Bir Yabancı Dilde Bilgi Çıkarımı
Sonuçlarının Semantik
Temsilciliğinin Sağlanması
Ve Bir Uygulama

Ahmed A.TAEB
Master of Science Thesis

Computer Engineering Program
June, 2010

JÜRİ VE ENSTİTÜ ONAYI

Ahmed A. TAEB'in "**Bir Yabancı Dilde Bilgi Çıkarımı Sonuçlarının Semantik Temsilciliğinin Sağlanması Ve Bir Uygulama**" başlıklı **Bilgisayar Mühendisliği** Anabilim Dalındaki, Yüksek Lisans Tezi 18.06.2010 tarihinde, aşağıdaki jüri tarafından Anadolu Üniversitesi Lisansüstü Eğitim-Öğretim Yönetmeliğinin ilgili maddeleri uyarınca değerlendirilerek kabul edilmiştir.

	Adı Soyadı	**İmza**
Üye (Tez Danışmanı)	**: Prof. Dr. Yaşar HOŞCAN**	**.....**
Üye	**: Doç. Dr. C. Hakan KAĞNICIOĞLU**	**.....**
Üye	**: Yard. Doç. Dr. Cüneyt AKINLAR**	**…….....**

Anadolu Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ..... tarih ve ..... sayılı kararıyla onaylanmıştır.

Enstitü Müdürü

ABSTRACT

Master of Science Thesis

Foreign Language in the Information Inference Ensuring Results And An Application of Semantic Representation
Ahmed A. TAEB

Anadolu University
Graduate School of Sciences
Computer Engineering Program

Supervisor: Prof. Dr. Yaşar HOŞCAN
2010, 66 pages

In recent years, fast development and widespread use of the Internet with Web, accessible around the world has become the largest source of data. Stacks of information over the Internet as increased in accordance with Web visitors request the services can be provided, the improvement of Web site structure, development, and provide information of interest Inference increasingly seen as an issue.

The largest source of information as seen in the internet access information in front of the biggest obstacles in the knowledge that 90% of natural language is created, the natural language of computers yet adequately be unable to understand due to their page find the information out to people still are falling.

This study of natural language is used to obtain information from the method of information extraction is being developed based on. In this context, natural language as components of the adjective, noun, verb, adverb, etc.. using resources to obtain output from a qualitative study is intended category. This study will focus on Arabic language, because Arabic from right to left, very difficult language spoken and biridi. Written in the Java language and natural language Gate in a word processing functions, see the software, using the GATE platform software information, we will separate types

Keywords: Information Extraction, Language Natural Processing, Gate, Jape

ÖZET

Yüksek Lisans Tezi

BİR YABANCI DİLDE BİLGİ ÇIKARIMI SONUÇLARININ SEMANTİK TEMSİLCİLİĞİNİN SAĞLANMASI VE BİR UYGULAMA

Ahmed A. TAEB

Anadolu Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: **Prof. Dr. Yaşar HOŞCAN**

Son yıllarda İnternet'in hızla gelişmesi ve yaygın kullanımı ile Web, dünyada erişilebilir en geniş veri kaynağı haline gelmiştir. Internet'teki bilgi yığınları aşırı şekilde artarken, Web ziyaretçilerinin isteklerine uygun hizmetlerin sağlanabilmesi, Web site yapısının iyileştirilebilmesi, geliştirilebilmesi ve etkin olarak kullanılabilmesi gibi amaçları sağlamak için kullanılan Bilgi Çıkarım Metodu, gün geçtikçe daha çok ilgi çeken bir konu haline gelmiştir.

En büyük bilgi kaynağı olarak görülen İnternet'te, bilgiye ulaşabilmenin önündeki en büyük engel, içindeki bilgilerin %90'ının doğal dille oluşturulmuş olmasıdır. Bilgisayarların doğal dili henüz yeterli bir şekilde anlayamıyor olmalarından dolayı sayfalardaki bilgileri bulup çıkarmak yine insanlara düşmektedir.

Bu çalışma, doğal dillerden bilgiyi elde etmek için kullanılan metotlardan biri olan Bilgi Çıkarımı'na (Information Extraction) dayanarak geliştirilmektedir. Bu kapsamda doğal dillerin bileşenleri olan sıfatlar, isimler, fiiller, zarflar vb. kaynaklar kullanılarak nitel bir sınıflandırma çıktısı elde edilmes amaçlanmaktadır. Bu çalışmada odaklanılan yabancı dil Arapça olarak seçilmiş ve teknoloji olarak Java diliyle geliştirilmiş olan GATE platformu tercih edilmiştir.

Anahtar Kelimeler: Bilgi Çıkarımı, Doğal Dil İşleme (DDİ), GATE, Jape,

TEŞEKKÜR

Yüksek lisans eğitimim boyunca zengin bakış açısıyla beni aydınlatan danışman hocam Sayın Prof. Dr. Yaşar HOŞCAN'a yüksek lisans tez çalışmamda gösterdiği ilgi ve sabrından dolayı teşekkür ederim.

Sayın hocam Yard**.** Doç. Dr. Cüneyt AKINLAR'a da tez dönemimdeki yardımları için ve beni engin bilgi ve tecrübeleri ile aydınlatan ve destekleyen hocama sonsuz teşekkürlerimi sunarım.

Anadolu Üniversitesi Bilgisayar Mühendisliği Anabilim Dalına ve bütün hocalarıma teşekkürlerimi sunarım.

Fikirleri ile beni destekleyen tüm bölüm hocalarıma ve araştırma görevlisi arkadaşlarıma teşekkür ederim.

Son olarak beni maddi ve manevi her konuda destekleyen, sonsuz sevgi ve ilgisini esirgemeyen sevgili annem, eşim, kardeşlerim ve arkadaşlarıma teşekkürlerimi sunarım.

Ahmed A.TAEB

Haziran, 2010

İÇİNDEKİLER

SİMGELER ve KISALTMALAR DİZİNİ

NLP	: Nutural Language Processing
DDİ	: Doğal Dil İşleme
W3C	: World Wide Web Consortium
PDF	: Postscript Document File
XML	: Extensible Markup Language
OWL	: Web Ontology Language
HTML	: Hyper Text Markup Language
RDF	: Resource Description Framework
RDFS	: Resource Description Framework Schema
URI	: Uniform Resource Identifier
FOAF	: Friend Of A Friend
OIL	: Ontology Interface Layer
DAML	: DARPA Agent Markup Language
SPARQL	: Protocol and RDF Query Language
B2B	: Business To Business
UB/SPSC2	: Universal Standard Products and Services
NL	: Nutural Language
SBM	: Syllable-Based Morphology
LBM	: Lexeme-Based Morphology
KDD	: Knowledge Discovery Databases
GATE	: General Architecture for Text Engineering
LR	: Language Resource
PR	: Processing Resource
VR	: Visual Resource
GKA	: Grafiksel Kullanıcı Ara yüzü
ANNIE	: A Nearly-New Information extraction System
POS	: Part Of Speech

JAPE : Java Annotation Patterns Engine

RHS : Right Hand Side

LHS : Left Hand Side

ŞEKİLLER DİZİNİ

ÇİZELGELER DİZİNİ

1. Giriş

Bilgisayar ve İnternet tabanında şekillenen yeni teknoloji, bilgiye daha hızlı ulaşmayı sağlamaktadır. Böylece haber ve etkinlikler birkaç saniye içinde dünyanın dört bir yanına internet aracılığıyla yayılırken, bunlara karşı verilen tepki ve açıklamalar anında yer bulmaktadır. Bilgiye evden veya ofisten dışarı çıkmadan ulaşılabilir. Günümüzde insanoğlu, yeryüzünde bulunan bütün kütüphanelere, önündeki tuşlarla ve bir ekran yardımı ile ulaşabilmektedir. Artık bilgiye ulaşmak için olanaklar ve fırsatlar her gün gelişmekte ve değişmektedir.

Bu hızlı ve sınırsız bilgi ortamı bilginin yer ve zamandan bağımsız paylaşılmasını sağlarken aynı zamanda bilgi kirliliğine de neden olabilmektedir. Bilinçli ya da bilinçsiz olarak sunulan yanlış bilgiler internetten hızla yayılırken doğru bilgileri de gölgelemektedir. Örneğin; bir mail hesabı düşünülürse mail hesabına istenmeyen birçok mesaj gelmektedir ve bu mesajların içeriğini öğrenmek için bütün gelen mesajları tek tek açıp incelemek gerekmektedir. Bu da zaman kaybına yol açmaktadır. Bu sorunu çözmek için yapılacak olan işlem, tanınmayan mail adreslerinden gelen mesajları "Spam" olarak filtrelemektir.

Web'te asıl sorun bilgisayarın metinleri kavrayamamasıdır. Web sayfaları da geçen bilgileri anlamayabilir, Örneğin; Web'te arama motoru aracılığıyla bir meyve türü aranırsa "Elmanın faydaları" gerekli veya gereksiz elmayla ilgili birçok bilgi çıkar ve bu bilgilerden, elma sirkesi, elma şekeri veya elma adıyla herhangi bir şehir, sokak, mahalle, insan adı gibi birçok bilgi çıkabilir, aynı zamanda bu bilgilerin arasında meyve anlamıyla elmanın faydaları bilgisi de vardır. Bulunan bilgilerin çoğu "Elmanın Faydaları" anlamıyla alakasız bilgilerdir bu sonuç ise bir bilgi kirliliğine yol açmaktadır. Bu durumdaki kullanıcı daha az sonuç ve özet bir bilgiye ulaşmak için gelişmiş arama motorlarını kullanır.

Bu bilgi kirliliğinin nedeni, arama motorunun bulunmak istenen sözcüğü anlamsız bir terim olarak algılamasıdır. Bunun nedeni arama motorlarının kelimeleri anlamlarına göre arayamamasıdır. Bu tür sorunlar teknolojinin olumsuz

taraflarıdır ve bu olumsuzluklara karşı elbette aynı kalitede ve etkinlikte çözümler de sürekli olarak geliştirilmektedir. Peki, bu bilgi kirliliğinin nedeni nedir?

1.1. Bilgi Kirliliği (Information Pollution)

Bilgi kirliliği, İnternete sunulan aşırı miktardaki bilgi, bilginin işleyişini ve kullanıcılar tarafından algılanmasını kısmaktadır; çünkü bir bilginin gerçeklik değerini ne kadar düşükse kişiler tarafından önem taşıması da o kadar düşüktür[1].

Dünyanın bilgiye ulaşım ve iletişim alanında yaptığı en büyük devrim internet teknolojisidir[2]. Dünya giderek artan bilgi bombardımanı altında kalmakta ve insanoğlu düşünerek, yorumlayarak ve konuşarak bilgi üretmektedir. Üretilen ve iletilen bilgi miktarı artarken bilgi kirliliği de hızla artmaktadır. E-mail hesapları gereksiz iletilerle dolduğunda, hızla gelişen bir iletişim sitesi olan Facebook'ta çeşitli uygulamalarla gelen fotoğraf ve yazılara, birçok forum sitesinden her konuda yapılan yorumlara bakıldığında, bu bilgi kirliliğine örnek verilebilinir. Örneğin, İnternet sitelerine bakıldığında, tamamen uydurma sayısal verilerle, tamamen aldatıcı ve kaynağı belli olmayan bilgilerle karşılaşılmaktadır. Bütün bu tür bilgilerin kaynağı veya gönderen kişinin gerçek adresi bulunmamaktadır. Ayrıca bu tür mesajlar alındığında mesajın altına not düşülerek, Lütfen! Bu mesajı herkese gönderin (forward) diye özellikle belirtilmiş bir not bulunmaktadır. Böyle durumlarda doğru ve yanlışın ayırt edilmesi zorlaşmakta ve hata yapma olasılığı artmaktadır. Bilginin yoğunluğu, kalitesizliği gibi bilginin yanıltıcı olması da kullanıcının aldığı ve ürettiği doğru bilginin de büyük bir bölümünün kaybolmasına neden olmaktadır. Bazı durumlarda ise bilgi kirliliği, edinilmiş ve kullanmakta olan doğru bilginin güvenilirliğinin sarsılmasına neden olabilmektedir. Bugünlerde bilgi adeta bir nesneye dönüştürülmüş durumdadır. Bilgi alınıp satılan bir eşya muamelesi gibi görülmektedir. Bu da yeni bir soruna yol açmaktadır; aşırı bilgi yüklenmesi…

1.1.1. Bilgi kirliliği sebepleri?

- Hızla artan yeni bilgi üretimi.
- Verinin internet üzerinden çoğaltımının ve iletiminin kolaylıkla sağlanabilmesi.
- Bilginin birçok değişik kanaldan gelmesi (örneğin telefon, e-posta, anlık mesajlaşma, RSS).
- Büyük miktarda depolanan bilgiler.
- Bilgide mevcut olan çelişkiler ve yanlışlıklar.
- Farklı bilgi işleme ve karşılaştırma için yöntem eksikliği.
- Bilgilerin ilişki, bağlantıları veya herhangi bir genel yapı olmaması.[3]

1.1.2. Aşırı bilgi yüklenmesi (Information Overload)

Aşırı Bilgi Yüklenmesi, Alvin Toffler [4] tarafından ortaya çıkarılmış bir terimdir. E-maillerin kontrol edilmesi ile başlayan, arabadaki nevigasyon sisteminden telefonundaki birçok uygulamaya kadar devam eden teknoloji alışkanlığı, gittikçe kontrolünün olmadığı imkansız bir bilgi yüklenmesine sebep olmaktadır. Günümüzde insanlar bitirmesi gereken önemli projeler ve yapılacak birçok önemli işi olmasına rağmen hiç farkında olmadan saatlerce bilgisayar başında blogları okuyarak, friendfeed, facebook ve twitter gibi sosyal sitelerde takılarak zamanlarını boşa harcamaktadır. Bunun nedeni; kullanıcının hızla birçok bilgiye ulaşması ve yoğun bir bilgi bombardımanıyla karşı karşıya kalmasıdır. Söz konusu bilgi ve malûmat (enformasyon) gibi kavramları masaya yatırmaktan ziyade; insanlar gerçekten bilgi bombardımanı karşısında çaresizlerdir. Günümüz teknolojisinin konularından biri de artık, bu ‘bilgi bombardımanı’ karşısında insanların karşılaştığı sorunlardır. ‘Bilgi zehirlenmesi’, ‘Tekno-stres’ ve ‘Bilgi yorgunluğu sendromu’ gibi kavramlarla ortaya bu sorun çıkmaktadır [5].

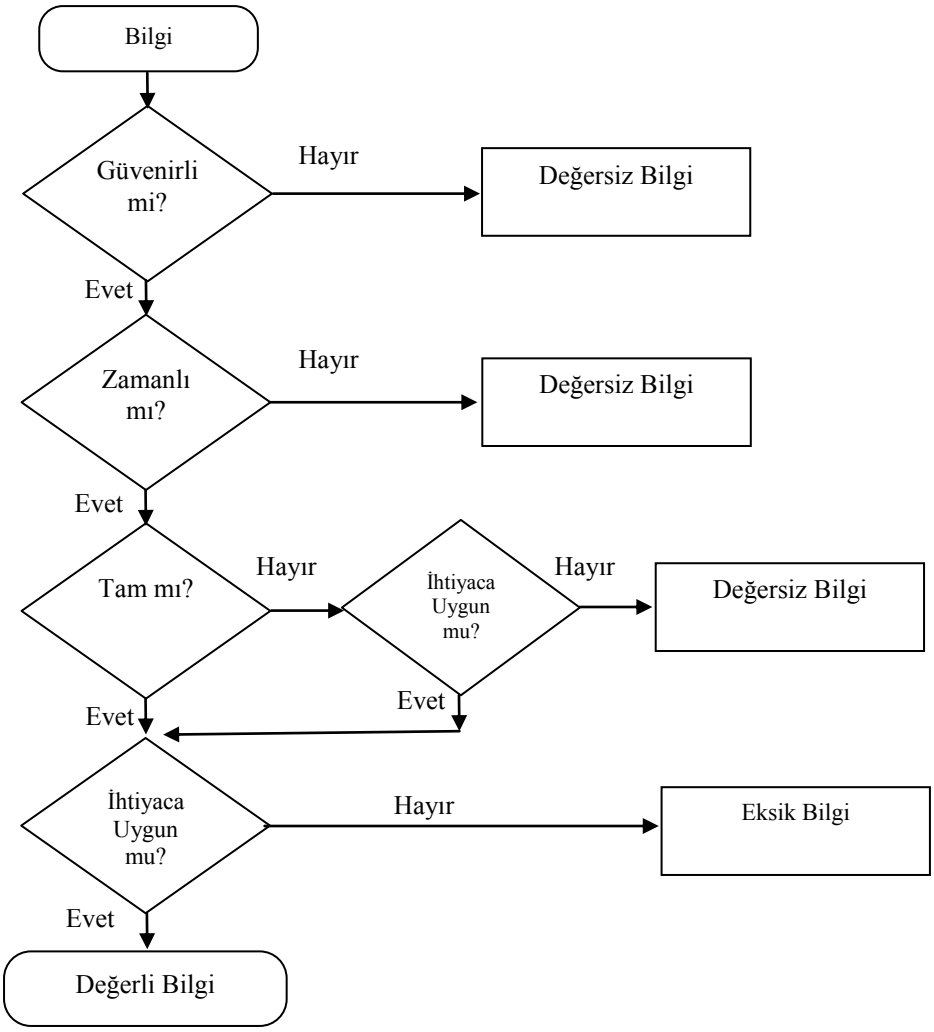
Olay ve bilgilerin, zihinde hızlı akışı, beyinde yarattığı karmaşa arttıkça, insan beyni bunu kaldıramamakta ve insanların bilgiyi analiz gücü kırılıp beyin adeta felce uğramaktadır. Bu durumun nedeni ise, insan beyninin, bilgisayar gibi 'multitask' olmaması, yani aynı anda birden çok işlemi yapacak şekilde tasarlanmamış oluşudur. Buna çağın hastalığı denir ve ismi de "Information Overload" (Aşırı Bilgi Yüklenmesi)'dir.

1.2. Bilginin Kalitesi (Information Quality)

Bilginin temel kullanım amacı, karar alma fonksiyonuna destek olmak olduğuna göre değeri de alınan kararın değerine bağlı olacaktır. Bilginin değerini, bilgiden beklenen sonuçlara göre belirlemek, kuşkusuz ki gerçekçi bir yaklaşımdır. Bilginin değerini belirleyen nitelikler: Zamanındalık, güvenilirlik, yerindelik, eksiksizlik, ekonomiklik, uygunluk vs. olarak sayılmaktadır. Şekil 1.1'de verilen aşamalar, bilgiyi değerlendiren ölçütlerdir.

Bilginin değeri ya da kalitesi konusunda etkili önemli bir faktörde hatadır. Hata zamanında tespit edilmemişse düzeltmek çok güçtür. Bilginin hatalı olmasının nedenleri:

- Yanlış bilgi toplanması
- Bilgi işlemenin yanlış yapılması
- Bilginin işlenmemesi ya da kaybedilmesi
- Yanlış kaynaktan bilgi temin edilmesi
- Bilgi işlem hataları
- Bilginin kasten bozulması



Şekil 1.1 Bilginin Değerini Ölçme Akış Şeması

1.3. Semantik Web (Semantic Web)

Semantik; anlamı araştırmak demektir, Yunanca "Semantikos" yani gerçek anlam kelimesinden gelmektedir [6,7]. Web içeriklerinin sadece doğal dillerde değil, aynı zamanda ilgili yazılımlar tarafından anlaşılabilir, yorumlanabilir ve kullanılabilir bir biçimde ifade edilmesidir. Böylece bu yazılımlar verinin kolayca bulunmasını, paylaşılmasını ve bilginin birleştirilmesini amaçlayan, gelişen bir internet eklentisidir. Semantik web teknolojileri açık standartlar sayesinde anlamlı verinin, doküman içeriğinden ayrıştırılmasını sağlamaktadır. Eğer bilgisayar bir dokümanın semantiğini anlar ise, artık o sadece dokümanın içerisindeki karakterleri algılamaz, dokümanın anlamını da öğrenmektedir [8]. Semantik web'teki temel amaç iyi tanımlanmış ve bağlaştırılmış olan bilgilerin ve

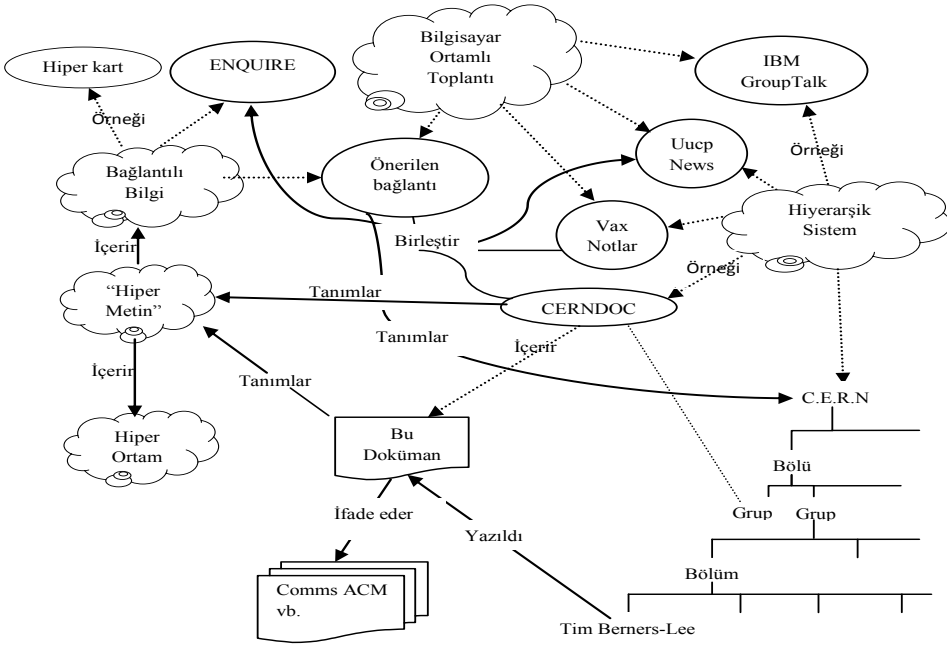
servislerin web ortamında kolay bir şekilde bilgisayarca okunabilir ve anlaşılabilir olmasını sağlayacak standartların ve teknolojilerin geliştirilmesidir.

Web üzerindeki bütün bilgi ve verilerin açıklamalar ile ilişkilendirilmesi gerekmektedir. Semantik Web, dünya üzerindeki bilgileri tek bir platformda toplamayı amaçlayan, ilgili süreçlerin bilgisayarlar tarafından web üzerinden otomatik olarak yönetilmesini sağlayan bir uygulamadır.

Semantik Web verilerin uygulamalar arası, kurumlar arası, topluluklar arası paylaşılabilmesi için bir altyapı sağlamaktadır. Semantik web, verinin bilgisayarlar tarafından anlamlandırıldığı bir veri paylaşımıdır. Semantik Web'ler, bilgiyi birbirine değişik tipte bağlarla bağlanmış bir düğüm şeklinde göstermektedir. Düğümler kavramları, bağlar ise kavramlar arasındaki ilişkileri gösterir Semantik teknolojiler anlamı ontolojiler kullanarak anlatılmaktadır.

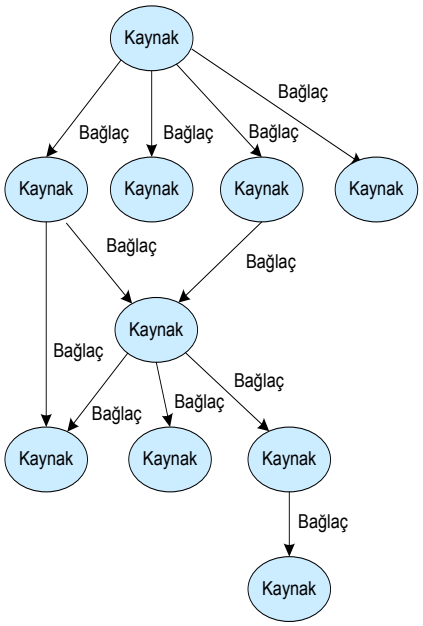
Semantik web yeni ve ayrı bir web olmayıp, bilgilere iyi tanımlanmış anlamların verildiği, bilgisayarların ve insanların birlikte çalışmalarına imkân veren bugünkü web'in bir uzantısıdır [9]. Web ilk olarak 1994' de Tim Berners-Lee [10] tarafından Uluslarası Web Konferansı'nda dünya üzerindeki Web bağlantılarının sürekli boş duran, sadece istendikleri zaman yönlendiren birer araç oldukları varsayımından yola çıkarak ortaya atılmıştır.

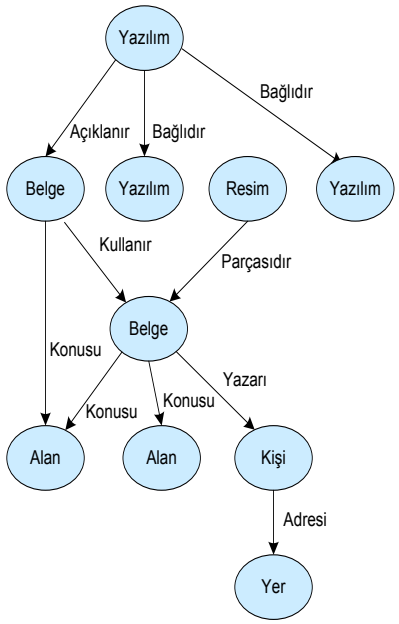
Semantik web kavramı, bugünkü web'in temelini oluşturan URI, HTTP ve HTML gibi yapılarını tasarlayan ve bulan kişi olan Tim Berners-Lee [10,11] tarafından öne sürülmüş ve mevcut web ortamının geliştirilerek potansiyel kullanımı için web'in gelecek adımı olarak düşünülmektedir. Tim Berners-Lee tarafından düşünülen orijinal web modeli Şekil 1.2'de verilmektedir [10,11]. Bilgi bölümleri arasında; includes (içerir), describes (tanımlar) ve wrote (yazdı) gibi ilişkiler görülmektedir. Maalesef, kaynaklar arasındaki bu ilişkiler hali hazırda var olan Web üzerinden elde edilememektedir. Böyle ilişkileri yakalayabilecek teknoloji, Kaynak Tanımlama Çerçevesi (Resorce Description Framework, RDF) olarak adlandırılmaktadır.



Şekil 1.2 Tim Berners-Lee tarafından düşünlen orjinal web modeli

Klasik web ile Semantik web arasındaki temel fark (Şekil 1.3) ve (Şekil 1.4)'de karşılaştırılarak anlaşılabilir [10,11]. (Şekil 14)'te nesneler, birbirlerine bağlanırken bir eylem ile niteliği pekiştirilmektedir. Nitelik eylemi için bir Semantik Web kullanılmıştır. Oysa (Şekil 1.3)'te nesneler arasında sadece bir bağlantı vardır. Bağlantının eylemi belirtilmemiştir.





Şekil 1.3 Klasik Web Bugünkü Web

Şekil 1.4 Semantik Web in nesneler arasındaki bağlantı

Bu amaçla çeşitli çalışmalar yürütülmektedir. İnternet teknolojileri geliştiren World Wide Web Consortium (W3C) bünyesinde bir süredir "Semantik (anlambilim) web" denilen bir metot üzerinde çalışmalar yapılmaktadır [11]. Web üzerindeki dinamik siteler bir veritabanına bağlı çalışırlar. Genel olarak web sitelerinde çok büyük yatırımlar yapılmadığı sürece, birebir eşlenik verileri göstermek üzere tasarlanmıştır, matematiksel sonuçlara bağlı kalarak sonuçlar döndürürler. Bu da çoğu kez, site ziyaretçilerinin arama sonucunun ve zamanlarının kaybolmasına neden olmaktadır. Örnek olarak; bir e-ticaret sitesinde, kullanıcılar (müşteriler) web sitesinin sınıflandırma yapısında tanımlanmış olan ürün reyonlarına bağlı kalarak aradıkları ürünü bulmak için çoğu kez zaman kaybına uğrarlar. Eğer aradıkları ürün hakkında sınıflandırma bilgisine sahip değillerse, site içerisinde bulmaları zor olabilir ve alışverişten vazgeçebilirler. Temel olarak internetteki milyonlarca bilginin bağlamlarına göre algılanarak birbirleri ile değişik bağlamlarda defalarca yeniden tasniflenmesinden oluşan "Semantik web", internet kullanıcılarına aradıklarına daha kolayca ve bilgi bombardımanına maruz kalmadan ulaşmasını hedeflenmektedir.

Semantik Web, gelişen bilgi (knowledge) kümelerinin oluşturduğu, herkesin bildiği ya da global olarak sorularına net ve doğru yanıtlar bulabildiği bir veri deposudur [12]. Bu bilgi kümeleri sadece ortam (media) nesnelerini (Web sayfaları, resimler, görüntülü içerikler, vb.) değil aynı zamanda insanlar, yerler, organizasyonlar ve olayları da kapsamaktadır [13]. Basit bir ifade ile anlatmak gerekirse Semantik Web, veriye daha fazla tanım veya anlam katarak veriyle kullanıcının karşılıklı etkileşimini ve internet üzerinde iş yapma biçimini değiştirmektedir.

1.3.1. Semantik web'e neden ihtiyaç vardır ?

Semantik Web amacı, internetteki milyonlarca bilgiyi tasniflemek; sistem, bilgileri bağlamlarına göre algılayarak, aranan bilgiyi bulunduğu ham bağlam ile ilişkilendirmektedir. Sistemde, semantik (anlambilim) metotları kullanılarak veriler birbirleri ile değişik bağlamlarda defalarca yeniden tasniflenmektedir.

İnternetteki milyonlarca veri birbirleriyle bir şekilde ilintilendirilebilir; tek bir bilgi birden fazla bağlama dâhil olabilir. Elbette, kimi zaman veriler birbirleri ile daha az ilgili olurken, kimi zamansa bağlamın çapı genişletilerek veri sayısı defa yükseltilebilir. Semantik Web'in ana fikri temel olarak, diğer belgelerle ilişkili PDF veya XML formatındaki dokümanların meta-datalarını (verilerle ilgili veriler) WEB üzerinden kullanmaktır.

Semantik Web, noktadan noktaya yapılan linklerin ötesinde; kişiler, yerler ve kavramlar üzerine kurulu yönlendirmelere olanak sağlarken kullanım esnasında veriyi otomatikleştirme, bütünleştirme ve yeniden kullanmaya imkân vermektedir.

Şu anki bilgi teknolojisi mimarilerinin karşılaştığı çok önemli birkaç sorunu çözebileceğinden Semantik Web'e ihtiyaç duyulmaktadır. Bu sorunların başında bilginin aşırı yüklenmesi, anlamsız içeriklerin oluşması gibi sorunları sayılabilir [14]. Farklı kaynaklardan toplanan bilgilerin formatını (yazı boyutu, paragraf arası boşluk v.b.) tanımlayan, ancak bir belgenin anlamıyla ilgili ipucu vermeyen HTML'de yazılmış bilgiler bulunmaktadır. Bu sorunlar için Semantik Web teknolojileri temelde veriyi tanımladıkları için çözüm olmaktadır.

Çok farklı tipteki verileri orijinal formatlarında tek bir havuzda tutabilen XML; bilgiye hızlı, kolay ve ortamdan bağımsız olarak erişebilme imkânı sunmaktadır. Günlük yaşantıda kullanmakta olduğu verilerin %80'ini oluşturan ve yapısal olmama özellikleri nedeniyle kendi bulundukları medya dışında veri özelliklerini koruyamayan (kelime işlem, elektronik tablo çıktıları, PDF dokümanları, ses, resim v.b.) [15] farklı tipteki verilerin, uyumuna gerek duymadan hiyerarşik bir yapıda kullanılabilmelerine imkân tanımaktadır. Bu tanımlı verilerin hızlı bir şekilde sorgulanabilmelerini sağlamaktadır. Öncelikle veri transferinin kolaylaşmasını ve verinin içerik bilgisiyle saklanabilmesini hedefleyen XML, içerik ve sunum bilgilerini birbirinden ayırır. Bu özelliğiyle de HTML' den farklılaşır. Sonuç olarak XML için aşağıdaki temel özellikler sayılabilir:

- XML yapılandırılmış belge ve verilerin evrensel formatıdır.
- XML, Semantik Web'in söz dizimsel temel tabakasını oluşturmak için kullanılmaktadır. Semantik Web için özellikleri sağlayan tüm diğer teknolojiler, XML üzerinde yapılacaktır. XML üzerinde diğer Semantik Web teknolojilerinin (Kaynak Tanımlama Çerçevesi, RDF v.b.) tabakalandırılmasını sağlamak, temel bir ara işlem yapılabilirlik düzeyini garanti eder [14,15]. XML'nin üzerine inşa edildiği teknolojiler, Unicode karakterler ve Düzgün Kaynak Tanımlayıcılarıdır (Uniform Resource Identifier, URI). Unicode karakterler, XML'nin uluslararası karakterleri kullanmasına izin verir.
- XML yeterli midir? Yanıt maalesef hayırdır; çünkü XML sadece sözdizimsel ara işlerin yapılabilmesini sağlar. Başka bir deyişle bir XML dosyasını paylaşmak içeriğe anlam kazandırır. Ancak sadece her iki taraf elaman adlarını tanıyıp anladığında bu durum gerçekleşir. Örneğin bir şeye **<fiyat>12$</fiyat>** ve bu alanı **<ücret>12$</ücret>** olarak ifade edildiğini düşünülürse; bu iki alan arasındaki ilişkiyi ifade eden ontoloji gibi Semantik Web teknolojileri eklenmez ise bu iki alanın aynı anlama geldiğinin bilgisayar tarafından anlaşılması imkânsızdır.

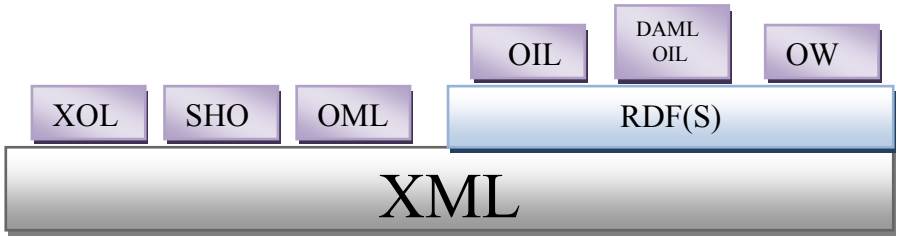
1.3.2. Kavramsal Dil

Kavramsal dilin (Conceptual Language) temel elemanları, sınıflar (Concepts) ve bu sınıfların özelliklerini belirlemeye yarayan rollerdir (roles). Bir kavramsal dil, var olan sınıf ve rolleri kullanarak, karmaşık sınıf ve roller tanımlayabilme yeteneğiyle değerlendirilmektedir.

1.3.3. Semantik web teknolojileri

Semantik Web, yapay zekâ olmamakla beraber yapay zekâ teknolojileri ile kullanabilen bir teknolojidir. Bu teknoloji bilgisayarın anlayabileceği akıllı veriler ortaya çıkarır. Bilgisayarın anlayabileceği akıllı veri kavramı, sadece bilgisayarın mevcut iyi tanımlanmış veriler üzerinde iyi tanımlanmış işlemler yaparak iyi tanımlanmış bir problemi çözebilme yeteneğini ifade eder. Bilgisayarlardan

insanların dilini anlamalarını istemek yerine, insanlardan bilgiyi daha düzenli tanımlamak için daha fazla çaba göstermelerini istemek akıllı verinin daha kolay oluşturulmasını sağlar[16]. Semantik Web teknolojilerini oluşturan önemli öğeler (Şekil 1.5)'de hiyerarşik bir şekilde gösterilmektedir. Bu teknolojileri bünyesinde barındıran sisteme, Semantik Web Teknolojileri denir. Semantik Web teknolojileri günümüzde uygulamaya konulan WEB 2.0 denilen yapıyı oluşturmaktadır.



Şekil 1.5 Semantik Web teknolojisi kullanılarak arama yapan sistem

Bu teknolojinin tüm basamakları birbirleriyle bağlantılıdır. XML hariç hepsinin kendi içinde sınıf modeli mevcut olmaktadır. Bu model farklı değişim formatları olan RDF/XML, N-Triples, N3 ve Turtle için baz teşkil etmektedir. Semantik dünyanın kalbini ise Ontolojiler oluşturmaktadır, bu yüzden de OWL (Web Ontology Language) yani Web Ontoloji Dili tüm açıklamaları ve ilişkileri yapabilmesi için kullanılacak teknoloji olacaktır. Semantik web teknolojilerinin kullanımına bir örnek FOAF (Friend Of A Friend) uygulamasıdır [17].

Semantik Web'in de tıpkı bilim gibi bir gelişim süreci sonunda oluşacağı görünmektedir, yani; en başta, mükemmel ve tam olarak işleyen bir sistem yerine, temel işlevleri gerçekleştiren basit ve sağlam bir sistemle başlayarak daha sonra bu sistemi daha karmaşık bir yapıya kavuşturmak hedeflenmiştir. Öncelikle bilginin gösterimi için bir biçim olan HTML üretilmiş ve bu bilginin transferi için http protokolü yazılmıştır. Daha sonra bunların üstüne bu bilginin yapısal bilgisini ve anlamını ekleyebilmek için XML ve RDF dilleri eklenmiştir. Son aşamada ontoloji tanımlamalarını oluşturan, işaretleme dilleri (OIL, DAML+OIL, OWL) verilmektedir. Her aşamada baştaki çekirdek sisteme yeni özellikler ekleyerek sistem genişletilmektedir. Bununla beraber önceden eklenen her sistem de tıpkı

son eklenen sistem gibi gelişmesine devam edecektir. Örneğin günümüzde OWL Dili ile ilgili çalışmalar sürerken, RDF dili ile ilgili çalışmalar da sürmektedir. Hatta OWL "Candidate Recommendation"[18] aşamasına gelmişken; RDF hala "Working Draft"[18] aşamasındadır. Yani gelişme sürecinde daha geri aşamadadır. Sonuç olarak, Semantik Web'in hayata geçirilebilmesi için yaklaşık 20 yıllık bir sürenin geçmesi gerektiği tahmin edilmektedir .

1.3.3.1. Web XML dili (Extensible markup language)

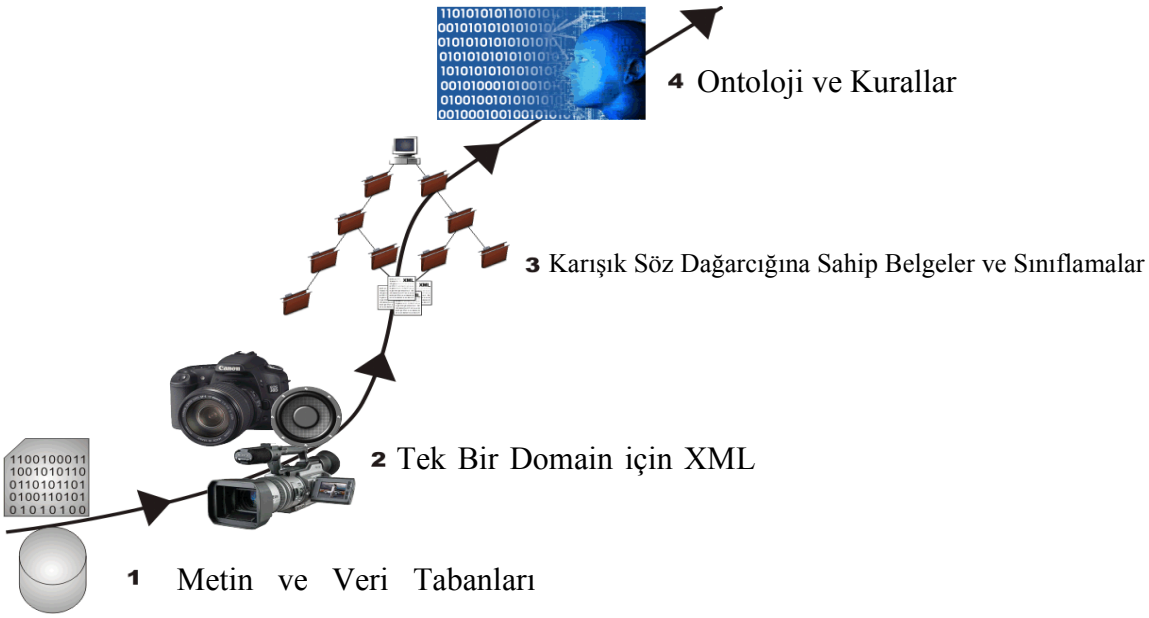
XML (Extensible Markup Language) Semantic Web'in en önemli yapı taşlarından biridir. Semantik Web XML dili (Extensible Markup Language) ile veriyi tanımlarken, uygulamalardan verilere doğru bir güç kaymasını meydana getirmiştir. Bağımsız bir kuruluş olan W3C (World Wide Web Consortium) organizasyonu tarafından tasarlanan ve herhangi bir kurumun tekelinde bulunmayan XML'in ana kullanım nedeni; organizasyon içinde ve dışında veri değişiminin sağlanmasıdır. Bu bakış açısından XML, birlikte çalışabilirlik sağlayan önemli bir araçtır. XML dört temel konuda başarı ile kullanılmaktadır [19] :

- XML uygulama bağımsız veri ve belge yaratmaktadır.
- Üst veri (meta-data) ortamı için standart bir gösterim sunmaktadır.
- Veri ve belge için ortak yapısal standartlar sunmaktadır.
- XML sınanmış bir teknolojidir

XML hem bir dil hem de bir teknoloji olarak, bir verinin biçimlendirilmesi, tanımlanması ve verilerin yapılandırılmasında kullanılmaktadır. Dolayısı ile veriler standart bir şekilde tanımladığından, web'te veya herhangi iki program arasında veri alış verişi kolaylaşmaktadır. Bu özellikleri nedeniyle XML, Semantic Web'in geliştirilmesinde önemli bir konuma sahiptir.

Bu yaklaşım, Semantik Web'i daha anlaşılır hale getirmektedir. Verilerin bilgisayarlar tarafından işlenebilmesinin tek yolu veriyi daha akıllı hale getirmekten geçmektedir. Bütün Semantik Web teknolojileri, akıllı verileri

oluşturmaya yönelik, sistemli (Şekil 16)' deki gibi bir çizelge boyunca verinin ilerlemesini sağlarlar [20] .



Şekil 1.6 Akıllı veriye doğru ilerleme süreci

Dört aşamanın sırası; en düşük anlamlı verilerden, bilgisayarların sonuçlar çıkarabileceği, yeterince anlamlı verilere doğru ilerlemeyi göstermektedir. Bu dört aşamadan kısaca bahsetmek gerekirse:

(a) Metin ve veri tabanları XML öncesi

Çoğu verinin bir uygulamaya ait olduğu ilk aşamadır. Bu aşamada veriler akıllı değildir.

(b) Tek bir domain için XML belgeleri

Verilerin özel bir alanda uygulama bağımsızlığına kavuştuğu aşamadır. Veri tek bir alandaki uygulamalar arasında hareket edebilecek kadar akıllıdır.

Bunu şu şekilde örneklendirebiliriz: Emniyet, otelde kalan müşterilerin verilerini tanımlama yaparak suç hareketi verisini ortaya çıkarabilir. Burada müşteri bilgilerini XML olarak tanımlayan otel uygulamaları ve emniyet

birimlerinin kullandığı ortak veri tanımlama sitemi bu sorunların üstesinden gelebilmektedir.

(d) Karışık Söz Dağarcığına Sahip Belgeler ve Sınıflamalar

Bu aşamada veri, çoklu alanlardan oluşturulabilir ve doğru biçimde hiyerarşik bir sınıflamaya yerleştirilebilir. Hatta verinin keşfedilmesi için sınıflama sistemi kullanılabilir. Sınıflandırma sistemi oluşturulurken, kategoriler arasındaki basit ilişkiler verileri birleştirmek için kullanılabilir. Bu nedenle bu aşamada veri kolayca keşfedilebilecek ve mantıklı olarak diğer verilerle birleştirilebilecek kadar akıllı durumdadır.

(c) Ontoloji ve Kurallar

Bu aşamada yeni veriler, mevcut verilerden mantık kurallarına göre ayrıştırılabilir durumdadır. Veriler, mantıksal hesaplamalar ve somut ilişkiler, karmaşık düzenlemelerde tanımlanabilecek kadar akıllıdır. Bu, daha atomik bir seviyede verinin birleştirilmesini, yeniden birleştirilmesini ve verinin çok ince ayrıntılı analizinin yapılmasını sağlar.

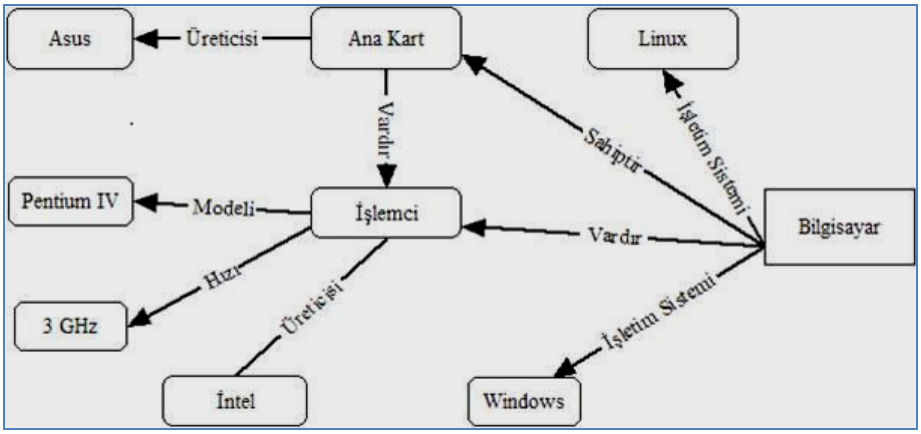
1.3.3.2. Web servisleri

Web servisleri; İnternet, intranet ve extranet üzerinde XML ve standart web protokollerini kullanarak uygulama birlikteliğini sağlayan, bilgiye erişimi kolaylaştıran, tanımlayan ve bilgiyi ortaya çıkaran yazılım uygulamalarıdır. Web servisleri, uygulamalar arasında entegrasyonu ve birlikteliği sağlayarak; iş yapmayı kolaylaştıran bir yapı sunmaktadır. Örneğin; otel bulma web servisi, uçak bileti aracılığı, araba kiralama web servisi vb. [21]. Web servisleri, iş süreçlerinin ve yazılımların entegrasyonunu sağlarken, grafik kullanıcı ara yüzünden ve gösterimden tamamen bağımsızdır. Farklı bilgisayar sistemleri arasında veri alış verişini kolaylaştıran web servisleri, yazılım ürünleri için standartlar geliştirmekte ve firmalar arası ticaret (B2B), sipariş, sigorta kontrolleri, finansal bilgi paylaşımı ve tedarik zinciri yönetim sistemlerinde işletmenin sınır tanımaksızın gerçek zamanlı işlem yapmasına olanak sağlamaktadır.

Gelecek yıllarda çeşitli ilgi alanlarında kullanılacak olan web servisleri, Semantik Web ile uyumlu hale gelerek, gerekli yapısal entegrasyonu sağlayacaktır. Böylelikle web servisleri, Semantik Web vizyonunun ilk önemli uygulama alanı haline gelecektir.

1.3.3.3. Ontoloji

Ontoloji, varlıkları ilişkileri ile birlikte tanımlayan felsefecilerin kullandığı bir sözcüktür ve Semantik Web' in en temel bileşenidir (Şekil 1.7) [22]. Ontolojiler belirli bir alandaki bilgilerin "paylaşılan ve genel bir anlamının" oluşmasına imkân verir.



Şekil 1.7 Varlıkların İlişkileri

Web Ontolojisi, web üzerindeki bir alanda (domain, özel bir konuya ait bilgi alanı), paylaşılabilinir bilgiye ulaşmak isteyen ihtiyaç sahiplerine nesnelerin kurallı tanımını yaparak ortak kelimeler ve anlamlar sunmaktadır [22]. Başka bir ifadeyle, Ontolojiler herhangi bir alanda standart olarak kullanılacak ortak ve paylaşılan sözcük kümelerini (vocabulary) veya terminolojiyi belirler. Ontolojiler ontoloji dilleri (RDFS, DAML+OIL, OWL …) ile tanımlanır. Ontoloji geliştirme araçları (editörleri), ontolojilerin görsel olarak kolayca tanımlanmasını sağlar. Ontoloji geliştirmeye konu olabilecek alan örnekleri; Gıda ontolojisi, araba kiralama ontolojisi, turizm, kara taşımacılığı, doğal gaz boru hattı bakım ontolojisi gibi çeşitli uygulama alanlarından verilebilir. Bu alanlar genelde sosyal bilimlerin, özelde de işletme bilimlerinin alanına girmektedir.

1.3.4. Web ontoloji dili

Ontoloji dilleri, ontoloji geliştirmek, tanımlamak, çeşitlemek ve ontolojilere web ortamındaki nesneleri tanımlamak için kullanılmaktadır. OWL, bilginin içeriğini sadece insanlara gösteren değil, bunun yanında bilgisayarlar tarafından işlenebilmek üzere tasarlanmıştır. Ayrıca Semantik Web dilleri ontolojilerin ve ontolojilerle web ortamındaki nesnelerin (kaynakların) tanımlanmasını sağlar.

W3C tarafından 2002 yılında geliştirilen OWL (Web Ontoloji Dili, Web Ontology Language) yaygın olarak kullanılmaktadır ve RDF (Kaynak Tanımlama Çerçevesi, Resource Description Framework) bilgisayarlarca işlenecek verinin anlamını temsil edecek olan veri modelinin düzenlenmesini sağlamaktadır. RDFS (RDF Schema) gösterimi ise, RDF veri modelini genişleterek, alanda kullanılacak sözcük kümesini nesneler ve nesneler arası ilişkiler, özellikler ve özelliklerin alabileceği değerler açısından tanımlamaktadır. Yukarıdaki ontoloji dillerinin Web üzerindeki standartların belirtilmesinde önemli rol oynayan W3C organizasyonu tarafından geliştirilmiş olması, ileriye yönelik uygulama sürecinde yaygın kullanım alanı bulacağı olarak yorumlanmaktadır [23]. Ayrıca ABD Hükümetinin desteği ile DAML (DARPA Agent Markup Language) ve Avrupa Birliği tarafından geliştirilen OIL (Ontology Interface Layer) ontoloji geliştirme dilleri de tanımlanmıştır. ABD ve AB tarafından ilk olarak 2000 yılında geliştirilen ve 2001 yılında da son sürümü yayınlanan DAML+OIL dili de pek çok araştırıcı tarafından kullanılmaktadır [24]. Stanford Üniversitesi tarafından geliştirilmiş Protege ontoloji editörü de, ontoloji geliştirilmesi konusunda kolaylık sağlamaktadır. Grafik arayüzü sayesinde ontolojiler görsel olarak tanımlanmakta ve böylelikle de alan modellenebilmektedir.

1.3.4.1. RDF (Resource Description Framework)

RDF (Resource Description Framework) yani Kaynak Tanım Çerçevesi meta-data model olarak tasarlanmış bir veri modelidir. RDF meta-data model, kaynaklar hakkında, RDF terminolojisinde triples (üçlüler) olarak adlandırılan, bu model web ortamındaki nesnelerin (kaynakların), kaynak özelliklerinin ve özellik değerlerinin tanımlanması fikrine dayanır. Ayrıca RDF ifadelerinde yer alan nesne, özellik ve değer üçlüleri RDF'in temelini oluşturur. Mesela Türkçe'deki şu ifadede 'New York, NY kısaltılmış posta koduna sahiptir.', bilgisini RDF olarak gösterilişi, özellikle formatlanmış bir metin üçlüsü olarak ifade edilebilir 'New York' özne, 'kısaltılmış posta koduna sahiptir' yüklem, 'NY' ise nesne olarak kabul edilebilir.

1.3.4.2.RDFS (RDF Schema)

RDF veri modeli web ortamındaki kaynaklar, isimlendirilmiş kaynak, özellikleri ve değerleri üçlülerini temel alan basit bir gösterim yöntemidir. RDFS, RDF veri modelini genişleten bir tür sistem olup; RDF ile ifade edilen modelin sözlüğünün oluşturulmasında kullanılır. Bu sözlükte, ilgili alanda kullanılacak olan varlıklar, varlıklar arasındaki ilişkiler, özellikler, özellikler arasındaki ilişkiler ve özelliklerin alabilecekleri değerler tanımlanır. Aşağıda verilen örnekte bir RDFS örneği Türkçe'ye çevrilerek verilmiştir.

***RDFS* örneği:**

```
<?xml version="1.0"?>
<rdf:RDF

xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xml:base="http://www.cicekler.fake/cicekler#">
<rdf:Description rdf:ID="Çiçek">
<rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-
schema#Class"/>
</rdf:Description>
<rdf:Description rdf:ID="Papatya">
<rdf:type rdf:resource="http://www.w3.org/2000/01/rdf-
schema#Class"/>
```

```
<rdfs:subClassOf rdf:resource="#Çiçek"/>
</rdf:Description>
</rdf:RDF>
```

Örnekte “Papatya, bir Çiçek’tir” ifadesi modellenmiştir. Bu ifadede “Çiçek” ve “Papatya” varlık adları olup, bu varlılar arasında kalıtım (is-a) ilişkisi vardır. RDFS, RDF sözlükleri oluşturmak için basit yetenekler sunmaktadır. Gelişmiş ontolojiler oluşturabilmek için bu yetenekleri genişleten üst seviye dillere ihtiyaç duyulmuştur.

1.3.4.3. SPARQL (Protocol and RDF Query Language)

SPARQL Semantik web için bir sorgulama dili ve veri erişim protokolüdür. W3C tarafından RDF veri modeli için tanımlanmıştır. RDF sorgulama dilleri üzerindeki çalışmalar son birkaç yıldır devam etmektedir. Bu süreç içerisinde RDQL, Squish, Versa gibi farklı yaklaşımların kullanıldığı diller de geliştirilmiştir. RDQL ve Squish’e benzer olarak SQL sözdizimini örnek alan bir dil olan SPARQL, kendisine geniş bir kullanım alanı bulmuştur. RDF ve OWL sorgulama araçlarının büyük çoğunluğu SPARQL desteği ile sunulmaktadır. Elementlere ait isim, sembol, renk, atom numarası ve kütle bilgilerinin yer aldığı periyodik tablo, veri seti üzerinden, elementlere ait isim, atom numarası ve renk bilgilerini, atom numarasına göre sıralanmış biçimde çekebilmek için tanımlanmış örnek sorgu aşağıdaki örnekte verilmiştir.

SPARQL Örnek Kullanım:

```
PREFIX table:
<http://www.daml.org/2003/01/periodictable/PeriodicTable#>
SELECT ?name ?number ?color
WHERE
{
?element table:name ?name.
?element table:symbol ?symbol.
?element table:atomicNumber ?number.
OPTIONAL { ?element table:color ?color. }
}ORDER BY ?number
```

1.3.4.4. DAML (DARPA Agent Markup Language) +OIL(Ontology Interface Layer)

Semantik Web'in temelini oluşturan ontolojileri tanımlamak için, RDFS şema dilinin yeteneklerini genişleten üst seviye dillere gereksinim duyulmaktadır. RDF (S)' in bir üst seviye katmanı olarak DAML (DARPA Agent Markup Language), OIL (Ontology Interface Layer), DAML+OIL ve OWL (Web Ontology Language) ontoloji dilleri tanımlanmıştır. DAML+OIL şu aşamada en gelişmiş ve olgunlaşmış bir dil olarak görünmektedir. DAML dili Amerikan hükümetinin desteklediği bir çalışma sonucunda Ağustos 2000'de yayınlanmıştır. OIL (Ontotoloji Interface Layer) Avrupa Birliği IST programı çerçevesinde geliştirilmiş bir dildir. Bu iki dilin yapılarını birleştirmek için Amerika ve Avrupa Birliği'nce oluşturulan ortak komite DAML+OIL dilini geliştirerek Aralık 2000'de yayınlamıştır. DAML+OIL'in en son versiyonu Mart 2001'de yayınlanmıştır. İlk yayın tarihinden itibaren DAML+OIL birçok Semantik Web araştırmacısının ilgisini çekmiş ve yoğun bir kullanım bulmuştur. Şu anda değişik alanlar için DAML+OIL ile geliştirilmiş yaklaşık 250 adet ontoloji ve 60 adet bu dile özel geliştirme aracı bulunmaktadır.

1.4. Anlamsal Web'in Uygulama Alanları

Ontolojiler, B2B (işletmeden işletmeye) alanındaki bilgilerin yönetilmesi ve e-ticaret alanında önemli bir rol üstlenme potansiyeline sahiptir. Büyük elektronik ticaret gruplarının veya birlikteliklerinin standartlaşmış operasyonlar yapabilmeleri için uygulamaların anlamsal olarak bilgileri paylaşabilmesi gerekmektedir. Bu da yeni hizmet ve ürünlerle sürekli gelişen, özellikle de üretici ve dağıtıcılar arasındaki ilişkilerin organize edildiği süreçleri daha verimli hale getirmektedir. Tedarikçilerle bağlantıların ve birlikteliklerinin dinamik olarak oluşturulması, otomatik iş süreçleri, şeffaf pazarlama, ürünlerin online olarak konfigüre edilmesi gibi işletme konuları, bu değişimden en fazla yararlanacak olan süreçler olarak görülmektedir. Örneğin UB/SPSC2 (Universal Standard Products and Services) ve UCEC3 gibi ürün ve hizmetlere uluslararası standartlar getiren ve bunların niteliklerini tanımlayan organizasyonlar, aslında yatay

ontolojilerdir ve B2B süreçlerinde önemli konumları vardır. RosettaNet gibi bütün endüstriye açık, e-iş için gerekli standartları oluşturulduğu ve iş süreçleri için ortak bir dilin sağlandığı birliktelikler, bilgi teknolojileri alanında elektronik bileşenlerin, yarı iletkenlerin tanımlandığı dikey ontolojilerdir [25].

Arabuluculuk, e-ticaretin önemli işlevlerinden birisidir. Hızla gelişen elektronik pazaryeri, alıcı ve satıcıları buluşturan sanal ortamlardır ve dinamik bir ekonomik değer değişim sistemini desteklemektedir. Gelişmiş arabuluculuk hizmetleri verebilmek için zengin ve esnek bir üst veri bilgisine sahip olmak gereklidir ve RDF gibi Semantik Web ile ilişkili teknolojileri kullanarak çeşitli modellemelerin yapılabileceği anlaşılmaktadır [26].

2. Bilgi ÇIKARIMI SİSTEMİ

Bilgi çıkarımı konusu, genellikle bir metin üzerinde doğal dil işleme metodunu kullanarak belirli kriterdeki bilgileri elde etmeyi hedefler [27]. Kullanıcının önceden tanımladığı şablonlar ile dokümanları tarayarak dolduran sistemlerdir. Şablonlar bir veri tabanı da olabilir. Örneğin firmaların web sitelerinden firma profillerini bulup daha önceden yapısı verilmiş olan bir veri tabanı tablosuna yazma işlemi bu tür bir işlemdir. Amaç çok miktardaki veriyi otomatik olarak işleyen bir yazılım üreterek insan müdahalesini en az seviyeye indirmektir.

Bilginin çıkarılacağı ortam genellikle yazılı metinlerdir ancak bu metinlerin bulunacağı kaynaklar değişebilir; örneğin veri tabanı, internet, dokümanlar veya taranmış metinler gibi verinin kaynağını oluşturabilir. Bilgi kelime anlamı olarak verinin işlenmiş halini ve anlaşılabilir veriyi ifade etmektedir. Örneğin resmi devlet kurumların yayınladığı haberler veri kaynağı olarak kabul edilebilir.

Bilgi çıkarım işleminin en zor adımlarından birisi de veriyi işlerken belirli bir yapıya oturtmaktır. Örneğin internet üzerinde yayınlanan verilerin herhangi bir standart yapısı bulunmamakta, veriler dağınık halde istenildiği gibi yayınlanmaktadır. Bu verilerin düzenli bir hale getirilmesi için XML ve benzeri teknolojilerden faydalanarak bilgi çıkarımı işleminin basitleştirilmesi hedeflenmektedir. Ayrıca çok sayıda yazılım, bilgi çıkarımına alt yapı hazırlamak amacıyla çeşitli kaynaklardan veri toplayarak düzenlemektedir.

Internet'teki bilgiye ulaşmanın önündeki en büyük engel; internetteki bilgilerin %90'ının doğal dilden oluşmasıdır [28]. Bilgisayarın doğal dili henüz yeterli bir şekilde anlayamıyor olmasından dolayı sayfalardaki bilgileri bulup çıkarmak yine insanlara düşmektedir. Ancak incelenecek belgelerin çok fazla olması durumunda bu işlemin çok zaman alacağı da bir gerçektir. Bilgi çıkarım sistemleri, kullanıcının yerini alarak her hangi bir konuda incelemeyi yapmakta ve sadece kullanıcının istediği sonuçları döndürmektedir.

Bilgi miktarı ve bilgi miktarının artış hızı sürekli artmaktadır. Bu kadar çok bilgiyi bulmak, işlemek ve kullanmak için insanın çok fazla çaba sarf etmesi gerekir. Kullanıcının ihtiyacı olan bilgiyi, büyük bilgi yığınlarının içinden bulup çıkarma işlemini bilgisayarlara yaptırabilmek, her geçen gün daha fazla istenmektedir. Bu sayede hesap makinelerinin insanın işlem gücüne yaptığı katkıyı, bu tür sistemlerde insanın bilgiye erişimine yapacaktır.

2.1. Doğal Dil (NL)/ Natural Language

İnsanlar arasında iletişimi sağlayan sözlü ve yazılı kurallar dizisidir. Dil içerisinde sesler, işaretler, semboller, kelimeler, cümleler ve paragraflar kullanılır [29,30].

2.1.1. Dilbilim (Linguistic)

Dilbilim, iletişimin en yaygın ve en temel aracı olan insan dilinin sistematik yapısını, bireysel ve toplumsal özelliklerini kuramsal ve uygulamalı olarak inceleyen sosyal bilim dallarından biridir. Bilimin giderek gelişmesi sonucu, günden güne modern dilbilim incelemeleri değişik alt-alanlar bağlamında yürütülmekte, "dil" olgusu disiplinler arası ele alınarak daha geniş bakış açıları ile de ele alınmaktadır [31]. Dillerin nasıl yapılandığını ve kullanıldığını araştıran bilim alanıdır.

Dilbilimin temel inceleme alanları genel olarak ikiye ayrılabilir : (1) Küçük-ölçekli (micro) incelemeler, (2) Büyük-ölçekli (macro) incelemeler [31]. Birinci grup incelemeler daha çok dilin sesbilim (phonology), sesletim bilgisi (phonetics), biçimbilim (morphology), sözcük bilgisi (lexicology), sözdizimi (syntax), anlambilim (semantics) bağlamındaki yapısal ve işlevsel özelliklerini ele alır. İkinci grup incelemeler ise dilin bireysel ve toplumsal özelliklere bağlı olarak ortaya çıkan görünümlerini araştırır.

Bir dilde, düşünce ve duyguların anlatılabilmesi için kelimelerin ne şekilde bir arada kullanılabileceğini gösteren kurallar dizisine gramer denir. Gramer, sözdizimi (syntax) ve anlambilimi (semantics) denilen iki alt parçadan oluşur [32].

Sözdizimi (syntax), cümle içinde kelimelerin, isim, fiil, zarf, sıfat vb. gibi hangi görevler ile yer alacağını inceler. Diğer bir anlatım ile sentaks kelimelerin bir cümle içinde yer alırken uyulması gereken bir metottur. Semantik ise dil içindeki kelimelerin anlamı ve birbiri ile iletişimi üzerinde çalışan bir bilim alanıdır. Semantik söyleneni analiz etmeyi, anlamayı ve yorumlamayı sağlar.

NL ile çalışılırken bilgisayar için gerekli olan genel kültür bilgi tabanı, alana özel bilgi tabanı ve göreve özel bilgi tabanı sağlansa dahi, dilin farklı kullanımlarından kaynaklanan belirsizlikler ortaya çıkar. Bu nedenle NL üzerine çalışmak oldukça zordur.

Örneğin:

Hasan: Bu gece, kim daha şanslı, İstanbul mu Eskişehir mı?
Ali: Bence ES ES , ya sen ne diyorsun?
Hasan: Olmaz öyle şey...

Yukarıda verilen örnekte böyle bir konuşma, çok şey ifade etse de bilgisayar için oldukça karmaşık bir anlama sahiptir. Anlaşılması ve yorumlanması zordur.

2.1.2. Doğal dil işleme (NLP) Natural language processing

Doğal Dil İşleme, yaygın olarak NLP (Natural Language Processing) olarak bilinen Yapay zeka ve Dil bilimin alt kategorisidir. Türkçe, İngilizce, Almanca, Fransızca, Arapça gibi doğal dillerin (insana özgü tüm diller) işlenmesi ve kullanılması amacı ile araştırma yapan bilim dalıdır [32].

Uzman Sistemler ve NLP yani Doğal Dil İşleme, doğal dillerin kurallı yapısının çözümlenerek anlaşılması veya yeniden üretilmesi amacını taşır. Bu çözümlemenin insana getireceği kolaylıklar, yazılı dokümanların otomatik

çevrilmesi, soru-cevap makineleri, otomatik konuşma ve komut anlama, konuşma sentezi, konuşma üretme, otomatik metin özetleme, bilgi sağlama gibi birçok başlıkla özetlenebilir. Bilgisayar teknolojisinin yaygın kullanımı, bu başlıklardan üretilen uzman yazılımların gündelik hayatın her alanına girmesini sağlamıştır. Örneğin, tüm kelime işlem yazılımları birer imla düzeltme aracı taşır. Bu araçlar aslında yazılan metni çözümleyerek dil kurallarını denetleyen doğal dil işleme yazılımlarıdır. Batı dillerinde SAPI (Microsoft şirketinin konuşma sentezleyici üretmek amacı ile satışa sunduğu geliştirici program) tabanlı Konuşma sentezleyici bileşenleri, yazılımcıların multimedia (çoklu ortam) sunuları hazırlamaları için hizmete sunulmuştur. Konuşma ve komut anlama yazılımları ise insan ve bilgisayar arasındaki klavye, fare gibi veri girişi aygıtlarını ortadan kaldıracak yazılımlardır. Bu gelişmeler makine-insan iletişiminde yeni ve devrimci değişimlere yol açacak ve bilgisayarların daha çok insan tarafından kabul görmesine yol açacaktır. Yapay Zeka ve Doğal Dil İşleme Gelecekte, konuşma sentezleyiciler ve konuşma anlama alanındaki gelişmeler ve makine-insan iletişiminin gelişmesi, insanın makineden beklentilerini yükseltecektir. İnsanlar makinelerin kendisini anlamalarını isteyecek, karmaşık kullanımı olan makineler pazar bulamayacaktır. Giderek gelişen ve insanı anlayan makinelerin daha zeki olması, insanın yaşam kalitesini yükselteceğinden, bu makinelerin vazgeçilmez olması kaçınılmazdır. Zeki makine kavramı, yapay zeka çalışmalarının hızlanmasına yol açmıştır. Geleceğin en önemli sektörlerinden biri olan yapay zeka ile insanın iletişim kuracağı tek araç dildir [32].

2.1.3. Dilin morfolojisi

Dil bilime terim olarak 1859 yılında August Schleicher [26]tarafından kazandırılan morfoloji, dilde biçimi oluşturan öğelerin türlerini tanımlar ve özetle dil bilgisi kuralları denilen biçimsel öğelerin sınıflandırmasını yapar.

2.1.3.1. Morfolojik çözümlemede analitik yaklaşımlar

Doğal dil işleme çalışmalarında, anlam bütünsel çözümleme yapabilmek için, bazı yaklaşımlar belirmiştir. Bu yaklaşımlar iki süreçten oluşur. Sözdizimsel (Semantic) Analiz ve sözdizimini (Syntax) [32].

Sözdizimini (Syntax): veya cümleyi oluşturan morfolojik öğelerin hiyerarşik kurallara uyumunu karşılaştırarak ölçümlemektedir. Böylece söz diziminin anlamlı olup olmadığının ölçülebilmesi için düzenleyici bir süreç gerçekleşmiş olur. Türkçede cümleler en genel şekliyle özne, nesne ve yüklem bileşenlerinden oluşur. Cümleye eklenmek istenen anlamlar arttıkça cümleler, özne, yer tamlayıcısı, zarf tamlayıcısı, nesne ve yüklem gibi bileşenleri içerir. Ayrıca cümlenin anlamını kuvvetlendiren cümle dışı bileşenler de (bağlaç, edat, vb) cümlede bulunabilir. Bunlara örnek olarak ile, için, ama, çünkü kelimeleri verilebilir. Türkçede özne ile yüklem cümlenin temel bileşenleridir ve genelde tüm cümlelerde yer alırlar. Yer tamlayıcısı, zarf tamlayıcısı, nesne gibi bileşenler bazı cümlelerde yer almayabilirler veya bazı cümlelerde sadece biri, bazılarında sadece ikisi bulunabilir. Bu bileşenlerin cümle içindeki sıralanışları da değişebilir. Bilgisayarla doğal dilin modellenmesinde anlamsal analizden önce kelimelerden oluşturulan yapının cümle olup olmadığının test edilmesi faydalıdır. Bu işlem sentaktik eşleştirme işleminde anlamsız eşleşmelerin önlenmesine faydalı olur.

Anlambilimsel (Semantic) analiz: sözdizimini oluşturan morfolojik öğelerin ayrılmasıdır yani, sözdizimsel analiz ile anlam taşıyan kelimelerin sınıflandırılması işleminden sonra gelen anlamlandırma veya anlama sürecidir. Bu süreçte anlam taşıyan kelimelerin, ekler ve cümle hiyerarşisi içindeki konumlarının saptanması sayesinde birbirleri ile ilişkileri kurulabilir. Bu ilişkiler anlam çıkarma, fikir yürütme gibi ileri seviye bilişsel fonksiyonların oluşturulmasında ham bilgi olarak kullanılacaktır. Yapay Konuşma Morfolojik çözümleme aşamalarından sonra sözdizimsel kurgu veya yapay konuşma süreci ile yapay zekâya veya uzman sistemlere iletişim becerisi kazandırılacaktır. Sözdizimsel çözümlemenin tersi süreçlerden oluşan birleştirme sürecinde, önceki süreçlerde ele geçen bilgi yine morfolojik kurallar dâhilinde birleştirilir.

2.1.4. Doğal dil işleme genel uygulama alanları

Bilgisayarların insan dilini algılamaları gereken her yerde kullanılabilir, muhtemel uygulama alanları şu şekilde sıralanabilir:

- İnternet Ortamında gittikçe artan dokümanların değerlendirilmesinde
- Uluslar arası Çalışan şirketlerin müşteri profillerini belirlemede
- Elektronik Ticarette
- Savunma ve İstihbarat Alanlarında (Güvenlik ve suçlu teşhisi)
- Yabancı Dil Öğretiminde
- Makine Çevirisinde
- Elektronik Sözlüklerde
- İmla hatalarının otomatik düzeltilmesinde
- Film ve Sinema Sektöründe
- Mobil Telefonların Konuşma Algılama Sistemlerinde
- Otomatik Özet Çıkarmada
- Bilgi Aramada
- Görme engellilerin bilgisayar kullanmalarında

2.1.5. Doğal dil işleme ve esas uygulamaları

Doğal dil işleme; insanların kendi aralarında iletişim için kullandıkları dilin bilgisayarlar yardımıyla Aşağıda çeşitli doğal dil işleme uygulamalarından örnekler verilmiştir.

- Metin Sınıflandırma: Bir metni önceden belirlenmiş kategorilerden hangisine ait olduğunu bulan sistemlerdir.
- Yazar Tanıma: Bir metnin yazarının bulan sistemlerdir.
- Dil Tanıma: Bir metnin hangi dille yazıldığını belirleyen sistemlerdir.
- Konu Belirleme: Bir metnin hangi konuda yazılmış olduğunu belirleyen sistemlerdir.
- Bilgi Çıkarımı: Bir metinden daha önceden belirlenmiş şablonların doldurulmasını gerçekleştiren sistemlerdir.
- Özet Çıkarımı: Bir metnin özetini çıkaran sistemlerdir.
- Soru Cevaplama: Kullanıcısının sorduğu sorulara cevap bulan sistemlerdir.

Doğal dilleri işlemek için birçok araç geliştirilmiştir. Doğal Dil isleyen yazılım araçlarıyla, uygulamada nerelerde kullanıldıkları çizelge 2.1'de

gösterilmiştir[33]. Yukarıdan aşağıya doğru kelimelerin anlamlarına olan ihtiyaç artmaktadır.

Çizelge 2.1 Doğal dil işleme araçları ve uygulamaları arasındaki ilişki.

NLP araçları	METİN	Uygulamalar
Morfolojik Analiz		Yazım Denetleme
Kelimelerin Türlerini Bulma		Gramer Denetleme
Birliktelikleri Bulma(Tamlama vb.)		Doküman Erişim
Sözdizimsel Analiz		Doküman Sınıflandırma
Kelimeler Arası İlişkileri (öğeleri) Bulma		Bilgi Çıkarımı
Etiketleme		Özet Çıkarımı
Şüpheli Durumların Çözümlenmesi		Soru Cevaplama
Anlamsal Analiz		Diyalog Sistemleri
Zamir Çözümleme		Otomatik Tercüme
Söylem Analizi	ANLAM	

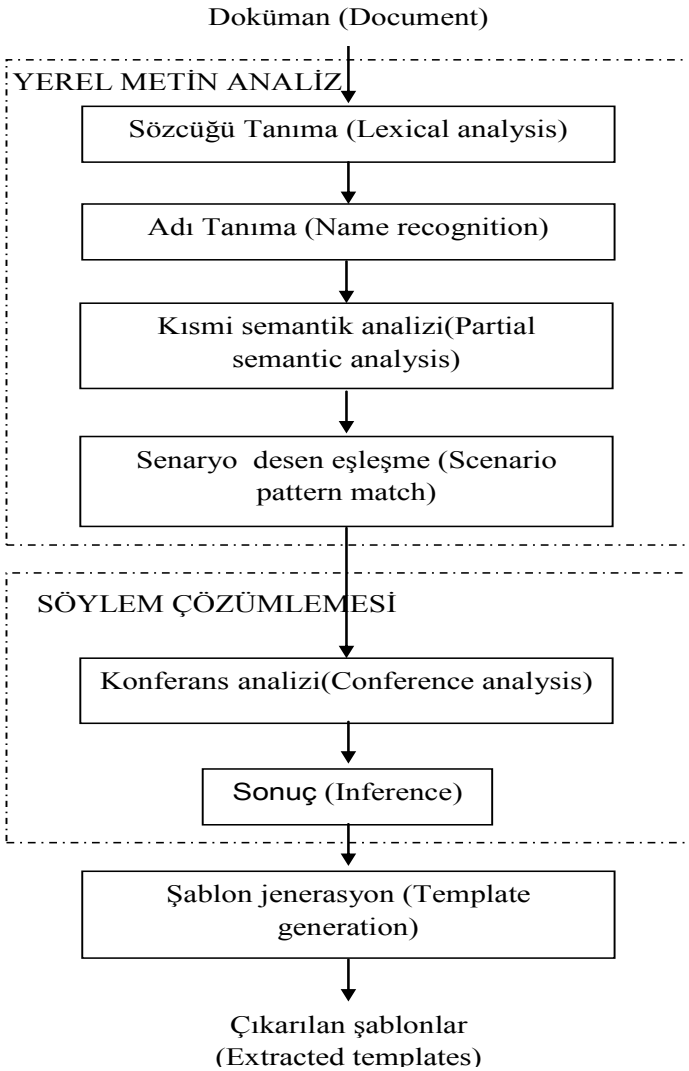
2.2. Bilgi Çıkarımın Bünyesi (Information Extraction Structure)

Bilgisayar bilgiyi iki biçim olarak saklamaktadır, yapılandırmış biçim (Structured) biçim ve yapılandırılmamış (Unstructured) biçim. Yapılandırmış biçim bilgiyi genelden kategorize eder ve anlamlı bir şekilde saklar dolayısıyla işleyişi kolay olur. Ancak büyük miktarda olan bilgiler örneğin kitaplar, haberler, raporlar ve diğer tip dokümanlar doğal dilde ve yapılandırılmamış biçim olarak saklanmaktadır. Oysa yapılandırılmamış biçim, bilişim sektörünün karşılaştığı büyük problemlerden biridir[33].

Bilgi çıkarım teknolojisi bu tür sorunları çözmektedir. Doğal dilde bilgi çıkarımının hedefi, yapılandırılmamış biçimdeki bilgiyi yapılandırılmış bilgiye çevirmektir. Bilgi çıkarımı teknolojisi, büyük miktarda yapılandırılmamış metinden gerekli veri parçasını alarak otomatikman anlamlı hale getirmektedir.

Bilgi çıkarımı farklı uyguluma alanlarında kullanılmaktadır. Bu uygulamalardan: Çıkarma, toplama, görselleştirme, karşılaştırma, arama,

indeksleme, sınıflandırma, çeviri, sorgu cevap (question formulating and query answering), tümevarım formüle soru, görev tanımı ve bilgi tabanı oluşturma [34].



Şekil 2.1 Bilgi Çıkarım Algoritmasının Genel Yapısı

Bilgi çıkarımında farklı uygulama alanları için genel bir algoritma üretmek karmaşık ve zordur; çünkü yapılandırılmış bilgide, biçimler her ne kadar birbirine benzese de, her uygulama alanın kendi yapı biçimine göre ihtiyaç duyduğu farklı bilgi parçaları gereklidir. Herhangi bir bilgi çıkarma algoritmaların genel yapısı (Şekil 2.1)'de [34] göstermektedir.

3. ARAPÇA DİLİ (ARABIC LANGUAGE)

İslamiyet'in Arapların dışında yaygınlaşmasıyla birlikte, bu dinin kutsal kitabının dili olan Arapça, sadece Türkler için değil, diğer milletler için de önemli ve öğrenilmesi gereken bir dil olarak kabul edilmiştir. Bu Arapçanın önemli dillerden biri olarak kabul edilmesi sonucunu doğurmuştur. Türk-Arap ilişkilerinin tarihi boyutu, zaman zaman göz ardı edilse dahi, Arapça; günümüzde de önemini ve dünya dilleri arasındaki etkinliğini gittikçe artıran bir dildir. Zira Arapça 22 Orta Doğu ülkesinde 330 milyona yakın bir nüfus tarafından konuşulan bir dildir. Ayrıca 24 Arap olmayan Müslüman ülkede de, 1 milyara yakın bir nüfus tarafından kullanılan bir dildir. Ayrıca yaklaşık %2 oranında, yani 6.5 milyon İngiliz tarafından konuşulan bir dildir [36]. Petrol üretimi ve petrokimya endüstrileri sebebiyle dünyanın ilgisi birçok Arap ülkesinin ekonomileri üzerindedir. Uluslararası ticaret, politika bilimi, uluslararası hukuk ve kültür tarihi öğrencileri, Arapça öğrenerek çok şey kazanabilirler. Antik arkeoloji ve Mısır'daki piramitler, sfenksler gibi tarihi eserler ve Arapça'nın edebi yoğunluğu, Arapça öğreniminin önemini artıran öğelerdir. Arapça ilahiyat fakültelerinin temel derslerinin başında gelmektedir. Ona bu niteliği; İslami kaynakların hemen tümünün Arapça olması ve bu dil bilinmeden bu alanda araştırma yapmanın imkânsız olması vermektedir. Arap Dili ve Yazısı Arapça, Afro-Asyetik (Hamito-Semitik)'dillerin alt grubundaki Semitik dil ailesine mensuptur [36]. Arapça, Arabistan yarımadası lehçeleri, Irak lehçeleri, Suriye lehçeleri, Mısır lehçeleri ve Kuzey Afrika lehçeleri gibi beş ana lehçe öbeğine ayrılır. Bu dil Arap Yarımadası'ndan Bereketli Hilal (The Fertile Crescent) boyunca Atlantik Okyanusu'na kadar ulaşan geniş bir alanda konuşulan dünyanın önemli dillerinden biridir [37]. Dünyada yaklaşık 215 milyon insanın anadili olan Arapça; bir milyarı aşkın müslümanın ibadet dili olmasının yanı sıra, Suudi Arabistan, Yemen, Birleşik Arap Emirlikleri gibi 22 Arap ülkesinin resmi dilidir. Arapça, büyük medeniyetler, kültür ve imparatorluklar doğuran dillerin başında gelir. Arapça'nın kullanımı 7. yüzyıla kadar Arap Yarımadası içine sınırlı kalmış, İslamiyetin

gelişiyle birlikte Arap yarımadasının dışında büyük bir hızla yayılarak, Irak, Suriye, Mısır ve Kuzey Afrika'yı kuşatmış, oradaki dillerin yerini almış ve bir kültür ve medeniyet dili olmuştur. Sonraki asırlarda İslami fetihlerin sürmesiyle Arapça doğuda Afganistan ve en batıda İspanya'ya kadar uzanan bölgede konuşulan bir dil haline gelmiştir. İslam'dan önceki "Cahiliye" diye adlandırılan dönemde, edebiyat özellikle de şiir çok üst düzeylere çıkmıştı. Ancak yine de yazma konusunda ileri seviyelere ulaşılmamıştı

Hz.Muhammed (s.a.s.) devrinde iki alfabe kullanılıyordu:

Nash: Kitap ve yazışmalarda kullanılan, yuvarlak harflerle ve bitişik olarak yazılmış olan el yazısı şeklidir.

Kufi: Çoğunlukla dekoratif amaçlar için kullanılan, keskin köşeli harfleri olan yazı şeklidir.

Arap alfabesi 28 harften oluşur. 28 harfli olan şimdiki alfabe; temel olarak harflerin üzerine ya da altına koyulan işaretlerle (hareke)belirtilen sesli ya da sessiz harflerden oluşur. Bu işaretler genelde kullanılmamalarına rağmen, ortaokul kitaplarında ve Kuran'ın tüm basımlarında yer alır. Diğer Semitik diller gibi Arapça da sağdan sola doğru yazılır. Alfabe Farsça, Peştuca, Urdu ve Sindi gibi diğer birçok dilde de kullanılır. Arapçada harfler; tek başlarına, sözcük başında, sözcük ortasında ya da sözcük sonunda olmalarına göre değişik biçimler alırlar. Arapçada üç sözcük türü vardır: fiil, isim, harf ya da edat. Adların eril ve dişil biçimleri vardır. Konuşulan Arapça doğal olarak ülkeden ülkeye değişir. Fakat Kuran dili olan klasik Arapça,7. yüzyıldan beri büyük ölçüde değişmeden kalabilmiştir.

Klasik Arapça (Eski Arapçada konuşulan dil); dilin standartlaştırılması ve geliştirilmesinde büyük bir itici güç olarak yer aldı. Farklı ülkelerden gelen eğitimli Araplar bir araya geldiğinde, genellikle klasik Arapça aracılığıyla iletişim kuruyorlardı. Günümüzde; Arap Yarımadası'nın Güney kıyısında güney Arapça olarak bilinen birçok lehçe konuşulur. Fakat Güney lehçeleri, Arap Yarımadası'nın kuzeyinde, o kadar farklıdır ki, güney Arapçası çoğu zaman ayrı bir dil olarak kabul edilir. Modern Arapça; temel kelimeler, morfoloji ve sözdizimi bütünlüğü bakımından, Kur'an'daki gibidir. Günümüzde yaygın olan

çoğu dil, Arapça'nın zengin kelime hazinesinden pek çok kelime almıştır. Örneğin; Türkçe'de Arapça kökenli birçok kelime bulunmaktadır. Ayrıca İngilizce'ye, Arapça'nın öneki olan -al ile başlayan birçok kelime geçmiştir. Bunlardan bazıları; algebra, alcohol, alchemy, alkali, alcove, ve albatrostur. Diğerleri ise; mosk, minaret, sultan, elixir, harem, girate, gazelle, cotton, amber, sofa, mattress, tariff, magazine, arsepial, syrup, sherbet ve artichoke gibi sözcüklerlerdir. "Coffee" de (kahve); İngilizce'ye, Türkçe ve İtalyanca yoluyla giren Arapça bir sözcüktür. "Assasin" (suikast) sözcüğü, "haşhaş bağımlıları" anlamındaki benzer bir Arapça sözcükten gelir.

Arapça, aşağıdaki ülkelerde konuşulur ve kullanılır: Cezayir, Bahreyn, Çad, Komor Adaları (Federal İslam Cumhuriyeti), Cibuti, Mısır, Etiyopya, Gazze, Şeridi, İran, Irak, Filistin, Ürdün, Kuveyt, Lübnan, Libya, Moritanya, Fas, Amman, Katar, Suudi Arabistan, Somali, Sudan, Suriye, Tunus, Türkiye, Birleşik Arap Emirlikleri, Amerika Birleşik Devletleri, Batı Şeria, Batı Sahra, Yemen Arap Cumhuriyeti.

3.1. Arapça Dilinin Çeşitleri

Arapça dili üç kısma ayrılmaktadır: Klasik Arapça, Modern Arapça ve Günlük Konuşma Dili. Günlük Konuşma dili de ülkeden ülkeye değişiyor Klasik Arapça bu dilin, en eski kısmıdır ve zamanla çok az bir değişikliğe uğramıştır, oysa bugünlerde bile, yazıda kullanılmaktadır. Klasik Arapçanın yazı dilinin, günümüzde hala kullanılması; katı dil bilgisi kuralları ve ses tonu sayesindedir. Modern Arapça diliyse, aslında klasik Arapçadan gelmektedir. Modern Arapça sadece yeni terimlerde, düşünce ve mefhumlarda kullanılmaktadır. Örneğin ingilizce Computer sözcüğünü, Arapça dilinde türetmek için yeni bir kelime oluşturmak gerekir. Dolayısıyla Computer kelimesinin tam karşıtı olarak Hasibe kelimesi türetilmiştir. Oysa “Hasibe” kelimesi, köken olarak klasik Arapça dilindeki “Hesab” kelimesinden gelmektedir. Modern Arapça dili, şuan Arapların ve Arapça konuşan toplumların kullandığı dildir. Son olarak Konuşma dili, her ülkede değişkenlik göstermektedir. Her dilin kendine ait mefhumları vardır, örneğin Domates kelimesi Irakta; domates olarak kullanmaktayken, Ürdün ve Suriye’de Banadora olarak kullanılır.

3.2. Arapçada Doğal Dil İşleme (Arabic Information Extraction)

Arapça dili, İngilizce ve diğer dilerden farklıdır. Dolaysıyla; Doğal Dil İşlemede daha farklı bir yaklaşım gerekmektedir. Diğer diller için anlam ayrımı, analiz sürecinin bir adımıdır, oysa Arapçada ana süreçtir. Doğal dil uygulamalarında Lexicon, her dilin omurgası sayılmaktadır; çünkü kelime ayrıştırma ve kelime türetme işleminin en temel elementlerinden birisidir. Dolayısıyla herhangi bir uygulama Lexicon olmadan çalışmamaktadır. Bütün Doğal dil işlemelerinde Lexicon işlemi gerekiyor[38]. Arapça dillinde Doğal Dil İşleme birkaç sorun ve engelle kaşı karşıya kalmaktadır:

- Genelde Latin dillerinde özel isimler ilk harfi büyük yazılmaktadır. Arapça dilinde makale ve günlük gazetelerde, çok sayıda özel isim bulunmaktadır ve bu özel isimleri metinden çıkarmak için özel kurallar gerekmektedir; Çünkü Arapçada büyük veya küçük harf yoktur.
- Arapça metinde, işaret eklemesiyle isim, fiile ve sıfatlara dönüşmektedir. Dolaysıyla anlam da değişmektedir. Örneğin aşağıda verilen iki kelime yazı dilinde aynı yazılmakta ama farklı anlama gelmektedir.

Örnek: كتب K(e)t(e)b(e) yazdı
كتب K(e)t(e)b kitaplar

- Genelde Arapça yayınlarda işaretler kullanılmamaktadır. Sadece Kuranı Kerimde ve ilkokul kitaplarında bulunmaktadır.
- Arapçada kök kelimenin baş, orta veya sonuna ek harf gelmesiyle, kök kelimeden başka bir kelime türetilmektedir. Örneğin aşağıdaki kelime iki parçadan oluşmaktadır: kelime başında olan parçacık edat, kelimeye ait bir harf olmamasına rağmen, edatın katılımıyla yeni bir kelime oluşmaktadır. Dolayısıyla bu yeni kelime başka bir anlam taşımaktadır.

Örnek: (fırsat) مناسبة
(vesilesiyle) ب + مناسبة

3.2.1. Morfolojik analizi

Kelimenin morfolojik analizi, kelime kökünün veya kelime gövdesinin morfolojik özelliklerinin atanması işlemi olarak tanımlanmaktadır. Bu tanımla şu şekilde gösterilmektedir; Sözcük türü (isim, fiil ve harf) ve bu üç türden çıkan alt bölümler; kelimenin cins sayısı (tekil, ikişer ve çoğul); sözcük irab durumu (açıklama); Ve sözcük gövdesini tanımlamak (öneki, sözcüğün ortasına konan ek ve sonek).

Morfolojik analizi geliştirmek için genelde dört metot kullanılmaktadır. Biricini metot (Syllable-based Morphology (SBM)) morfoloji analizi bu metodu kullanarak sözcükleri parçalara ayırmaktadır. İkinci metot (Root-Pattern Morphology) bu metotta; Morfolojik analizi kök ve fiillerin vezinlerine dayanarak yapmaktadır. Üçüncü metot (Lexeme-based Morphology (LBM)) morfolojik analizi; sözcüğün gövdesine dayanarak sözcüğü çekilmemiş duruma getirmektedir. Ve en son metot Arapçada dil bilgisi; grameri ve fiillerin vezinlerini kullanarak sözcükleri ayırmaktadır[39]. Bütün bu yöntemlerle, elle yazılan tablolarda sözcük kök, gövde ve çekimine dayanarak sonuçlar elde edilebilinir. Buna ilaveten morfoloji analizinde yapay zekâ yöntemi de vardır, o da Arapça dil kaynaklarını kullanarak, veri tabanını oluşturmaktadır.

Arapçanın morfolojik analizinde yaygın olarak kullanılan metot; (Tim Buckwalter Morphological Analyzer) metodudur. Bu metot elle hazırlanan sözcük ve morfolojik bilgiler, tablolarına dayanmaktadır. Bu tablolar da kök, önek ve sonek listelerini içermektedir[40,41]. Diğer sözcük kökü çıkarım sistemlerinde de (Khoja's Stemmer)Şirin KOCA sistemi, bu sistem, sözcük kökünü çıkartmak için sözlüğün en uzun önek ve soneklerini alarak, daha sonra tabloda bulunan fiillerin vezinleriyle karşılaştırır. Bu sistem bir çok özelliğe sahiptir. Örneğin: kelime işaretleri, üçlü ve dörtlü kökler, noktalama işaretleri, tanımlayıcı edatları ve bir listede 168 stop sözcükleri (Stop Words), yer almaktadır. Bu sistem çok hata vermemesine rağmen, bilgi çıkarım uygulamalarında kullanılmaktadır. Dolaysıyla bilgi erişme sistemleri için gelişim sonuçları elde etmiş oldu [42]. Bütün bu çalışmalar daha önceden yapılan projelerde yer almaktadır. Yapılan projeler

yeterli isteği karşılamadığı ve semantik web'te sınırlı olduğu için bu defa başka yöntemler ortaya çıkmaktadır.

3.3. İsim ve Fiil Etiketleme (Name and Veb Tagging)

Arapçada, isimleri ve filleri ayırt etmek için bazı belirgin işaretler vardır. Bu işaretlerden birisi sözcüklerdeki önektir; ekler genelde fillerde kullanılır ve bazen de isim ve fillerde kullanılır. Birkaç araştırma projesinde bu tekniği kullanarak metinden sözcük çıkarma başarıyla sonuçlanmıştır. Andrei Mikheev [43], bu tekniği ve tam otomatik edinme kurallarını kullanarak, sözbölük etiketlemelerinde (part of speech tags) ve bilimin önek ve sonek tahmininde bulundu. Bazı tahmin kuralları genelde önek morfoloji ve sonek morfoloji kuralları içermektedir. Zhang ve Kim [44], morfolojik sözcük işlevi, otomatik kural öğrenme için bir sistem geliştirdiler. Bu sistemde, sözcük diziyi üç parçaya ayırarak ve daha önceden yazılmış eğitim yazılarından benzer örnekleri alarak, sistemde temel morfoloji özellikleri oluşturulur. Arapça dilinde sözcükleri ayırt etmek için, fiillerin vezinleri, fonksiyonu en önemli kılavuzlardan biri sayılmaktadır. Bu fiillerin vezinleri bazıları sadece isimler için kullanılmakta; bazıları da sadece filler için kullanılmaktayken, bir kısmı da hem isim hem de filler için kullanılmaktadır. Gramer kurallarının en önemli etiketlemelerinden olan bir takım etiketlerin özellikleri şöyledir; bazı etiketler isim ve filleri ayırt etmek için kullanılmaktadır, bu etiketlemelerden bir diğeri de edat etiketlemesidir (Prepositional). Bu edatlar tıpkı İngilizcede kullanılan edatlar gibidir. Bazı edatlar isimleri belirliyor ve bazıları da filleri.

3.4. Özel İsimleri İşaretleme

Doğal dil uygulamalarını desteklemek amacıyla, özel isimler için yapısal giriş inşa etmek değeri çok önemli değildir. Bu genel isim, fiil ve sıfatların tanıtımı ve analiz işlemleri kadar önemlidir. Özel isimler ve semantik kategorileri, metni anlamak ve bilgi çıkarımı için önemli bilgilerdir [45,46]. Ayrıca bilgi geri alma sisteminde de kullanılmıştır [47]. Yapısal semantik ve bilgi geri alma sistemi arasındaki ilişkinin yararlılığını bir çok çalışma göstermektedir[48,49,50]. Yapısal- semantik ilişkilerinin önemi, başka uygulamalarda da gösterilmektedir.

Örneğin soru sorma ve cevaplama sistemleri [51]. Rau [52], bilgi alma sisteminde savunuyor ki, bir metinde, büyük oranda bilinmeyen kelimeler sadece özel isimler değil. Lakin aynı zamanda bir metin içeriğini ayıklamak için. Çok önemli bir kaynaktır, ayrıca bir metinde söz konusu olan bilgiyi tanımlamak ve konuyla ilişkisi olan belgeleri tespit etmek içinde bir hayli önem arz etmektedir. Wacholder'ise [53], metinde özel isimleri çıkarmada ki engelleri kaldırmak için, birkaç farklı türde semantik ve yapısal belirsizlikleri analiz etmiştir. Jong-Sun Kim ve Evens [54] kişisel isim ve değer isimleri, Wall Street dergisinden ayıklamak için bir doğal dil işleme sistemi inşa etmiştir. Al-Raya gazetesinden bulunan özel isimleri aşağıdaki çizelge 2.2 tablolar şeklinde sınıflandırılmaktadır:

Çizegle 2.2 Özel İsimler Sınıflandırma

Özel İsimler:

İsim	Meslek	Organizasyon	Uyruk
M.Evens	Profesör	İİT	Amerika

Organizasyon İsimleri:

İsim	Tür/Tip	Yer	Servis/Hizmet
İİT	Üniversite	Chicago	Eğitim
Byte	Dergi	Amerika	Bilgisayar

Yer(Politik İsimler):

İsim	Tür/Tip	Yer	Dil
Chicago	Şehir	Illinois	İngilizce
Illinois	Eyalet	Amerika	İngilizce

Yer(Coğrafi İsimler):

İsim	Tür/Tip	Yer
Nil	Nehir	Afrika
Atlantık	Okyanus	Dünya

Ürünler

İsim	Türü	Yapım_yeri
Toyota	Araç	Japonya
Compaq	Bilgisayar	Amerika

Tarih

İsim	Tarih_Parça	Ugulanan_Zaman
Eylül	Aylar	9th
Noel	Tatiller	Aralık

Kategori (milliyet, dil, din, etnik, parti, vb.)

İsim	Türü	İlgili
Amerikalı	Milliyet	Amerikalılar
Arapça	Dil	Araplar

3.5. Arapçada Metin Madenciliği Sistemi

Veritabanı sistemleri, bilgileri yapılandırılmış veri şeklinde saklar ve kullanıcı bilgileri ulaşmak veya sorgulamak için sorgu yöntemlerini kullanır. Metin madencilik olarak tanınan Knowledge Discovery Databases (KDD) Araştırmacıları [55,56], üstü kapalı ve önceden bilinmeyen veriler için yeni bir bilgi çıkarım tekniği sağladılar. bu teknik; potansiyel olarak yapılandırılmış veritabanlarından özel amaçlı bilgi çıkarmak için yararlıdır [57,58]. Bu alanda, kalıp olarak yapılandırılmış veritabanlarında otomatik keşif için öğrenme makinesi uygulamakta ve istatistiksel analiz teknikleri içermektedir. Elektronik olarak yapılandırılmamış olan: İnternette, Dahili Ağlarda ve e-postada, veri hacminin sürekli büyümesine rağmen Knowledge Discovery Databases'in (KDD) son on yıl içinde yaptığı çalışmalar sayesinde, metin madencilik alanında yapılandırılmış veritabanına odaklanıldı [59]. Yeni bir teknoloji olarak ortaya çıkan Metin madenciliği, belgelerde metin kalıpları bulmayı hedefliyordu. Yapılandırılmış veri kalıpları bulmayı amaçlayan madencilikle, yapılandırılmamış veri analizi karşılaştırıldığında, metin madenciliği işlemi daha zordur.

İnternette bulunan, çok miktarda yapılandırılmamış Arapça bilgileri nedeniyle; Arapça metin madenciliği bu durumlarda önemli rol almaktadır.

4. GATE (General Architecture for Text Engineering)

4.1. Giriş

GATE, Sheffield üniversitesi tarafından hazırlanan ve FSF (Özgür Yazılım Vakfı) altında LPGL (Library General Public License) [60] lisanslı olarak Doğal Dil İşleme için geliştirilen bir uygulamadır. GATE, açık kaynak, Java dilinde yazılan, destekli her hangi bir platformda çalışmaktadır. Ayrıca Unix, Windows ve MacOS işletim sistemlerinde test edilip ve çalışmaktadır. GATE geniş kapsamlı birçok uygulamalara desteklemektedir bu uygulamalardan biriside Bilgi Çıkarımıdır. GATE standart olarak 50 üzerinde plugin ve 70 tür kaynak (processing resource) içermektedir [61]. Örneğin bu kaynaklardan da bazıları: Tokeniser, Sentence Splitter, POS tagger ve Gazetteer'dir. GATE, XML, HTML, PDF, MS Word, email ve açık metin gibi girdi formatları desteklemektedir. GATE yazılımı, yazı formatı olarak UNICODE kullanmakta ve bu sayede bütün dilleri desteklemektedir. GATE veritabanı olarak Oracal ve PostgersQL kullanmaktadır [61].

4.2. GATE Entegrasyonu

Başka projelerden yararlanarak, GATE performans gücünü artırmış ve sistem platformunu bütünleşmiştir:

- Bilgi Alma (Information Retrieval): Lucene (Nutch, Solr) [62], Google ve Yahoo APIs [63], MG4J (Managing Gigabytes for Java) [64] metin arama motorları;
- Makine Öğrenmesi (Machine Learning): Weka [65], MaxEnt [66], SVMLight [67], vb.;
- Ontoloji Desteği (Ontology Support): Sesame ve OWLIM [68];
- Araştırma (Parsing): RASP [69], Minipar [70], ve SUPPLE [71];
- Ve UIMA [72], Wordnet, Snowball [73].

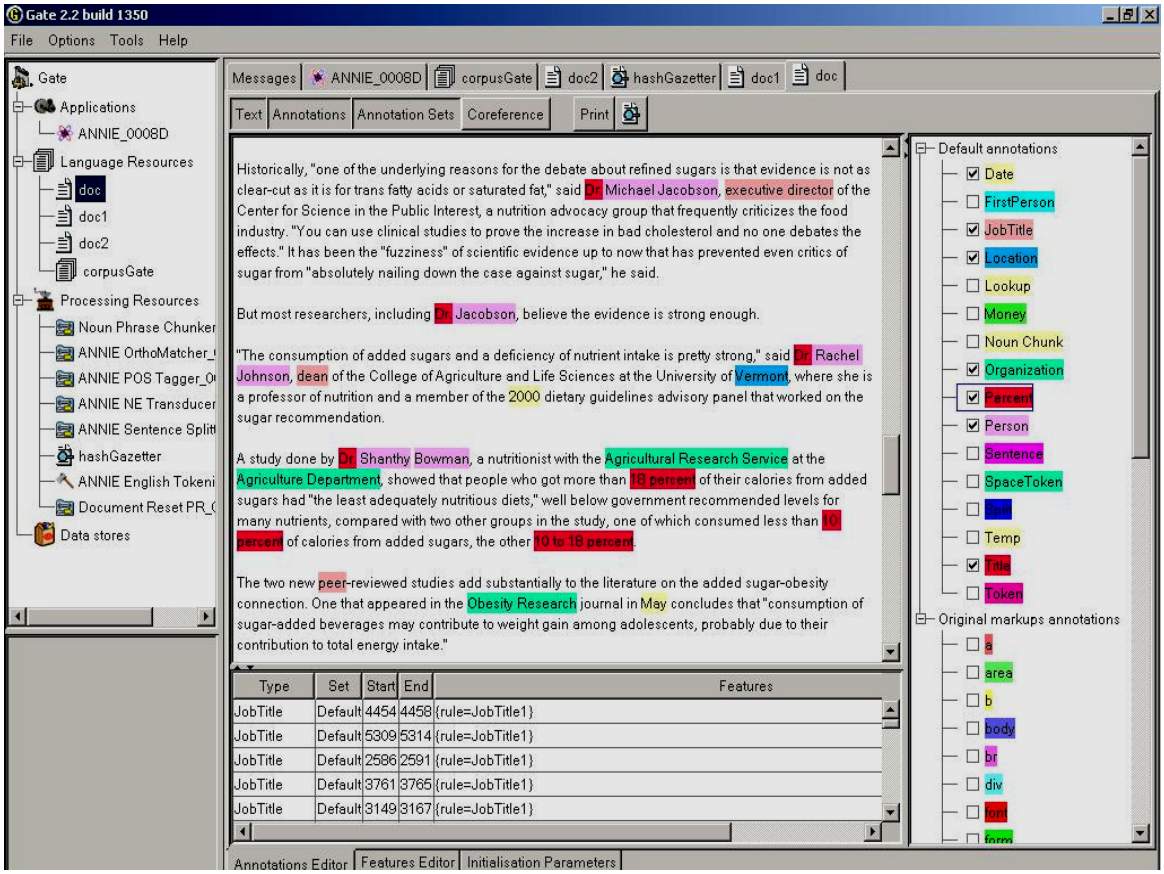
GATE arka planda üç parçadan oluşmaktadır:

- Mimari: bu parça, Dil İşleme sistem bileşenlerini göstermektedir.
- Yapısı, kütüphane veya SDK (Software Development Kit): bu parça Java ile yazılmış ve Linux, Windows ve Solaris işletim sistemler üzerinde test edilmiş.
- Yapı üzerinde inşa edilmiş grafik geliştirme ara yüz.

4.3. GATE Mimarisi

GATE mimarisi Doğal Dil İşleme ile ilgili bütün fonksiyonları içermektedir, tokeniser, sentence splitter, POS tagger, gazetteer vb. Ayrıca bu unsurlar Doğal Dil İşleme dünyasında Kaynak (Processing Resource) olarak ayrılmaktadır. Bileşenleri parçaları yeniden iyi tanımlanmış arayüz ve popüler bir mimari formu tanımak için, Sun'ın Java Beans ve Microsoft'un Net kullanılmaktadır. GATE sistemi üç kaynaktan oluşmaktadır:

- Dil kaynağı (Language Resource (LR)): Sadece veri kaynaklara işaret etmektedir. Genelde Bu bölüm doküman, sözlük, corpus, ve ontolojileri saklar. Bu kaynaklara ulaşmak için GATE gerekli servisleri sağlamaktadır [73].
- İşleme kaynağı (Processing Resource (PR)): Bu kaynağın esas özelliği, programlı veya algoritmatik olmasıdır. Örneğin: Çeviriciler, ayrıştırıcılar veya konuşma belirleyici. Genelde İşleme kaynağı (PR), Dil kaynağını (LR) içerir, DDİ'de bilinen Tagger modülünde her zaman sözlük (lexicon) bulunmaktadır. Örneğin manası açık olmayan bir sözcük olursa, bu durumda sistem eşanlamlara veya sözlüklere başvurulur. Dil ve İşleme kaynakları veri deposunda saklanır ve bu kaynaklara ulaşmak için veri deposundan çağırılır.
- Görsel kaynak (Visual Resource (VR)): GKA (Grafiksel Kullanıcı Arayüzü) bileşenleri düzenleme bileşenleri ve diğer görsel parçalaradır. Şekil 4.1'de anlatılan kaynaklar gösterilmektedir.



Şekil 4.1 GATE platformu ve kaynaklar

4.4. GATE'in Yapısı

GATE geliştirme konusunda, sistemde Dil işleme kaynağını görevini yapan araçlar vardır. Bu araçlarda yapılan geliştirmeleri, GATE Yapısı farklı kaynaklarla bağlayarak sistem yeni bir özelliğe sahip olur, ancak bu işlem ekstradan sistemin bünyesini değiştiren bir kod değildir. Bu konu daha detaylı olarak bir sonraki kodlama JAPE (Java Annotation Patterns Engine) bölümünde açıklanacaktır.

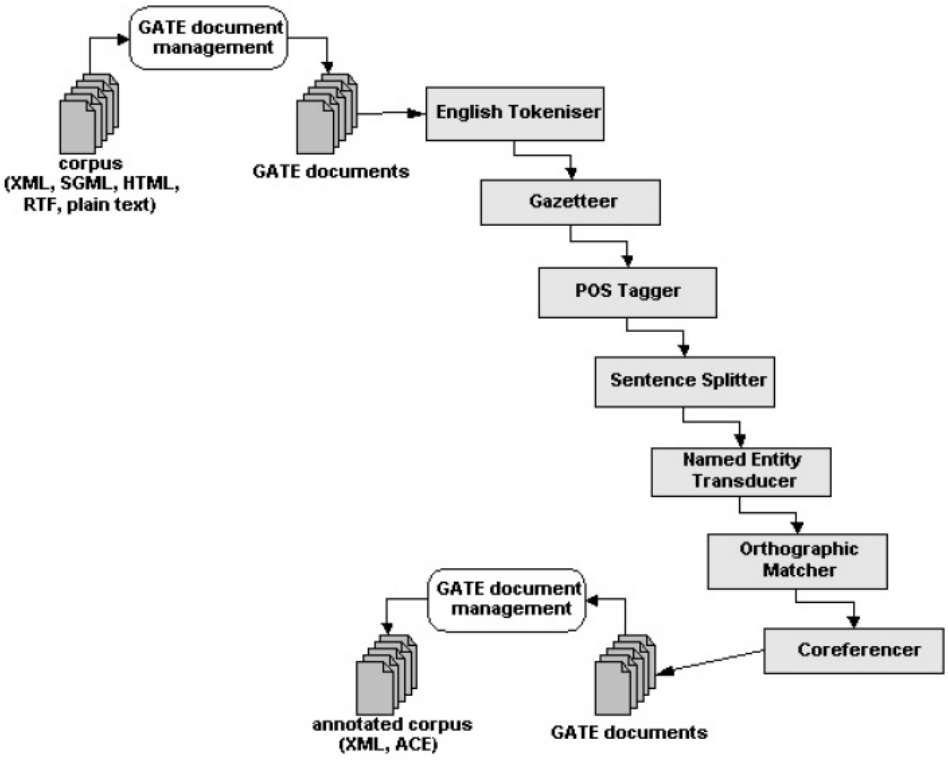
GATE yapısı üç parçadan oluşmaktadır:

- Dil kaynağı (Language Resource (LR)) yazılım programlama ara yüzü
- İşleme kaynağı (Processing Resource (PR)) yazılım programlama ara yüzü
- Ve Görsel kaynak (Visual Resource (VR)) yazılım programlama ara yüzü.

4.5. GATE ve Bilgi Çıkarımı

GATE, bilgi çıkarımla ilgili farklı dillerde ve sorun alanlarında birçok projelerde kullanılmıştır. GATE, bilgi çıkarım konusunda, uygulamaların oluşmasına kolaylık sağlayan bir açık kaynak platformdur, ayrıca çeşitli yaklaşımlarda bulunmaktadır. Örneğin Anlamsal arama, kural tabanlı, istatistiksel tabanlı ve makine öğrenme yöntemleri.

GATE sisteminde, bilgi çıkarımının bileşen seti gömülü olarak bunmaktadır. Bu sete ANNIE (a Nearly-New Information extraction System) denilir. Bu set, Hamish Cunningham, Valentin Tablan, Diana Maynard, Kalina Bontcheva ve Marin Dimitrov tarafından geliştirilmiştir [74]. ANNIE seti açtık metin ve dosyalar üzerinde bilgi çıkarmak için aşağıdaki şekilde verilen modüllerden oluşmaktadır (Şekil 4.2) [75,76].



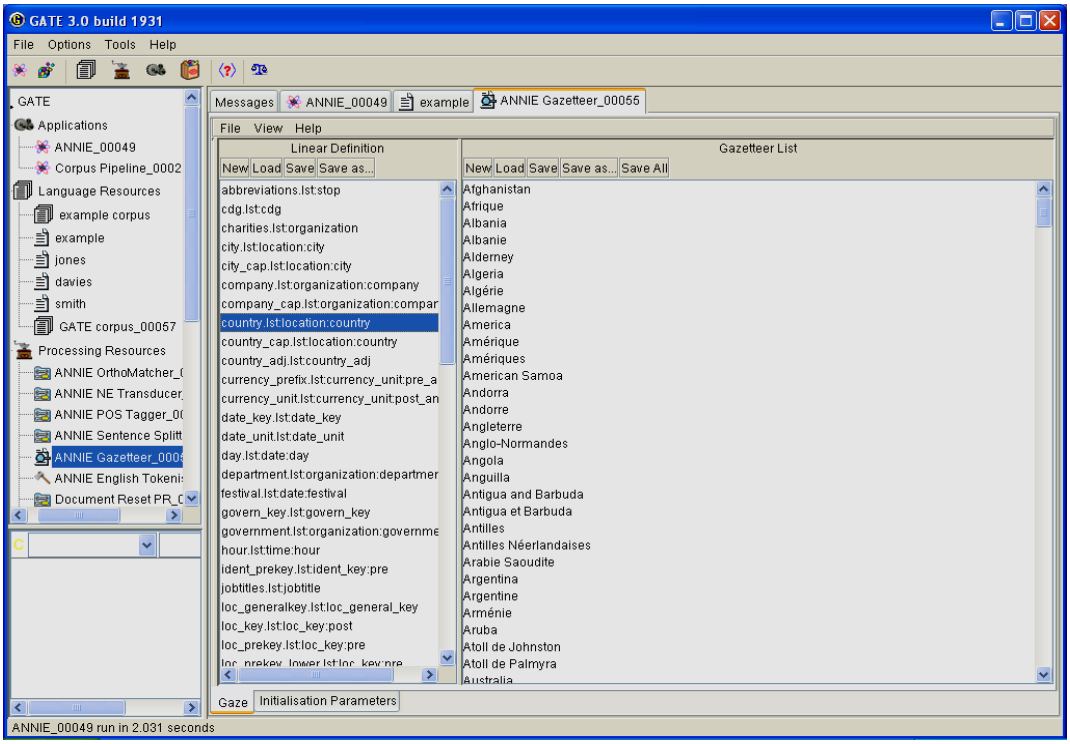
Şekil 4.2 ANNIE Set Bileşenleri

GATE'te gömülü olarak ANNIE modülleri, bilgiyi metinden çıkarmak için ek açıklamalarıyla (Annotations) olarak göstermektedir. Şekil 4.5'de bu ek açıklamalar gösterilmektedir. ANNIE sürecine devem etmek için sisteme XML, HTML, Email vb. formatlarında dosya alınmaktadır. Dosyaları sisteme alındıktan sonra sistem tarafından bu dosyalar GATE dosyası olarak görünmektedir. Bu dosyalar üzerinde işlemi başlatmak için, iki yöntem kullanılır.

Birinci yöntem Otomatik işlem ve ikinci yöntem ise GATE platformun sunduğu araçlarla, kullanıcı geliştirme yöntemdir. Kullanıcın yöntemi detaylı olarak bir sonraki bölümde bahsedilecektir. GATE platformuna işlem kaynakları PR (Processing Resource) yüklendikten sonra, Otomatik yöntem veya ANNIE süreci başlamış olur.

Şekil 4,2'de görüldüğü gibi süreç ilk önce Tokeniser modülüyle başlıyor. Temel olarak Tokeniser modülü, DDİ (Doğal Dil İşleme) dünyasında ilk işlem aşamasıdır. Bu modül, metni önce analiz edip ve sonra bileşenlerine ayıran bir yazılımdır. Bu bileşenler sözcüklere, noktalama işaretlerine vb bileşenlerine ayrılır. Bu öğelerin her birine "token" denir. Kısaca söylemek gerekirse bir "tokeniser" girdi olarak bir metni alır ve bunu parçalar. Bu modülde metin üzerinde yapılan işlemler daha sonra diğer programlarda kullanılır. Bu işlemden sonra Gazetteer modüllü gelir.

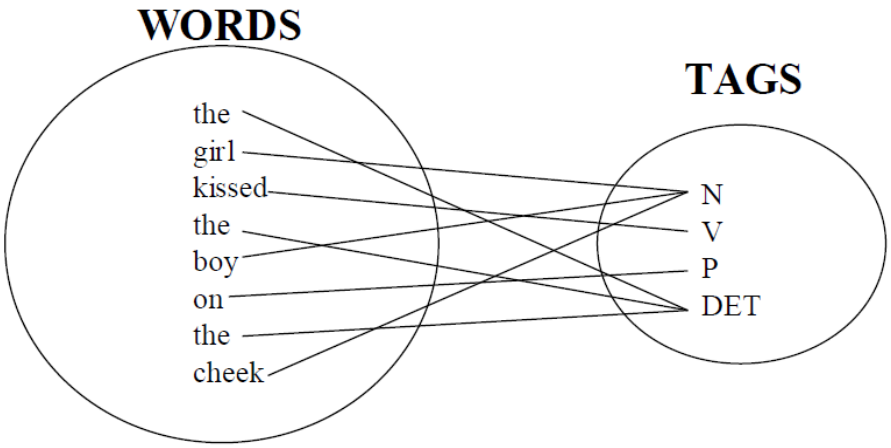
Gazetteer modül, bir dosya olarak sözcükler içermektedir, bu sözlükler şehir, organize, gün, özel tarih vb. isimler liste şeklinde ve her sözcük tek satırla dosyada yer alır, Bu dosyalara isim olarak Gazetteer dosyası denilir [77] (Şekil 4,3). Genelde Gazetteer dosyası iki dosyadan oluşmaktadır, birinci dosya indeks olarak ayrılır ve sistemde bulunan Gazetteer liste adlarını içerir bu dosya genelde list.def uzantısıyla ayrılmaktadır. Aşağıdaki Şekil 4,3'de gösterilen her liste üç özelliğe taşımaktadır, bu özellikler: majorType (Büyük Tipi), minorType (Yandal Tip) ve language (Dil). Bu Liste özellikleri genelde JAPE'te kurallarında kullanılır. Bir sonraki JAPE bölümlünde detaylı bir şekilde bu özelliklerden bahsedilecek.



Şekil 4.3 Gazetteer Indeks ve liste Dosyaları

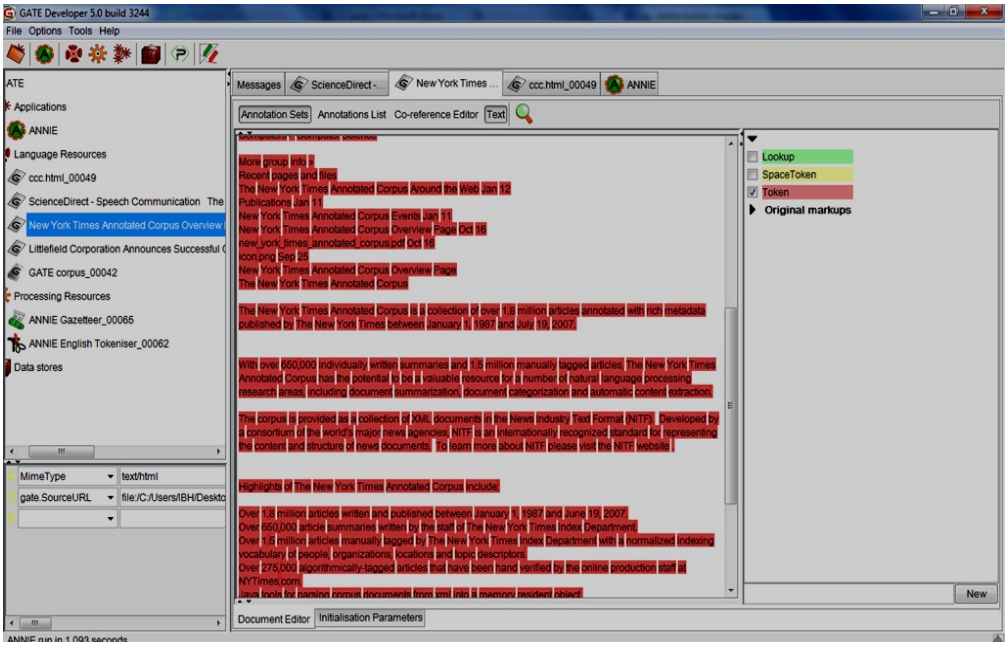
Bir sonraki modül ise cümle ayırıcı (The sentence splitter), açık metinleri segmentler şeklinde cümlelere ayırmaktadır. Cümle ayırıcı modülü, Gazetteer'de bulunan kısaltmalar listesinin kullanarak cümlenin sonunu ayırtmaktadır. Cümle sonu bazen nokta, bazen de soru veya önlem gibi işaretlerle bitmektedir.

Bir sonraki modül Konuşma birimlerinin (part of speech (POS)) etiketlemesi (tagging), Doküman içerisinde yer alan her bir kelime POS olarak adlandırılır. Her POS kendisine verilen TAG'ler (etiket) ile ifade edilir ve sözdizimsel yapı çıkarılırken kullanılır. Benzer sözdizimsel davranışlar ile kelimelerin sınıflandırılması syntactic veya grammatical categories veya parts of Speech (POS) olarak adlandırılır [78]. POS'ları hem İngilizce hem de Türkçe için (şekil 4.4) [78] gösterilmektedir. Aşağıdaki verilen şekilde örnekte TAG'lerin bulunan kısaltmalar N (isim),V (fiil), P (edat/ilgeç) ve DET (ismi belirler) belirtmektedir.



Şekil 4.4 İngilizce Cümlenin Ögeleri (Part Of Speech POS)

olarak bulunan ek açıklamalar (Annotate) ve JAPE'te kurallarında düzenli ifadeleri temsil etmektedir. Örneğin bir dokuman üzerinde ANNIE modülleri uygulandıktan sonra standart olarak paragraflar, boşluklar, token, cümle sonları vb. Şekil 4.5. Yukarıda anlatılan Bilgi çıkarımda standart olarak sistem tarafından kullanılmaktadır. Şimdi sistemden kullanılan ikici yöntem ise, GATE'in sunduğu araçlarla kullanıcı.



Şekil 4.5 Semantic Tagger ek açıklamalar

bazı modüller üzerinde yaptığı değişiklikler ve geliştirmelerdir. Bu yöntemde kulacını yaptığı geliştirme PR (işleme kaynağın)'da üzerinde olacaktır, bu çalışmada daha önceki bölümlerde bahsedildiği Arapça dille ilgili Bilgi Çıkarım ve Arapça dil dilbilgisi kurallarına dayanarak yapılmıştır.

4.6. ANNIE'de Yeni Bir Uygulama Oluşturmak

Genellikle her yeni bir uygulamada, ANNIE'nin bütün bileşenlerini kullanır. Temel olarak ANNIE bileşenlerinden Tokeniser, sentence splitter ve orthomatcher dil ve uygulama olarak bağımsızlardır. Başka bir değişle her hangi bir dile bağlı değildir bütün konuşulan dilleri üzerinde çalışmaktadır. Ayrıca sadece bir sistem çalışmıyor, bütün Dİİ sistemlerde çalışmaktadırlar[79]. POS Tagger ise dil olarak bağımsız değil ama uygulama olarak bağımsızdır. Burada başlangıç noktası olarak Gazetteer ve JAPE en çok fazla önem taşımaktadır. Bunlara dayanarak bu tez çalışmada bu iki modül üzerine çalışılmıştır.

4.6.1. Gazetteer geliştirme

Gazetteer dosyalarını değiştirmek için kullanılan paltform, Microsoft Windows işletim sistemi sunduğu notepad uygulaması veya GATE ekibi tarafından hazırlanan Unicode Editor aracı kullanılmıştır. Gazetteer dosya oluşturulduktan sonra, dosyayı UTF-8 kodu olarak kayıt olması gerekmektedir. GATE'te her dilin kendine ait bir Gazetteer indeks dosyası vardır, bu dosya içinde Gazetteer'in içerdiği liste özelliklerini barındırıyor. Genelde bu dosya list.def olarak adlandırılır, daha önceki bölümde bahsedildiği gibi her Gazetteer dosyanın özelliğe vardır bu özellikler üçe ayrılır: majorType (Büyük Tipi), minorType (Yandal Tip) ve language (Dil). Aşağıdaki verilen örnekte ilk sütün liste adına işaret etmektedir. İkinci sütünse MajorType bölümüne işaret etmektedir, bu bölüm liste türünü göstermektedir. Üçüncü sütün MinorType bölümüne işaret ediyor bu bölüm ise listenin yan özelliklerini tanımaktadır. Ve en son dördüncü sütünse Dili bölümüne göstermektedir.

```
monthen.lst:date:month:en
monthde.lst:date:month:de
season.lst:date:season
```

Bu özellikler majorType, minorType ve language genelde yeni bir uygulama yazıldığında JAPE'te gramer kurallarında kullanılır. Bu özelliklerin yararı Annotation (ek açıklamalar) ulaşmak için Gazetteer listelerde bulunan sözcükleri, metin üzerinde eşleştirerek bu özelliklerine kullanılır. Örneğin; metinde bulunan gün adı çıkarmak için önce Gazetteer indeks dosyada bulunan özellikleri araçlığıyla ve JAPE'te "Lookup" komutlarla önce MajorType "Tarih" , MinorType "Gün" ve dil eşleştirdikten sonra Gazetteer liste gün dosya çağırılır. Bu özelliklerinde başka yararı ise eğer sadece MajorType "Tarih" olsaydı o zaman birden çok Gazetteer liste dosya beraber çağırılırdı bu da sistemin yavaşlamasına neden olurdu. Çünkü Ay, hafta, mevsim ve özel yıl isimler Tarih kategori olarak ayrılmaktadırlar. Ayrıca bu özellikler Dil özelliği Gazetteer listesi hangi dile ait olduğunu göstermektedir.

Örneğin : `monthen.lst:date:month:en`
bu örnekte Gazetteer liste ismi, MajorType "date", MinorType "month" ve en son kısım dil, bu örnekte İngilizcedir.

4.6.2. JAPE (Java Annotation Patterns Engine)

JAPE veya (Java ek açıklamalar desen motoru). JAPE eşleştirme işlemi için kullanılan bir dildir. JAPE gramerleri bir yürütülür programlar kümesinden ibarettir [80]. Genelde JAPE grameri iki tarafı vardır; sol (LHS) ve sağ (RHS) denilir. LHS ve RHS, "-->" işaretlerle ayrılmaktadır. Gramerin LHS tarafı ek açıklamaları tanıtmak için desenler, düzenli ifade ve operatörler (*, ?, +, |) kullanılır, operatörler anlamları örneklerde gösterilmektedir. LHS tarafında, her desen parantez arasında tanımlanır ve her desenin sonunda bir etiketi (Label) vardır. Bu etiket araçlığıyla LHS'te bulunan ek açıklamalar bilgileri RHS tarafına transfer edilir, tespit edilen desenleri üzerinde yapılan eylemler ve ek açıklamalar manipülasyon ifadeler içerir.

Örneği:

{Token.string=="of"} Bu desen dizi metinin ifade etmektedir

{Token.kind==number} Bu desen ise Token özelliği tanımaktadır

{Lookup.minorType==month} bu desen ek açıklamalar için Gazetteer'terden sözcük kategorisini tanıtmaktadır.

{Lookup.majorType==location}:Location, location olarak etiketlenen bir desen.

Aşağıda verilen örnekte, LHS tarafında tanımlanan desen, RHS tarafta Location olarak etiketlenmiş ayrıca yeni bir ek açıklama türü türetmiş, bu ek açıklama adı Enamex türü ve değeri Location ayrıca kural özelliği GazLocation'dir[80].

```
Rule: GazLocation
(
{Lookup.majorType==location}
):Location-->
   :Location.Enamex={kind=" location", rule=GazLocation}
```

Yeni bir uygulama yazmak için JAPE gramerinde Macro'lar kullanılır. Macro, bir kuralın düzenli ifadelerle genel olarak tanımlama içeren bir kural parçasıdır. Bu kural parçası genelde programın başına yazılır ve daha sonra programın farklı yerlerinden çağırılır. Macroların yararı ise birden fazla kullanılan kuralların yeniden yazılmamasıdır. Buna ilaveten bazı Macro'lar diğer Macrolar tarafından çağrılabilinir.

Gramer kuralları temel olarak iki türe ayrılır. Birinci tür kurallar, Gazetteer "Lookup" arama motoru kullanmaz. Bu tür kurallar oldukça basit formatlarla tanımlanır. Belirsizlik için daha az potansiyelleri vardır. Aşağıdaki verilen örnekte kural basit bir formatla ifade edilmiştir; çünkü IP adresi tanıtmak için tek kural yeterlidir.

```
Rule:IPAdress
(
{Token.kind==number}
{Token.string=="."}
{ Token.kind==number }
{ Token.string=="."}
{ Token.kind==number }
{ Token.string=="."}
{ Token.kind==number }
):ipAddress-->
   :ipAddress.Address={kind="ipAddress"}
```

İkinci tür kurallar daha sık kullanılır, bu kurallar Gazetteer listeleri kullanmaktadır ve daha geniş olasılık kapsar. Örneğin bir tarih tanımlamak için farklı tarih yazma format vardır.

Örneğin: "The late '80s", "Monday", "99 BC", "mid-Noveber","1980-81" Ve "from March to April".

Bu tür kurallar belisizlik için daha büyük potansiyel vardır. Bu durumlarda, belirsizliği azaltmak için sık kullanılan formatlar kullanılır. Örneğin yıllarla ilgili sıkça kullanılan bağlamlar vardır, metinde yıllar kolayca bulunması için bu tür bağlamlar aşağıdaki verilen örnekte kullanılmıştır.

```
Rule:YearContext
({Token.string=="in"}|{Token.string=="by"})
(YEAR) // bu bir Macro daha önce programın başında
tanıtılmış
:date--> date.Timex={kind="date",rule="YearContext"}
```

İkinci tür kurallarda sıra ve öncelikle yürütülmesine önem verilmektedir. Aşağıda verilen örnekte Gazetteer liste kullanılıyor ayrıca bu kuralın aynı gramer dosyada başka kurallara göre önceliği (Priority) 20 olarak gösteriliyor. Kuralın öncelik Sayısı ne kadar büyükse o kuralın uygulaması daha yüksektir. Önceliklik (Priority) değeri genelde kullanıcı tarafında verilir, Eğer kurallara öncelik değeri verilmezse sistem tarafında varsayılan olarak bütün kurallara öncelik değer (-1) verilir.

```
Rule: SportsCategory
Priority: 20
( {Lookup.majorType == "Sports"}
): label -->
:label.Sport = {rule= "SportsCategory" }
```

JAPE Gramerinde kuralları kolayca uygulaması için birkaç kontrol tarzı vardır: Once, First, All, Brill ve Appelt, bu kontroller gramer dosya başına yazılır.

Kontroller genelde daha fazla ve geniş olasılıklıları kapsayan kurallarda kullanılır. Bu çalışmada Arapça bilgi çıkarım için kullanılan kontroller, Brill ve Appelt. Appelt kontrol çalışma mekanizması gramerde ilk kural metinle eşleşirse o kural bütün doküman üzerinde uygulanacaktır. Örneğin birden fazla tarih yazılış formatı vardır ve gramer dosyasında bütün tarih formatları kurallar geçiyor. Bu durumda eğer ilk kural, metinde ilk tarih formatıyla eşleşirse bütün dokuman üzerine sadece o kural uygulanır başka kuralları uygulanamaz sonuçta eğer metinde birden fazla tarih formatı geçiyorsa sadece bir tarih formatı bulunur diğer tarih formatlar bulunmaz. Bu kontrolde öncelilik ” Priority” komutu önemlidir çünkü eğer örneğin bir dokumanda sık kullanılmayan bir tarih formatı varsa ve bu format ilk başta bir kuralla eşleşirse sonuç itibariyle diğer sık kullanılan tarih formatların bulunmaz, o yüzden bu kontrolde öncelik, önce sık kullanılan formatlara verilmektedir.

Diğer kontrol tarzı ise Brill, bu kontrol ayni dokuman üzerine birden fazla kural uygulamasına izin veriyor. Bu kontrolde öncelik komutuna gerekmez çünkü bütün kurallar üzerine eşleşme yapılıyor.

4.7. Uygulama İçin Yapılan Çalışmalar

Yeni bir uygulama için, bu tez çalışmasında birçok işlem yapılmıştır. Bu işlemlerin çoğu örneklerle gösterilmektedir.

4.7.1. Arapça tokenizer kuralları

Arapça Tokenizer’ın diğer dillerle farklılıkları vardır. Örneğin Arapçada büyük veya küçük harflerin olmaması, yazının sağdan sola olması ve noktalama işretlerinin ayrı olmas. Bu tez çalışmasında sözcükleri birbirlerinden ayırmak için İngilizce Tokenizer kurallarından esinlenerek yeni bir kural yazılmıştır. Aşağıdaki örnekte, verilen Tokenizer kuralı İngilizce diline aittir.

Rule: İngilizce Tokenizer

```
"UPPERCASE_LETTER" (LOWERCASE_LETTER)* >
oken;orth=upperInitial;kind=word;

"UPPERCASE_LETTER" (UPPERCASE_LETTER)+ >
Token;orth=allCaps;kind=word;

"LOWERCASE_LETTER" (LOWERCASE_LETTER)* >
oken;orth=lowercase;kind=word;
```

Rule: Arapça Tokenizer

```
("UPPERCASE_LETTER”|”LOWERCASE_LETTER")(LOWERCASE_LETTER|
UPPERCASE_LETTER)* > Token;orth= mixedCaps;kind=word;
("UPPERCASE_LETTER”| “LOWERCASE_LETTER ")
(UPPERCASE_LETTER| LOWERCASE_LETTER)+ > Token;orth=
mixedCaps;kind=word;
```

Bu kuralda yapılan değişiklik ile Arapça metinlerde ilk harfin büyük veya küçük olması dikkate alınmaz. "UPPERCASE_LETTER | LOWERCASE_LETTER" deseni, parçada bulunan “|” (OR) operatörü kullanarak bu işlem gerçekleştirir. İngilizce Tokenizer kurallında verilen (LOWERCASE_LETTER)* deseni ilk harften sonra bir veya birden fazla küçük harfin gelebileceğini gösteriyor. Arapça kurallında bunu iptal etmek için böyle bir değişiklik yapılmıştır; (LOWERCASE_LETTER|UPPERCASE_LETTER)* . İlk harften sonra bir veya birden fazla dizi harfler gelir. Bu dizide büyük veya küçük harfin dikkate alınmaması için “|” (OR) operatörü kullanılmıştır. Bu yapılan işlemler kuralın LHS tarafında gerçekleşiyor. Kuralın RHS tarafında yapılan değişiklikler ise sadece İngilizce Tokenizer kurallında bulunan “orth=upperInitial” eylemini, “orth= mixedCaps” olarak değiştirmek olmuştur. Bu kural Arapça ve İngilizcede birinci harfin ardından hiç harf gelmemesini veya birden fazla harfin gelmesini temsil etmektedir.

("UPPERCASE_LETTER"(UPPERCASE_LETTER)+>Token;orth=allCaps;kind =word;) bu kural İngilizce Tokinezer’da ilk harfin büyük ve sonrasındaki harflerinde büyük olması gerektiğini göstermektedir. Bu kural üzerinde değişiklik yaparak aşağıda verilen Arapça Tokinezer kuralı elde edilmiştir.

("UPPERCASE_LETTER"| "LOWERCASE_LETTER ")
(UPPERCASE_LETTER| LOWERCASE_LETTER)+ > Token;orth=
mixedCaps;kind=word;

Arapçada bu kural birinci kuraldan farkı, ilk harften sonra bir veya birden fazla harf gelmesidir. Ayrıca "orth=allCaps" bu eyleminin "orth= mixedCaps" ile değişmesidir.

4.7.2. Gazetteer listeleri

Bu çalışmada, yeni bir Gazetteer listesi oluşturmak için yeni bir indeks listesi oluşturulmuştur. Bu indeks dosyası list.def olarak adlandırılmıştır ve standart olarak bütün dillerde ayni adı taşımaktadır. Gazetteer listeleri oluşturmak için, Microsoft'un sunduğu Notepad uygulaması kullanılmıştır. Ayrıca GATE sisteminde var olan metin düzenleme hizmeti kullanılmıştır. İndeks dosyası oluşturma esnasında liste adı ve kategoriler, Arapça dil kaynaklarına dayanılarak hazırlanmıştır. Aşağıdaki verilen örnekte bu indeks çalışması gösterilmektedir:

```
city.lst:location:city:arabic
city_world.lst:location:city:arabic
country.lst:location:country:arabic
country_world.lst:location:country:arabic
currency.lst:money_unit::arabic
date_key.lst:date_key::arabic
days.lst:date:day:arabic
facility.lst:facility::arabic
female_names.lst:person:female:arabic
location_other.lst:location:other:arabic
male_names.lst:person:male:arabic
months.lst:date:month:arabic
monuments.lst:facility:monument:arabic
mountains.lst:location:mountain:arabic
oceans_seas_islands.lst:location:other:arabic
numbers.lst:numbers:arabic
hour.lst:hour:arabic
organisations.lst:organisation::arabic
percent.lst:percent::arabic
places.lst:location:other:arabic
rivers.lst:location:river:arabic
surnames.lst:person:surname:arabic
time.lst:time:arabic
```

```
time_ampm.lst:time_ampm
time_key.lst:time_key:arabic
titles.lst:person:title
year.lst:date:year:Arabic
```

Yukarıda verilen örnekte her satır bir Gazetteer dosya ismini taşıyor ve bu dosyalar içeriğin ismine göre kategorize edilmiştir.

Örneğin ilk satırda verilen örnekte:

```
city.lst:location:city:arabic
```

Arap ülkelerinde bulunan şehir adlarını içermekte ve bu isimler "Location" kategorisi olarak ayrılmaktadır. Ayrıca Arabic ile bir dil özelliği belirtilmiştir. İndeks dosyası oluşturduktan sonra Gazetteer liste işlemleri başlar. Bu işlem, Gazetteer listelere tek satırda isimler girilmesidir. Bu çalışmada girilen isimler, Arapça sözcük kaynaklarına dayanarak yazılmıştır. Geliştirilen Gazetteer listelerinde, birçok eklemeler yapılmıştır, bu eklemeler genelde isimler üzerindedir. Daha önceki bölümlerde bahsedildiği gibi, Arapçada büyük ve küçük harflerin olmamasından dolayı isimlerin yazılışı metinde yerden yere değişir. Örneğin; Irak'ta bir şehir adı olan "Süleymaniye", iki şekilde yazılabilir. Birincisi "سليمانية" diğeri ise "السليمانية" 'dir. Bu iki ismin farkı"ال" (Alf Lam), önek olmasıdır. Bu nedenle Arapça kaynaklardan yararlanarak yapılan uzun bir çalışmadan sonra, birçok isimlere önek ekleyerek Gazetteer listeler kullanıma hazır hale getirilmiştir. Buna benzer çalışmalar tarihlerin yazılış formatları üzerinde de yapılmıştır. Örneğin, Çoğu Arap ülkesinde hem Hicri hem de Miladi takvim kullanılır. Dolayısıyla iki tarih bir arada birkaç formatla yazılır. Aşağıda verilen örnekte tarih formatı bazen rakamla, bazen yazıyla, bazen de bağlamla yazılmaktadır.

أربعاء, 10/7/00
أربعاء, 10/كانون اول/00
أربعاء, 10 كانون اول, 2000
أربعاء, عاشر من كانون اول,2000
أربعاء, كانون اول, 10, 2000
أربعاء 10 كانون اول 2000

Bu çalışmada, sıkca kullanılan tarih formatları yazılmıştır. Ayrıca farklı formatları da internet üzerinde Arapça yayın yapan haber, gazete, kurumlar vb. sitelerden alarak formatları Gazetteer listelere girilmiştir. Gazetteer dosyaları üzerinde yapılan başka çalışmalar ise bağlamlar üzerinedir. Örneğin, metinlerde saat formatını fark etmek için saat değerinden önce ve sonra "في", "من", "الساعة", "صباحا", "مساءا", "ظهرا","PM","AM" vb. gibi bağlaçlar gelir. Yalnız bu bağlaçlar başka anlamlarda da olabilir. Örneğin "في البيت" " evde" kelimesindeki "في" "fi" burada Turkçedeki "de, da" anlamında kullanılmıştır. Dolayısıyla Gazetteer listeleri, indeks dosyalarında bulunan kategorilere göre ayrılmıştır.

GATE platformunda Gazetteer PR işleme kaynağını yükledikten sonra sadece dillerle ilgili indeks dosyalar sisteme yüklenir. Yukarıda verilen örnekte Gazetteer dosya kategorileri önemli rol almaktadır çünkü JAPE'te kodlar bu kategorilere göre yazılmaktadır.

4.7.3. JAPE kuralları

Bu tezde JAPE kullanarak yapılan çalışmalar aşağıda verilen örnekle anlatılmaktadır.

```
Rule:   Location1
1//     Priority: 25
2//    (
3//
({Lookup.majorType == loc_key, Lookup.minorType == pre}
4//    {SpaceToken})?
5//    {Lookup. majorType == country}
6 //   ({SpaceToken}
7//
Lookup.majorType == loc_key, Lookup.minorType == post})?
8//    )  :locName -->
9//:locName.Location = {kind = "location", rule = "Location1
"}

Rule: Location2
1//   Priority: 20
2//  ({Lookup.majorType == country}) :location  -->
3// :location.Name = {kind = "location", rule=GazLocation}
```

Yukarıda verilen örnekte, iki kural bir dokuman üzerinde uygulanacaktır. Sisteme girilen dokumanda "بحر الصين" (Çin Denizi) diye bir sözcük grubu geçmektedir ve bunu çıkarmak için gerekli işlemler yapılacaktır. Daha önceki Gazetteer çalışmalarda, "الصين" (Çin) sözcüğü Gazetteer'de listesinde "Country" olarak ve "بحر" (deniz) sözcüğü ise Gazetteer listesinde "Loc_key" bağlam olarak tanımlanmıştır. Sisteme Gazetteer listeler ve JAPE kuralları yüklendikten sonra, sistem "Location1" kuralını uygulayacaktır; çünkü "بحر الصين" ilk kuralla eşleşmektedir. Bununda sebebi "Priority" değerinin daha yüksek olmasıdır. Bu durumda "Location2" kuralı uygulanmayacaktır ve yazılmasına gerek yoktur; çünkü JAPE'te verilen operatörler aracılığıyla birçok kural bir kuralda toplanabilinir. Yukarda verilen birinci kuralın (Location1) 4 ve 7. satırlarında, bir Gazetteer ve Token deseni geçmektedir. Bu iki modül, Gazetteer ve Token, yukarıda bahsedildiği gibi her cümlede geçmektedir. Yalnız burada önemli olan soru işareti"?" dır. Bu operatör ve benzerleri bu tez çalışmasında kullanılmıştır. Şimdi Location1 kurallarına bakılırsa, 2, 3 ve 4. satırlarındaki kurallar, metinde bağlam olup olmadığını ve bağlamın ardından boşluk gelmesi gerektiğini kontrol etmektir. Bu durumda "بحر" sözcüğü ardından boşluk geliyor yani bu kural metinde bulunan sözcükle eşleşiyor.

Operatörün rolünü daha detaylı bir şekilde anlamak için, farz edelim ki eğer "Loc_key" veya "TokenSpace" desenlerinden herhangi birisi olmasaydı kural nasıl işlerdi? Yukarıda "Location1" kuralının 4. satırının sonunda bulunan soru işaretinin "?" anlamı, eşleşme işleme sırasında bağlam veya boşluğun olmasına göre bir ya da sıfır değer döndürmesidir. Yani eğer eşleşme sırasında bağlam ve ardından boşluk bulunmazsa bu desenle ilgili hiçbir işlem yapılmayacaktır. O zaman eşleşme işlemi "Location1" kuralında bir sonraki desene geçer yani 5 satıra geçer. Bu satırda Gazetteer deseni olarak yazılmış "الصين", "Çin" sözlüğü eşleştiriyor ve kategori "Country" olarak bulunuyor. İşlem devam ederek, 6. satırda boşluk olmasını kontrol ediliyor. Burada son iki desen olan 5 ve 6. satıra bakılırsa, satır sonlarında herhangi bir operatör bulunmamaktadır. Satır aralarında herhangi bir operatör olmadığı için bu iki desenin ardı ardına gelmesi gereklidir eğer bu şart olmazsa "Location1" kuralı iptal olur ve hemen diğer eşleşme işlemi olan "Location2" kuralına geçer.

Yukarıda verilen örneğe göre eşleşme işlemi devam ediyor ve 7. satıra geçiyor. Burada "الصين" "Çin" sözcüğü ardından herhangi bir bağlam olup olmadığını kontrol ediyor. Bu örneğe göre bağlam bulunmuyor. Ve son olarak satır 8'de, bu kurala "locName" etiketi veriliyor. Burada JAPE gramer kurallarında ve daha önceki bölümlerde bahsedildiği gibi LHS tarafı bitiyor ve RHS tarafı başlıyor. RHS tarafıysa bu kural üzerinde yapılacak eylemleri gösteriyor. Burada etiketlenen "locName", "Location" 'den türüyor ve verilen özellikler, tür "Location" ve kural "Location1" olarak değerlendiriliyor. Bunun anlamı "الصين بحر" "Çin Denizi" sözcüğü ve benzerleri "locName" olarak sistem tarafından ayrılıyor. Bu örnekte iki konuya değiniliyor; birisi operatörlerin rolü, diğeri ise JAPE kurallarda " Priority" öncelilik komutudur. "Priority" rolü aşağıda verilen paragrafta anlatılmıştır.

Yukarda verilen örneğe dayanarak, eğer metinde sadece "الصين" sözcüğü olsaydı kurallar nasıl çalışırdı? Bu durumda iki kuralında uygulanması mümkün, çünkü iki kuralda eşleşme gerçekleşiyor. Birinci kuralda"Location1", 4 ve 7 satırlarda soru işaretler "?" bulunuyor ve sadece "الصين" sözcüğü kontrol ediyor. Bu yüzden bu kuralın uygulaması mümkündür. Şimdi ikinci kurala "Location2"ye bakıldığında, aynı eşleşme işlemi gerçekleşiyor. Bu durumda hangi kural uygulanacak sorusu sorulursa; kural "Location1" uygulanacaktır. Çünkü bir gramer dosyasında birden fazla kural, metinde ayni sözcük veya cümleyle eşleşiyorsa, bu durumda sistem "Priority" değerine bakacaktır ve hangi kuralın öncelik değeri yüksekse o kuralı uygulanacaktır[80].

4.8. Kullanıcı Yöntemiyle Arapça Örnek Uygulama

Kullanıcı yöntemi aşağıda verilen bir örnekle gösterilmektedir.

سيعرض على القناة الأرضية ثمانية مسلسلات هي : مسلسل (انتقام الوردة) **في الساعة الحادية عشرة والنصف قبل الظهر** , مسلسل (الهشيم) **الساعة الثالثة بعد الظهر** , مسلسل (باب الحارة) **في الساعة السادسة وأربعين دقيقة**, مسلسل (الوزير وسعادة حرمه) في الساعة السابعة وخمس وثلاثين دقيقة, (صدى الروح) في التاسعة والنصف مساءً ,مسلسل (فسحة سماوية) في الحادية عشرة والنصف مساءً , مسلسل

(وشاء الهوى) في الثانية عشرة وخمسين دقيقة عدا يوم الخميس , أما مسلسل (مرايا2006) فسيعرض في الساعة الثانية صباحاً.

Yukarıda verilen örnek paragrafta, kalın yazı olarak işaretlenen vakitleri çıkarmak amacıyla uygulama aşağıda verilen aşamalarla geçmektedir.

4.8.1. Tokenizer işlemi

Bu aşamada, Tokenizerın standart kurallarından yararlanarak, sözcükleri ayırmak için daha önce Tokinezer çalışmasında geliştirilen kurallar kullanılmıştır. Tokeniser işlemi, yukarıda verilen paragrafta uygulanmış olup, aşağıda verilen örnekteki gibi "token"lere ayırmaktadır:

Token, string=" في ",	kind=word,	length=2, orth= mixedCaps
Token, string=" الساعة",	kind= word,	length=6, orth= mixedCaps
Token, string="الحادية",	kind=word,	length=7, orth= mixedCaps
Token, string="عشرة",	kind=word,	length=4, orth= mixedCaps
Token, string="و",	kind=word,	length=1, orth= mixedCaps
Token, string="النصف",	kind=word,	length=4, orth= mixedCaps
SpaceToken,string=" ",	kind=space,	length=1
Token, string="قبل",	kind=word,	length=3, orth= mixedCaps
Token, string="الظهر",	kind=word,	length=5, orth= mixedCaps
Token, string="الثالثة",	kind=word,	length=7, orth= mixedCaps
Token, string="بعد",	kind=word,	length=3, orth= mixedCaps
Token, string="خمسين",	kind=word,	length=5, orth= mixedCaps
Token, string="اربعين",	kind=word,	length=6, orth= mixedCaps
Token, string="دقيقة",	kind=word,	length=5, orth= mixedCaps
Token, string="التاسعة",	kind=word,	length=6, orth= mixedCaps
Token, string="مساءاً",	kind=word,	length=6, orth= mixedCaps
Token, string="صباحا",	kind=word,	length=5, orth= mixedCaps

4.8.2. Gazetteer işlemi (List lookup)

Bu aşamada eşleşen sözcükleri, "Lookup" arama motoruyla, saat terimlerini paragrafta bulma amacıyla Gazetteer listelerinde aranılır. Daha önce

Gazetteer çalışmalarında zamanlarla ilgili kategoriler oluşturulmuştur. Bu örnekte zamanla ilgili Gazetteer listeleri aşağıda verilen dosya listelerinde aranmaktadır:

```
1// numbers.lst:numbers:arabic
2// hour.lst:hour:arabic
3// time.lst:time:arabic
4// time_key.lst:time_key:arabic
```

Birinci liste Arapçada yazılı olarak rakamları içermektedir. yukarıda verilen paragrafta saatlerle beraber kullanılan " ثلاثة,اثنان,واحد " yani " bir, iki, üç" rakamları ve devamı bu listede yer almaktadır. Örneğin "أربعين" bu sözcük "kırk" anlamına gelmektedir ve paragrafta dakika olarak geçmektedir. Yazılı olarak rakamları gösteren sözlükler genel bir listede toplanılmıştır. İkinci liste ise Arapçada yazılı olarak saatleri içerir. Burada saat listesi yerine niye number.lst kullanılmadı sorusu sorulursa?, Çünkü saatlerle ilgili özel yazı tipleri vardır ve bu yazı tipleri sadece saatlerle kullanılır. Örneğin "الثالثة " bu sözcük önek "elif lam" a ve sonek "te" alarak "üç"ü temsil etmektedir. Ayrıca listede bulunan sözcüklerin sayısı, rakamlar listesine göre daha azdır. Bu sayede aranan sözcük sistem tarafından daha çabuk bulunur. Üçüncü liste saatlerle beraber kullanılan bağlamlardır. Örneğin bu listedeki "dakika, saniye, çeyrek vb." gibi ifadeler saatlerle ve başka aramalarda da kullanılır. Dördüncü liste ise ayni zamanları ifade etmekle kullanılan sözcükleri içermektedir. Örneğin "bu gün, bu sabah, bu akşam, gece, öğlenden sonra vb.". Yukarıda verilen liste adları bir sonraki aşamada JAPE kurallarında kullanılmıştır.

4.8.3. JAPE gramer kuralları

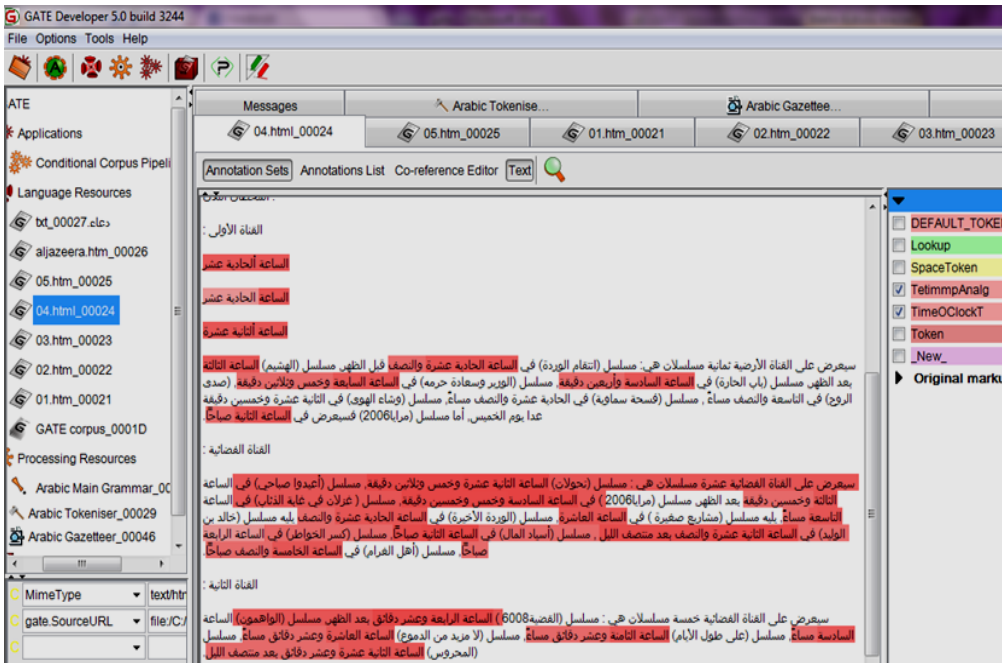
Bu aşamada, önce bütün genel kuralların desenleri Macrolar halinde yazılmıştır. Sonra bu Macrolar kurallar içinde kullanılmıştır. Aşağıda verilen örnekte, saatlerle ilgili Macro'ları göstermektedir:

```
Macro: COMMA
({Token.string == ","})

Macro: TIME_KEY
```

```
// “bu akşam” `هذا المساء`
(
 {Lookup.majorType == time_key}
)
Macro: TimeOClock
// “saat on”  `الساعة العاشرة `

(
({Token.string=="الساعة"}|{Token.string=="ساعة"})
 {Lookup.majorType == hour}
)
Macro: TMniutSe
//واربعون ثانية  "ve kırk dakika "
(
({Token.string==" و"})*
 {Lookup.majorType == numbers}
({Lookup.majorType == numbers})*
 {Lookup.majorType==time}
)
Macro: TimeAnalogueM
// الساعة الحادية عشرة والنصف”saat on bir buçuk”
(
(TimeOClock)
({Token.string==" و"})*
( {Lookup.majorType == numbers})*
( {Lookup.majorType==time})*
(TMniutSe)*
(TIME_KEY)*
)

Yukarıda verilen paragrafta saat olarak yazılan vakitleri
bulmak için aşağıda verilen kurallar kullanılmıştır.

Rule: TimeOClockR
Priority: 25
(
(TimeOClock)
)
:time
-->
 :time.TimeOClockT={kind = "positive", rule = "TimeOClockR"}

//Time Rules

Rule: TimeAnalogue
// الساعة الحادية عشرة والنصف
Priority: 50
```

```
(
(TimeAnalogueM)
)
:time
-->
:time.TetimmpAnalg = {kind = "positive", rule =
"TimeAnalogue"}
```

Bu üç aşama gerçekleştikten sonra uygulamayı çalıştırmak için GATE platformu kullanılmıştır. Sisteme Tokenizer, Gazetteer listeleri ve JAPE'le yazılan kurallar yüklendikten sonra söz konusu olan bilgi çıkarım işlemi, paragraf üzerinde uygulanmış olup, GATE platformunda elde edilen sonuçlar (Şekil 4.6)'te gösterilmektedir.



Şekil 4.6 GATE Platformda Uygulanan Tokenizer, Gazetteer ve JAPE Kuralları

5. Sonuç ve Öneriler

Bilgiye erişimde Internet'in, Internet'teki bilgiye erişimde ise arama motorlarının önemi tartışılmazdır. Ancak arama motorlarından sadece bilginin adresi öğrenilebilmektedir. Aradığımız bilgiye tam olarak ulaşabilmemiz için, site içinde yine bir arama yapmamız gerekebilmektedir. Buna ek olarak aradığımız bilgiyi belirtmek için kendi dilimizi (doğal dilimizi) değil, bilgi kaynağı olan sitenin kullandığı dilden anahtar kelimeler seçilmelidir. Bu anahtar kelimeler seçmek için önce metinlerden gerçek bilgiyi elde etmek gerekiyor.

Bu çalışmada, Arapça metinlerde Bilgi Çıkarım için yeni bir uygulama yazılmıştır. Bu uygulama GATE platformu üzerinde geliştirilmiştir. Bu uygulama üzerinde iki ayrı yöntem uygulanmıştır.

Birincisi Otomatik Yöntem diğeri ise GATE platformunun sunduğu araçlarla Kullanıcı Yöntemidir. Otomatik yöntem Arapça metinler üzerinde pek etkin bir sonuç vermediği için Kullanıcı Yöntemi kullanılmıştır. GATE'in sunduğu araçlar sayesinde bu yeni uygulama Arapça metinler üzerinde daha etkin bir sonuç elde edilmiştir. Bu uygulama, daha önce yapılan Arapça Bilgi çıkarım uygulamalarında farkı ise, bu uygulama bütün işletim sistemlerinde çalışabilmesidir. Ayrıca bir Plugin olarak, başka Bilgi Çıkarım uygulamalarının da eklenip çalıştırılabilmesidir.

Geliştirmiş olduğumuz bu uygulamada karşılaşılan sorunlar ise Gazetteer listelerin uyguluma için yeteri kadar veritabanı olarak hazır olmaması ve haber sunan birçok yayın sitelerinden güncellenen yeni modern Arapça terimlerin eksik olmasıdır, bu uygulamayı belli bir noktadan sonra geliştirilen GATE uygulamasının ilerleyememesine neden olmaktadır. Eğer sürekli yeni terimler veya bağlamlar Gazetteer listelerine eklenirse, bu uygulama tam verimli bir performansta çalışır. GATE'ten alınan veriler Arapça'da bir arama motoru oluşturmak için, akıllı veri olarak kullanılabilir.

KAYNAKÇA

[1] The world as information: overload and personal design Yazar: Robert D. Abbott, http://books.google.com.my/books .

[2] İnternet Ve Bilgi Kirliliği,http://www.kritize.net/yazarlar/burak-baskan/213-internet-ve-bilgi-kirliligi

[3] Hoover's Handbook of Private Companies 2005 Yazar: Hoover's Incorporated, http://books.google.com.my/books?id=xoAT77z3mv0C&printsec=frontcover&hl=tr#v=onepage&q=&f=false

[4] Alvin_Toffler, http://en.wikipedia.org/wiki/Alvin_Toffler, http://www.tnewfields.info/Articles/str.htm

[5] BİLGİYE ERİŞİM VE BİLGİ KİRLİLİĞ, http://www.netpano.com/makale/?makale=718.

[6] Akarsu, Bedia (1998), Felsefe Terimleri Sözlüğü, İnkılap Yayınları

[7] Aksan, Doğan (1978), Anlambilimi ve Türk Anlambilimi, Ankara, Ankara Üniversitesi Dil ve Tarih-Coğrafya Fakültesi Yayınları, 2. Baskı, 199 s.

[8] Cardoso, J., Sheth, Amit (2006). Semantic Web Services, Processes and Applications. Springer. ISBN 0-387-30239-5.

[9] Foundations of intelligent systems: 14th international symposium, ISMIS 2003 .Yazar: Ning Zhong, http://books.google.com/books

[10] http://tr.wikipedia.org/wiki/Tim_Berners-Lee, Kim, W. and J. Seo. 1991. Classifying schematic and data heterogeinity in multidatabase systems. IEEE Computer 24(12): 12-18.

[11] http://www.w3.org/People/Berners-Lee/ Berners-Lee, T., Hendler, J. and Lassila O. 2001. The Semantic Web: A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities. Scientific American, 284 (5): 34-43.

[12] Hawke, S.: How the Semantic Web Works, 04/2002, http://www.w3.org/2002/03/semweb/

[13] Guha, R. ve McCool R.: TAP: A Semantic Web Platform, 2003, http://tap.stanford.edu/tap.pdf

[14] SEMANTİK WEB TEKNOLOJİLERİ, ab.org.tr/ab08/bildiri/14.doc

[15] XML NEDİR?, www.emo.org.tr/ekler/150ccc6069bea6b_ek.doc

[16] A. Maedche and S. Staab, "Discovering Conceptual Relations from Text," Proc. European Conf. Artificial Intelligence (ECAI-00), IOS Press, Amsterdam, 2000, pp. 321-325.

[17] Ontology learning for the semantic Web Yazar: Alexander Maedche, http://books.google.com.tr/books.

[18] FENSEL D., HENDLER J., LIEBERMAN H., WAHLSER W. (2003): "Spinning the Semantic Web", The MIT Press.

[19] A. Maedche and S. Staab, "Discovering Conceptual Relations from Text," Proc. European Conf. Artificial Intelligence (ECAI-00), IOS Press, Amsterdam, 2000, pp. 321-325.

[20] http://www.w3.org/People/Berners-Lee/ Berners-Lee, T., Hendler, J. and Lassila O. 2001. The Semantic Web: A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities. Scientific American, 284 (5): 34-43.

[21] WEB'ĠN GELECEĞĠ: ANLAMSAL WEB, eab.ege.edu.tr/pdf/8_1/C8-S1-M11.pdf

[22] Ontology learning for the semantic Web Yazar: Alexander Maedche, http://books.google.com.tr/books

[23] Savtek 2006 3. SAVUNMA TEKNOLOJ_LER_KONGRES, www.kho.edu.tr/...Bildirileri/SAVTEK2006_Cilt2_Degerlendirme_Bildirileri.pdf

[24] Akyokuş S., "Anlamsal Web, Anlamsal Web Dilleri ve Araçları", http://vdb.gib.gov.tr/edirnevdb/kultur/ppt/anlamsal_web_rdf_dc_owl.ppt, (07.03.2007).

[25] Berners-lee T., Hendler, J., Lassila, O. (2001): "The Semantic Web",Scientific American, vol. 184, no: 5, Mayıs 2001

[26] Trastour, D., Bartolini, C., Acstillo, J.G., (2001): "A Semantic Web Approach to Service Description for Matchmaking of Services", HP Company.

[27] Bilgi Çıkarımı (Information Extraction) ,http://www.bilgisayarkavramlari.com/2008/03/24/bilgi-cikarimi-information-extraction/

[28] Opportunities in Natural Language Processing, http://nlp.stanford.edu/~manning/talks/OracleNLP.pp

[29] Doğal Dil İşleme Nedir? (NLP) yapay zeka metotlarını kullanarak bilgisayar ile doğal dilde iletişim, http://elektroteknoloji.com/Elektrik_Elektronik/Teknik_Yazilar/Dogal_Dil_isleme_Nedir_NLP_yapay_zeka_metotlarini_kullanarak_bilgisayar_ile_dogal_dilde_iletisim.html.

[30] Foundations of statistical natural language processing By Christopher D. Manning, Hinrich Schütze, www.books.google.com

[31] What is linguistics, http://www.idb.hacettepe.edu.tr/turkish/the_department/what_is_linguistics/

[32] Doğal dil işleme, http://tr.wikipedia.org/wiki/Doğal_dil_işleme

[33] Engels, Robert & Bremdal, Bernt. Information Extraction: State-of-the-Art Report, Norway, 2000. INTERNET UZERĐNDE CALISAN BİR DOĞAL DİL İŞLEME UYGULAMASI:SORU CEVAPLAMA SİSTEMİ http://www.ce.yildiz.edu.tr/mygetfile.php?id=1570

[34] Blumberg, Robert & Atre, Shaku. The Problem with Unstructured Data, DM Review Magazine, February 2003, www.ug.bcc.bilkent.edu.tr/~acer/proposal.doc

[35] العالم العربي أو الوطن العربي الكبير, http://ar.wikipedia.org/wiki/الوطن_العربي

[36] Afro-Asyatik diller, http://tr.wikipedia.org/wiki/Afro-Asyatik_diller

[37] Ahlswede, T.E., and Evens, M., 1988,"Generating a Relational Lexicon from a Machine Readable Dictionary", InternationalJoumal of Lexicography, Vol. 1, No. 3, pp.214-237.

[38] Soudi, A., Bosch, A. V. D. & Nenmann, G. (Eds.) (2007) Arabic Computational Morphology: Knowledge-Based and Empirical Methods, Springer Netherlands.

[39] Thabet, N. (2004) Stemming the Qur'an. COLING 2004, Workshop on computational approaches to Arabic script-based languages.August 28,2004.

[40] Buckwalter, T. (2004) Buckwalter Arabic Morphological Analyzer Version 2.0. Linguistic Data Consortium,catalog number LDC2004L02 and ISBN 1-58563-324-0.

[41] Larkey Leah. S. AND CONNELL MARGRATE. E. (2001). Arabic information retrieval at UMass. In Proceedings of TREC 2001, Gaithersburg: NIST, 2001.

[42] Mikheev A., 1997. "Automatic Rule Induction for Unknown-Word Guessing"

[43] Zhang, B.-T. and Kim, Y.-T., 1990. "Morphological Analysis and Synthesis by Automated Discovery and Acquisition of Linguistic Rules". Proceedings of the 13 the International Conference on Computational Linguistics (COLING-90), pp. 431-435.

[44] Wolinski, F., Vichet, F., and Dillet, B., 1995. "Automatic Processing of Proper Names in Text". Proceedings of the 7 th Conference of the European Chapter of the Association for Computational Linguistics, Dublin, Ireland, pp. 23-30.

[45] Cowie, J., and Lehnert, W., 1996. "Information Extraction", Communications of the ACM, Vol. 39, No. 1, pp. 83-92.

[46] Paik, W., Liddy, E. D., Yu, E., and Mckenna, M., 1993. "Categorizing and Standardizing Proper Nouns for Efficient Information Retrieval", In B. Boguraev and J. Pustejovsky, eds, Corpus Processing for Lexical Acquisition, MIT Press, Cambridge, Mass, pp.44-54.

[47] Evens , M., Vandendorpe, J., and Wang Y., 1985, "Lexical-Semantic Relations in Information Retrieval". In Humans and Machines. S. Williams, (ed.), Ablex, Norwood, NJ, pp.73- 100

[48] Nutter, J. T., Fox, E., and Evens, M., 1990, "Building a Lexicon from Machine-Readable Dictionaries for Improved Information Retrieval", Literary and Linguistic Computing, Vol. 2, No. 5, pp.l-18.

[49] Abu-Salem, H., 1992, A Microcomputer Based Arabic Bibliography Information Retrieval System v, ith Relational Thesauri (Arabic-IRS). Unpublished Ph.D. Dissertation, Computer Science Department, Illinois Institute of Technology, Chicago, IL.

[50] Evens, M., and Smith, R., 1978. "A Lexicon for Computer Question Answering System". American Journal of Computational Linguistics, Microfiches 81, pp. 16-24, and 83, pp. 1-98

[51] Rau, L. F., 1991. "Extracting Company Names from Text", Proceedings of the Seventh Conference on Artificial Intelligence Applications, Feb. 24-28, Miami Beach, Florida, pp.29-32

[52] Wacholder, N., Ravin, Y., and Choi, M., 1997 "Disambiguation of Proper Names in Text" Proceedings of the Fifth Conference on Applied Natural Language Processing, Mar 31- Apr 3 Washington, DC, pp.202-208.

[53] Kim, J-S., and Evens, M., 1995. "Extracting Personal Names from the Wall Street Journal", Proceedings of the 6 ~ Midwest Artificial Intelligence and Cognitive Science Society Conference, Carbondale, IL, April 21-23, pp. 78-82

[54] Witten I, et al, 1999. Text mining: A new frontier for lossless compression". Proceedings of Data Compression Conference, Snowbird, Utah.

[55] Feldman R, 1998. Knowledge Management: A Text Miming Approach. Proceedings of the 2nd International Conference on Practical Aspects of Knowledge Management (PAKM98), Basel, Switzerland.

[56] Ichimura Y. et al, 2001. Text Mining System For Analysis of a Salesperson's Daily Reports. Proceedings of Pacific Association for Computational Linguistics (PACLING 2001), Kitakyushu, Japan.

[57] Frawley W. et al, 1991. Knowledge Discovery in Databases: An Overview. Knowledge Discovery in Databases. MIT Press, Page 1–27.

[58] Semio Corporation, 1999. Text Mining and the Knowledge Management Space. Semio Corporation White Paper www.semio.com

[59] Karar Destek Sistemleri ve Uzman Sistemler, http://www.muhasebedergisi.com/muhasebe-makaleleri/karar-destek-sistemleri-ve-uzman-sistemler-2.html

[60] GNU Lesser General Public License, http://en.wikipedia.org/wiki/GNU_Lesser_General_Public_License

[61] Multilingual adaptations of ANNIE, a reusable information extraction tool, http://gate.ac.uk/sale/gate-flyer/2009/gate-flyer-4-page.pdf, ,http://portal.acm.org/citation.cfm?id=1067789

[62] Apache Lucene, http://lucene.apache.org/java/docs/

[63] Crockford on JavaScript , http://developer.yahoo.com/

[64] Managing Gigabytes for Java, http://law.dsi.unimi.it/index.php?option=com_content&task=view&id=31&Itemid=42

[65] Weka, http://www.cs.waikato.ac.nz/ml/weka/

[66] Maxent software for species habitat modeling, http://www.cs.princeton.edu/~schapire/maxent/

[67] SVMlight Support Vector Machine, http://svmlight.joachims.org/

[68] The RASP System, http://www.informatics.susx.ac.uk/research/groups/nlp/rasp/

[69] MINIPAR HOME PAGE, http://webdocs.cs.ualberta.ca/~lindek/minipar.htm

[70] A Practical Parser for Natural Language Engineering Applications, http://www.dcs.shef.ac.uk/~mark/nlp/pubs/index.html

[71] A Practical Parser for Natural Language Engineering Applications, http://www.dcs.shef.ac.uk/~mark/nlp/pubs/index.html

[72] Apache UIMA, http://incubator.apache.org/uima/

[73] WordNet, http://wordnet.princeton.edu/

[74] GATE Information Extraction, http://gate.ac.uk/ie/

[75] Multisource and Multilingual Information Extraction, gate.ac.uk/talks/bcs-03-cheltenham.ppt

[76] ANNIE: a Nearly-New Information Extraction System ,http://gate.ac.uk/sale/tao/splitch6.html#chap:annie

[77] Gazetteer, http://gate.ac.uk/sale/tao

[78] Sözdizimsel Analiz (Syntactic Analysis), http://www.ce.yildiz.edu.tr/mygetfile.php?id=2567

[79] Creating a new application from ANNIE, www. gate.ac.uk/sale/talks/annie-tutorial.ppt.

[80] GATE JAPE Grammar Tutorial, http://gate.ac.uk/sale/thakker-jape-tutorial/GATE%20JAPE%20manual.pdf