

Development of Ontology Knowledgebase for Archive Management Systems

Mehmet Ertugrul*, Ali Gunes

Istanbul Aydin University, Engineering Faculty, Department of Computer Engineering, 34295, Istanbul, Turkey

•Received Date: 09 Apr 2017 •Revised Date: 17 Jul 2017 •Accepted Date: 28 Jul 2017 •Published Online: 02 Aug 2017

Abstract

Digital Archive Management Systems (DAYS) is the most important software tools that provide to manage the millions of digital documents in the administration of the General Directorate of Foundations in electronic environment and share with other institutions. The most important operation in such systems is document search; the aim of it is able to find the required document(s) correctly and quickly for the processes of the institutional functioning and organization. In accordance with the services provided by the foundations, it is expected that DAYS software will search a desired document from millions of documents in the shortest possible time and present it to the user as soon as possible. For this reason, there is a need to enhance the existing DAYS systems used in the General Directorate of Foundations in our country with artificial intelligence approaches. In this study, an ontology-based document search module is proposed for the existing DAYS used by the General Directorate of Foundations to manage their digital archives. The module can work in the backplane, like an additional background module or a Web service, integrated into the DAYS. The proposed system has two basic outputs; Archive Management System Ontology (AYSO) knowledge base and a semantic-based document search engine. However, since the system is quite wide, the article has only mentioned the development of a semantic information base, which is the Archive Management System Ontology and the working principle of the system general.

Keywords

Digital Archive Management Systems, Ontology, Semantic Web, Intelligent Search Engines

*Corresponding Author: Mehmet Ertugrul, m_ertg@hotmail.com

Arşiv Yönetim Sistemleri için Ontoloji Bilgi Tabanı Geliştirilmesi

Mehmet Ertuğrul*, Ali Güneş

İstanbul Aydın Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği, 34295, İstanbul, Türkiye

*Gönderi Tarihi: 09 Nis 2017 •Düzeltilme Tarihi: 17 Tem 2017 •Kabul Tarihi: 28 Tem 2017 •Çevrimiçi Yayın Tarihi: 02 Ağu 2017

Özet

Sayısal Arşiv Yönetim Sistemleri (SAYS), Vakıflar Genel Müdürlükleri idaresinde bulunan milyonlarca adet sayısal belgenin, elektronik ortamda yönetilmesi ve diğer kurumlar ile paylaşılmasına olanak sağlayan en önemli yazılım araçlarıdır. Bu tür sistemlerde en önemli işlev, belge arama olup; amacı, kurum işleyişi ve düzeni için aranan ilgili bir belgeyi/belgeleri doğru ve hızlı bir şekilde bulmaktır. Vakıflarda verilen hizmetin gereği, SAYS yazılımlarından milyonlarca belge içinden istenilen bir belgenin en kısa zamanda etkin bir üslupla aranması ve kullanıcıya en kısa zamanda sunması beklenir. Bu nedenle, ülkemizde Vakıflar Genel Müdürlükleri bünyesinde kullanılan mevcut SAYS sistemlerinin, yapay zekâ yaklaşımlarıyla yükseltilmesine ihtiyaç duyulmaktadır. Bu çalışmada, sayısal arşivleri yönetmek için Vakıflar Genel Müdürlükleri tarafından kullanılan mevcut SAYS için ontoloji tabanlı belge arama modülü önerilmiştir. Bu modül, SAYS'ne entegre olabilecek, ek bir arama modülü ya da bir ağ servisi gibi arka planda çalışabilecektir. Önerilen sistemin iki temel çıktısı bulunmaktadır; Arşiv Yönetim Sistemi Ontolojisi (AYSO) bilgi tabanı ve semantik tabanlı belge arama motoru yazılımıdır. Fakat sistemin oldukça geniş olmasından dolayı, bu makalede sistemin sadece Arşiv Yönetim Sistemi Ontolojisi (AYSO) olan anlamsal bilgi tabanının geliştirilmesi süreçlerinden ve sistemin genel çalışma prensibinden bahsedilmiştir.

Anahtar Kelimeler

Sayısal Arşiv Yönetim Sistemleri, Ontoloji, Anlamsal Ağ, Akıllı Arama Motorları

*Sorumlu Yazar: Mehmet Ertuğrul, m_ertg@hotmail.com

1. GİRİŞ

Vakıflar Genel Müdürlüğü idaresinde bulunan milyonlarca belge sayfanın, hızlı ve güvenilir şekilde elektronik ortama aktarılması, elektronik ortamda yönetilmesi, diğer kurumlar ile paylaşılması için Sayısal Arşiv Yönetim Sistemleri (SAYS) kullanılmaktadır. Bu tür sistemlerde, en önemli fonksiyonlardan biri de belge aramadır. Verilen hizmet gereği, milyonlarca belge içinden istenilen bir belgenin, en kısa zamanda, anlamsal tabanda akıllı bir üslupla bulunabilmesi önemli bir gereksinimdir. Bu nedenle, özellikle Web tabanlı büyük veri yönetim sistemlerinde anlamsal arama motorları yaygın araştırma çalışmaları arasındadır [1-5]. Bilindiği gibi bilgi sistemlerinin inşasında kullanılan günümüz Web teknolojisi (Web 2.0) [6] bilginin anlamının, makinalar tarafından anlaşılmasına ve bilgiler arasında anlamsal bağlar kurulmasına olanak sağlayamamaktadır. Web 2.0 teknolojisi ile mevcut Web ortamında sunulan bir sayfanın veya içeriğinin, bilgisayarlar tarafından anlaşılabilir olması zordur ve yeni bir yaklaşımı gerektirmektedir. Bu nedenle, Tim Berners-Lee ve arkadaşları, Web ortamında bulunan sadece insanların algıladığı bu dosya veya belgelerin, bilgisayarlar tarafından işlenerek anlamlandırılması ve en uygun bilgiye ulaşabilmesini sağlayan, Anlamsal Ağ (Semantic Web) ya da Anlamsal Veb (AV) [7] yaklaşımı üzerine çalışmalar yapmışlardır. AV'in amacı, Web ortamında bulunan farklı yapıdaki bilgilerin ve aralarındaki bağların bilgisayarlar tarafından anlaşılır hale getirilmesini sağlamaktadır. Kısaca, AV; makinelerin, kullanıcı kelimelerinin "anlamalarını anlamasına imkân sağlayan" ve böylece karmaşık insan taleplerine "yanıt vermeyi" amaçlayan bir Web teknolojisidir.

Bu çalışmada, sayısal ortama taşınmış olan Vakıflar Genel Müdürlüğü idaresine ait milyonlarca belgenin, bir SAYS kullanıcı ara yüzü üzerinden, kolaylıkla aranmasını ve mevcut veri kaynaklarından bilgi keşfi kapsamında kullanılabilirliğini AV teknolojisi sayesinde sağlayacak bir sistem mimarisi önerilmiştir. Böylece, SAYS arkasında çalışabilen, daha gelişmiş bir arama modülünü ile kullanıcıların ihtiyaç duyduğu tarihi bir belge veya dokümanın anlamsal yollarla bulunabilmesi amaçlanmıştır. Sadece dokümanın arşivlendiği dosya, envanter numarası vb. bilgilere göre arama yapmak yerine, belgelerin içerik bakımından da aranabilmesi düşünülmüştür. Bu nedenle, bu çalışmada, Vakıflar Genel Müdürlüğü idaresindeki, resim formatında arşivlenmiş milyonlarca belgenin, bilgisayarların anlayacağı formata dönüştürülebilmesi için bir SAYS arama modülü yazılımı ve genel bir arşiv ontoloji bilgi tabanı [8-9] geliştirilmiştir. Böylece her belgenin, kendisine özgü ve aynı zamanda standart bir formatta bir ontoloji dosyası saklanabilecek ve anlamsal arama esnasında kullanılması düşünülmüştür.

Bu çalışmada, diğer yapılan ulusal ve uluslararası benzer çalışmalardaki ontoloji yapıları incelenmiş ve önerilen sistemin kendisine özel bir Arşiv Yönetim Sistemi Ontolojisi (AYSO) bilgi tabanı geliştirilmiştir. Bu makalede, ASYO bilgi tabanının geliştirilmesindeki tüm süreçler ve sistemin çalışma modeli hakkında bilgi sunulmuştur. Sistemin arama mekanizmasının geliştirilmesi süreçleri hakkındaki detaylı bilginin ele alınması gelecek makale çalışmalarında yapılacaktır.

Çalışma süresince, incelenen benzer diğer çalışmalarda, farklı doküman türleri (gazete, dergi, makale, tarihsel doküman ve belgeler veya devlet süreçlerinin girdi/çıktıları belge türleri vb. gibi) üzerinde çalışmak durumunda kalan kurum veya kuruluşlara özel ontoloji yapıları ve arama mekanizmaları düşünülmüştür. Bu ontolojiler sayesinde, kurumların belge arama sürecinde, belgelerin içeriğini makinelerce anlama sağlanabildiğinden, daha verimli sonuçlar üretilebilecektir. Aşağıdaki çalışmalar, genellikle ontoloji tabanlı doküman arama sistemlerine yönelik çalışmaların özetidir.

İlk incelenen proje, MUNINN projesi olup, çok disiplinli, çok uluslu bir akademik araştırma projesidir [10]. MUNINN projesinin amacı, sayısallaştırılmış belgelerin arşivlerini almak, yazılı veriyi büyük miktarda veri madenciliği kullanarak ayıklamak ve elde edilen bilgiyi yapılandırılmış veri tabanlarına dönüştürmektir. Proje kapsamında geliştirilen *Documents Ontolojisi*; FOAF, CC Works ve Dublin Core gibi diğer yaygın ontolojilerin birleşiminden yaratılmıştır.

Diğer incelenen proje, FinnONTO 2.0 ontolojisi olup, Finlandiya Milli Kütüphanesi için geliştirilen bir arşiv ontolojisidir. Finlandiya Milli Kütüphanesi, 900.000'den fazla eski gazetenin bulunduğu ve bu tarihi gazetelerin sayısal arşivlerinin içerdiği bir kütüphanedir. Bu arşiv, 1771-1890 yıllarında Finlandiya'da yayınlanan her gazeteyi içermektedir [11]. AV teknolojisi ile ulusal FinnONTO ontolojisini geliştirilmiş ve bu ulusal arşivdeki sayısal dokümanlara ve tarihsel gazetelere anlamsal özellikler kazandırılmıştır. İlâveten, FinnONTO [12] ontoloji üzerinde çalışması için bir anlamsal arama motoru geliştirilmiştir. “Hako Anlamsal Arama Motoru”, Helsinki Teknoloji Üniversitesi'nin Semantic Computing Research grubunda Joonas Laitio ve Jussi Kurki tarafından FinnONTO 2.0 projesi kapsamında geliştirilmiştir. Hako, bir Resource Description Framework (RDF) [13] dosyasını girdi olarak alır ve bazı materyal-özellikli konfigürasyonlarla RDF materyali için bir arama ara yüzü olan bir web uygulaması açar. Önceden tanımlanmış özellikleri ek bilgi olarak aramaya katar ve

böylece Finlandiya Milli Kütüphanesinin tarihi gazete arşivinde yazılmış eski makaleleri anlamsal tabanda arayabilmeyi sağlar.

Diğer bir sayısal arşiv ontolojisi ise STOLE ontolojisidir ve İtalyanca "STORİA LEgislative della pubblica amministrazione Italiana" nın kısaltmasıdır. "İtalyan Kamu İdaresinin Yasama Tarihi" anlamına gelmektedir. STOLE, tarih araştırmacılarına yardım etmeyi amaçlayan, arşiv dokümanlarındaki veriyi düzenleyen, gerekli bilgileri toplamayı ve tarihsel belgelere kolaylıkla erişmeyi sağlayan ontoloji tabanlı bir sayısal arşiv projesidir [14].

Diğer bir sayısal arşiv ontolojisi ise, İçişleri Bakanlığı Nüfus ve Vatandaşlık İdaresi Dijital Arşiv Projesi (NVI DAP) projesi olup; Erk yazılım, TÜBİTAK Ulusal Elektronik ve Kriptoloji Araştırma Enstitüsü, T.C. İçişleri Bakanlığı Nüfus ve Vatandaşlık İşleri (NVI) ortaklığında geliştirilen Ulusal Bilim Fonu-KAMAG projesidir ve %100 devlet tarafından fonlanmış Ontoloji tabanlı sayısal arşiv projesidir [15]. Bu projede, kurum arşivine özel belge ontolojisi geliştirmişlerdir. Bu ontolojiden faydalanarak, Osmanlıca el yazısı tanımlama yapılması için ARGE çalışmaları yapmışlardır.

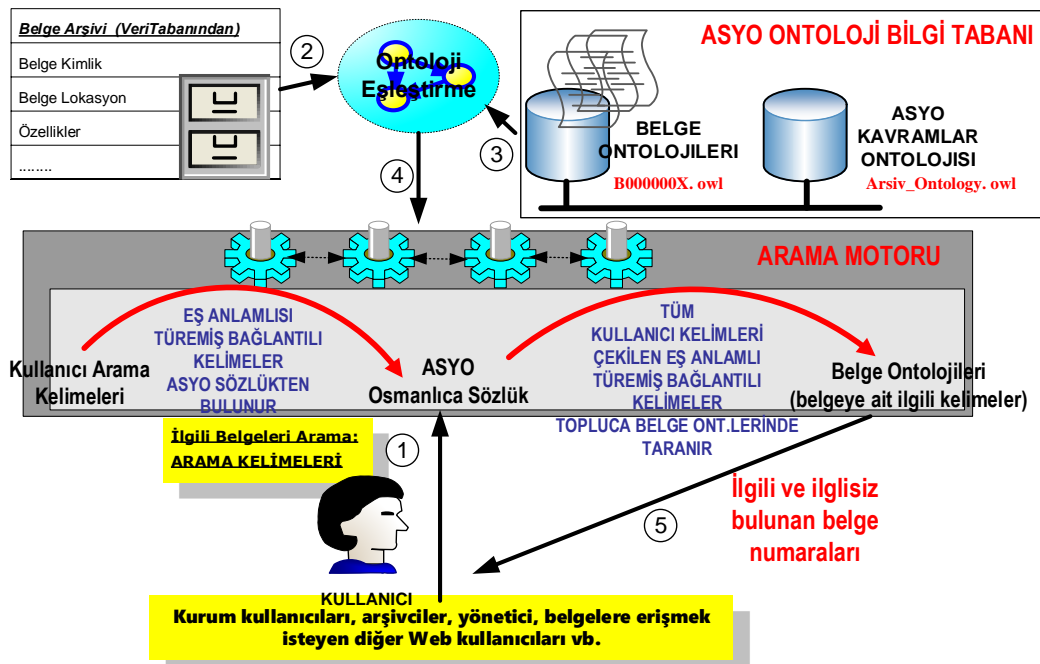
Son olarak, AYSO bilgi tabanını geliştirirken diğer bir faydalanılan çalışma, Semantik Belge Yönetim Sistemi (SDMS) [16] çalışması olmuştur. Bu sistem için bir ontoloji bilgi tabanı geliştirilmiş, Semantik Doküman Ontolojisi (SDO) olarak literatüre geçmiştir. Sistemin amacı, Web üzerindeki dokümanların tıpkı ofis dokümanları gibi işlenmesine olanak sağlamaktır. Sistemde, üç farklı belge türü temel alınmıştır: Kâğıt tabanlı, sayısal ve semantik belgeler. Semantik belgeler, sistemde benzersiz olarak duran ve anlamsal bağlamlarla tanımlanmış bileşik bir kaynak türüdür [16].

2. SİSTEM ÇALIŞMA PRENSİBİ

İnternet ortamında anahtar kelime girilerek, aranılan bilgiye erişim sunan, arama motorları; günümüz Web sitelerinin taranması için kullanılan temel araçlardır. Web sitelerinde ilgili veriyi bulmak için kullanılan bu araçlar, ne yazık ki içerik tabanlı bilgi yönetiminde verim sağlayamamaktadır. Bunun nedeni, anahtar kelime girilerek yapılan bilgi arama sürecinde kişi ve kurumların konuyla doğrudan ilgili bilgilere ulaşması zordur. Aynı zamanda, Web siteleri için anahtar kelime girilerek yapılan arama sayesinde ilgili belgelere ulaşım, hem zaman hem de enerji kaybı anlamına gelmektedir. Çünkü anahtar kelimeye dayalı aramalar, farklı anlamlara sahip, fakat yazılışları aynı olan terimlerin yer aldığı konuyla ilgili olmayan gereksiz birçok bilgiye erişim sağlar. Aynı şekilde, aynı anlama gelip farklı terimlerle ifade edilen bilgilere de erişimi engeller. Diğer bir sorun ise, Web sitelerinde aranan bilgilere ilişkin terminolojideki

tutarsızlıklar olabilmektedir. Günümüz Web tabanlı bilgi sistemlerinde sıklıkla karşılaşılan bu sıkıntılar, bilgi yönetimi sistemlerinde önemli zayıflıklar olduğuna işaret etmektedir. Genelde bu günümüz bilgi sistemleri Web 2.0 teknolojisi ile yapılandırılmıştır. Bu nedenle, Web 3.0 yani AV teknolojileri kavram ve özellik bağlamlarına odaklanarak bu olumsuzluklara çözüm olarak ortaya atılmıştır.

Şekil 1' de görüldüğü üzere, sistemin Web arama hizmetine bağlı olacak olan Anlamsal Arama Motoru bir uygulama ara yüzden giriş ile mümkün olacaktır (Adım 1). Bu ara yüz üzerinden yetkili veya yetkisiz çeşitli kullanıcı tipleri (kurum kullanıcıları, arşivciler, yönetici ve Web üzerinden belgelere erişmek isteyen diğer tüm kullanıcılar, müşteriler, araştırmacılar vb.) sisteme giriş yapabileceklerdir. Bir erişimle, kurumun sayısal arşivine (Adım 2) yapılan sorgular, kullanıcıların girdiği anahtar kelimeler ile başlayacaktır. Bu anahtar kelimeler, sistemin ontoloji bilgi tabanındaki (Belge Ontolojileri ve ASYO) bilgi tabanlarındaki tanımlı kelimeler ile kıyaslanacaktır (Adım 3 & Adım 4).



Şekil 1. Sistemin genel çalışma prensibi.

Belge Ontolojileri, kurumun sayısal arşivinde bulunan her belgenin, kendisine has bir ontoloji dokümanıdır. Bir belge ontolojisinde, ait olduğu belgeyi tanımlayan; özellik tanımları, içeriğine yönelik ilgili kelimeler ve diğer bağlı olduğu anlamsal özellik tanımlarını içermektedir. Belge Ontolojileri sayesinde, bir belgenin, yapılan herhangi bir arama için bir sonuç belge olup olmadığına karar verilmesi için gereklidir.

Öncelikle anlamsal arama süreci, kullanıcının girdiği kelimelerin, genel sistem ontolojisi olan ASYO içindeki bir terimler sözlüğü içinde aranması ve bağlı olduğu eş anlamlısı, zıt anlamlısı,

türemiş kelime vb. anlamsal bağlarının algılanması ile başlatılır (Adım 4). Daha sonra, toplanan tüm arama kelimeleri, diğer deyişle; kullanıcı terimleri, eş anlamlısı, zıt anlamlısı, türemiş kelime vb. bağlantılı bulunan terimler topluca, Belge Ontolojilerinde aranmaya başlar. Bu noktada, taranan her belge ontolojisi için, önceden tanımlanmış bu bağlar aracılığıyla (örneğin; eş anlamlısı, zıt anlamlısı, türemiş kelime vb.) ilgili veya ilgisiz sonucu, yine önceden ontolojide tanımlı yapay zekâ kuralları sayesinde döndürülebilmektedir (Adım 5).

3. KULLANILAN ARAÇLAR VE METOTLAR

Sistem yazılımı ve ASYO bilgi tabanının geliştirilmesinde bazı yazılım araçları, dil ve kütüphaneleri kullanılmıştır (Tablo 1). Sistemin ASYO bilgi tabanı geliştirilirken OWL (Web Ontology Language) [17] dili kullanılmıştır. ASYO bilgi tabanına bağlı iki alt ontoloji birimleri (Belge Ontolojileri ve Arşiv_Ontology.owl arşiv kavramlar ontolojisi) yaratılmıştır. Kullanıcının sisteme girdi olarak gönderdiği, belge arama sorguları ve belgelerden içerik verisi çıkarsamaları, bu ontoloji kesitleri üzerinde, sorguların yürütülmesi ile yapılmaktadır. ASYO bilgi tabanı, Protégé 5.0 editörü [18] kullanılarak, OWL dilinin son sürümü olan, OWL 2.0 formatında [19] yaratılmıştır. Ayrıca önerilen sistemin yazılım uygulaması, Java program dili ile Netbeans 8.0 [20] platformunda geliştirilmiştir. Netbeans’de sistemin Web arama servislerinin oluşturulması, Web servisleri yolu ile, ASYO bilgi tabanına kullanıcı sorgularının taşınması ve istenilen bilgi çıkarımlarının yapılması sağlanmıştır. Java kullanılarak, ontolojideki mevcut bilginin okunması, güncellenmesi, ontolojiye yeni aksiyomlar yazılması veya silinmesini vb. işlemler; OWL API (Ontology Parser) [21] kütüphanesi ile yapılmıştır. Ayrıca belge arama işleminde, bazı anlamsal kurallar tanımlanmıştır. Bu kurallar, ontoloji üzerinde konuşlanan bir kural bilgi tabanı yapısında olup; Semantic Web Rule Language (SWRL) [22] dili kullanılarak yaratılmıştır.

Tablo 1. Sistemin Semantik Arama Motoru ve ASYO bilgi tabanının geliştirilmesinde kullanılan araç, dil ve kütüphaneler.

Kullanılan	Araç, Dil ve Kütüphaneler	Neden Gerekli	Geliştiren Kurum	Kullanılan Yer
Diller	OWL [17]	Ontoloji yaratmak için kullanılan dildir.	Stanford Üniversitesi	-
	Semantic Web Rule Language (SWRL) [22]	Ontolojide kurallar yaratmak için kullanılan dildir.	Stanford Üniversitesi	-
	Java	Sistemin yazılımı için tercih edilen programlama dilidir.	Java Sun/Oracle	-
Yazılım Platformları	NetBeans [20]	Arama Motorunu Java dilinde geliştirmek.	Java Sun/Oracle	Bilgisayara yüklenmiştir.
	Protégé Ontoloji Editör [18]	ASYO bilgi tabanı geliştirmek.	Stanford Üniversitesi	Bilgisayara yüklenmiştir.

Kütüphaneler	JDK 7	NetBeans platformunda Java yazmak için gereklidir.	Java Sun/Oracle	NetBeans 8.0 içine kütüphane olarak eklenmiştir.
	OWL API [21]	NetBeans de Java kodlarından Ontolojiye işlem yapmak için.	Manchester Üniversitesi	NetBeans 8.0 içine kütüphane olarak eklenmiştir.
	PELLET API [23]	NetBeans de Java kodlarından Ontolojide var olan kuralları çalıştırıp çıkarsınmış sonuçları Java ortamına taşımak için.	Stanford Üniversitesi	NetBeans 8.0 içine kütüphane olarak eklenmiştir.

Sistem mimarisinde kullanılan bazı metotlar:

- Ontoloji kullanılarak anlamsal eşleştirme ‘Semantic Matchmaking’ ve SWRL ile veri çıkarımı,
- Jaro-Winkler algoritması [24,25],
- Düzenli ifadelerdir (Regular-expression).

Yukarıda listelenen bu metotlar;

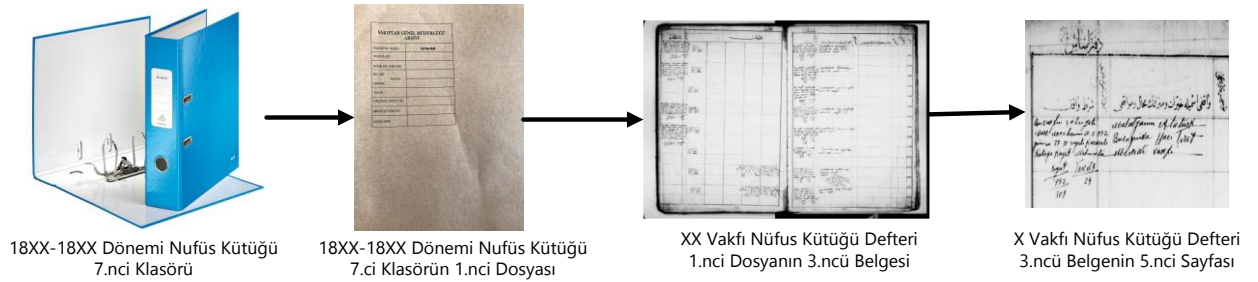
- Kullanıcıların belge ararken girdikleri arama kelimeler (örneğin; Osmanlı dönemi anlaşmalar, vb.) ve,
- Her belge ontolojisindeki önceden tanımlı o dokümana ait anahtar terimler (örneğin: Karlofça antlaşması vb.) ve o belgenin özellik tanımları (dokümanın yıl, yazar, tarihi dönem bilgileri vb.) ve,
- Son olarak, genel sistem ASYO ontolojisi içindeki terimler sözlüğünde bulunan eş anlamlısı, zıt anlamlısı, türemiş kelime vb. anlamsal bağlarının algılanması ve anlamsal eşleştirme metotları ile ASYO ontoloji tabanında kıyaslanması sağlanmıştır.

Yapılan sorgu ile belge üzerindeki tüm anlamsal tanımlamalar kıyaslandıktan sonra anlamsal tabanda bir bağlantı bulunmuşsa, o belge listeye **“ilgili”** olarak sonuç atanmaktadır. Eğer anlamsal olarak hiçbir bağlantı bulunamadıysa gelecek diğer doküman incelemesine geçilmektedir. Ontoloji üzerindeki kavram tanımları ve kavramlar arası tanımlanan bağlar gelecek bölümlerde detaylı olarak gösterilmiştir.

4. SİSTEMİN ONTOLOJİ BİLGİ TABANININ GELİŞTİRİLMESİ

AV, yazılımların, kullanıcılarına daha iyi hizmet sağlayabilmeleri için, yazılımların mevcut veri tabanları üzerine anlamsal bir bilgi tabanı katmanı koymayı gerektirmektedir. Bu bilgi katmanının, insanların anlamasından ziyade, daha çok yazılım fonksiyonları tarafından anlaşılması ve kendi kendine karar verebilmesi veya yeni veri çıkarsamaları yapabilmesi açısından gereklidir. Bu çalışma kapsamında geliştirilen ontoloji, mevcut bir sayısal arşiv yönetim sisteminin veri tabanı üzerinde çalışabilir ve aynı zamanda bu ontoloji üzerinde çalışan bir semantik arama motoru modülü tarafından da sistemin mevcut belge arama fonksiyonu

arkasında entegrasyonu mümkün olabilir. Bu çalışmada, ASYO bilgi tabanının yapısal kurgusu Vakıflar Genel Müdürlükleri idaresince kullanılan klasik SAYS işlevlerini dikkate alınarak oluşturulmuştur. Bilindiği üzere, Vakıf arşivleri klasörlerden oluşur. Klasörler, ilgili arşiv dosyalarını içerir. Bu klasör ve arşiv dosyaları genelde belli kategori, dönem veya tarih aralığına göre oluşturulmaktadır. Her dosyada değişik sayıda belge içerir. Her belge farklı sayıda sayfa içermektedir. ASYO yaratılırken bu kurguya bağlı kalınarak temel bir yapı oluşturulmuştur. Şekil 2’de verilen örnek de görüldüğü üzere, 18XX-18XX döneminde ait bir klasör, o klasör içinde 7.nci dosya, o dosya içinde 3.ncü belge ve o belge içinde herhangi bir sayfa örnek teşkil etmesi bakımından verilmiştir.



Şekil 2. ASYO bilgi tabanındaki arşivleme ve belge yönetimindeki genel kurgu.

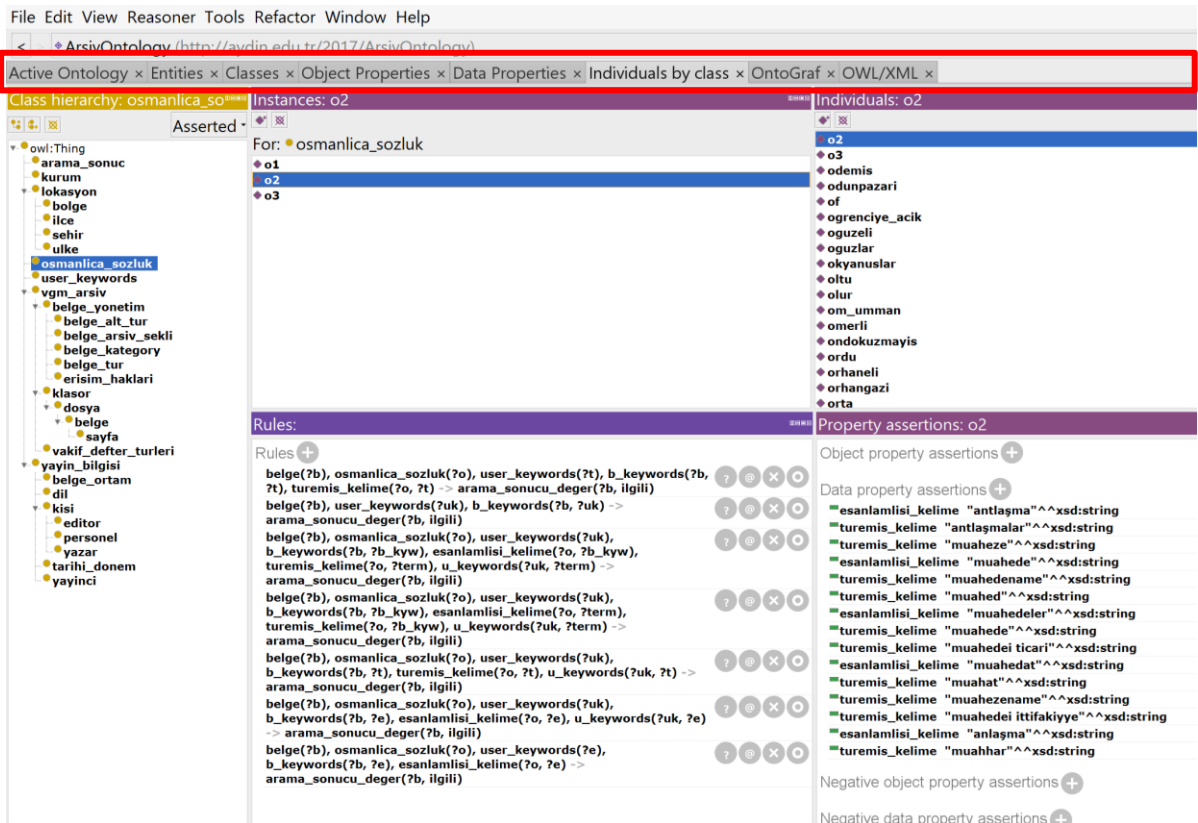
Şekil 3’de geliştirilen AYSO bilgi tabanının Protégé ontoloji editörü üzerindeki genel görüntüsü verilmiştir. AYSO’nin hazırlanmasındaki yürütülen temel adımlar aşağıdaki gibidir:

- Klasör, Dosya, Belge ve Sayfa kavramlarına ait Fişler/Etiketlerin belirlenmesi,
- AYSO içindeki kavram ya da sınıf yapılarının (Ontology Classes) oluşturulması,
- AYSO’nin içinde belirlenen sınıf yapılarının arasındaki özelliklerin (ObjectType ve DataType properties), önceden belirlenmiş Fişler/Etiketlere göre kurgulanması ve yaratılması,
- AYSO’nin içinde belirlenen sınıf yapılarının yapılarına ait veri elemanlarının (Ontology Individuals) oluşturulması,
- Ontoloji Tabanlı Gelişmiş Arama Motoruna ait Anlamsal Kural Tabanının SWRL dili ile geliştirilmesi,
- AYSO’nin doğrulaması testleri yapılmasıdır.

Şekil 3, bir sayısal arşiv yönetim sistemi alanındaki tüm genel kavramları (*örneğin; belge, klasör, dosya, belge-kurum, lokasyon, belge orijin, vb.*), kavramlara ait veri öğelerini (değerlerini) (*örneğin; belgeNo:000001, belgeŞehir:Ankara, belgeKurum:Merkez Kütüphane vb.*) ve kavramlar arası özellik tanımlarını (*örneğin; belgeNo, belgeKlasörNo,*

belgeYazarı, *belgeDönemi*, vb.) gibi anlamsal bağları ve ilişkileri ifade etmesi açısından verilmiştir. AYSO bilgi tabanı oluşturulurken, OWL ontoloji dilinin; *owl: class*, *rdfs:subClassOf*, *owl:DatatypeProperty*, *owl:ObjectProperty* vb. gibi anlamsal ilişki tanımları kullanılmıştır (bkz. Şekil 3’de kırmızı alan).

ASYO bilgi tabanı şu anda, sadece; Vakıflar Genel Müdürlüklerinin kullandığı sayısal arşiv yönetim sistemlerinin genel yapısını kapsamaktadır, fakat zaman içerisinde ASYO bilgi tabanı daha çok geliştirebilir. Kısacası, arşiv dokümanlarında kullanılan diller (Osmanlıca, Farsça, vb.), ve bu dillere ait genel terimler veya sözlükler vb. bilgiler ontolojiye ancak zamanla ilave edilmesiyle belli bir doygunluğa ulaşılabilir.



Şekil 3. AYSO bilgi tabanın Protégé editörü üzerindeki genel görüntüsü.

Şu anki geliştirilen AYSO bilgi tabanı içinde, iki tür ontoloji yapısı düşünülmüştür:

1. Arşiv Yönetim Sistemlerinin genel yapısını içeren “Arsiv_Ontology.owl” adı verilen genel bir ontolojidir. Bu ontoloji içinde, Arşiv Yönetim Sistemlerine ait tüm kavramlar, ilişkiler, kavramlara ait veri öğeleri, veri öğeleri arasındaki mevcut aksiyomlar, ve SWRL yapay zekâ arama kuralları tanımlanmıştır. Aynı zamanda, yeni ontolojiler üretmek için temel bir ontoloji (Upper Ontology) veya bir kalıp ontoloji olarak da düşünülebilir.

2. Belge Ontolojileri (B000000x.owl): Kuruma yeni gelen bir belgeyi, arşiv personelinin bir ara yüz üzerinden kurumun veri tabanına kayıt etmesi beklenir. Bu kayıt esnasında, sistemin bir fonksiyonu olan, Ontology Generator (Ontoloji Üretici) modülü sayesinde, o belgeye ait özgün bir ontoloji dosyası otomatik olarak üretilmektedir. Her belge için özgün bir ontoloji dosyası üretilirken, Arsiv_Ontology.owl ontoloji kalıbı kullanılır. Sonunda, o belgeyi tanımlayan, B000000X.owl adında bir ontoloji dosyası yaratılır ve sistemin veri tabanına atılır. Kullanıcıların gelecek sorgularında, arama esnasında kullanılmak üzere tutulur.

4.1. Arsiv_Ontology.owl Yaratılması

Önceki bölümde, bir arşiv belgesi, dosyası veya klasör yapısına ait tüm kavramları, ilişkileri, kavramlara ait veri öğelerini, veri öğeleri arasındaki mevcut aksiyomları ve SWRL yapay zekâ arama kurallarını, “Arsiv_Ontology.owl” adı verilen bir ontoloji altında tanımlandığından bahsedilmiştir. Bu anlamsal tanımlamalar, Arsiv_Ontology.owl adı verilen ve AYSO'nin genel ontoloji yapısı ifade eden, bu ontoloji altında: 31 temel sınıf ve alt sınıf (Şekil 4), 30 veri türü özellik (Şekil 5), 28 nesne türünde özellik (Şekil 6), yaklaşık 1347 veri öğesi (NamedIndividual- bkz. Şekil 7) ve 7 adet anlamsal arama kuralı (Şekil 8) yaratılmıştır. AYSO'nin kalıp ontolojisi olan, Arsiv_Ontology.owl ontolojisinde yaratılan tüm özellik ve kural tanımları Tablo 2’de sunulmuştur.

Tablo 2. AYSO ontolojisinin büyüklüğü ve içerdiği kavramlar hakkındaki veriler.

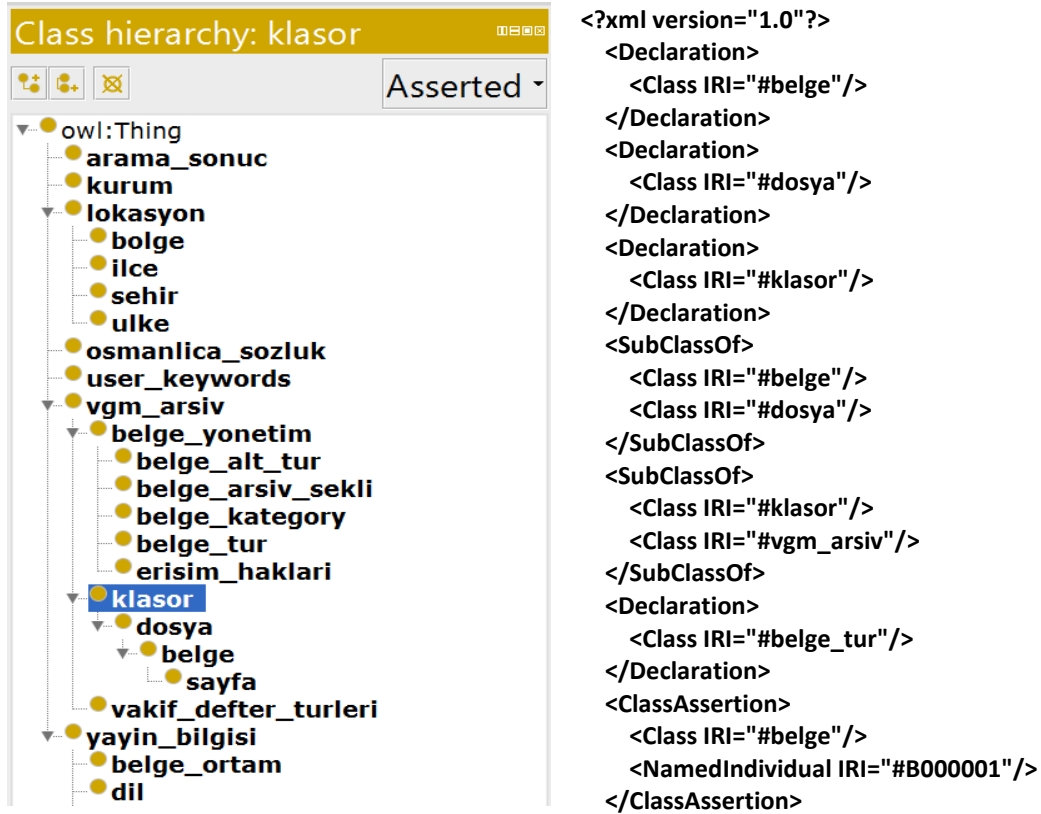
owl:class, rdfs:subClassOf	54	owl: Thing, arama_sonuc, kurum, lokasyon, bolge, ilce, sehir, ulke, osmanlica_sozluk, user_keywords, vgm_arsiv, belge_yonetim, belge_alt_tur, belge_arsiv_sekli, belge_kategory, belge_tur, erisim_haklari, klasor, dosya, belge, sayfa, vakif_defter_turleri, yayin_bilgisi, belge_ortam, dil, kisi, editor, personel, yazar, tarihi_donem, yayinci
ObjectProperty	17	arama_sonucu_deger, b_altTuru, b_arsivSekli, b_bolge, b_dili, b_dosyasiKodNo, b_duzenleyenKurum, b_editoru, b_erisimHakki, b_eseri, b_excellSiraNo, b_geldigiKurum, b_ilce, b_kategory, b_kayitEdenPersonelNo, b_klasorununKodNo, b_ortami, b_sehir, b_tarihiDonemi, b_turu, b_ulke, b_vakifDefterTuru, b_yayincisi, b_yayinlamistir, b_yazari, d_klasorununKodNo, k_belgeyilcerir, k_dosyayilcerir
DatatypePropert y	22	b_arkaKapagi, b_dijitalizeTarihi, b_duzenleyenKisi, b_ilkSayfasi, b_kayitNo, b_keywords, b_klasorunAdi, b_onKapagi, b_orjinalKayitNo, b_sayfaAdeti, b_tarihAraligi, b_tarihi, b_url, b_vakifaGelisTarihi, b_yeniKlasorKodNo, d_acilisTarihi, d_belgeAdeti, d_gomlekSiraNo, d_keywords, d_mikroFormNo, d_orjinalNo, d_tasnifKodu, d_varakAdeti, esanlamlisi_kelime, s_byteDegeri, s_sayfaNo, s_x_pixel, s_y_pixel, turemis_kelime, u_keywords
NamedIndividual	1347	Arama motoru çalıştıkça değişkenlik göstermektedir.

SWRL Kural Tabanı	7	<p>*Kural 1: belge(?b), osmanlica_sozluk(?o), user_keywords(?t), b_keywords(?b, ?t), turemis_kelime(?o, ?t) -> arama_sonucu_deger(?b, ilgili)</p> <p>*Kural 2: belge(?b), user_keywords(?uk), b_keywords(?b, ?uk) -> arama_sonucu_deger(?b, ilgili)</p> <p>*Kural 3: belge(?b), osmanlica_sozluk(?o), user_keywords(?uk), b_keywords(?b, ?b_kyw), esanlamlisi_kelime(?o, ?b_kyw), turemis_kelime(?o, ?term), u_keywords(?uk, ?term) -> arama_sonucu_deger(?b, ilgili)</p> <p>*Kural 4: belge(?b), osmanlica_sozluk(?o), user_keywords(?uk), b_keywords(?b, ?b_kyw), esanlamlisi_kelime(?o, ?term), turemis_kelime(?o, ?b_kyw), u_keywords(?uk, ?term) -> arama_sonucu_deger(?b, ilgili)</p> <p>*Kural 5: belge(?b), osmanlica_sozluk(?o), user_keywords(?uk), b_keywords(?b, ?t), turemis_kelime(?o, ?t), u_keywords(?uk, ?t) -> arama_sonucu_deger(?b, ilgili)</p> <p>*Kural 6: belge(?b), osmanlica_sozluk(?o), user_keywords(?uk), b_keywords(?b, ?e), esanlamlisi_kelime(?o, ?e), u_keywords(?uk, ?e) -> arama_sonucu_deger(?b, ilgili)</p> <p>*Kural 7: belge(?b), osmanlica_sozluk(?o), user_keywords(?e), b_keywords(?b, ?e), esanlamlisi_kelime(?o, ?e) -> arama_sonucu_deger(?b, ilgili)</p>
-------------------	---	--

EK A’da Arsiv_Ontology.owl’un OWL 2 formundan kesit tablo halinde sunulmuştur. Gelecek bölümde, Protégé’deki görselleri ve ilgili OWL 2 formundan kesitler paylaşılmıştır.

a. Protégé Editörü ile “Class” Sınıfların Yaratılması

Şekil 4’de, AYSO’nin kalıp ontolojisi olan, Arsiv_Ontology.owl ontolojisi içinde tanımlanan arşiv kavramlarına ait görsel (solda) ve OWL 2 kodu (sağda) verilmiştir.



The image shows a screenshot of the Protégé ontology editor. On the left, a class hierarchy for 'klasor' is displayed. The hierarchy starts with 'owl:Thing' and includes classes like 'arama_sonuc', 'kurum', 'lokasyon', 'bolge', 'ilce', 'sehir', 'ulke', 'osmanlica_sozluk', 'user_keywords', 'vgm_arsiv', 'belge_yonetim', 'belge_alt_tur', 'belge_arsiv_sekli', 'belge_category', 'belge_tur', 'erisim_haklari', 'klasor', 'dosya', 'belge', 'sayfa', 'vakif_defter_turleri', 'yayin_bilgisi', 'belge_ortam', and 'dil'. The 'klasor' class is highlighted in blue. On the right, the corresponding OWL code is shown, including declarations for 'belge', 'dosya', 'klasor', and 'belge_tur', and a class assertion for 'belge' with a named individual 'B000001'.

```
<?xml version="1.0"?>
<Declaration>
  <Class IRI="#belge"/>
</Declaration>
<Declaration>
  <Class IRI="#dosya"/>
</Declaration>
<Declaration>
  <Class IRI="#klasor"/>
</Declaration>
<SubClassOf>
  <Class IRI="#belge"/>
  <Class IRI="#dosya"/>
</SubClassOf>
<SubClassOf>
  <Class IRI="#klasor"/>
  <Class IRI="#vgm_arsiv"/>
</SubClassOf>
<Declaration>
  <Class IRI="#belge_tur"/>
</Declaration>
<ClassAssertion>
  <Class IRI="#belge"/>
  <NamedIndividual IRI="#B000001"/>
</ClassAssertion>
```

Şekil 4. a) Protégé üzerinde “Class” sınıfların görünümü b) Bazı sınıfların OWL formu.

b. Protege Editörü ile “DataType” Tipindeki Özelliklerin Yaratılması

Şekil 5’de, Arsiv_Ontology.owl ontolojisi içinde tanımlanan, arşiv kavramlarının veri türü özelliğindeki ilişki tanımlamalarına (örneğin; “Belge” → *b_safaAdeti* → “xsd:Int” (Kişi)) ait görsel (solda) ve OWL 2 kodu (sağda) verilmiştir.

Active Ontology x Entities x Classes x Object Properties x Data Properties x Instances

Data property hierarchy: b_url

Annotations: b_url

Description: b_url

```

<Declaration>
  <DataProperty IRI="#url"/>
</Declaration>
<Declaration>
  <DataProperty IRI="#klasorKodNo"/>
</Declaration>
<DataPropertyDomain>
  <DataProperty IRI="#belgeAdeti"/>
  <Class IRI="#dosya"/>
</DataPropertyDomain>
<DataPropertyDomain>
  <DataProperty IRI="#dijitalizeTarihi"/>
  <Class IRI="#dosya"/>
</DataPropertyDomain>
<DataPropertyDomain>
  <DataProperty IRI="#keywords"/>
  <Class IRI="#belge"/>
</DataPropertyDomain>
<DataPropertyRange>
  <DataProperty IRI="#keywords"/>
  <Datatype abbreviatedIRI="xsd:string"/>
</DataPropertyDomain>

```

Şekil 5. a) Protégé editöründe “Data Type” veri türü özelliklerinin görünümü
b) Bazı “Data Type” özelliklerinin OWL formu.

c. Protege Editörü ile “ObjectType” Tipindeki Özelliklerin Yaratılması

Active Ontology x Entities x Classes x Object Properties x Data Properties x Instances

Object property hierarchy: b_dili

Annotations: b_dili

Characteristics: b_dili

```

<ObjectPropertyDomain>
  <ObjectProperty IRI="#b_yazari"/>
  <Class IRI="#belge"/>
</ObjectPropertyDomain>
<ObjectPropertyRange>
  <ObjectProperty IRI="#b_arsivSekli"/>
  <Class IRI="#belge_arsiv_sekli"/>
</ObjectPropertyRange>
<ObjectPropertyRange>
  <ObjectProperty IRI="#b_bolge"/>
  <Class IRI="#bolge"/>
</ObjectPropertyRange>
<ObjectPropertyDomain>
  <ObjectProperty IRI="#b_dili"/>
  <Class IRI="#belge"/>
</ObjectPropertyDomain>
<ObjectPropertyRange>
  <ObjectProperty IRI="#b_dili"/>
  <Class IRI="#dil"/>

```

Şekil 6. a) Protégé editöründe “ObjectType” nesne türü özelliklerinin görünümü
b) Bazı “ObjectType” özelliklerinin OWL formu.

Şekil 6’da, Arsiv_Ontology.owl ontolojisi içinde tanımlanan, arşiv kavramlarının nesne türü özelliğindeki ilişkisel tanımlamalarına (örneğin; “*klasor:Belge*” → *b_dili* → *yayin_bilgisi:dil*) ait görsel (solda) ve OWL 2 kodu (sağda) verilmiştir.

d. Protégé Editörü ile “Individual” Sınıf Elemanların Yaratılması

Şekil 7’de, Arşiv_Ontology.owl ontolojisi içinde tanımlanan belge veri öğelerine ait tanımlamalara (örneğin; bir belge öğesi için, “*belge:B000000X.owl*” → *b_dili* → “*dil:Osmanlica*”) ait görsel (solda) ve OWL 2 kodu (sağda) verilmiştir.

OWL 2 kodu:

```

<DataPropertyAssertion>
  <DataProperty IRI="#turemis_kelime"/>
  <NamedIndividual IRI="#o2"/>
  <Literal IRI="#string">muahhar</Literal>
</DataPropertyAssertion>
<DataPropertyAssertion>
  <DataProperty IRI="#esanlamlisi_kelime"/>
  <NamedIndividual IRI="#o3"/>
  <Literal IRI="#string">agza</Literal>
</DataPropertyAssertion>
<DataPropertyAssertion>
  <DataProperty IRI="#esanlamlisi_kelime"/>
  <NamedIndividual IRI="#o3"/>
  <Literal IRI="#string">muharebe</Literal>
</DataPropertyAssertion>
<DataPropertyAssertion>
  <DataProperty IRI="#esanlamlisi_kelime"/>
  <Literal IRI="#string">savaş</Literal>
</DataPropertyAssertion>

```

Şekil 7. a) Protégé editörü üzerinde “Individual” sınıf elemanlarının görünümü

b) Bazı “Individual” özelliklerinin OWL formu.

e. Protégé Editörü ile SWRL Semantik Arama Kurallarının Yaratılması

Şekil 8’de, Arşiv_Ontology.owl ontolojisi içinde tanımlanan SWRL kuralları gösterilmiştir.

SWRL Kuralları:

```

belge(?b), osmanlica_sozluk(?o), user_keywords(?t),
b_keywords(?b, ?t), turemis_kelime(?o, ?t) ->
belge(?b), user_keywords(?uk), b_keywords(?b, ?uk) ->
arama_sonucu_deger(?b, ilgili)
belge(?b), osmanlica_sozluk(?o), user_keywords(?uk),
b_keywords(?b, ?b_kyw), esanlamlisi_kelime(?o, ?b_kyw),
turemis_kelime(?o, ?term), u_keywords(?uk, ?term) ->
arama_sonucu_deger(?b, ilgili)
belge(?b), osmanlica_sozluk(?o), user_keywords(?uk),
b_keywords(?b, ?b_kyw), esanlamlisi_kelime(?o, ?term),
turemis_kelime(?o, ?b_kyw), u_keywords(?uk, ?term) ->
arama_sonucu_deger(?b, ilgili)
belge(?b), osmanlica_sozluk(?o), user_keywords(?uk),
b_keywords(?b, ?t), turemis_kelime(?o, ?t), u_keywords(?uk, ?t) ->
arama_sonucu_deger(?b, ilgili)
belge(?b), osmanlica_sozluk(?o), user_keywords(?uk),
b_keywords(?b, ?e), esanlamlisi_kelime(?o, ?e), u_keywords(?uk, ?e)
-> arama_sonucu_deger(?b, ilgili)
belge(?b), osmanlica_sozluk(?o), user_keywords(?e),
b_keywords(?b, ?e), esanlamlisi_kelime(?o, ?e) ->
arama_sonucu_deger(?b, ilgili)

```

Şekil 8. Protégé editörü üzerinde SWRL semantik arama kurallarının görünümü.

Ontolojideki bu kurallar; kullanıcının girdiği arama kelimelerini, *Arsiv_Ontology.owl* ontolojisindeki Osmanlı terimler sözlüğü içinde eş anlamlısı, zıt anlamlısı, türemiş kelime vb. anlamsal bağlantılara erişir. Aynı zamanda, belge ontolojilerinin de içine girip, belgelere ait anahtar kelimeler içinde de arama yapmayı sağlar. Bu noktada, Şekil 8’de gösterilen kurallar çalıştırılıp ve “*ilgili*” belge sonucu çıkarsanır ise, o belgenin id’si sistem tarafında çekilir.

4.2. Belge Ontolojilerinin (B00000000X.owl) Yaratılması

Diğer ontoloji yapısı, Belge Ontolojileridir (B00000000X.owl). Bu ontolojiler, kurum üyesi tarafından, emsalsiz bir id (X id olsun) değeri ile, Şekil 9’daki gibi sunulan bir ara yüzden, belgenin detaylarını sisteme girer ve o anda yazılım tarafından B000000X.owl adlı bir ontoloji dosyası otomatik olarak üretilir. Bu ontoloji, sadece bu belgeye özel olup; ona ait verileri içerir, örneğin; *belge_kategori* nesne özelliğinin ne olacağı veya *belge_arsiv_sekli* özelliğinin ne olacağı; *sayısal mı*, *kopya basım mı* veya *kâğıt mı* vb. değerlerin tanımlandığı andır. Sürece göre; Vakıf’a aktarılan her kâğıt türündeki belgenin, kurum üyelerince, önce resmi taranıp sayısal arşiv veri tabanına yeni bir numara ile aktarılır. Bunun yanı sıra, Şekil 9’da ki gibi bir ekrandan, belge verileri aktarılırken, o anda bu belgenin özelliklerini makinalarca anlaşılabilir formatta, diğer deyişle OWL 2 formatına, dönüştürür.

T.C. BAŞBAKANLIK VAKIFLAR GENEL MÜDÜRLÜĞÜ	
KAYIT YAPAN PERSONEL NO	VP00135
TANIMLANACAK BELGE NO	B0000001
BELGENİN TARİHİ	01/01/1982
DİJİTALLEŞTİRME TARİHİ	15/02/2017
VAKIFA GELİŞ TARİHİ	14/02/2017
BELGENİN GELDİĞİ KURUM	Vakıflar Genel Müdürlüğü Merkez Kutuphanesi
BELGEYİ DÜZENLEYEN KURUM	Vakıflar Genel Müdürlüğü Merkez Kutuphanesi
BELGEYİ DÜZENLEYEN KİŞİ	Belirsiz
BELGE TARİH ARALIĞI	1982
TARİHİ DÖNEMİ	tömm_tc
BELGE DİLİ	türkçe
BELGE KATEGORİSİ	Belirsiz
BELGE TÜRÜ	kitap
BELGE ALT TÜRÜ	diğer_kitaplar
BELGE DEFTER TÜRÜ	Belirsiz
BELGE ÜLKE	tr_türkiye
BELGE BÖLGE	ic_anadolu
BELGE ŞEHİR	ankara
BELGE İLÇE	Belirsiz
BELGENİN YAZARI	Resat Yazıcı
BELGE EDITÖRÜ	Belirsiz
BELGENİN YAYINCISI	Belirsiz
SAYFA ADETİ	565
BELGE ÖN KAPAK	Belirsiz
BELGE ARKA KAPAK	Belirsiz
BELGE İLK SAYFA	Belirsiz
BELGE ERİŞİM HAKKI	herkese_acik
BELGE ARŞİV ŞEKLİ	dijital_resim
BELGE ORTAMI	dijital

BELGE ANAHTAR KELİMELER	
Türkiye Dış İlişkiler, Antlaşmalar, Ortadoğu Antlaşmalar, Türkiye Dış İlişkiler ve Ort...	

BELGE ÖN RESİM	

BELGE KLASÖR KOD NO (SıraNo)	
BELGE KLASÖR KOD NO	113
BELGE KLASÖR YENİ NO	K2016_113
BELGE KLASÖR ADI	Türkiye Dış İlişkiler
BELGE ORJİNAL NO	0003807
BELGE LİNK URL	vgm.gov.tr/yordambt/yordam.php?dilsecim=0
DOSYA TASNİF KODU	VGM
BELGE DOSYA NO	D341/YAZ
DOSYA GÖMLEK SIRA NO	Belirsiz
DOSYADAKİ BELGE ADETİ	12
DOSYADAKİ VARAK ADETİ	8
DOSYA AÇILIŞ TARİHİ	01/02/2010
DOSYA ORJİNAL NO	81/1
MİKRO FROM NO	0003807
DOSYA ANAHTAR KELİMELER	Türkiye Dış İlişkiler, Antlaşmalar

Şekil 9. Her bir belge ögesi bir belge ontoloji dosyasının otomatik oluşturulması.

Böylece, herhangi bir anlamsal arama motoru, bu belge üzerinde tanımlanmış çeşitli özelliklere göre değişik çapta anlamsal aramalar yapabilecektir. Hatta eldeki tanımlı bu veriler ile belge anlamsal aramaya uyarlanmış mevcut SWRL kural seti çalıştırılarak; bu belge için, yeni veriler çıkarsamaya olanak sağlar.

Şekil 9’da tanıtılan “B0000001.owl” belgesinin sistem tarafından o an da otomatik olarak üretilen OWL 2 dosyası EK B’de gösterilmiştir. Belge ontolojileri üzerinden, herhangi bir belgeye yönelik anlamsal sorgulamalar ve çıkarsamalar, yazılımlar tarafından yapılarak, kullanıcının ihtiyacı olan belge olup olmadığına karar verilir. Karar mekanizması için oluşturulan sistemin yapay zekâ kural setinin oluşturulması ve çalıştırılması hakkında bilgi gelecek bölümde bahsedilmiştir.

5. SİSTEMİN YAPAY ZEKÂ KURALLARININ PROTÉGÉ EDITÖRÜ ÜZERİNDE SWRL İLE YARATILMASI

SWRL, kural tanımlamak için W3C konsorsiyumuna sunulan bir Web standardı olup, tündengelim özelliğine sahip güçlü bir kural tanımlama dilidir. SWRL kuralları atomlardan oluşup, her bir atom; OWL ontoloji sınıflarından (OWL classes), (OWL object, annotation or datatype properties) özelliklerinden, OWL veri öğelerinden (OWL individuals) yapılarından oluşmaktadır. Geliştirilen kurallar, ontolojinin bir parçası olarak yerleşiktir. Web servislerinde konuşlanan bir çıkarsama yazılımı; önceden tanımlanmış bu kuralların üzerinde sorgulanan veri öğelerini yürütür. Çıkarsama yazılımları, ontoloji çıkarsama (reasoner) kütüphaneleri ile geliştirilmektedir; Pellet [23], Hermit [26], Fact++, RacerPro bunlardan bazılarıdır. Geliştirilen arama motorunda, AYSO bilgi tabanına gömülü kural tabanında önceden tanımladığımız kurallara göre “*ilgili*” belge sonuç çıkarsaması yapılmıştır. Bu nedenle, Java ortamında geliştirilen sistemin arama motoru servisi yazılımında, Stanford Üniversitesinin geliştirmiş olduğu Pellet Reasoner API kütüphanesi kullanılmıştır. Pellet çıkarsayıcısı, SWRL kurallarını, DL-Safe kavramlarını kullanarak yorumlayabilmektedir. Bunun anlamı, çıkarsama esnasında, ontolojiye yüklenen veri öğeleri (individuals), SWRL kuralları üzerinde yürütülür ve yeni veri çıkarsaması yapılabilmektedir.

Sistemin kurallarını anlatmaya başlamadan önce; kısaca SWRL kural yapısını anlamak gerekir, daha sonra sistemin SWRL tabanlı çıkarsama mekanizmasının detayları sunulacaktır. SWRL yapısında ‘ \Rightarrow ’ sembolü kullanılmaktadır ve; ‘ \Rightarrow ’ sembolünden önce mantıksal bağlaçlarla birleştirilmiş bir Antecedent (Öncül) ve, ‘ \Rightarrow ’ sembolünden sonra yine mantıksal bağlaçlarla birleştirilmiş Consequent (Kural Sonucu) içermektedir. Aslında bu kurallar daha çok 'If-Then (Eğer-İse)' mantığını içermektedirler. Örneğin, Eşitlik 1’de ki gibi bir kural;

$$A (? x) \wedge B (? x) \Rightarrow C (? x) \quad (1)$$

'Eğer x değişkeni A sınıfından ise ve Eğer x değişkeni B sınıfından ise \Rightarrow çıkarsama sonucu: x değişkeni ayrıca C sınıfındandır' anlamına gelmektedir. Yukarıdaki örnekte, Antecedent (Öncül) kısmında pozitif bağlaçlar taşımaktadır. Diğer deyişle, atomların birleşimi içermektedir; $atom \wedge atom \dots \rightarrow atom \wedge atom$. Bir atom, Yüklem (Predicate) sembolü içerebilir, örneğin P diye ifade edelim. Ayrıca, bazı parametreler içerebilir, örneğin; $par_1, par_2, \dots par_n$. Yüklem sembolü olan P ; OWL sınıflarını, nesne özelliklerini veya veri öğelerini (OWL individuals) temsil edebilir. P bazı parametreler içerebilir; veri öğeleri veya değişkenler gibi: $P(par_1, par_2, \dots par_n)$. Tablo 3' de sistemin geliştirilen kural tabanından (7 adet SWRL kuralı - bkz. Tablo 2, Kural 1), genel SWRL yapılarını içeren bir örnek paylaşılmıştır;

Tablo 3. KURAL-1 adlı SWRL kuralının yapısal açıklaması

KURAL-1: belge(?b), osmanlica_sozluk(?o), turemis_kelime(?o, ?t), user_keywords(?t), b_keywords(?b, ?t) -> arama_sonucu_deger(?b, ilgili)

belge(?b): '?b' bir belge ögesi ise ve,

osmanlica_sozluk(?o): '?o' bir Osmanlıca kelime ögesi ise ve,

turemis_kelime(?o, ?t): '?o' kelime ögesi için tanımlanmış türemiş kelimeler (?t) var ise ve,

b_keywords(?b, ?t): '?b' belgesine tanımlanmış anahtar kelimeler varsa ve bunlardan en az biri '?t' türemiş kelimesi ile aynı ise ve,

user_keywords (?t): kullanıcının girdiği en az bir kelime bu '?t' türemiş kelimelerden en az biri ile aynı ise ve,

arama_sonucu_deger(?b, ilgili): '?b' belgesi için döndürülecek arama sonucu **"ilgili"** olacaktır.

Daha öncede bahsedildiği gibi, her kural, atomların mantıksal bağlaçlarla birleşiminden oluşmaktadır. Örneğin, bir **<ClassAtom>** etiketi ve bir değişken etiketi **<Variable IRI="urn:swrl#b"/>** (*b* değişkeni tüm belge öğeleri için tanımlandı) kullanılarak, **belge (?b)** şeklinde gösterilmektedir (EK C'de 3-6.ncı satırlar). Benzer mantıkla, birçok SWRL etiketi ile OWL formundaki kurallar makinaların anlayacağı formatta oluşturulabilmektedir:

<ObjectPropertyAtom>

<ObjectProperty IRI="#arama_sonucu_deger"/>

<Variable IRI="urn:swrl#b"/>

<NamedIndividual IRI="#ilgili"/>

</ObjectPropertyAtom>

Tablo 3’de bahsedilen SWRL kuralının OWL 2.0 formu EK C’deki tablonun sol tarafında verilmiştir. Sağ tarafında ise başka bir SWRL kuralı, türemiş kelimeyi bulmak yerine, eş anlamlı kelimeyi bulmaya dayalı farklı bir kuraldır.

Kullanıcı tarafından sorgu esnasında girilen arama kelimeleri; yukarıda bahsi geçen arama kuralları üzerinde çalıştırılacak; daha sonra, kural sonucunda her belge için bir “*ilgili*” belge veya “*ilgisiz*” belge çıkarsama sonucu döndürülmektedir. Sistemde milyonlarca belge olması halinde bile, her belgenin kendisine ait ontoloji dosyası olduğundan ve o anda bu dosyalardaki sadece ilgili etiketlere giriş yapılacağından, sistemin normal çalışma hızını çok kötü oranda etkilemeyeceğine inanıyoruz. Sistemin performans çalışmaları için gerekli çalışmalara devam edilmesi ve araştırma sonuçlarının gelecek makalelerde yayınlanması planlanmaktadır. Ayrıca, gelecek makale çalışmasında, bu kuralları çalıştıran sistemin arama motoru için yaratılan yazılımların detaylı olarak yayınlanması planlanmaktadır.

6. SONUÇLAR

Ülkelerin Vakıflar Genel Müdürlüğü idaresinde bulunan milyonlarca adet belgenin, hızlı ve güvenilir şekilde elektronik ortama aktarılması, elektronik ortamda yönetilmesi ve diğer kurumlar ile paylaşılması için Sayısal Arşiv Yönetim Sistemleri (SAYS) kullanılır. Bu tür sistemlerde, en önemli fonksiyonlardan biri, kurum işleyişi ve düzeni için ilgili belge/belgeleri doğru ve hızlı bir şekilde aramaktır. Vakıflarda verilen hizmet gereği, elzem olan SAYS’lerinde, milyonlarca belge içinden, istenilen belgenin en kısa zamanda, doğru ve akıllı bir üslupla aranması önemli bir işlemdir. Ülkemizde ve günümüzde, bu sahadaki arama süreçlerinde, yapay zekâ çözümlerine nadiren rastlanmaktadır. Bu nedenle, bu çalışmada, Vakıflar Genel Müdürlüklerinin kullandığı mevcut SAYS’ne entegre olabilecek yapay zeka tabanlı geliştirilmiş ek bir arama modülü önerilmiştir.

Bu tür sistemlerin modellenmesinde, içerik tabanlı arama yapabilmek için, bilgi tabanı kullanımına ihtiyaç duyulmaktadır. İçeriklerin kavramlaştırılması ve kavramlar arası özelliklerin önceden tanımlanması ancak bilgi tabanları ile yapılabilmektedir. Makinaların anlayacağı şekilde yorumlanmış anlamsal bilgi tabanlarının kullanımı sayesinde, geliştirilen yazılımlar anlamlı çıkarımlar yapabilmektedir. Bu nedenle, anlamsal bilgi tabanları üzerinde kurgulanmış yazılımlar, kullanıcılara aradıklarını sunarken daha mantıklı ve ilgili sonuçlar getirebilmektedir.

AV teknolojisinin en önemli katkısı olan Ontoloji Bilgi Tabanları, Web içeriklerinin sadece doğal dillerde değil, aynı zamanda ilgili yazılımlar tarafından anlaşılabilir, yorumlanabilir ve

kullanılabilir bir biçimde ifade edilebileceği, böylece bu yazılımların veriyi kolayca bulmasını, paylaşmasını ve bilgiyi birleştirmesini sağlamayı amaçlayan, gelişen bir internet eklentisi olarak bilinmektedir. Bu nedenle, bu çalışmada, kullanıcıların ihtiyaç duyduğu bir belge veya dokümanı bulmak için, mevcut SAYS yazılımlarına entegre olabilecek, arşivleme alanına özelleştirilmiş bir ontoloji bilgi tabanı geliştirilerek, yararlı ve yenilikçi bir yaklaşım ortaya konulmuştur. Böylece, sadece dokümanın arşivlendiği dosya, envanter numarası vb. bilgilerin aranması yerine içerik bakımından da arama sağlanmıştır.

Sistemin ontoloji bilgi tabanı olan Arşiv Yönetim Sistemi Ontolojisi (AYSO) iki alt tip ontoloji yapısı içermektedir. Birinci ontoloji; AYSO genel yapısını taşıyan *Arsiv_Ontology.owl* adlı genel bir ontolojidir (Upper Ontology). İkinci tip ontoloji ise; *Belge Ontolojileri*'dir. Kurumun sayısal arşivinde bulunan her belgenin, kendisine has bir ontoloji dokümanı olup — *o belgeyi tanımlayan özellik tanımları, içeriğine yönelik ilgili/anahtar kelimeleri ve diğer bağlı olduğu anlamsal özellik tanımları içermesi* — sayesinde o belge üzerinde yapılan bir arama için, doğru belge olup olmadığına karar verilmesi sağlanır. Geliştiren bir yazılım yardımıyla, SAYS üzerinde yapılan bir anlamsal arama sonucunda, sadece **“ilgili”** sonucunu içeren arşiv belgeleri ve id'leri bir liste olarak kullanıcıya sunulmaktadır. Kurum üyesi, bu listeden uygun bulduğu ve seçtiği belgenin; *kayitEdenPersonelNo, tasnifKodu, belgeYeniKlasorKodNo, orjinalKayitNo, belgeKlasorunAdi, tarihAraligi, dijitalizeTarihi, gomlekSiraNo, mikroFormNo, url, vakifaGelisTarihi String, yayınlanmaTarihi, belge keywords* vb. diğer tüm bağlı verisine aynı ara yüz üzerinden ulaşabilmektedir.

Sistem ontolojisinin, KKTC Vakıflar Genel Müdürlüğü'nün yakın gelecekte planladığı SAYS projesindeki belge arama motorunun sunulduğu ara yüzün arkasında eklenmesi planlanmaktadır. Sonuç olarak, yapılan değerlendirmeler sonucunda, Vakıflar Genel Müdürlüğü idaresinde bulunan kütüphaneler veya kurumlar genelde arama motorlarını sözdizimsel yapıda modellediklerinden ilgisiz belgeler döndürebilmekte veya ilgili belgeleri anlam bakımından bakmadığı için kaçırmaktadır. Bu çalışmada, sunulan ASYO bilgi tabanı üzerinde konuşlanacak olan herhangi bir anlamsal arama motoru sayesinde; belge sorgusuna girilen arama kelimelerinin, ontolojideki mevcut kavramsal bağlara (diğer deyişle; türemiş kelime özelliğine, eş anlamlı özelliğine, tekil-çoğul vb. anlam ilişkilerine) göre arama yapmasına olanak sağlayacaktır. Böylece, ilgili belgeler ıskalanmayacak ve dönen **“ilgili”** belge sayısı diğer sözdizimsel arama yapan çevrimiçi kütüphanelere göre daha fazla olabilecektir.

KAYNAKLAR

- [1] O. Menemencioglu, İ. M. Orak, *Semantik Web Arama Motoru Üzerine Bir İnceleme*. ISITES 2014: 2nd International Sym. On Innovative Technologies in Engineering and Science, (2014) Karabuk, Turkey. Accessed February 2016.
- [2] D. E. Rose, D. Levinson, *Understanding user goals in Web search*. 13th international conference on World Wide Web, New York, (2004) 13-19.
- [3] M. Levene, *An introduction to search engines and Web navigation*. John Wiley & Sons, (2011).
- [4] M. d'Aquin, L. Ding, E. Motta, *Semantic Web Search Engines*. In Handbook of Semantic Web Technologies, Springer, (2011) 659-700.
- [5] G. Madhu, D.A. Govardhan, D.T. Rajinikanth, *Intelligent Semantic Web Search Engines: A Brief Survey*. arXiv preprint arXiv:1102.0831, (2011).
- [6] T. O'reilly, *What Is Web 2.0*. (2005).
- [7] T. Berners-Lee, J. Hendler, O. Lassila, *The semantic web*. Scientific American 284.5, (2001) 28-37.
- [8] T. Gruber, *Ontology*. Encyclopedia of Database Systems (Springer-Verlag). Liu, Ling; Özsu, M. Tamer, eds. ISBN 978-0-387-49616-0, <http://tomgruber.org/writing/ontology-definition-2007.htm>, (2008).
- [9] B. Chandrasekaran, J. R. Josephson, V. R. Benjamins, *What are ontologies, and why do we need them?*. IEEE Intelligent Systems and their applications 14.1, (1999) 20-26.
- [10] *The MUNIN Project-An overview*. MUNIN Final Event, Hamburg, Germany, Dipl.-Wirtsch.-Ing. University MUNIN Project Coordinator:Hans-Christoph Burmeister, (2015).
- [11] E. Ahonen, E. Hyvonen, *Publishing historical texts on the semantic web-a case study*. In: Semantic Computing, ICSC'09. IEEE International Conference on, IEEE, (2009) 167-173, IEEE. DOI: 10.1109/ICSC.2009.9
- [12] E. Hyvonen, K. Viljanen, J. Tuominen, K. Seppälä, *Building a national semantic web ontology and ontology service infrastructure—the FinnONTO approach*. In: European Semantic Web Conference. Springer Berlin Heidelberg, (2008) 95-109.
- [13] O. Lassila, R.S. Ralph, *Resource Description Framework (RDF). Model and Syntax*. W3C Recommendation, <https://www.w3.org/TR/WD-rdf-syntax-971002/>, Version 1, February (1997), Accessed April 2017.
- [14] G. Adorni, M. Maratea, L. Pandolfo, L. Pulina, *An ontology for historical research documents*. In: International Conference on Web Reasoning and Rule Systems, Springer International Publishing, (2015) 11-18. DOI: 10.1007/978-3-319-22002-4 2.
- [15] B. Arslan, *İçişleri Bakanlığı Nüfus ve Vatandaşlık İdaresi Dijital Arşiv Projesi (NVİ DAP)*. <http://www.erkyazilim.com.tr/blog/konu/projeler>. Accessed April 2017.
- [16] S. Nešić, *Semantic document model to enhance data and knowledge interoperability*. In: Web 2.0 & Semantic Web. Springer US, (2010) 135-160.
- [17] OWL, Web Ontology Language Overview., W3C Recommendation, Deborah L. McGuinness, Frank van Harmelen, Stanford University, Online: <https://www.w3.org/TR/owl-features/>, (2004) Accessed February 2016.

- [18] Protégé OWL Ontology Editor. Protégé 4. 1 tool website, Stanford University. <http://protege.stanford.edu/>. Accessed 30 Jan 2016.
- [19] OWL 2.0, OWL 2 Web Ontology Language Document Overview. W3C Recommendation, Online: <http://www.w3.org/TR/owl2-overview/>, 11 December 2012, Accessed February 2016.
- [20] Netbeans Java Framework, <https://netbeans.org/downloads/8.0/>, Accessed April 2017.
- [21] M. Horridge, S. Bechhofer, *The OWL API: a Java API for working with OWL 2 ontologies*. Proceedings of the 6th International Conference on OWL: Experiences and Directions- Volume 529. CEUR-WS. org, (2009).
- [22] SWRL, A Semantic Web Rule Language Combining OWL and RuleML, W3C Member Submission 21 May 2004. Accessed February 2016.
- [23] E. Sirin, B. Parsia, B.C. Grau, A. Kalyanpur, Y. Katz, *PELLET: A practical owl-dl reasoner*. Web Semantics: science, services and agents on the World Wide Web, (5) (2007) 51-53.
- [24] W.E. Winkler, *The state of record linkage and current research problems*. Technical report, U.S. Census Bureau, (1999).
- [25] W.E. Winkler, Y. Thibaudeau, *An application of the Fellegi-Sunter model of record linkage to the 1990 U.S. decennial census*. Technical report, U.S. Bureau of the Census, Washington, D.C., Statistical Research Report Series RR91/09, (1991).
- [26] R. Shearer, B. Motik, I. Horrocks, *Hermit: A highly-efficient OWL reasoner*. In: OWLED, (432) (2008).

EK A: AYSO bilgi tabanı Arsiv_Ontology.owl dosyasından kesit

"ARSIV_ONTOLOGY.OWL" : SISTEMIN UPPER ONTOLOJİ BİLGİ TABANI

1	<Ontology	48	<Declaration>
2	xml:base="/aydin.edu.tr/ArsivOntology"/>	49	<DataProperty IRI="#d_gomlekSiraNo"/>
3	<Declaration>	50	</Declaration>
4	<Class IRI="#ahkam_defterleri"/>	51	<Declaration>
5	</Declaration>	52	<DataProperty IRI="#d_keywords"/>
6	<Declaration>	53	</Declaration>
7	<Class IRI="#arama_sonuc"/>	54	<Declaration>
8	</Declaration>	55	<DataProperty IRI="#d_mikroFormNo"/>
9	<Declaration>	56	</Declaration>
10	<Class IRI="#belge"/>	57	<Declaration>
11	</Declaration>	58	<DataProperty IRI="#d_orijinalNo"/>
12	<Declaration>	59	</Declaration>
13	<Class IRI="#belge_arsiv_sekli"/>	60	<Declaration>
14	</Declaration>	61	<DataProperty IRI="#d_tasnifKodu"/>
15	<Declaration>	62	</Declaration>
16	<Class IRI="#belge_kategory"/>	63	<Declaration>
17	</Declaration>	64	<DataProperty IRI="#d_varakAdeti"/>
18	<Declaration>	65	</Declaration>
19	<Class IRI="#belge_ortam"/>	66	<Declaration>
20	</Declaration>	67	<DataProperty IRI="#esAnlamlisi_kelime"/>
21	<Declaration>	68	</Declaration>
22	<Class IRI="#belge_tur"/>	69	<ClassAssertion>
23	</Declaration>	70	<Class IRI="#belge_tur"/>
24	<Declaration>	71	<NamedIndividual IRI="#tez"/>
25	<ObjectProperty IRI="#arama_sonucu_deger"/>	72	</ClassAssertion>
26	</Declaration>	73	<ClassAssertion>
27	<Declaration>	74	<Class IRI="#belge_arsiv_sekli"/>
28	<ObjectProperty IRI="#b_altTuru"/>	75	<NamedIndividual IRI="#tipki_basim"/>
29	</Declaration>	76	</ClassAssertion>
30	<Declaration>	77	<ClassAssertion>
31	<ObjectProperty IRI="#b_arsivSekli"/>	78	<Class IRI="#belge_kategory"/>
32	</Declaration>	79	<NamedIndividual IRI="#trikomo"/>
33	<Declaration>	80	</ClassAssertion>
34	<ObjectProperty IRI="#b_dili"/>	81	<ClassAssertion>
35	</Declaration>	82	<Class IRI="#dil"/>
36	<Declaration>	83	<NamedIndividual IRI="#turkce"/>
37	<ObjectProperty IRI="#b_dosyasiKodNo"/>	84	</ClassAssertion>
38	</Declaration>	85	<ClassAssertion>
39	<Declaration>	86	<Class IRI="#vakif_defter_turleri"/>
40	<ObjectProperty IRI="#b_duzenleyenKurum"/>	87	<NamedIndividual
41	</Declaration>	88	IRI="#vakfiye_defterleri"/>
42	<Declaration>	89	</ClassAssertion>
43	<ObjectProperty IRI="#b_erisimHakki"/>	90	<ClassAssertion>
44	</Declaration>	91	<Class IRI="#erisim_haklari"/>
45	<Declaration>	92	<NamedIndividual IRI="#yayinciya_acik"/>
46	<ObjectProperty IRI="#b_geldigiKurum"/>	93	</ClassAssertion>
47	</Declaration>	94	<ClassAssertion>

EK B: “B0000001” belgesinin (“B0000001.owl”) ontolojisinin OWL 2 formundan kesit

“B0000001.OWL” : SISTEME KAYDEDILEN YENI BIR BELGENIN ONTOLOJISI

1	<?xml version="1.0"?>	52	<ObjectPropertyAssertion>
2	<Ontology xml:base=""/.../ArxivOntology"/>	53	<ObjectProperty IRI="#b_ortami"/>
3	<ClassAssertion>	54	<NamedIndividual IRI="#B0000001"/>
4	<Class IRI="#belge"/>	55	<NamedIndividual IRI="#dijital"/>
5	<NamedIndividual IRI="#B0000001"/>	56	</ObjectPropertyAssertion>
6	</ClassAssertion>	57	<DataPropertyAssertion>
7	<ObjectPropertyAssertion>	58	<DataProperty IRI="#b_keywords"/>
8	<ObjectProperty IRI="#b_erisimHakki"/>	59	<NamedIndividual IRI="#B0000001"/>
9	<NamedIndividual IRI="#B0000001"/>	60	<Literal datatypeIRI="#string">Türkiye Dış
10	<NamedIndividual IRI="#herkese_acik"/>	61	İlişkiler Antlaşmalar</Literal>
11	</ObjectPropertyAssertion>	62	</DataPropertyAssertion>
12	<ObjectPropertyAssertion>	63	<DataPropertyAssertion>
13	<ObjectProperty IRI="#b_geldigiKurum"/>	64	<DataProperty IRI="#b_keywords"/>
14	<NamedIndividual IRI="#B0000001"/>	65	<NamedIndividual IRI="#B0000001"/>
15	<NamedIndividual IRI="#vakiflar_genel_mudurlugu_	66	<Literal
16	merkez_kutuphanesi"/>	67	datatypeIRI="#string">Antlaşmalar</Literal>
17	</ObjectPropertyAssertion>	68	</DataPropertyAssertion>
18	<ObjectPropertyAssertion>	69	<DataPropertyAssertion>
19	<ObjectProperty	70	<DataProperty IRI="#b_keywords"/>
20	IRI="#b_kayitEdenPersonelNo"/>	71	<NamedIndividual IRI="#B0000001"/>
21	<NamedIndividual IRI="#B0000001"/>	72	<Literal datatypeIRI="#string"> Türkiye Dış
22	<NamedIndividual IRI="#vp00135"/>	73	İlişkiler </Literal>
23	</ObjectPropertyAssertion>	74	</DataPropertyAssertion>
24	<ObjectPropertyAssertion>	75	<DataPropertyAssertion>
25	<ObjectProperty	76	<DataProperty IRI="#b_keywords"/>
26	IRI="#b_klasorununKodNo"/>	77	<NamedIndividual IRI="#B0000001"/>
27	<NamedIndividual IRI="#B0000001"/>	78	<Literal datatypeIRI="#string"> Türkiye Dış
28	<NamedIndividual IRI="#113"/>	79	İlişkiler ve Ortadoğu</Literal>
29	</ObjectPropertyAssertion>	80	</DataPropertyAssertion>
30	<ObjectPropertyAssertion>	81	<DataPropertyAssertion>
31	<ObjectProperty IRI="#b_ortami"/>	82	<DataProperty IRI="#b_keywords"/>
32	<NamedIndividual IRI="#B0000001"/>	83	<NamedIndividual IRI="#B0000001"/>
33	<NamedIndividual IRI="#dijital"/>	84	<Literal datatypeIRI="#integer">565</Literal>
34	</ObjectPropertyAssertion>	85	</DataPropertyAssertion>
35	<ObjectPropertyAssertion>	86	<DataPropertyAssertion>
36	<ObjectProperty IRI="#b_sehir"/>	87	<DataProperty IRI="#b_tarihi"/>
37	<NamedIndividual IRI="#B0000001"/>	88	<NamedIndividual IRI="#B0000001"/>
38	<NamedIndividual IRI="#ankara"/>	89	<Literal
39	</ObjectPropertyAssertion>	90	datatypeIRI="#string">01/01/1982</Literal>
40	<ObjectPropertyAssertion>	91	</DataPropertyAssertion>
41	<ObjectProperty IRI="#b_tarihiDonemi"/>	92	<DataPropertyAssertion>
42	<NamedIndividual IRI="#B0000001"/>	93	<DataProperty IRI="#b_vakifaGelisTarihi"/>
43	<NamedIndividual IRI="#tbmm_tc"/>	94	<NamedIndividual IRI="#B0000001"/>
44	</ObjectPropertyAssertion>	95	<Literal
45	<ObjectPropertyAssertion>	96	datatypeIRI="#string">14/02/2017</Literal>
46	<ObjectProperty IRI="#b_turu"/>	97	</DataPropertyAssertion>
47	<NamedIndividual IRI="#B0000001"/>	98	<DataPropertyAssertion>
48	<NamedIndividual IRI="#kitap"/>		<DataProperty IRI="#b_yeniKlasorKodNo"/>
49	</ObjectPropertyAssertion>		<NamedIndividual IRI="#B0000001"/>
50	<ObjectPropertyAssertion>		<Literal
51	<ObjectProperty IRI="#b_yazari"/>		datatypeIRI="#string">k2016_113</Literal>
	<NamedIndividual IRI="#B0000001"/>		</DataPropertyAssertion>
	<NamedIndividual IRI="#resat_yazici"/>		<DataPropertyAssertion>
	</ObjectPropertyAssertion>		<DataProperty IRI="#d_acilisTarihi"/>

EK C: Arşiv_ontology.owl içinde mevcut iki SWRL kuralının OWL 2.0 formları

TÜREMİŞ KELİME BULMAK İÇİN ÇIKARSAMA KURALI		EŞ ANLAMLI KELİME BULMAK İÇİN ÇIKARSAMA KURALI	
KURAL-1: belge(?b), osmanlica_sozluk(?o), turemis_kelime(?o, ?t), user_keywords(?t), b_keywords(?b, ?t) -> arama_sonucu_deger(?b, ilgili)		KURAL-2: belge(?b), osmanlica_sozluk(?o), user_keywords(?uk), b_keywords(?b, ?e), esanlamlisi_kelime(?o, ?e), u_keywords(?uk, ?e) -> arama_sonucu_deger(?b, ilgili)	
1	<DLSafeRule>	1	<DLSafeRule>
2	<Body>	2	<Body>
3	<ClassAtom>	3	<ClassAtom>
4	<Class IRI="#belge"/>	4	<Class IRI="#belge"/>
5	<Variable IRI="urn:swrl#b"/>	5	<Variable IRI="urn:swrl#b"/>
6	</ClassAtom>	6	</ClassAtom>
7	<ClassAtom>	7	<ClassAtom>
8	<Class IRI="#osmanlica_sozluk"/>	8	<Class IRI="#osmanlica_sozluk"/>
9	<Variable IRI="urn:swrl#o"/>	9	<Variable IRI="urn:swrl#o"/>
10	</ClassAtom>	10	</ClassAtom>
11	<ClassAtom>	11	<ClassAtom>
12	<Class IRI="#user_keywords"/>	12	<Class IRI="#user_keywords"/>
13	<Variable IRI="urn:swrl#t"/>	13	<Variable IRI="urn:swrl#uk"/>
14	</ClassAtom>	14	</ClassAtom>
15	<DataPropertyAtom>	15	<DataPropertyAtom>
16	<DataProperty IRI="#b_keywords"/>	16	<DataProperty IRI="#b_keywords"/>
17	<Variable IRI="urn:swrl#b"/>	17	<Variable IRI="urn:swrl#b"/>
18	<Variable IRI="urn:swrl#t"/>	18	<Variable IRI="urn:swrl#e"/>
19	</DataPropertyAtom>	19	</DataPropertyAtom>
20	<DataPropertyAtom>	20	<DataPropertyAtom>
21	<DataProperty	21	<DataProperty
22	IRI="#turemis_kelime"/>	22	IRI="#esanlamlisi_kelime"/>
23	<Variable IRI="urn:swrl#o"/>	23	<Variable IRI="urn:swrl#o"/>
24	<Variable IRI="urn:swrl#t"/>	24	<Variable IRI="urn:swrl#e"/>
25	</DataPropertyAtom>	25	</DataPropertyAtom>
26	</Body>	26	<DataPropertyAtom>
27	<Head>	27	<DataProperty IRI="#u_keywords"/>
28	<ObjectPropertyAtom>	28	<Variable IRI="urn:swrl#uk"/>
29	<ObjectProperty	29	<Variable IRI="urn:swrl#e"/>
30	IRI="#arama_sonucu_deger"/>	30	</DataPropertyAtom>
31	<Variable IRI="urn:swrl#b"/>	31	</Body>
32	<NamedIndividual IRI="#ilgili"/>	32	<Head>
33	</ObjectPropertyAtom>	33	<ObjectPropertyAtom>
34	</Head>	34	<ObjectProperty
		35	IRI="#arama_sonucu_deger"/>
		36	<Variable IRI="urn:swrl#b"/>
		37	<NamedIndividual IRI="#ilgili"/>
		38	</ObjectPropertyAtom>
		39	</Head>
		40	</DLSafeRule>