dolaylı bilginin geçerli olduğu nesne sayısı ikinci kısımda verilmektedir. Önerme ve sonuç kısımlarında genellikle özellik isimleri yer almaktadır. Dolaylı bilgi mavi renkte gösteriliyorsa bağlamda kurala uyan nesneler olduğunu, kırmızı renkte gösteriliyorsa mevcut bağlamda bu kurala uyan nesne olmadığını belirtmektedir.

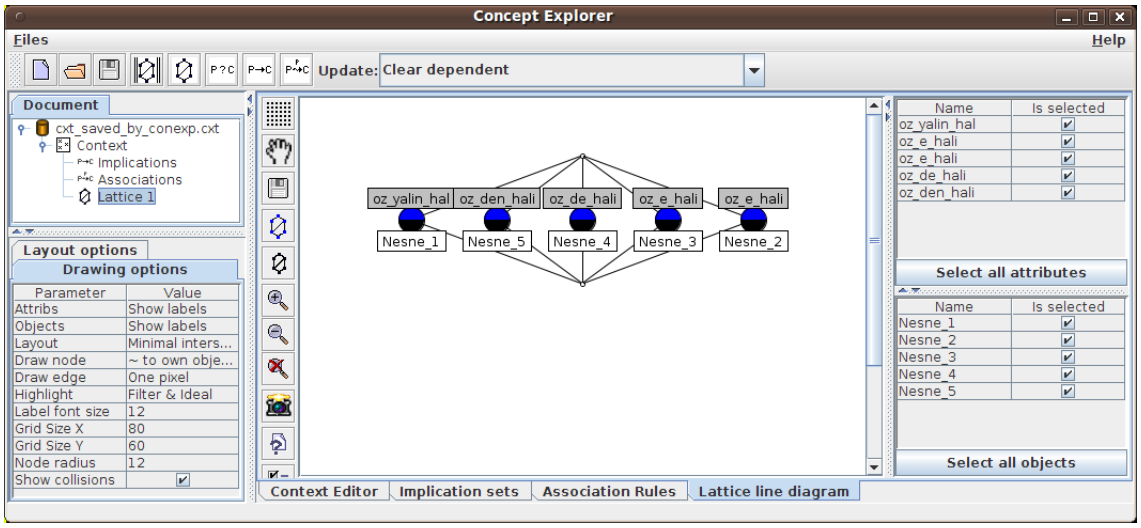
Yazılım özellik keşfine de izin vermektedir. Özellik keşfinin amacı, ilgilenilen alanın genelinde geçerli olmayan ancak kullanılan bağlamda geçerli olan bazı dolaylı bilgilerin olması probleminden kurtulmaktır. Kullanıcı ile etkileşimli bir prosedürdür. Kullanıcıya farklı özellikler arasında bir bağımlılık olup olmadığı sorulur. Sorulara "Evet" cevabı verilirse diğer soruya geçilir. Eğer "Hayır" cevabı verilmişse kullanıcının karşı örnek girmesi istenmektedir. Ancak yazılım bunun için de kendi hazırladığı örneği kullanıcıya sunmaktadır. Dolayısıyla bu hazır örneğe onay vermek de yeterlidir. Burada

bir bağımlılığa olumlu veya olumsuz cevap verilmesi, durumun genel olarak mevcut olması ile ilgilidir. Bölümdeki "4.6 Dolaylı Olarak Bulunan Bilgi (Implications)" başlığında durumu açıklayan örnek mevcut olduğu için burada daha fazla ayrıntıya girilmeyecektir.

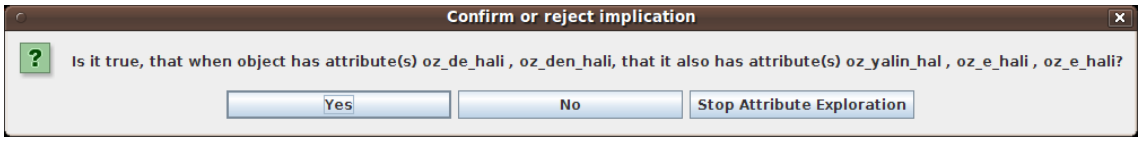
ConExp yazılımı ölçeklendirme yapamadığı için Tablo 4.7 ile verilen bağlam kullanılarak işlemler gerçekleştirilmiştir. Yapılan işlemlere ilişkin görseller aşağıdaki gibidir.



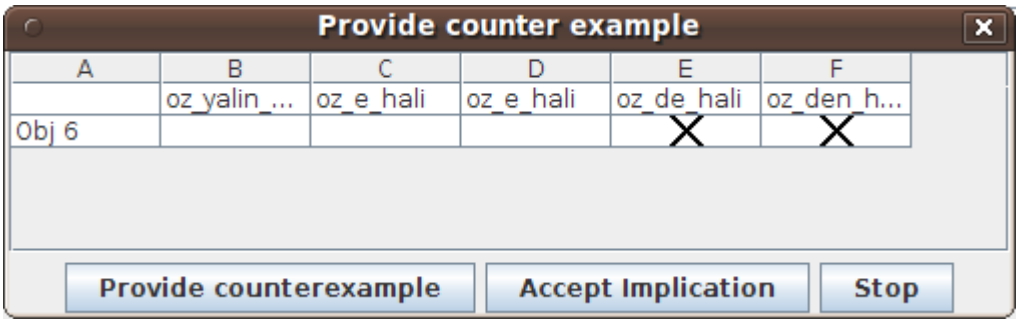
Şekil 4.10. ConExp ile bağlam oluşturma



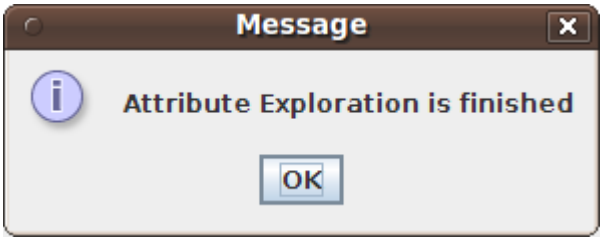
Şekil 4.11. ConExp ile elde edilen latis görselleştirmesi



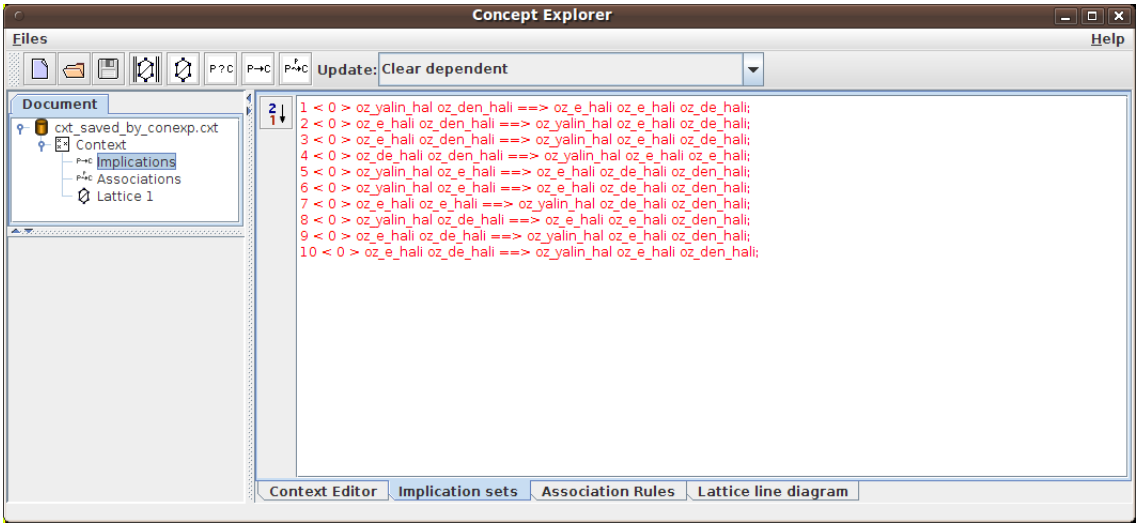
Şekil 4.12. ConExp ile dolaylı bilgi çıkarımı/özellik keşfi



Şekil 4.13. ConExp'te özellik keşfi sırasında karşı örnek isteme ekranı



Şekil 4.14. ConExp'te özellik keşfinin sonlandırılması ekranı

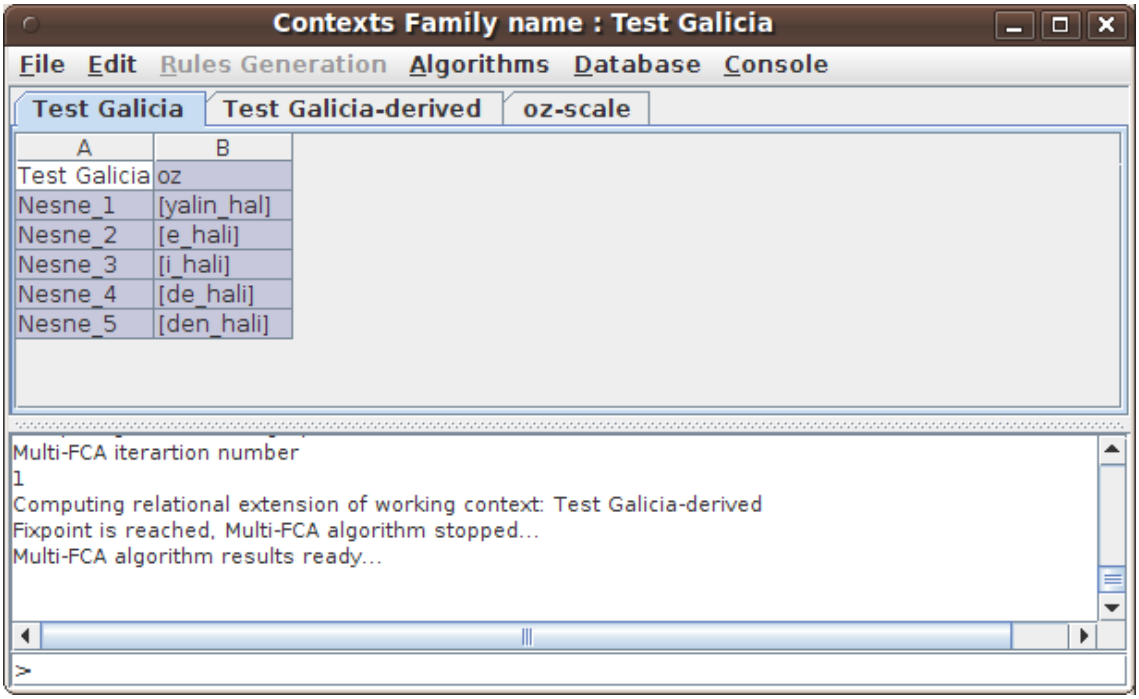


Şekil 4.15. ConExp ile elde edilen dolaylı bilgilerin görüntülenmesi

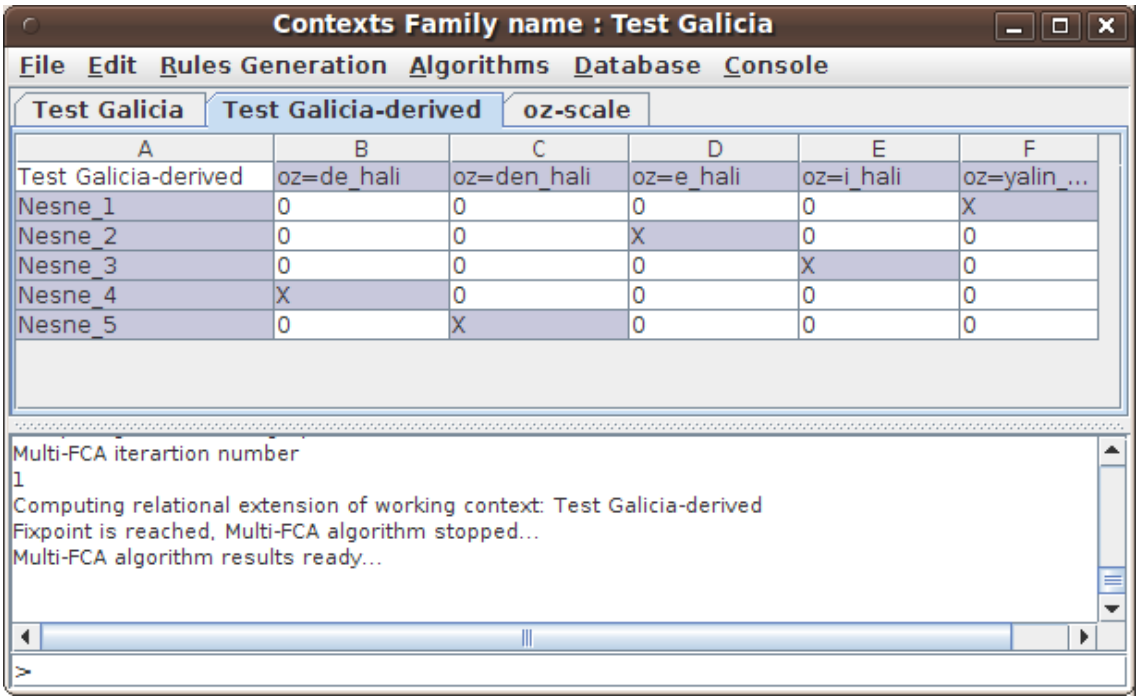
4.10.3 Galicia yazılımı

Galicia da Java tabanlı, bağlam oluşturma ve görselleştirme yazılımıdır. ConExp'de de olduğu gibi her iş için ayrı ayrı yazılımı yoktur, tek parçadır. Tek değerli ve çok-değerli bağlamlar Galicia ile analiz edilebilmektedir. Çok-değerli bağlamlar için elde edilen/ölçeklendirilen tek değerli/ikili bağlamlar yazılım tarafından kullanıcıya gösterilmektedir. Kendine ait RCF uzantılı İlişkisel Bağlam Ailesi (Relational Context Family) yapısı bulunmaktadır. Bu yapı sayesinde bağlamdaki nesneler arasındaki ikili ilişkiler de girilebilmekte, RCF formatı içinde saklanabilmektedir. Ayrıca, elde edilen latis gösterimi için diğer yazılımlardaki gibi iki boyutlu gösterimin yanında üç boyutlu (3D) gösterim imkanı da sunmaktadır.

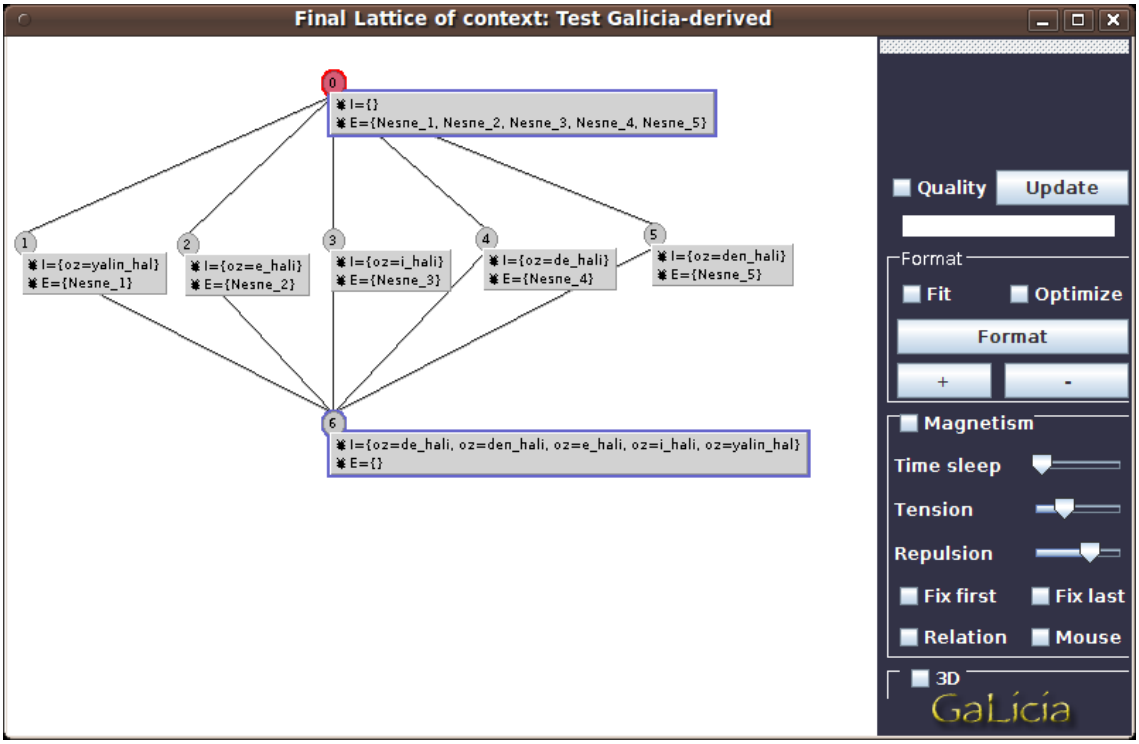
Çok-değerli bağlam girişine izin verdiği için yazılımın testi Tablo 4.6 ile verilen bağlam kullanılarak yapılmıştır. Bu bağlamın yazılım tarafından ölçeklenmesi ile elde edilen bağlam Tablo 4.7'de verilen ile aynıdır. Galicia, özellik adı ile ölçeklendirdiği değer ismini birleştirerek yeni bir isim oluşturmaktadır. Aynı yaklaşım Tablo 4.7'de de kullanılmıştır.



Şekil 4.16. Galicia ile çok-değerli bağlam girişi



Şekil 4.17. Galicia ile elde edilen ölçeklenmiş bağlam



Şekil 4.18. Galicia ile elde edilen latis gösterimi

4.10.4 ToscanaJ, ConExp ve Galicia'nın değerlendirilmesi

BKA camiası tarafından en sık kullanılan bağlam oluşturma ve görselleştirme yazılımlarının kullanımı esnasında karşılaşılan ve uygulamalar açısından önem arz edebilecek durumlar aşağıda karşılaştırmalı şekilde verilmiştir.

Her üç yazılımın kullanımı esnasında parametreli olarak çalıştırma özelliklerinin olmadığı dikkat çekmiştir. Dökümantasyonlarında olmasa bile yazılımlar ile gelen çalıştırma betikleri yerine doğrudan konsoldan çalıştırma denemeleri yapılmış ve ConExp ile ToscanaJ yazılımları örnek bir CXT dosya ile çalıştırılmaya çalışılmıştır. Ancak yazılımların hiçbiri kendilerine parametre olarak verilen dosyayı işleme almamışlardır. Bu durum uygulamaların otomatik olarak başlatılmasını engellemektedir. Öncelikle istenen yazılım çalıştırılmalı, ardından üzerinde işlem yapılmak istenen dosya

“Open” seçenekleri ile kullanılmaya başlanmalıdır. Bu noktada yazılımların açık kaynak kodlu olmalarından faydalanılarak istenen özelliğin eklemesinin yapılabileceği düşünülmektedir.

İncelemesi yapılan yazılımlar arasında sadece Galicia’nın çok-değerli bağlam girişlerine izin verdiği ve ölçeklendirme yapabildiği tespit edilmiştir. ToscanaJ yazılım grubundan Siena ile ilgili belgelerde ölçeklendirme yapılmasına ilişkin bilgiler bulunmuştur. Ancak yazılım verimli şekilde çalıştırılarak kullanılamadığı için bu özellik denenememiştir. Dolayısıyla incelenen yazılımlar arasında sadece Galicia sorunsuz şekilde ölçeklendirme yapabilmektedir. Bu durumdan, uygulama geliştirme aşamasında faydalanılmıştır. Farklı yazılımlar kullanılmak istense ve elle ölçeklendirme yapılsa bile doğrulama amaçlı olarak Galicia kullanılmıştır. Ayrıca Galicia’nın I(ntension) ve E(xtension) kümeleri şeklinde gösterimi ve 3D gösterim özelliği de zaman zaman tercih edilebilir.

Yazılımlar arasında özellik keşfi ve dolaylı bilgi çıkarımı, anlatıldığı üzere etkileşimli şekilde ve sadece ConExp tarafından yapılabilmektedir. Uygulamada son derece faydalı olan bu özellik geliştirilmiş olan filtre için de uygundur.

Her üç yazılım da bağlam girişi ve görselleştirme için kullanılabilmektedirler. Ancak farklı veri formatları kullanarak veri kaydetmeleri sebebi ile yazılımların birlikte kullanılabilmesi için veri çevrimleri gereklidir. BKA camiası da son yıllarda farklı yazılımların veri biçimlerinin birbirleri ile uyumlu hale getirilebilmesi ve birlikte çalıştırılmaları konusuna yönelmiştir. Özellikle veri çevrimi konusu bir sonraki başlıkta ayrıntılı olarak ele alınmaktadır.

Sonuç olarak, incelenen tüm yazılımlar bağlam girişi ve görselleştirme için kullanılabilir. Yazılımların sahip oldukları ek özelliklere ihtiyaç duyulması halinde yazılım seçimi buna uygun olarak yapılabilir. Ancak yazılımların birlikte kullanılması gereğinin ortaya çıkması halinde (her durumda mümkün olmamakla birlikte) veri çevrimleri yapılmalıdır.

4.10.5 FCA Stone ve BKA yazılımları arası veri dönüşümü

BKA yazılımlarının çeşitliliği ve farklı veri formatları kullanmaları sebebi ile yazılımlar arası veri dönüştürme işlemleri gerekli olmaktadır. Çünkü daha önce yazılımların anlatıldığı bölümde dile getirildiği gibi her yazılım (çok değerli bağlamları ölçeklendirebilme, farklı gösterimler oluşturmaya izin verme gibi) kendine has özelliklere sahiptir ve zaman zaman bu özelliklerden faydalanabilmek için yazılımlar arası geçiş yapmak gerekebilir. Böylesi durumlarda ya yazılımların aynı formatta veri kaydetmesi sağlanmalıdır ya da farklı formatlar birbirine dönüştürülebilir olmalıdır.

Yazılımların aynı formatta veri kaydetmek konusunda ortak bir standart takip etmemeleri sebebi ile dönüşümler konusu ağırlıklı olarak değerlendirilmektedir. Son yıllarda BKA alanında bu durum üzerine çalışmalar artmıştır. Özellikle (Priss, 2008a) ve (Andrews, 2009)'da duruma dikkat çekilmiştir.

FCA Stone yazılımı farklı formatlar arası veri dönüşümü yapmak üzere geliştirilmiş, Perl programlama dili ile yazılmış, platform-bağımsız bir konsol uygulamasıdır. Linux, Mac OS X ve Windows ortamında çalışabilmektedir. Uygulamanın grafik arayüzü yoktur. Ancak uygulama BKA alanındaki tüm formatları başarı ile çevirememktedir. Örneğin ToscanaJ yazılımının kavramsal sistem şeması olan CSX dosyalarını dönüştürememektedir. Aşağıdaki tablo (Priss, 2008a)'dan alınmıştır ve FCA Stone tarafından hangi formatın hangi işlem için desteklendiğini göstermektedir.

Tablo 4.8. FCA Stone tarafından desteklenen formatlar

extension	I/O	type	scope	Graphivz required?	comments
cxt	input/output	FCA format	only context	no	P. Burmeister's format
con					Colibri format
slf					Galicia format
bin.xml					Galicia format
tuples		tab separated values			Tuplewware format (like csv, but tab instead of comma + additional first line) only two column files supported
csv		comma separated values			used by databases/spreadsheets
csc		FCA format	context + lattice		F. Vogt's Anaconda format (lattice not implemented)
cex				no	ConExp, lattice not yet implemented
csx				no	ToscanaJ, lattice not yet implemented
fig	output only	vector graphics		yes for lattice	xfig
tex		latex			to be used with B. Ganter's fcastv, lattice not yet implemented
dot		graph format	only lattice	no	Graphviz format
gml				no	
gxl				yes	format availability depends on local Graphviz installation
svg		vector graphics			
jpg		raster graphics			
gif					
png					
ps		page description format			
pdf					

Tablonun en başında yer alan ve Peter Burmeister tarafından tasarlandığı için Burmeister format olarak da bilinen ve CXT uzantısını kullanan dosya formatı bir çok BKA yazılımı tarafından kullanılabilen ortak bir BKA bağlam formatıdır (Andrews, 2009). Ayrıca aynı kaynakta dosya formatı da ayrıntılı şekilde verilmektedir. İncelenen yazılımlardan ToscanaJ ve ConExp, Burmeister formatını okuyabilmektedirler. Ayrıca Galicia yazılımının kullandığı bazı formatların da FCA Stone tarafından (CXT formatına) dönüştürülebildiği yukarıdaki tablodan görülmektedir. Dolayısıyla CXT dosyaları ve dosya formatı uygulamalar arası geçiş için uygun görünmektedir.

4.10.6 Burmeister dosya formatı (CXT dosyaları)

Burmeister dosyaları olan CXT uzantılı dosyalar incelendiğinde, verilerin (Andrews, 2009)'da verilen dosya yapısına uygun şekilde yerleştirildiği görülmektedir. Ancak dikkat edilmesi gereken konu Burmeister format'ın çok-değerli bağlamlar için kullanılamayıp, sadece tek değerli bağlamlar için kullanılabilmesidir. Dolayısıyla, çok-değerli bağlamlar ancak ölçeklendirme yapıldıktan sonra bu formata dönüştürülebilirler.

Bir CXT dosyasının ilk satırında (muhtemelen Burmeister'in baş harfi olduğu için) "B" karakteri bulunmaktadır. Bir satır boşluktan sonra nesne sayısı üçüncü satırda, özellik sayısı da dördüncü satırda verilmektedir. Ardından bir satır boşluk bırakılarak her satıra sadece bir tane yazılmak koşulu ile önce nesne adları ardından da özellik adları verilmektedir. En son kısımda da tablo şeklinde verilen bağlamın her bir satırı bir satıra gelecek şekilde nesne-özellik ilişkileri kodlanmıştır. Bir nesne bir özelliğe sahip ise o kolon için "X" değeri, sahip olunmayan özellikler için de "." karakteri kullanılmıştır. Aşağıdaki bağlam ve bu bağlam için elde edilmiş olan CXT dosyası (Andrews, 2009)'dan alınmıştır. Ayrıca http://www.upriss.org.uk/fca/examples.html adresinde daha karmaşık latisler için bağlam, latis gösterimi ve CXT dosyalarını bulmak mümkündür.

Mushroom	bruises	gill-size-broad	gill-size-narrow	veil-type-partial	veil-type-universal	ring-number-none	ring-number-one	ring-number-two
mushroom1	×	×		×		×		
mushroom2	×		×	×				×
mushroom3			×	×		×		
mushroom4	×			×			×	
mushroom5			×	×		×		

Şekil 4.19. (Andrews, 2009)'daki örnek bağlam

```
B

5
8

mushroom1
mushroom2
mushroom3
mushroom4
mushroom5
bruises
gill-size-broad
gill-size-narrow
veil-type-partial
veil-type-universal
ring-number-none
ring-number-one
ring-number-two
XX.X.X..
X.XX...X
..XX.X..
X..X..X.
..XX.X..
```

Şekil 4.20. (Andrews, 2009)'daki örnek bağlamın Burmeister formatı

4.10.7 Diğer formatlar

Galicia yazılımının incelendiği bölümde çok-değerli bağlamların girişine izin verdiği ve ölçeklendirme yapabildiği belirtilmişti. Ölçeklendirme sonucu ikili hale getirilen bağlamlar "Export" seçeneği ile bir dosyaya kayıt edilebilmektedir. Bu şekilde elle ölçeklendirme yapmak yerine yazılım aracılığı ile yaptırılıp ardından da elde edilen ikili bağlam kullanılabilir. Belirtilen kullanım ile insan kaynaklı hatalardan da kurtulmak mümkün olacaktır. Bir ikili bağlamın doğrudan Galicia'ya girilip bu formatta kaydedilmesi de mümkündür. FCA Stone için verilen veri dönüşüm tablosunda yer alan SLF uzantılı dosyalar Galicia tarafından ikili bağlamların kaydedilmesinde kullanılmaktadır. Bu format sadece Galicia tarafından kullanıldığı için ancak (örneğin CXT'ye) dönüştürülerek diğer yazılımlar tarafından da bağlamın girdi olarak kullanılması sağlanabilir. FCA Stone SLF uzantılı dosyaları CXT formatına başarılı şekilde çevirebilmektedir.

Son olarak, virgülle ayrılmış değerler olarak geçen CSV (Comma Separated Values) ve nesne-özellik listesi olarak tanımlanan formatlar da ConExp tarafından

kullanılabilmektedir. (ConExp sadece ikili bağlamları kullanabildiği için dosyaların ölçeklendirmesinin yapılmış olması gerekmektedir.) Her iki dosya çeşidi de format olarak basit oldukları için, hem herhangi bir editör ya da bir yazılımcı tarafından hazırlanan bir program aracılığı ile oluşturulabilirler hem de CSV dosyaları Excel, OpenOffice Spreadsheet gibi hesap tablosu yazılımları tarafından üretilen veriler için elde edilebilir. Yapılan testlerde FCA Stone yazılımının CSV formatından CXT formatına dönüşümü doğru yapamadığı görülmüştür. Bu durum CSV formatlı verinin ConExp ile açılıp "Save as" seçeneğiyle CXT formatlı olarak kaydedilmesi şeklinde çözülebilmektedir. Böylelikle CSV dosyalarının, dolaylı olarak da hesap tablosu uygulamaları kullanılarak elde edilen verilerin, BKA analizinde kullanılması sağlanabilir. Nesne-özellik listeleri OAL uzantılı dosyalar şeklindedir. Bir editör aracılığı ile elle oluşturulmaları ya da bir yazılım ile hazırlanmalarına imkan verecek şekilde basit bir yapıları vardır. Nesne-özellik listesi dosya formatı ConExp kullanıcı kılavuzunda aşağıdaki gibi verilmiştir.

nesne_adı_1:özellik_1;özellik_2;…;özellik_N

nesne_adı_2:özellik_1;özellik_2;…;özellik_M

Yukarıdaki yapıdan da anlaşıldığı gibi her satıra bir nesne adı, nesne adının ardından ":" ve aralarına ";" konularak nesnenin sahip olduğu özellikler yazılır.

4.10.8 FCA Stone kullanım örnekleri

Yazılımları incelerken kullandığımız örneğimiz için Tablo 4.7'de verilen bağlamın farklı formatlarda görüntüleri aşağıda verilmiştir.

Galicia tarafından ölçeklendirme ile elde edilen SLF dosyası aşağıdaki gibidir :

```
[Lattice]
5
5
[Objects]
Nesne_1
Nesne_2
Nesne_3
Nesne_4
Nesne_5
[Attributes]
oz=de_hali
oz=den_hali
oz=e_hali
oz=i_hali
oz=yalin_hal
[relation]
0 0 0 0 1
0 0 1 0 0
0 0 0 1 0
1 0 0 0 0
      0 1 0 0 0
```

Yukarıda verilen ve Galicia tarafından üretilen SLF dosyasının FCA Stone yazılımı kullanılarak CXT haline dönüştürülmüş hali aşağıdaki gibidir.

```
B

5
5

Nesne_1
Nesne_2
Nesne_3
Nesne_4
Nesne_5
oz_yalin_hal
oz_e_hali
oz_e_hali
oz_de_hali
oz_den_hali
X....
.X...
..X..
...X.
....X
```

Kullanılan dönüştürme komutu sadece girdi dosyasını ve çıktı dosyasnı parametre olarak almaktadır. Dosya uzantılarının yazılması zorunludur. FCA Stone bu şekilde gerekli dönüşüme karar verebilmektedir. Yapılan testlerde SLF'den CXT'ye sorunsuz dönüşüm yapılmıştır. Ancak CSV dosyaları da destekleniyormuş gibi görünse

de başarılı sonuçlar alınamamıştır.

Örneğimizin MS Excel ya da OpenOffice Spreadsheet gibi bir yazılım ile tablo olarak girilip CSV olarak kaydedilmesi ile elde edilecek dosyanın bir editör yardımı ile açılmasıyla elde edilecek görüntü aşağıdaki gibidir :

```
;"oz_yalin_hal";"oz_e_hali";"oz_i_hali";"oz_de_hali";"oz_den_hali"
"Nesne_1";1;0;0;0;0
"Nesne_2";0;1;0;0;0
"Nesne_3";0;0;1;0;0
"Nesne_4";0;0;0;1;0
"Nesne_5";0;0;0;0;1
```

Örneğimizin ConExp'in desteklediği nesne-özellik listesi (Object Attribute List) dosyası olarak gösterimi aşağıdaki gibidir.

```
Nesne_1:oz_yalin_hal
Nesne_2:oz_e_hali
Nesne_3:oz_i_hali
Nesne_4:oz_de_hali
Nesne_5:oz_den_hali
```

Farklı formatların çıktıları incelendiğinde belli ve basit bazı desenleri takip ettikleri görülmektedir. Bu durumda MS Excel gibi belli bir yazılım kullanmadan da kullanıcı tarafından yazılan programların bu çıktıları üretmesi sağlanabilir. Ardından elde edilen dosya BKA görselleştirme yazılımlarından birine verilerek görselleştirilebilir. Böylelikle işlemlerin otomatizasyonunu sağlamak mümkün olabilir.

5. ÖZELLİKLERİN AYIRDEDİCİLİĞİNİN BİÇİMSEL KAVRAM ANALİZİ YARDIMI İLE DEĞERLENDİRİLMESİ

5.1 Giriş

Bu kısımda yapılacak uygulamanın adımları ayrıntılandırılacak ve uygulamadan elde edilen sonuçlar yorumlanacaktır. Uygulamanın ilk adımı kullanılacak veri setini seçmektir. İkinci olarak, kullanılacak özellikler ve alabilecekleri değerler listelenecektir. Üçüncü adımda, veri setindeki örnekler için özellik vektörleri oluşturulacaktır. Özellik vektörleri oluşturma işleminden sonra, son adımda, elde edilen vektörler biçimsel kavram analizi yazılımlarından Elba ile görselleştirilecek ve ortaya çıkan durumlar değerlendirilecektir. Yeri gelen bölümlerde konular ile alakalı tüm ayrıntılar verilecektir.

KAB alanında daha önce de sınırlı veri seti ile çalışmalar yapılmıştır. Örneğin, (Ng ve Lee, 1996)'da 200'den az kelimelik bir sözlük kullanmıştır. (Cowie vd. 1992) açık bir sözlük kullanmamakla birlikte testlerini elli cümle ile yapmışlardır. Yarowsky 12 kelime için KAB işlemi gerçekleştirmiş ve %92 başarı bildirmiştir. Ama bu başarı "interest" kelimesi için %72'ye düşmektedir. Benzer biçimde "interest" kelimesi için (Black 1988) %72, (Zernik 1990) %70, (Bruce ve Wiebe 1994) %79 başarı bildirmişlerdir. Her ne kadar %92 çok başarılı bir yüzde olarak görünse de "interest" kelimesi için hemen hemen diğer tüm sistemler aynı seviyede ve daha düşük başarı bildirmişlerdir. Ayrıca (Yarowsky, 1995) ve (Brown vd. 1991) her çok anlamlı kelimenin sadece iki anlamının olduğu test derlemleri kullanmışlardır.

Uygulamada verilen kaynaklardakine benzer bir yaklaşım benimsenmiştir. Durum yeri geldikçe ayrıntılı şekilde verilmektedir.

5.2 Örneklemelerin Seçimi (Veri Setinin Oluşturulması)

Veri seti oluşturulurken (Prati vd., 2004)'teki yaklaşım kullanılarak yapay bir veri seti oluşturulmasının uygulama ve tez açısından uygun olduğuna karar verilmiştir. Durum belirtilen kaynakta aşağıdaki gibi verilmektedir:

"… çalışmamızı bir grup yapay veri seti üzerinde geliştiriyoruz. Yapay veri setleri kullanılmasının ardındaki fikir, analiz etmek istediğimiz değişkenlerin hepsini tam olarak kontrol edebilmektir. Eğer bu değişkenleri kontrol edemezsek, yanıltıcı sonuçlar üretme riski altında, sonuçlar maskelenebilir ya da anlaması ve yorumlaması güç bir hale gelebilir." (Prati vd., 2004)

Yapay veri seti kullanımına karar verildikten sonraki ilk adım KAB için kullanılacak kelimeyi ve farklı anlamlarını belirlemek olmuştur. Farklı anlamların belirlenmesi için Türk Dil Kurumu'na ait Güncel Türkçe Sözlük kullanılmıştır. Seçilen "yüz" kelimesi için verilen anlamlar arasından beş tanesi kullanılmak üzere seçilmiş ve anlam etiketleri belirlenmiştir. Kullanılan açıklamalar belirtilen siteden alınarak Ek-B'de verilmiştir. Uygulamamız için seçilen anlam etiketleri aşağıdaki gibidir :

Tablo 5.1. Örneklemeler ve atanan anlam etiketleri

Anlam etiketi	**Örnek kullanım**
İlgilenmek	Yüz vermedi.
Not	Sınavdan yüz aldı.
Yüzey	Suyun yüzü kirli idi.
Dış kaplama	Yorganın yüzü kirli idi.
Utanma	Konuşacak yüzü yoktu.
İnsan organı	Çocuğun yüzü kirli idi.

Üzerinde öğrenme işlemi gerçekleştirilecek olan yapay veri seti toplamda yedi cümle ve üç farklı gruptan oluşmaktadır. Aşağıda tablolar halinde örneklemeler bulunmaktadır.

Tablo 5.2. İlk grupta yer alan örneklemeler ve anlam etiketleri

Grup - I		
Sıra No.	**Örnek cümle**	**Anlam etiketi**
1	Suyun yüzü pislik içindeydi.	Yüzey
2	Yorganın yüzü pislik içindeydi.	Dış kaplama
3	Çocuğun yüzü pislik içindeydi.	İnsan organı

Tablo 5.3. İkinci grupta yer alan örneklemeler ve anlam etiketleri

Grup - II		
Sıra No.	**Örnek cümle**	**Anlam etiketi**
1	Arkadaş Ayşe'ye kızgın olduğu için yüz vermedi.	İlgilenmek
2	Öğretmen Ayşe'ye kızgın olduğu için yüz vermedi.	Not

Tablo 5.4. Üçüncü grupta yer alan örneklemeler ve anlam etiketleri

Grup - III		
Sıra No.	**Örnek cümle**	**Anlam etiketi**
1	Hiç bir konuda konuşmak için yüzü yoktu.	Utanma
2	Hiç bir dersten övünmek için yüzü yoktu.	Not

Örneklemelerin gruplara ayrılmasının sebebi, değerlendirme yaparken ayrı ayrı ele alınacak olmalarıdır. Ayrıca yapay veri seti kullanma sebebi açıklanırken belirtildiği gibi, tüm değişkenleri kontrol edebilmek adına bu şekilde hareket edilmiştir. BKA ile görselleştirme yapılıp sonuçlar yorumlanırken durum daha iyi anlaşılacaktır.

5.3 Kullanılacak Özelliklerin Seçilmesi

İkinci sırada yapılması gereken, kullanılacak özelliklerin seçilmesidir. Konu ile ilgili olarak özellikle KAB literatüründe bolca kullanılan ve işe yaradıkları farklı uygulamalarla belirlenmiş özelliklerin bir alt kümesi kullanılacaktır. Bir alt küme kullanılmasının sebebi ise çıkarımların daha kolay şekilde yapılmasıdır. BKA ile ilgili bölümde verilen yazılımlar kullanılırken, özellik vektörleri elle girilecektir. Bu sebeple uygulamanın sonuçlarının kolay anlaşılabilir olması için az miktarda özellik kullanılacaktır. Gelecekte yapılacak olan çalışmalar başlığı altında da belirtildiği gibi işlemlerin otomatize edilmesi durumunda özellik miktarının fazla olması problem olmayacaktır. Ancak bu aşamada tezin anlaşılabilirliği ve açıklık adına belirtilen basitleştirilmiş yaklaşım kullanılacaktır.

Kullanılacak olan özellikler aşağıda listelenmektedir. Ancak her özellik her durumda kullanılmayacaktır. Özellikler ile ilgili kısaltmalar ve karşılıkları aşağıdaki gibidir :

Tablo 5.5. Kullanılan kısaltmalar ve karşılıkları

Kısaltma	Açıklama
ABK	Anlamı Belirginleştirilecek Kelime
ÖK	Önceki Kelime
SözR	Sözdizimsel Rol
GramR	Gramatikal Rol
Hal	Hal bilgisi
TemR	Tematik Rol
KTB	Kelimenin Tür Bilgisi
Eşdizim	Eşdizimli kelime
SemTür	Semantik Tür

Verilen kısaltmalardan faydalanarak kullanılması planlanan özellikler için oluşturulan tablo aşağıdaki gibidir :

Tablo 5.6. Kullanılacak özellikler ve kullanılacakları yerler

	SözR	GramR	Hal	TemR	KTB	Eşdizim	SemTür
ABK	+	+	+	+	+	+	-
ÖK	+	+	+	+	+	-	-

Tabloda "+" ile ifade edilenler kullanılacak, "-" ile ifade edilenler ise kullanılmayacak olan özelliklerdir.

Kullanılması planlanan özellikler, özellik vektöründeki alanlara karşılık gelmektedir. Bu alanların alabileceği farklı değerler de aşağıda verilmiştir. Uygulamamız ve örneklemelerimiz göz önüne alınarak her özellik için kullanılabilecek tüm değerler değil sadece örneklemelerimiz için geçerli olabilecek olanlar kullanılmıştır. Bunun sebebi, BKA ile görselleştirme yapılırken her yazılımın çok-değerlibağlamlar/veriler ile çalışamaması nedeniyle tek değerli hale getirilmesi zorunluluğudur. Kısaca, tek değerli hale getirme esnasında özellik sayısının çok fazla artmasını engellemek için bir özelliğin alabileceği her değer değil uygulamada kullanılacak olan değerler ile yetinilmiştir. Görselleştirme kısmında bu durum tekrar ele alınacaktır.

Tablo 5.7. Kullanılacak özellikler ve kullanılacakları yerler

Özellik	Alabileceği Değer Sayısı	Değerler
SözR	2	İsim öbeği – Fiil öbeği
GramR	2	Özne – Nesne – Tümleç
Hal	7	Yalın, -e, -i, -de, -den, sahiplik, ilgi
TemR	3	Agent – Patient -
KTB	3	İsim – Fiil – Sıfat
Eşdizim	Sınırsız	Sınırsız
SemTür	2	Canlı – Cansız

5.4 Uygulamanın Yapılışı

Öncelikle ilk örnekleme grubu için uygulama yapılacak, ardından diğerleri için de durum tekrarlanacaktır. İlk uygulamada tüm ayrıntılar verilecek, diğerlerinin yapılışı esnasında ortak durumlar için tekrar anlatım yapılmayacaktır.

İlk gurup için önceki bölümde listelenen özellikler ve aldıkları değerler aşağıdaki gibidir :

Grup – I :

a-) Suyun yüzü pislik içindeydi.

ANLAM : Yüzey

Tablo 5.8. Yüzey anlamı için ilk gruptaki örneklemenin özellik vektörü

	SözR	GramR	Hal	TemR	KTB	Eşdizim	SemTür
ABK – yüzü	İsim öbeği	Özne	İlgi	Patient	İsim	-	X

b-) Yorganın yüzü pislik içindeydi.

ANLAM : Dış kaplama

Tablo 5.9. Dış kaplama anlamı için ilk gruptaki örneklemenin özellik vektörü

	SözR	**GramR**	**Hal**	**TemR**	**KTB**	**Eşdizim**	**SemTür**
ABK – yüzü	İsim öbeği	Özne	İlgi	Patient	İsim	-	X

c-) Çocuğun yüzü pislik içindeydi.

ANLAM : İnsan organı

Tablo 5.10. İnsan organı anlamı için ilk gruptaki örneklemenin özellik vektörü

	SözR	**GramR**	**Hal**	**TemR**	**KTB**	**Eşdizim**	**SemTür**
ABK – yüzü	İsim öbeği	Özne	İlgi	Patient	İsim	-	X

Dikkat edilecek olursa özelikle son kısımdaki Semantik Tür bilgisi tüm örneklemeler için işaretlenmeden bırakılmıştır. Bu durum "X" karakteri ile ifade edilmiştir. Benzer şekilde Eşdizim özelliği de hiç bir örnekleme için doldurulmamıştır. Ancak Semantik Tür bilgisinden farklı olarak, herhangi bir eşdizimli kelime belirlemesi yapılmadığı ya da eşdizimli kelime olmadığı için durum "-" ile gösterilmiştir. Bu noktada Semantik Tür bilgisi bilerek kullanılmamıştır. BKA uygulamasının sonucunda neden bu şekilde davranıldığı net şekilde görülecektir.

5.5 Biçimsel Kavram Analizi ile Özelliklerin Değerlendirilmesi

Biçimsel Kavram Analizinin iki ana bileşeni ilgili bölümde de anlatıldığı gibi nesneler ve özelliklerdir. Tez boyunca çalışma alanı KAB ile sınırlandığı için, KAB işlemine tabi tutulan kelimelerin anlamlarının her biri uygulamada ayrı birer nesne olacaktır. Kullanmaya karar verilen özellikler de BKA'da kullanılan özellikler olarak değerlendirilecektir. Dolayısıyla uygulamada kullanılacak biçimsel bağlamlar aşağıdaki gibi olacaktır :

Tablo 5.11. Çok-değerliden tek değerliye çevrilmiş biçimsel bağlamın yapısı

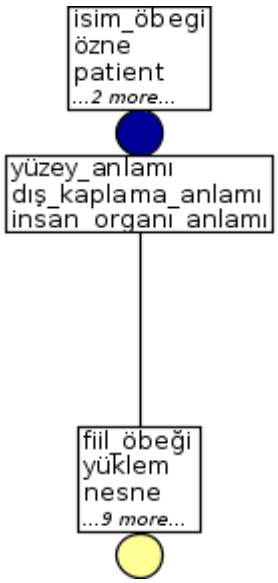
Nesne	**Özellik_1**	**Özellik_2**	**…**	**Özellik_N**
Anlam_1	X	X		X
Anlam_2	X	X		X
…				

Biçimsel Kavram Analizi ile alakalı bölümde belirtildiği üzere görselleştirme için kullanılan yazılımların hepsi, çok-değerli bağlamlar ile çalışamamaktadır. Bu sebeple ölçeklendirme yapılmalıdır. Akla gelebilecek en pratik çözümlerden biri her bir özelliğin her bir değerinin ayrı ayrı yazılması ile tek değerli bir bağlam oluşturulmasıdır. (Konu BKA ile ilgili bölümde ayrıntılı olarak ele alınmıştır. Ayrıca, doğrulama için Galicia yazılımının ölçeklendirme yeteneği de kullanılmıştır.) Bu şekilde elde edilen ve ilk gruptaki üç örnekleme için ortak olan bağlam aşağıdaki gibi olmuştur.

Tablo 5.12. Kullanılacak özelliklerin ikili değerlere dönüştürülmesiyle elde edilen biçimsel bağlam

Özellik adı	Değer
İsim öbeği	+
Fiil öbeği	-
Özne	+
Nesne	-
Yüklem	-
Yalın_hali	-
-e_hali	-
-i_hali	-
-de_hali	-
İyelik	-
İlgi	+
-den_hali	-
Agent	-
Patient	+
İsim	+
Sıfat	-
Fiil	-
Canlı	X
Cansız	X

İlk gruptaki örneklemeler/yukarıdaki tablo için Elba aracılığı ile bağlam oluşturularak elde edilen latis aşağıdaki gibidir:



5.6 Biçimsel Kavram Analizi ile Elde Edilen Latisin Yorumlanması

Şekilde görüldüğü gibi "yüz" kelimesine ait tüm anlamlar aynı düğümün altına gelmiştir ve bu düğümde ortak olan beş tane özellik vardır. Bu özellikler sırası ile isim öbeği içinde olma, özne olma, ilgi ekine sahip olma, tematik rol olarak patient olma ve kelime türü olarak "isim" olmak seklindedir. Sayılan tüm özellikler bütün anlamlar için ortaktır. Bu sebeple düğümün üstünde özelliklerin listesi, alt kısmında da bu özelliklere sahip nesneler (KAB uygulaması açısından anlamlar) bulunmaktadır. Ayrıca özellik tablosunda "-" şeklinde işaretlenen özellikler de en alttaki düğümde verilmiştir. Bu düğümde hiç nesne bulunmamaktadır. BKA açısından bakıldığında en alt düğüm özel bir anlama sahiptir. Kullanılan bütün özellikleri gösteren nesneler bu düğümde toplanmaktadır. (Konu ile ilgili ayrıntılar BKA ile ilgili bölümde verilmiştir.) KAB uygulaması açısından böylesi bir durum mümkün değildir. Çünkü çok değerliden tek değerliye dönüştürülen bağlamda aynı anda tüm özelliklere sahip olunması mümkün değildir. Böylesi bir durum tüm özellikleri sağlayan bir anlam olması demek olacaktır. Ortaya çıkması halinde derhal kullanılan özellikler ve öğrenme verisi üzerindeki işaretlemenin gözden geçirilmesi gerektiği gibi sonuçlar ortaya çıkacaktır.

Mevcut şekle bakıldığında, her üç anlamında aynı düğüm altında toplanmasını mevcut özellikler kullanılarak eldeki veri seti ile öğrenme yapıldığı durumda elde edilen özellik vektörlerinin "yüz" kelimesinin bu üç anlamını ayırdetmeye yetmeyeceği şeklinde yorumlanabilir. Denetimli makine öğrenmesi bölümünde bahsedildiği gibi öğrenme verisinde işlem yapıldıktan sonra test verisi üzerinde çalışmaya hiç başlamadan BKA tabanlı bu filtre uygulandığında elde edilecek sonuç, mevcut özellik vektörleri ile test verisi üzerinde KAB işlemi yapmanın kazanç sağlamayacağıdır. Çünkü, (uç bir durum olarak değerlendirilebilecek olsa bile) eldeki durumda KAB işleminin sonucu farklı anlamların ayrıştırılamaması olacaktır.

Bu noktada değerlendirilmesi gereken durumların bir listesi aşağıda verilmiştir :

1. Kullanılan özellikler yeterli değil midir?
2. Özelliklerin değerlerinin verilmesi sırasında hata mı yapılmıştır?
3. Üzerinde öğrenme gerçekleştirilen veri setinde dengesizlik mi vardır?

4. KAB işlemi için belirlenen anlamlar arasında alt-üst ilişkisi gibi birbirini kapsama durumu mu söz konusudur?
5. Kullanılan özellikler arasında alt-üst ilişkisi gibi birbirini kapsama durumu mu söz konusudur?

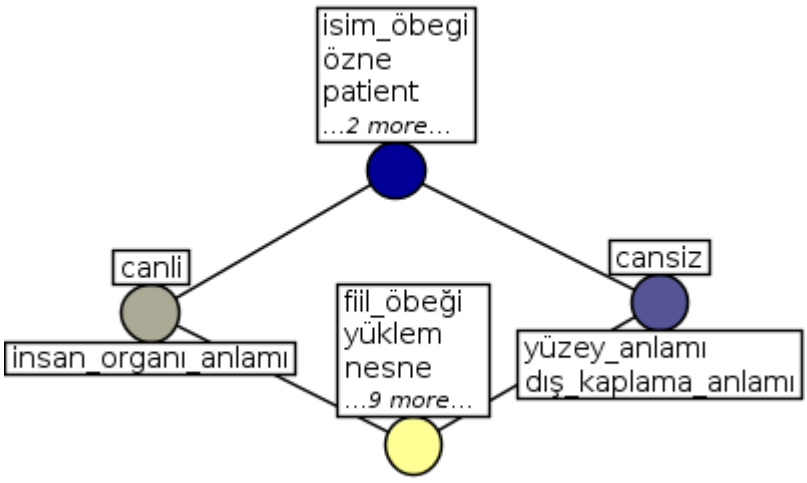
Sebep her ne olursa olsun denetimli makine öğrenmesinin bir sonraki adımına geçilmemesi gereklidir.

Yukarıda sıralanan durumların sırası ile incelnmesi aşağıdaki gibidir:

1. Kullanılan özelliklerin yeterli olmaması : KAB ve/veya makine öğrenmesi için seçilen özellik kümesi hedeflenen amaç için yetersizdir. Mevcut durumda kullanılan özelliklerin ayırdedici gücü yoktur. Önerilen filtrenin amacı da bu durumu belirleyebilmektir. Kullanılan özellikler değiştirilmeli ya da yeni özellikler eklenerek işlem tekrarlanmalı ve bu süreç ayırdedici gücü olan özellikler kümesine ulaşılana kadar devam etmelidir.
2. Özelliklerin değerlerinin verilmesi sırasında hata yapılması : Özellik vektörlerinin doldurulması konusunda ayrıştırıcı vb. yazılımlardan faydalanılabileceği gibi konu ile ilgilenen uzmanların elle işaretleme yapabileceği daha önce belirtilmiş idi. Bu sebeple bir insan hatası ya da yazılımın doğru şekilde işleyemediği bir durum olup olmadığını kontrol etmek gerekebilir.
3. Üzerinde öğrenme işlemi gerçekleştirilen veri seti içinde dengesizlik vardır : Veri seti dengeleme konusu makine öğrenmesi alanında üzerinde önemle durulan bir problemdir. Çünkü veri içindeki dengesiz dağılım öğrenme algoritmasını yanıltıp belli bir yöne doğru hata yapmasına sebep olabilir. Konu ile alakalı olarak (Japkowicz, 2000)'de açıkca dengesizliğin performansı kötüleştirdiği ve bir çok alanda rastlandığı için son derece önemli bir problem olduğu belirtilmektedir. Ancak tez boyunca KAB ve özellik seçimi konusu ile ilgilenildiği için bu alana girmeyecek ve veri setinin dengeli olduğu varsayılacaktır.

4. KAB işlemi için belirlenen anlamlar arasında alt-üst ilişkisi gibi birbirini kapsama durumu söz konusudur : Bu durum anlam belirlemesi yapılırken örtüşen anlamlar için farklı etiketler verilmesi sebebi ile oluşabilir. İki ayrı durum göz önüne alınmalıdır: İlk olarak, anlam seçimi yapılan kaynakta (tezimiz açısından TDK Güncel Türkçe Sözlük) verilen anlamlar çok ince ayrımlarla birbirinden farklıdır ve bu durum normal bir kullanıcı için bile zor anlaşılır olabilir. (Benzer bir durum WordNet için geçerlidir.) Uygulamamızda kullanılan "yüz" kelimesi için ayrık anlamlar seçilerek bu problemin yaşanması bilinçli olarak engellenmiştir. İkinci olarak, seçilen ve örtüşen anlamlar için farklı etiket belirlenmesi sebebi ile kullanılan ve ayırdetme açısından yetersiz olan özelliklerin kendilerini bu şekilde göstermiş olmasıdır. Bu konu ile de tez boyunca ilgilenilmeyecektir.
5. Kullanılan özellikler arasında alt-üst ilişkisi gibi birbirini kapsama durumu söz konusudur : Kullanılması planlanan özellikler arasında farkedilmeme ya da hatalı seçim sonucu ortaya çıkan bir kapsama ilişkisi vardır. Bu duruma da tez boyunca girilmeyecektir.

Kullanılan örnekleme kümesi için şu ana kadar kullanılmayan Canlı – Cansız özelliklerini de işaretleyerek yani bu özellikleri de özellik kümemize ekleyerek ortaya çıkan durum yeniden incelenebilir. Ekleme yapıldıktan sonra elde edilecek BKA tabanlı gösterim aşağıdaki gibi olacaktır.



Şekil 5.2. Canlı-cansız ayrımının eklenmesi ile elde edilen kavram latisi

Şekilden de anlaşılabileceği gibi Canlı – Cansız şeklindeki Semantik Tür bilgisi özellikler arasına eklendiğinde "insan organı" anlamı ayırdedilebilir hale gelmektedir. Her ne kadar yüzey ve dış kaplama anlamları hala ayırdedilebilir olmasalar da benzer şekilde özellik ekleme vb. bir yöntem uygulandıktan sonra filtre aracılığı ile değerlendirme işlemi gerçekleştirilerek gerçekten ayırdedici özellikler ihtiva eden bir küme ile öğrenme ve sınıflandırma yapılması sağlanabilir.

5.7 Biçimsel Kavram Analizi Tabanlı Filtre İle Ortaya Çıkabilecek Durumlar

Bu kısımda verilen örnekten bağımsız olarak görselleştirme ile ortaya çıkabilecek durumlar ve bu durumlara ilişkin teorik bilgiler verilecek, ardından hem verilen örneğin yeniden değerlendirilmesi hem de diğer örneklerin incelemesi ve değerlendirilmesi yapılacaktır.

İşlem kolaylığı açısından ve verilen örneklemelerde de takip edilen yöntem olarak bazı kabuller yapılmıştır:

1. En fazla iki farklı anlam için durum değerlendirmesi yapılacaktır.
2. Gösterimde nesneler, nesne1, nesne2 şeklinde ve özellikler de özellik1, özellik2 şeklinde sıralanacaktır.
3. Kullanılacak özellik sayısında bir sınırlama yoktur.

Temel kabullere uygun olarak ortaya çıkabilecek durumlar için BKA tabanlı görselleştirmeler ve yorumlar aşağıdaki gibidir. Anlatımda öncelikli olarak elde edilen görselleştirmeler, ardından da görsellere ilişkin yorumlar verilecektir.

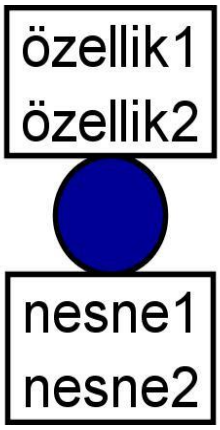
5.7.1 Ayırdedici özellik olmaması durumu :

Bu duruma ilişkin ortaya çıkabilecek farklı bağlamlar mevcuttur. Bunların bir kısmı görselleştirilerek açıklama yapılacaktır. Kullanılacak olan ilk biçimsel bağlam aşağıdaki gibidir :

Tablo 5.13. Ayırdedici özelliklerin olmadığı durumda ortaya çıkabilecek bir bağlam

	özellik1	Özellik2
nesne1	X	X
nesne1	X	X

Verilen bağlam sonucu ortaya çıkan şekil aşağıdaki gibidir :



Şekil 5.3. Tablo 5.13 ile verilen biçimsel bağlama ilişkin kavram latisi

Elde edilen görsel şu şekilde yorumlanabilir : Eldeki tüm nesneler için kullanılan bütün özellikler ortaktır. KAB açısından değerlendirilecek olursa bir kelimenin iki farklı anlamı için kullanılması planlanan bütün özellikler aynı değerlere sahiptir. Dolayısıyla mevcut durumdaki özellik vektörleri ile ya da yapılmış olan öğrenme ile bu iki anlamı ayırdetmek mümkün değildir.

Bu durumda yapılabilecekler ise şu şekilde sıralanabilir : Eldeki özelliklerin veri seti içinde doğru şekilde kodlandığının kontrol edilmelidir, yeni özellikler eklenerek tekrardan ayırdediciliğin tespit edilmesi ve bu işlemin ayırdediciliği olan bir özellik

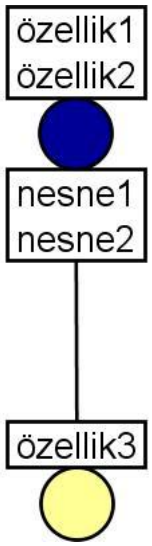
kümesi elde edilene ya da kullanılabilecek tüm özellikler kümeye eklenene kadar devam edilmelidir. Eğer kullanılması planlanan tüm özellikler eklendiği halde ayırdediciliği olan bir özellik kümesine ulaşılamıyor ise tüm süreç (kullanılması planlanan özellikler, örneklemeler, öğrenme verisi, seçilen anlamlar vb.) gözden geçirilmelidir.

Diğer bir bağlam aşağıdaki gibi olabilir :

Tablo 5.14. Ayırdedici özelliklerin olmadığı durumda ortaya çıkabilecek başka bir bağlam

	özellik1	özellik2	özellik3
nesne1	X	X	
nesne1	X	X	

Verilen bağlama ilişkin elde edilen şekilsel gösterim aşağıdaki gibidir :



Şekil 5.4. Tablo 5.14 ile verilen biçimsel bağlama ilişkin kavram latisi

Elde edilen görsele ilişkin olarak bir önceki yorumlama aynen geçerlidir. Bunun dışında ise özellik3'ün hiçbir nesne ile eşleşmemesinden KAB açısından bu özelliğin eldeki anlamlar için işaretlenmesinin gereksiz olduğu sonucu çıkartılabilir. Dolayısıyla mevcut durum için özellik vektöründen çıkarılmasında sakınca yoktur. Böylelikle özellik vektöründe bir küçülme ve işlem zamanından kazanç sağlanacaktır.

Bu noktada şunu söylemek yanlış olmayacaktır : Hangi durum olursa olsun BKA açısından özel bir duruma sahip olan en alttaki düğüme ilişkin özellikler kullanılmamaktadır. Çünkü en alttaki düğüm tüm özelliklere sahip olan bir anlamı gösterecektir ki KAB uygulaması açısından böylesi bir durum mümkün değildir. Dolayısıyla özellik vektöründen çıkarılmasında sakınca yoktur ve işlem zamanı açısından kazanç sağlanacaktır. En alttaki düğüm için belirtilen durum bundan sonraki gösterimler için de aynen geçerli olduğundan bir daha ayrıca ele alınmayacaktır.

5.7.2 Ayırdedici özellik bulunması durumu

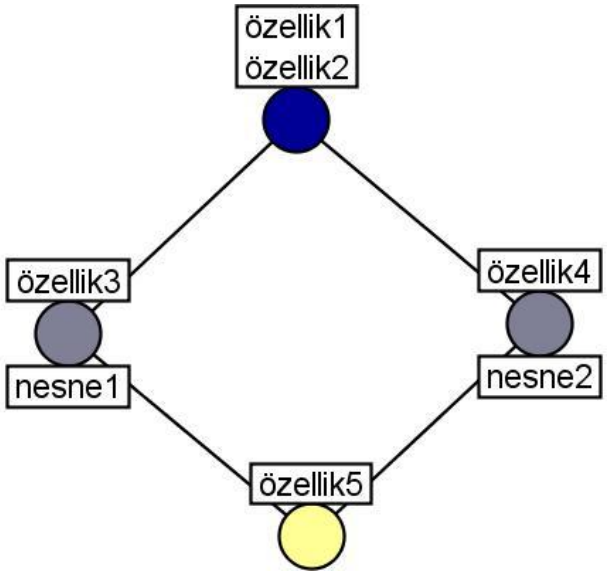
Ayırdedici özellik bulunmasına ilişkin en önemli durum farklı nesnelerin farklı düğümler altında yer almasıdır. Bu duruma ilişkin tek örnek verilecektir. Belirtilen özellik sağlandığı müddetçe (bağlam ve latisin büyüklüğü ne olursa olsun) ayrı düğümler üzerinde yer alan özellikler ayırt edici özellikler olarak değerlendirilecektir.

Gösterim için kullanılacak olan örnek bağlam aşağıdaki gibidir :

Tablo 5.15. Ayırdedici özelliklerin olduğu durumda ortaya çıkabilecek bir bağlam

	özellik1	özellik2	özellik3	özellik4	özellik5
nesne1	X	X	X		
nesne2	X	X		X	

Verilen bağlama ilişkin olarak elde edilen latis tabanlı gösterim aşağıdaki gibidir:



Şekil 5.5. Tablo 5.15 ile verilen biçimsel bağlama ilişkin kavram latisi

Yukarıdaki gösterimden çıkarılabilecek sonuçlar şunlardır : Özellik3 nesne1 için, özellik4 ise nesne2 için ayırdedicidir, dolayısıyla mutlaka özellik vektöründe bulunmalıdır. Özellik1 ve Özellik2 her iki nesne için de ortaktır. Dolayısıyla doğrudan bir ayırdediciliği yoktur. Duruma göre özellik vektöründen çıkarılması düşünülebilir. Duruma göre denmesinin sebebi özellik1, özellik2 ve özellik3 birlikte kullanılarak nesne1'e yani KAB açısından bir anlama ulaşılabilir. Eğer daha sonra üzerinde çalışılacak test verisinde özellik1 ve özellik2'ye sahip olmayan ancak özellik3 ve özellik4'e sahip olan bir örnekleme ile karşılaşılırsa mevcut durum karar vermede önemli bir rol oynayacaktır. Yine önceki başlıkta olduğu gibi en alt düğüme ilişkin olan özellik5 hiç kullanılmamıştır, ayırdediciliği yoktur ve özellik vektöründen çıkarılabilir.

Bu noktaya kadar anlatılanları özetlemek gerekirse, BKA ve dolayısıyla latis tabanlı gösterimler aracılığı ile çıkarılabilecek sonuçlar aşağıdaki gibi listelenebilir:

- En üst düğümde yer alan özellikler tüm nesneler için ortaktır. Dolayısıyla duruma göre özellik vektöründen çıkarılmasına karar verilebilir.
- En altta yer alan düğümde yer alan özellikler KAB uygulamaları açısından hiç bir nesneye yani anlama karşılık gelmemektedir. Hiç bir ayırdediciliği

yoktur. Dolayısıyla işlem zamanı gibi maliyetler açısından özellik vektöründen çıkarılmalıdır.

- Ayırdediciliği olan özellikler ayrı nesnelerin bulunduğu ayrı düğümlerde yer almaktadır/almalıdır. Özellik vektöründen kesinlikle çıkarılmamalıdırlar.
- Eğer görselleştirme sonucu ayırdedici özellik olmadığı sonucuna ulaşılırsa, kullanılabilecek başka özellikler özellik vektörüne eklenerek durum tekrar değerlendirilmelidir. Bu işlem eldeki bütün özellikler kullanılana ya da ayırdedici özellikler bulunana kadar devam etmelidir.
- Eğer kullanılması planlanan bütün özellikler vektöre dahil edildiği halde ayırdedici durumlar ortaya çıkmıyorsa, kullanılan öğrenme verisi örneklemelerin özellikleri açısından dengesiz olabilir. Ayrıca, özellik işaretlemeleri konusunda hata yapılmış olması ihtimali de mevcuttur. Dolayısıyla her iki durum da gözden geçirilmeli ve filtre ile desteklenen öğrenme işlemi tekrarlanmalıdır.

Yukarıda belirtilen durumlar nesne ve özellik sayısına göre değişiklik göstermemektedir. Ortaya çıkan latis ve gösterimi ne kadar farklı, büyük, karmaşık vb. olsa da olması gereken temel durumlar listelendiği gibidir. Bu noktadan sonra ilk örnekleme grubunda olduğu gibi üç ve daha fazla anlam için ortaya çıkabilecek genel durumlar tek tek ele alınmayacaktır.

Verilen örneklemelerin kalanı için uygulamalar konusunda sonraki bölümde bilgi verilmektedir. Sonuçlar ve yorumlar açısından bu noktaya kadar bahsedilenlerden farklı bir durum ortaya çıkmamıştır.

5.8. Uygulamanın Devamı

Şu ana kadar sadece örneklemelerden Grup-I için, anlamı belirginleştirilecek kelime açısından ve sadece Elba ile elde edilen görselleştirmeler verilmiştir. Bu bölümde sırasıyla Grup-I için "önceki kelime" özellikleri için yapılan inceleme verilecek ve değerlendirme yapılacaktır.

5.8.1 Grup-I için önceki kelimeden faydalanılması durumu

Grup – I :

a-) Suyun yüzü pislik içindeydi.

ANLAM : Yüzey

Tablo 5.16. "suyun" kelimesi için özellik vektörü

	SözR	**GramR**	**Hal**	**TemR**	**KTB**
ÖK – suyun	İsim öbeği	Özne	Sahiplik	Patient	İsim

b-) Yorganın yüzü pislik içindeydi.

ANLAM : Dış kaplama

Tablo 5.17. “yorganın” kelimesi için özellik vektörü

	SözR	**GramR**	**Hal**	**TemR**	**KTB**
ÖK – yorganın	İsim öbeği	Özne	Sahiplik	Patient	İsim

c-) Çocuğun yüzü pislik içindeydi.

ANLAM : İnsan organı

Tablo 5.18. “çocuğun” kelimesi için özellik vektörü

	SözR	**GramR**	**Hal**	**TemR**	**KTB**
ÖK – çocuğun	İsim öbeği	Özne	Sahiplik	Patient	İsim

Örneklemeler için elde edilen özellik vektörleri birbirinin aynı olduğu için ortaya çıkacak olan latis gösterimi Şekil 5.1 ile özdeştir. Aynı durum Gup-II ve Grup-III için de geçerlidir. Her ne kadar özellik vektöründe hal bilgisi olarak –i hali ya da yalın hali gelmiş olması gibi küçük farklar oluşsa da belirtildiği gibi yapısal olarak latisler özdeştir. Dolayısıyla kullanılması planlanan özellikler ile KAB işleminin başarıya ulaşması mümkün değildir.

5.9 Filtrenin Kullanımı

Bu bölümde uygulamalı olarak işleyişi anlatılmaya çalışılan filtre denetimli makine öğrenmesi bölümünde anlatıldığı gibi öğrenme işleminin ilk adımından hemen sonra eldeki özelliklerin amaçlanan ayırdetme işlemi için yeterli olup olmadıklarını değerlendirmede kullanılacaktır. Değerlendirme, özellikler ve nesneler arasındaki ilişkilerin latis teorisine göre görselleştirmesi sonucu elde edilen görseller aracılığı ile yapılacaktır. Elde edilen görsellerde dikkat edilmesi gereken en önemli ayrıntı nesnelerin her birinin ayrı düğümlerde yer alması gerektiğidir. Eğer birden fazla nesne aynı düğümde yer alıyorsa bir sonraki adıma geçmemek gereklidir. Çünkü bu nesneleri (ya da verilen KAB uygulaması için kelime anlamlarını) kullanılan özellik kümesi aracılığı ile ayırdetmek mümkün olmayacaktır.

6. SONUÇLAR VE GELECEĞE YÖNELİK ÇALIŞMALAR

6.1 Giriş

Geliştirilen uygulama ve filtre, Biçimsel Kavram Analizinden (BKA) faydalanarak, makine öğrenmesi alanındaki önemli problemlerden biri olan özellik seçimi ve değerlendirilmesi konusunda yararlı sonuçlar elde edilmesini sağlamıştır. Kullanılacak olan özelliklerin seçimi için, bir takım istatistiki yöntemlere başvurarak özellik vektöründe küçültmelere gidilmesi gibi ve daha başka yaklaşımlar mevcuttur. Ancak BKA'nın latis tabanlı yapısı sayesinde matematiksel bir kesinlikle ve görsel şekilde elde edilen sonuçlar hem veri görselleştirmesinin avantajlarından faydalanarak bilgi keşfine olanak sağlamakta hem de karmaşık yapıları daha anlaşılır hale getirmektedir. Bilindiği gibi bazı öğrenme algoritmaları kendi içlerinde, kullandıkları özellikleri değerlendirmekte ve faydasız olduğunu düşündükleri özellikleri elemektedirler. Fakat verilen filtre, daha önce de belirtildiği üzere, tembel öğrenici sınıfındaki algoritmalarda olduğu gibi ilgisiz özelliklerden etkilenen ve bunları eleme yeteneğine sahip olmayan algoritmalar ile birlikte kullanıldığında performansa olumlu katkısı olacaktır. Bölümün geri kalanında uygulama ve filtre aracılığı ile elde edilen sonuçların kısa bir değerlendirmesinin ardından mevcut halin daha da iyileştirilebilmesi için yapılabilecekler değerlendirilecektir.

6.2 Elde Edilen Sonuçların Değerlendirilmesi

Uygulama esnasında BKA aracılığı ile elde edilen latis tabanlı görsellerin değerlendirilmesi sonucu önceki bölümde de verildiği gibi

- En üstteki düğümde yer alan ortak özelliklerin zorunlu olmamakla birlikte (istendiği takdirde) özellik vektöründen çıkarılarak vektörün küçülmesinin sağlanabileceği,
- Hiçbir örneklemede rastlanılmayan, en alt düğümde kalan özellikler kullanılmayarak özellik vektörünün küçültülebileceği,
- Aynı düğüm altında birden fazla nesnenin ya da KAB için kullanılan anlamların olması uygulamada bir problem olduğunu işaret eder. Çünkü üzerinde çalışılan kelimenin anlamları arasında hiyerarşik bir yapı ya da kapsama ilişkisi yoksa (anlamlar ayrık ise) böylesi bir durum ortaya çıkmamalıdır. Her nesne (yani anlam) ayrı bir düğümde bulunmalıdır. Aksi takdirde;
    - Özellik işaretlemesi ve buna bağlı olarak vektör oluşturma işleminde hata yapılmış olabilir. Öğrenme verisinin kontrol edilmesi, hata bulunması durumunda düzeltme yapılması ve öğrenme işleminin tekrarlanması gerekeceği,
    - Eldeki tüm özellikler kullanıldığı halde ayırdetme işlemi yapılamıyorsa öğrenme verisinde problem olabileceği,
    - Özellik işaretlemesi konusunda hata yapılmamışsa ve öğrenme verisinin yapısında problem yoksa kullanılan özellikler aynı düğümde gösterilen anlamları ayırdetmek için yeterli değildir. Yeni özellikler eklenmeli ve öğrenme işlemi tekrarlanarak durumun yeniden değerlendirilmesi gerekeceği,

şeklinde çıkarımlarda bulunulup gereken işlemler yapılabilir.

Elde edilenler ve tespitler göz önüne alındığında BKA'nın makine öğrenmesi konusunda özelliklerin değerlendirilmesi amacıyla kullanımının faydalı olduğu sonucuna varılabilir. Bir filtre olarak ilgisiz özelliklerden etkilenen bir öğrenme algoritması ile entegre edildiğinde, daha test verisi üzerinde işlem yapılmadan işlemin başarılı olup olamayacağı konusunda fikir sahibi olunabilir. Tezde uygulama alanı olarak seçtiğimiz Kelime Anlamı Belirginleştirme söz konusu olduğunda da kelimelerin anlamlarının doğru şekilde sınıflandırılması ya da sınıflandırma doğru şekilde yapılamayacaksa önceden bunun görülmesi sağlanabilir.

Bildiğimiz kadarı ile BKA'nın KAB alanında böyle bir uygulaması yoktur. Dolayısıyla tezin en büyük katkısı, yapılan KAB uygulamalarında yoğun şekilde kullanılan denetimli makine öğrenmesi tekniklerinin bu filtre ile entegre edilerek kullanılmasının işlem zamanı ve performans açısından faydalı sonuçlar doğuracağını göstermesidir.

6.3 Gelecekte Yapılabilecek Çalışmalar

Önerilen filtre için yapılabilecek geliştirme ve iyileştirmeler aşağıda sıralanmıştır. Yapılabilecek bu çalışmalar, sonuçların daha hızlı ve kolay şekilde alınmasını sağlayacak, daha fazla özellik içeren, daha karmaşık durumlarda kullanımı kolaylaştıracak ve böylece işlevselliği arttıracak yöndedir.

Uygulamada özellik seçimi ve öğrenme verisi üzerinde gerçekleştirilen öğrenme işlemi sonrasında elde edilen özellik vektörleri BKA yazılımlarına elle girilerek görselleştirme yapılmaktadır. Bu esnada da BKA'nın çok değerli (multi-valued) değişkenlerle çalışamaması sebebi ile özellikler iki değerli hale getirilmektedir. Bu çevrim bir uzman tarafından yapılmaktadır ve kullandığımız yöntem biçimsel bağlamdaki özellik sayılarını arttırarak bu işlemi yapmaktadır. Özellik sayısının artması hem bağlamın yazılıma girilmesi sırasında fazla zaman harcanmasına hem de daha fazla hata yapılmasına sebep olabilecek bir durumdur. Dolayısıyla, öğrenme sonrası elde edilen özelliklerin çok değerli halden iki değerli hale çevrimi ve çevrim sonrasında görselleştirme yazılımının anlayacağı bir formata getirilerek görselleştirmenin yapılması şeklinde iki ayrı iş bulunmaktadır. Bu işler başarı ile yapıldığı takdirde görselleştirme sırasındaki insan müdahalesi ortadan kaldırılarak hem daha hızlı hem de daha hatasız işlem yapılması sağlanabilir. Bu konuda BKA yazılımlarının anlatıldığı bölümde bahsedilen ve ölçeklendirme yapabilen yazılımlardan faydalanılabilir. Özellikle görselleştirme yazılımının anlayacağı bir formata dönüşüm konusunda farklı yaklaşımlar benimsemek mümkündür. Veritabanı kullanımı, BKA ile ilgili Burmeister

vb. farklı dosya formatlarının incelenerek uygun bir çevirici yazılımın kullanılması mümkün olabilir.

BKA yazılımlarının incelendiği bölümde de belirtildiği gibi görselleştirme yazılımlarının konsoldan parametre ile çalıştırılamamaları bir dezavantajdır. Ancak açık kaynak kodlu yazılımların değiştirilmesi ile konsoldan parametre ile çalışma ve kullanıcı müdahalesi olmadan bazı işlemlerin yapılması sağlanabilir. Bu özelliklerin eklenmesinden sonra, öğrenme işlemi ile elde edilen özellik vektörlerinin geliştirilen bir program tarafından görselleştirme yazılımlarının tanıdığı (CXT gibi) bir formatta kaydedilmesi ve kaydedilen bilginin latis haline getirilmesi ile sürecin büyük bir bölümü otomatize edilebilir.

Uygulamadaki çıkarımlar, görsellere bakılarak yani bir uzman tarafından yapılmaktadır. BKA tabanlı latis gösterimlerinden kavram çıkarımı için ConExp (Concept Explorer) benzeri bir yazılım aracılığı ile insan müdahalesi ortadan kaldırılabilir. Mevcut durumda bu yapılmamıştır. Adı geçen yazılım vb. diğer yazılımlar incelenerek bu durumun ne derece mümkün olduğu ve başarımı değerlendirilebilir.

Uygulamada kullanılan örneklerin ve özelliklerin sayısı bilinçli olarak sınırlı tutulmuştur. Bunun sebebi, yeri geldikçe de açıklandığı gibi kontrol edilemeyen değişken ve durumların çalışmayı engellemesine mani olmaktır. Ancak makine öğrenmesi uygulamalarında kullanılan özellik sayısı son yıllarda oldukça artmıştır. Dolayısıyla daha önce anlatılan iyileştirmeler yapılarak özellik sayısı uygulamalardaki yüksek rakamlara çıkarıldıktan sonra bu hali ile başarımın test edilmesi yerinde olacaktır.

Son olarak, geliştirilen filtrenin başarımının kullanılan diğer yöntemlerle kıyaslanması amacıyla bir kıyaslama bulunmamaktadır. Daha önce yapılmış olan bir uygulamanın önerilen filtreden de faydalanılarak tekrarlanması ve sonuçların kıyaslanması, sağladığı faydaların daha net görülmesini sağlayacaktır.

REFERANSLAR

Agirre E., Edmonds P. (2006), Introduction in Agirre and Edmonds (eds.) Word Sense Disambiguation: Algorithms and Applications, 1-28, Springer.

Agirre E., Stevenson M. (2006), Knowledge Sources for WSD in Agirre and Edmonds (eds.) Word Sense Disambiguation: Algorithms and Applications, 217-251, Springer.

Akın A. A., Akın M. D. (2007), Zemberek, an open source NLP framework for Turkic Languages.

Almuallim H., Dietterich T. G. (1991), Learning with many irrelevant features, Proceedings of the 9th National Conference on Artificial Intelligence, AAAI-91, pp:547-552, Araheim, CA, AAAI Press.

Alpaydın E. (2004), Introduction to Machine Learning, The MIT Press.

Andrews S., (2009), Data Conversion and Interoperability for FCA, In CS-TIW 2009, pp. 42-49, http://www.kde.cs.uni-kassel.de/ws/cs-tiw2009/proceedings_final_15July.pdf

Atalay N. B., Oflazer K., SAY B. (2003), The Annotation Process in the Turkish TreeBank, Proceedings of 11th Conference of the EACL-4th Linguistically Interpreted Corpora Workshop- LINC, Hungary.

Aydın Ö., Tüysüz M. A. A., Kılıçaslan Y. (2007), Türkçe İçin Bir Kelime Anlamı Belirginleştirme Uygulaması, Elektrik,ElektronİK,Bilgisayar, Biyomedikal Mühendisliği 12. Ulusal Kongresi, Eskişehir Osmangazi Üniversitesi, Eskişehir.

Becker P. (2004), ToscanaJ User Manual, http://www.wormuth.info/ICFCA04/ToscanaJ_User_Manual.pdf

Becker P., Correira J. H., (2005), The ToscanaJ suite for implementing conceptual information systems, Ganter et al. (Eds.) : Formal Concept Analysis, LNAI 3626, pp. 324-348, Springer-Verlag Berlin Heidelberg

Brown, P., S.A. Pietra, V.J.D. Pietra, and R. Mercer (1991), Word Sense Disambiguation Using Statistical Methods, In Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics, pp 264-270.

Burmeister P. (2003), Formal Concept Analysis with ConImp: Introduction to the Basic Features.

Caruana R., Freitag D. (1994), Greedy Attribute Selection, Proceedings of the Eleventh International Conference, 28 – 36, Morgan Kaufmann Publishers, New Brunswick, New Jersey.

Chen J. (2007), Towards high-performance word sense disambiguation : Combining rich linguistic knowledge and machine learning, VDM Verlag Dr. Müller.

ConExp - Concept Explorer, http://sourceforge.net/projects/conexp/, http://conexp.sourceforge.net/index.html

Cowie J., Guthrie J., Guthrie L.(1992), Lexical disambiguation using simulated annealing, In Proceedings of the International Conference on Computational Linguistics, pp 359-365.

Domingos P. (1996), Unifying Instance-Based and Rule-Based Induction, Machine Learning.

Edmonds P., 2005, Lexical Disambiguation, The Elsevier Encyclopedia of Language and Linguistics, 2nd Ed., ed. by Keith Brown, 607-23, Oxford: Elsevier

FCA Examples, http://www.upriss.org.uk/fca/examples.html

FCA Home Page, http://www.upriss.org.uk/fca/fca.html

FcaBedrock - A Formal Context Creator, http://sourceforge.net/projects/fcabedrock/files/

FcaStone - FCA File Format Conversion and Interoperability Software, http://fcastone.sourceforge.net/

Gale W., Church K. W., Yarowsky D. (1992), Estimating Upper and Lower Bounds on the performance of Word-Sense Disambiguation Programs, Proceedings of the 30h Annual Meeting of The Association for Computational Linguistics, 249-256.

Galicia - Galois Lattice Interactive Constructor, http://www.iro.umontreal.ca/~galicia/, http://sourceforge.net/projects/galicia/files/

Ganter B., Wille R. (1989). Conceptual scaling. In F. Roberts (Ed.), Applications of combinatorics and graph theory to the biological and social sciences. Berlin: Springer, 139-167

Ganter B., Wille R. (1997), Applied Lattice Theory : Formal Concept Analysis, In General Lattice Theory, G. Grätzer editor, Birkhäuser.

Ganter B., Wille R. (1998), Formal Concept Analysis Mathematical Foundations, Springer-Verlag.

Ide N. ,Veronis J. (1998), Word sense disambiguation: The state of the art, Computational Linguistics, 24:1, 1-40.

In-Close - fast Formal Concept miner, http://sourceforge.net/projects/inclose/

Japkowicz N. (2000), The class imbalance problem : Significance and Startegies, In Proceedings of the 2000 International Conference on Artificial Intelligence (ICAI).

John G. H, Kohavi R., Pfleger K. (1994), Irrelevant Features and the Subset Selection Problem, Machine Learning : Proceedings of the Eleventh International Conference, 121 – 129, Morgan Kaufmann Publishers, San Francisco, CA.

Kılıçaslan Y., Güner E. S., Yıldırım S. (2009), Learning-based Pronoun Resolution for Turkish with a Comparative Evolution, Computer Speech and Language

Kira K., Rendell L. A. (1992), The Feature Selection Problem : Traditional Methods and a New Algorithm, AAAI-91 Proceedings

Koller D., Sahami M. (1996), Toward Optimal Feature Selection, International Conference on ML, pp : 284 – 292.

Kononenko I. (1994), Esimating Attributes : Analysis and Extensions of RELIEF, European Conference on ML, pp : 171 – 182.

Kotsiantis S. B. (2007), Supervised Machine Learning: A review of classification techniques, Informatica.

Lindig C. (2000), Fast Concept Analysis, Working with Conceptual Structures - Contributions to ICCS 2000, Shaker Verlag

Lindig C., Sneltig G. (1997). Assessing Modular Structure of Legacy Code Based on Mathematical Concept Analysis. Proceedings of the 19th international conference on Software engineering, Boston, MA, USA, 349-359.

Löbner S. (2002), Understanding semantics, Arnold Publishers

Marques L. , Escudero G. , Martinez D. , Rigaue G.. (2006), Supervised Copus-Based Methods in Agirre and Edmonds (eds.) Word Sense Disambiguation: Algorithms and Applications, 167-216, Springer.

Mihalcea R. (2002), Instance Based Learning with Automatic Feature Selection Applied to Word Sense Disambiguation, International Conference on Computational Linguistics, Taiwan.

Mihalcea R. (2006), Knowledge-Based Methods in Agirre and Edmonds (eds.) Word Sense Disambiguation: Algorithms and Applications, 107-131, Springer.

Mitchell T. (1997), Machine Learning, McGraw Hill.

Mladenic D. (1998), Feature Subset Selection in text-learning, European Conference on Machine Learning (ECML), pp : 95 – 100.

Ng H. T., Lee H. B., (1996), Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach. In Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics, pages 40-47.

Oflazer Kemal, Bilge Say, Dilek Hakkani-Tür, Tür G.., (2003)Design for a Turkish Treebank. In A. Abeille (ed). Building and Exploiting Parsed Corpora. Kluwer Academic Publishers.

Oflazer, K., Say, B., Tur, D. Z. H., Tur, G.. (2002), "Building A Turkish Treebank, Invited Chapter In Building and Exploiting Syntactically-Annotated Corpora", Anne Abeille Editor, Kluwer Academic Publishers.

Old L. J. (1996), Homograph Disambiguation Using Formal Concept Analysis, In: Fourth International Conference on Formal Concept Analysis,, 13th-17th February 2006, Dresden, Germany

Palmer M., Ng H. T., Dang H. T. (2006), Evaluation of WSD Systems in Agirre and Edmonds (eds.) Word Sense Disambiguation: Algorithms and Applications, 75-106, Springer.

Pedersen T. (2006), Unsupervised Copus-Based Methods in Agirre and Edmonds (eds.) Word Sense Disambiguation: Algorithms and Applications, 133-166, Springer.

Prati, R.C., Batista, G.E., Monard, M.C. (2004): Class imbalance versus class overlapping: an analysis of a learning system behavior. In: Proc. 3rd Mexican Intl. Conf. on Artificial Intelligence, Mexico City, Mexico, pp. 312-321.

Priss U. (1996), Formal Concept Analysis in Information Science, Annual Review of Information Science and Technology (ARIST).

Priss U. (2004), Linguistic Applications of Formal Concept Analysis, in Proceedings of the First International Conference on Formal Concept Analysis (ICFCA 2003), Lecture Notes in Artificial Intelligence, Springer

Priss U., (2008a), FcaStone - FCA File Format Conversion and Interoperability Software, In: Croitoru M. et al. (Eds.), In CS-TIW 2008, pp. 33-43

Priss U., (2008b), FCA Software Interoperability,
http://www.upriss.org.uk/papers/cla08.pdf
Rancz K. T. J. et al., (2008), A software tool for data analysis based on Formal Concept Analysis, Studia Univ. Babeş-Bolyai, Informatica, Volume LIII, Number 2

Say B., Özge U., Oflazer K. (2002), Bilgisayar Ortamında Bir Derlem Geliştirme Çalışması,Akademik Bilisim Konferansi, Konya, Turkiye.

Say, B., Zeyrek, D., Oflazer K., Özge, U. (2002), "Development of a Corpus and a Treebank for Presentday Written Turkish", in Proceedings of the Eleventh International Conference of Turkish Linguistics.

Stevenson M. (2003), Word Sense Disambiguation : The case for combinations of knowledge sources, CSLI Publications.

TDK Türkçe Sözlük, http://www.tdk.gov.tr

Tilley T., (2004), Tool support for FCA, In Eklund (Ed.), Concept Lattices: Second International Conference on Formal Concept Analysis, Springer-Verlag, LNCS 2961, p. 104-111

Vafaie H., Imam I. F. (1994), Feature Selection Methods : Genetic Algorithms vs. Greedy-like Search, Proceedings of the International Conference on Fuzzy and Intelligent Control.

Wagner H. (1973) Begriff. In: Handbuch philosophischer Grundbegriffe. 191-209. München, Kösel.

Wille R. (1982) Restructuring lattice theory: an approach based on hierarchies of concepts. In: Rival, I. (ed.) Ordered Sets.445-470. Dordrecht-Boston, Reidel.

Wolff K. E.(1993), A First Course in Formal Concept Analysis, Faulbaum, F. (ed.) Proceedings SoftStat'93, Advances in Statistical Software 4, 429 – 438, Gustav Fischer Verlag

Wormuth B. (2004), Elba User Manual, http://www.wormuth.info/ICFCA04/Elba_User_Manual.pdf

Wormuth B., Becker P. (2004), Introduction to Formal Concept Analysis, 2nd International Conference of Formal Concept Analysis, February 23 – February 27, Sydney – Australia

Yang Y., Pedersen J. O. (1997), A Comparative Study on Feature Selection in Text Categorization, In ICML 97, Proceedings of the Fourteenth International Conference on ML, pp:412 – 420.

Yarowsky, D. (1995), Unsupervised Word Sense Disambiguation Rivaling Supervised Methods. In Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics. Cambridge, MA, pp. 189-196.

Yu L. ve Liu H. (2004), Efficient Feature Selection via Analysis of Relevance and Redundancy, Journal of Machine Learning Research.

Zemberek Projesi ana sayfası, http://code.google.com/p/zemberek/

Zemberek Projesi belgeleri, http://code.google.com/p/zemberek/wiki/CesitliDokumanlar

EKLER

Ek – A : İngilizce Kelimeler İçin Kullanılan Türkçe Karşılıklar

Annotation : İşaretleme

Artificial Intelligence : Yapay zeka

Binary : İkili

Classifier : Sınıflayıcı

Collocation : Eşdizimlilik

Computational : Hesaplamalı

Conceptual system editor : Kavramsal sistem editörü

Conceptual system schema : Kavramsal sistem şeması

Connectionist : Bağlantısal

Corpus : Derlem

Cross-validation : Çapraz doğrulama

Discourse : Konuşma/söylev

Embedded : Gömülü

Enumerative : Sayımlamalı

Generative : Üretici

Greedy : Aç gözlü

Heuristic : Sezgisel

Homograph : Eşyazımlı

Homonymy : Eşsesslilik

Hypertext : Hipermetin

Information Retrieval : Bilgi çekme

Irrelevant : İlgisiz

Knowledge-based : Bilgi tabanlı

Lazy learners : Tembel öğreniciler

Leave-one-out validation : Birini dışarıda bırak doğrulama

Lexeme : (Teknik manada) sözcük/kelime

Lexicon : Teknik manada sözlük

Machine learning : Makine öğrenmesi

Mapping : Eşleşme

Nested line diagrams : İçiçe çizgi diyagramları

Parser : Ayrıştırıcı

Part-of-speech tagger : Kelime türü işaretleyici

Part-of-speech tagging : Kelime türü işaretleme

Polysemy : Çokanlamlılık

Selectional Preferences : Seçimsel tercihler

Semantic Network : Anlambilimsel ağ

Semantics : Anlambilim

Sense : Anlam

Sense repository : Anlam deposu

Speech : Ses/konuşma

Subcategorization : Alt öğeleme

Supervised : Denetimli/Gözetimli

Supervisor : Uzman

Syntax : Sözdizim

Ten-fold cross validation : On katlamalı çapraz doğrulama

Thesauri : Eşanlamlılar sözlüğü

Training data : Öğrenme verisi

Unsupervised : Denetimsiz

Wrapper : Sarma

Ek – B : TDK'dan "yüz" Kelimesi İçin Alınan Açıklamalar

yüz (I)
isim

1 . Doksan dokuzdan sonra gelen sayının adı.
2 . Bu sayıyı gösteren 100 ve C rakamlarının adı.
3 . *sıfat* On kere on, doksan dokuzdan bir artık.
4 . Kere, kat vb. kelimeler ile birlikte kullanılarak yapılan işin çokluğunu abartılı bir biçimde anlatan söz: *"Hikmet Bey'in kurum ve edası, her zamankinden belki yüz kat üstündü."*- S. M. Alus.

Birleşik Sözler

yüzbaşı	yüzbeşlik	yüz binlerce	yüz binlik
yüz kere	yüznumara	yüz para	yüzyıl
yüzde yüz			

yüz (II)
isim

1 . Başta, alın, göz, burun, ağız, yanak ve çenenin bulunduğu ön bölüm, sima, çehre, surat: *"Bir güzel çocuk yüzüyle gülümsüyor."*- S. F. Abasıyanık.
2 . Yüzey: *"Suyun yüzünde."*- .
3 . Kesici araçlarda ağız: *"Bıçağın keskin yüzü."*- .
4 . Bir kumaşın dikiş sırasında dışa getirilen gösterişli bölümü.
5 . Yorgana ve yastığa geçirilen kılıf.
6 . Bir şeyin görünen bölümünde kullanılan kumaş: *"Yorgan yüzü. Kanepenin yüzü."*- .
7 . Birinin görülegelen veya umulan hoşgörürlüğüne güvenilerek gösterilen cüret: *"Ne yüzle? Yüzü olmamak."*- .
8 . Nedeniyle, sebebiyle: *"Bu yüzden Fuat Köprülü ile çatışmaya başlamışlardı gazetelerde."*- Y. Z. Ortaç.
9 . Yan, taraf.
10 . Bir yapının dışa bakan düşey yüzeylerinin her biri:

"Ön yüz. Yan yüz. Arka yüz."- .
11 . *mecaz* Utanma: *"Adamda yüz yok ki!"*- .

Atasözü, deyim ve birleşik fiiller

yüz bulmak	yüz bulunca astar istemek	yüz çevirmek	yüze çıkmak
yüze duramamak	yüze gelmek	yüze gülmek	
yüz etmek	yüze vurmak	yüz geri etmek	
yüz göstermek	yüz kızartmak	yüz kızdırmak	
yüz surat davul derisi (veya mahkeme duvarı)	yüz sürmek	yüz takınmak	
(bir şeye) yüz tutmak	yüz tutmak	(bir şeyin) yüzü açılmak	
yüzü asılmak	yüzü düşmek	(bir şey) yüzü görmemek	
yüzü gözü açılmak	yüzü gülmek	(birinin) yüzü kâğıt gibi olmak	
yüzü kalmamak	yüzü karışmak (veya allak bullak olmak veya alabora olmak)	yüzü kasap süngeriyle silinmiş	
yüzü kızarmak	yüzü kireç gibi olmak (veya ağarmak)	yüzü kireç kesilmek	
yüzünden akmak	yüzünden düşen bin parça olmak	yüzünden kan damlamak	
yüzünden okumak	yüzüne bağırmak	yüzüne bakamaz olmak	
yüzüne bakılacak gibi olmak	yüzüne bakılır olmak	yüzüne bakılmaz olmak	
yüzüne bakmamak	yüzüne bakmaya kıyamamak	yüzüne bir daha bakmamak	
yüzüne duramamak	yüzüne gözüne bulaştırmak	yüzüne gülmek	
yüzüne hasret kalmak	yüzüne kan gelmek	yüzüne karşı	
yüzüne su çarpmak	yüzüne tükürseler yağmur yağıyor sanır	yüzüne vurmak (veya çarpmak)	
yüzüne yazmak	(birinin) yüzünü ağartmak	yüzünü buruşturmak (veya ekşitmek)	

yüzünü duvara yapıştırmak	yüzünü gören cennetlik	yüzünü görmemek
(birinin) yüzünü gözünü açmak	yüzünü güldürmek	yüzünü kara çıkarmak
yüzünü karartmak	(birinin) yüzünü kızartmak	yüzünü kızartmak (veya kızdırmak)
yüzünün derisi kalın	yüzünün derisi yere geçmek	yüzünü şeytan görsün
(birinin veya bir şeyin) yüzünü unutmak	yüzünü yere getirmek (veya geçirmek)	yüzünüze güller
yüzü olmamak	yüzü sararmak	yüzü seçilmemek
yüzü sıcak olmak	yüzü soğuk olmak	yüzü suyu hürmetine
yüzü suyuna	(bir şeye) yüzü tutmamak	yüzü yazılı kalmak
yüzü yere gelmek (veya geçmek)	yüz verince astar istemek	yüz vermemek
yüz yapmak	yüz yazmak	yüz yüzden utanır

Birleşik Sözler

yüz akı	yüzbeyüz	yüz görümlüğü	yüz göz
yüz havlusu	yüz kalıbı	yüz kaplama	yüz karası
yüz kızartıcı suç	yüz kiri	yüz ölçümü	yüz sabunu
yüzsuyu	yüzüstü	yüz yazısı	yüz yüze
yüze gülücü	yüze soğurma	yüzü ak	yüzü asık
yüzü kara	yüzükoyun	yüzü pek	yüzü yerde
yüzü yumuşak	arayüz	arka yüz	çatık yüz
dış yüz	eğri yüz	ekşi yüz	güler yüz
içyüz	iç yüz	kara yüz	paralel yüz
ters yüz	o yüzden	şu yüzden	gökyüzü
ters yüzü	yeryüzü	yorgan yüzü	eli yüzü düzgün
eli yüzü temiz			

ÖZGEÇMİŞ

Mehmet Ali Aksoy TÜYSÜZ 1979 yılında İstanbul'da doğdu. İlk ve orta öğrenimini İstanbul'da tamamladıktan sonra 1996 yılında girdiği Trakya Üniversitesi Mühendislik – Mimarlık Fakültesi Bilgisayar Mühendisliği Bölümü'nden 2000 yılında mezun oldu. Aynı yıl Trakya Üniversitesi Fen Bilimleri Enstitüsü'nde Yüksek Lisansa, İstanbul Bilgi Üniversitesi'nde Araştırma Görevliliğine başladı. Yüksek lisansını 2004 yılında tamamlayan Mehmet Ali Aksoy TÜYSÜZ aynı yıl doktoraya başlamıştır ve Nisan 2006'dan beri TÜBİTAK – UEKAE'de çalışmaktadır.