Düzce Üniversitesi Bilim ve Teknoloji Dergisi

Derleme Makalesi

Dijital Kütüphanelerde Dokümanlardan Bilgi Geri Kazanımı için Kullanılan Güncel Teknolojiler: Derleme Çalışması

Alev MUTLU [a], Mohamed Amin ABDISAMAD [a], Osman KABASAKAL [a], Furkan GÖZ [a], Öztürk TÜFEKÇİ [b], Kerem KÜÇÜK [a,*]

[a] *Bilgisayar Mühendisliği Bölümü, Mühendislik Fakültesi, Kocaeli Üniversitesi, Kocaeli, TÜRKİYE*
[b] *Yapı Kredi Teknoloji, Yapı Kredi Bankacılık Üssü Kocaeli, TÜRKİYE*
** Sorumlu yazarın e-posta adresi: kkucuk@kocaeli.edu.tr*
DOI : 10.29130/dubited.796964

ÖZET

Son yıllarda, farklı konular için sunulan dijital bilgi kaynaklarının sayısı aşırı miktarda artmaktadır. Bu dijital bilgi kaynaklarına erişim desteği sunan sistemlerin birçoğu tarama, arama ve bilgi geri kazanımı araçlarına odaklanmıştır. Sayısal kütüphaneler, elektronik kitaplıklar ve Web sayfaları, bilgi erişimini iyileştirmek, belge koleksiyonlarını farklı anahtar kriterlere göre hiyerarşik olarak oluşturmak ve düzenlemek için yeni birçok açılım sunmaktadır. Farklı arama araçları, bilgi erişim teknikleri kullanılarak erişilebilen belgeleri düzenlemek, endekslemek ve özetlemek için yazılım tabanlı hizmetleri kullanarak daha kapsamlı bir doküman kapsamı sunulabilmektedir. Dijital kütüphanelerdeki arama mekanizmalarına uygulanan teknolojiler, doküman koleksiyonlarını yönetmek, anlamlı veri çıkarmak ve doküman ilişkilerinin belirlenmesi için farklı yöntem ve teknolojilerin kullanımını zorunlu kılmıştır. Özellikle belgeler arasındaki ilişki ne biçimleri ne de türleri ile açıkça tanımlanamamaktadır. Bu çalışma, sayısal kütüphaneler için belgelerin içeriğinden üst-veri çıkarımı, varlık isimlerinin elde edilmesi, anahtar kelimelerin elde erilmesi ve doküman benzerliklerinin oluşturulması için kullanılan yöntem ve teknikler için kapsamlı bir çalışma sunmaktadır.

Anahtar Kelimeler: *Doküman işleme, Üst veri çıkarımı, Varlık ismi tanıma, Anahtar kelime çıkarımı, Doküman benzerliği*

Current Technologies for Information Retrieval of Documents in Digital Libraries: A Survey

ABSTRACT

In recent years, the number of digital information sources available for different topics has grown enormously. Many of the systems that support access to these digital information resources are focused on scanning, searching and information retrieval tools. Digital libraries, electronic libraries and Web pages bring many new initiatives to improve information access, create and organize document collections hierarchically according to different key criteria. Different search tools; by using software-based services to organize, index and summarize documents that can be accessed using information retrieval techniques, a more comprehensive document coverage can be provided. In digital libraries; the techniques applied to search mechanisms have made it necessary to use different methods and technologies to manage document collections, to extract meaningful data and to determine document relationships. In particular, the relationship between documents cannot be clearly defined neither by their forms nor by their types. This study provides a comprehensive study of methods and techniques used for extracting

Geliş: 18/09/2020, Düzeltme: 24/11/2020, Kabul: 25/11/2020

metadata, named entity recognition, keyword extraction and obtaining document similarities from the content of the documents for digital libraries.

Keywords: *Document processing, Metadata extraction, Name entity recognition, Keyword extraction, Document similarity*

I. GİRİŞ

Elektronik kütüphane, bilginin elektronik olarak üretilip saklandığı ve kullanıma sunulduğu ortam olarak tanımlanır [1]. Bilgi teknolojilerinin gelişmesi ve İnternet erişiminin artması ile birlikte bu kavram farklı meslek gruplarının ve kişilerin ilgisini çekmiş ve çevrimiçi kütüphane, sanal kütüphane, masaüstü kütüphane, doküman yönetim sistemi gibi farklı isimlerle anılmaya başlanmıştır. Her ne kadar bu terimler farklı gruplara hitap eden ve de barındırdıkları dokümanların çeşidi ve boyutu bakımından farklı büyüklükte olsa da tüm bu yapıların sağlaması gereken ortak bazı özellikler vardır. Bu ortak özellikler doküman isimlendirme, üst veri çıkarımı, dokümana erişim kısıtlarının belirlenmesi, farklı doküman tipi ile çalışabilme ve doküman arama olarak sıralanabilir [2].

Doküman işlemede en temel problemlerden biri üst veri çıkarımıdır. Üst veri, bir veri kaynağının içerisindeki parmak izi verisi olarak tanımlanabilen anahtar bilgi kümesini içermektedir. Üst veri, herhangi bir dijital kütüphaneden bilgi alma işlemi sırasında arama motorlarına veya kişilere ilgisiz belgelerden gerekli ayrımı yapma konusunda yardımcı olarak kaynak keşfini önemli ölçüde geliştirmektedir. Üst verilerin önemi açıkça bilinse de verimli ve etkili uygulama araçları geliştirme çalışmaları halen devam etmektedir. Üst veri çıkarımı, dijital kütüphanelerdeki yüksek seviyedeki büyüme ve birçok farklı üst veri standardının geliştirilmesi nedeniyle karmaşık bir problemdir. Üst veri çıkarımı gerektiren çok sayıda dijital kaynak göz önüne alındığında, insan eliyle oluşturulan üst veri yaklaşımlarına güvenmek gerçekçi değildir.

Varlık ismi tanıma (VİT), bir dokümandan kişi, yer, organizasyon, tarih, para birimi gibi bilgilerin otomatik olarak çıkarılması işlemidir [3]. VİT sistemleri genellikle soru yanıtlamada, bilgi almada, ortak referans çözümlemede ve konu modellemede ilk adım olarak kullanılır. Bu alandaki ilk çalışmalar kural ve ontoloji tabanlı yöntemleri kullanırken, daha sonra makine öğrenmesi ve günümüzde de derin öğrenme yöntemleri sıklıkla kullanılmaktadır [3]. Bu nedenle, VİT alanındaki son gelişmeleri, özellikle minimum özellik mühendisliği ile son teknoloji ürünü performansa ulaşan makine ve derin öğrenme tabanlı VİT mimarilerini vurgulamak önemlidir.

Metin analizi işlemleri, otomatik anahtar kelime çıkarma ve metin özetleme gibi görevleri de içermektedir. Otomatik anahtar kelime çıkarma, modele bağlı olarak herhangi bir insan müdahalesi olmaksızın belgenin temel içeriğini en iyi şekilde yansıtabilen kelime veya kelime öbeklerinin seçilmesi işlemidir [4]. Otomatik anahtar kelime çıkarmanın hedefi, mevcut hesaplama becerilerinin, gücünün ve hızının dokümana erişim ve geri çağırma sorununa ek maliyetleri ortadan kaldırabilecek bir yaklaşım sunmasıdır. Bu alandaki ilk çalışmalar dokümanlar için hazır sözlüklerden anahtar kelime atamasına dayanırken, son dönemlerde daha çok çizge tabanlı gözetimsiz öğrenmeye dayalı anahtar kelime yöntemleri üzerinde çalışılmaktadır [5].

Dokümanlar arası benzerliklerin hesaplanması dijital kütüphanelerin bir diğer önemli işlevidir. Bu benzerliklerin hesaplanmasında üst veri, varlık isimleri ve anahtar kelimeler kullanılabilirken dokümanının tüm içeriği de bu hesaplama için kullanılabilir. Literatürde bu amaçla kullanılan Jaccard ve kosinüs benzerliği gibi basit istatistiksel yöntemlerin yanında kelime temsillerine dayanan yeni yaklaşımlar da mevcuttur.

Özellikle Türkçe literatür oluşturma noktasında katkı sunmayı amaçlayan bu çalışmada dijital kütüphanede kullanılacak metin işleme teknolojilerinin literatür araştırması sunulmaktadır. Üst veri çıkarımı, metin işlemede, özet çıkarımında ve doküman benzerliklerinin hesaplanmasında birincil aşamadır. VİT soru yanıtlama, bilgi alma, ilişki çıkarma vb. için doğal dil işleme (DDI)-(Natural Language Processing, NLP) sistemlerinde anahtar bileşendir. Bu nedenle, VİT çıkarımı işlemi için kullanılan farklı metodolojileri ve her metodoloji altında hangi algoritmaların kullanıldığı verilmiştir. Ayrıca, VİT algoritmalarının uygulandığı farklı alanlar hakkında da bilgi verilmiştir. Benzer biçimde otomatik anahtar sözcük çıkarımı için de çalışmalar verilmiştir. Doküman indeksleme için üst veri çıkarımı, varlık ismi çıkarımı, dokümandan anahtar kelime çıkarımı problemlerine yoğunlaşılırken; doküman arama için anahtar kelime tabanlı arama, dokümanların benzerliklerinin hesaplanması ve doküman zincirlerinin oluşturulması problemlerine yoğunlaşılmıştır.

Çalışmanın organizasyonu aşağıdaki biçimde düzenlenmiştir. Üst veri çıkarımı hakkındaki çalışmalar Bölüm 2'de sunulmuştur. Varlık ismi tanıma ile ilgili güncel çalışmalar Bölüm 3'te detaylı olarak irdelenmiştir. Bölüm 4 otomatik anahtar kelime çıkarma çalışmalarını özetlemektedir. Doküman benzerliklerinin elde edilmesi yöntemlerine ilişkin çalışmalar Bölüm 5'te verilmiştir. Son olarak, çalışmanın gelecek yönleri ile sonuçları Bölüm 6'da ortaya konulmuştur.

II. ÜST VERİ ÇIKARIMI

Üst veri, bir veri kaynağının içerisindeki parmak izi verisi olarak tanımlanabilen anahtar bilgi kümesini içermektedir. Bu veri kümesi verinin işlenmesi, yönetilmesi ve anlamlandırılması için önem arz etmektedir [6]. Bu tür veriler, elektronik kütüphanelerin temel elemanları olan metin dokümanları için yazar, başlık, konu kategorisi ve dosya boyutu olarak olabilirken; üst veriler resimler, video, web sayfaları, elektronik tablolar, açıklama ve anahtar sözcükler gibi yapılandırılmamış veriler için de kullanılabilmektedir. Üst veri çıkarımının önemli adımlarından ilki dijital verinin elde edilmesidir. Özellikle basılı metinler için üst verinin elde edilmesi önemli bir problemdir. Bu tür verilerin elde edilmesinde en etkili yöntem iyi bir tarama işlemi yardımıyla verilerin otomatik olarak elde edilmesi ile başlamaktadır. Bu nedenle, bilgileri, özellikle de metni görüntüden otomatik olarak almak ve depolamak için bir teknik gereklidir. Optik karakter tanıma (Optical Character Recognition, OCR), taranan belgelerden metinleri elde etmek için kullanılır. OCR, bir sistemin insan müdahalesi olmadan görüntülerden karakterleri veya alfabeleri tanımlamasını sağlayan süreçtir [7, 8].

Tipik bir OCR sistemi, analog belgeleri dijitalleştirmek için bir optik tarayıcıdan ve metin içeren bölgeler bulunduğunda sembolleri ayıklamak için bir segmentasyon işlemi bileşenlerinden oluşmaktadır. Çıkarılan semboller, özellik çıkarımını kolaylaştırmak için gürültüyü ortadan kaldıracak biçimde ön işlemden geçirilmektedir. Öğrenme aşamasında çıkarılan özellikler ile birlikte açıklamalar her bir sembol için değerlendirilmektedir. Son olarak, orijinal metnin kelimelerini ve sayılarını yeniden yapılandırmak için ilişkisel yapı kullanılmaktadır [9]. OCR, taranmış kâğıt belgeler, PDF dosyaları veya görüntüler gibi farklı belge türlerini, otomatik üst veri ayıklama yapabilmek için düzenlenebilir ve aranabilir verilere dönüştürmek için kullanılır [7]. Tablo 1'de sıklıkla kullanılan bazı OCR araçları ve bu araçların bazı özellikleri listelenmiştir.

Tablo 1. *Popüler bazı OCR araçları ve özellikleri*

OCR İsmi	Desteklediği Platform	Türkçe Desteği	Lisanslama
Tesseract	Linux, Windows, Mac OS	Var	Ücretsiz
CuneiForm	Linux, Windows, Mac OS	Var	Ücretsiz
ABBYY FineReader	Linux, Windows, Mac OS	Var	Ücretli
Google Cloud Vision	Linux, Windows, Mac OS	Var	Ücretli
Amazon Textract	Linux, Windows, Mac OS	Var	Ücretli
OmniPage	Linux, Windows, Mac OS	Var	Ücretli

Kural tabanlı üst veri çıkarımı yöntemlerinde, belgelerden üst verileri çıkarmak için önceden kurallar oluşturulmalıdır. Örneğin CiteSeer sistemi [10], bu yöntemi PDF belgelerinden [11] üst verileri çıkarmak için kullanmaktadır. Bununla birlikte, bu yöntem kuralları otomatik olmayan yöntemler ile belirlemek için ön işleme süreçler gerektirir ve kuralı oluşturanların uygulama alanı hakkında iyi bir bilgiye sahip olmasını gerektirir. Ayrıca farklı belgeler eklenirse yeni kurallar geliştirilerek, veri çıkarımının eksik bırakılmaması gereklidir. Bu yaklaşımın en problemli tarafı kuralların oluşturulması ve kuralların güncel tutulabilmesidir.

Şablon tabanlı yaklaşımlar üst veri çıkarımında izlenen diğer bir yoldur. Bu yaklaşım temel motivasyonu, aynı kuruma, organizasyona ya da benzer etkinliklere ait dokümanların aynı ya da çok benzer şablonlara sahip olacağıdır [12]. Şablon tabanlı üst veri çıkarımında, her şablona üst veriler için kurallar tanımlanmaktadır. Eğer üst verisi çıkarılacak olan dokümanın şablonu bilinen bir şablona benziyorsa, üst veri çıkarımı için o şablonun kuralları kullanılır; değilse yeni gelen dokümanın şablonu çıkarılır, kuralları yazılır ve bu yeni şablon bilindik şablonlara eklenir. Bu tür sistemlere [13, 14] örnek olarak verilebilir.

Öte yandan, makine öğrenimine dayalı yaklaşım, büyük ölçekli veriler için eğitim uygulayabilmektedir ve eğitilen veriler, insan müdahalesi olmadan otomatik olarak yeni belgelerle ilgilenebilme özelliğine sahiptir. Bu yöntem, kural tabanlı yaklaşımdan daha esnektir ancak yeterli hacimde eğitim verisinin mevcut olmasını gerektirir. Son on yılda, makine öğrenimine dayalı üst veri çıkarımı için birçok teknik geliştirilmiştir. Bununla birlikte, en popüler yöntemler koşullu rastgele alanlar (Conditional Random Fields, CRF) ve destek vektör makineleri (Support Vector Machine, SVM) [15] olarak bilinirler. CRF ve SVM [16-18] CiteSeer Plus ve Mendeley gibi büyük ölçekli sistemler tarafından bibliyografik üst verileri çıkarmak için kullanılmaktadırlar. Üst veriler, herhangi bir sayısallaştırılmış koleksiyondan farklı teknikler kullanılarak belirlenmiş karakter dizisi örüntüsü aranarak çıkabilmektedir [19]. Naive Bayes modeli (NB), SVM ve saklı Markov modeli (Hidden Markov Model, HMM), bir koleksiyonun farklı sayfalarındaki metin satırlarını sınıflandırarak belge düzenlerini tanımak için kullanılan birkaç popüler öğrenme modelinin örnekleridir [19, 20].

Birçok alanda olduğu gibi son üst veri çıkarımında da son zamanlarda derin öğrenme teknikleri kullanılmaya başlanmıştır [21, 22].

Literatürde birkaç yıl önce yazılmış belirli bir bilgi parçasını bulmak araştırmacılar ve akademisyenler için büyük bir zorluktur. Dolayısıyla, üst veri çıkarımı, çok sayıda belge ve literatür koleksiyonundan başlık, yazarlar ve yayın tarihi gibi yapılandırılmış üst verileri elde etmek için yararlı olmaya devam etmektedir [23, 24]. Bu durum göstermektedir ki; veriden üst veri çıkarmanın arkasındaki temel neden, üst verilerin verinin kendisinden daha önemli olmasıdır [25].

Örneğin üst veri çıkarımının kıymetli olduğu bankacılık dokümanlarında yapılan üst veri edinme çalışmaları tipik olarak kurum adı, referans numarası, konu, tarih ve benzeri bilgileri sayısallaştırılmış belgeden elde etmektir. Sayısallaştırılmış belgeden elde edilen üst veri, belgenin yazıldığı bağlamın daha derin bir şekilde anlaşılmasını sağlar [26]. Çıkarılan üst veriler, tarih, referans numarası, konu, tür ve belgeyi hangi kurumun yazdığına ilişkin aşağıdaki sorulara cevap vermektedir. Farklı türlerdeki sayısal belgelerden üst verileri otomatik olarak çıkaran birçok araç vardır [10]. Ancak üst veri çıkarma standartlarının eksikliğinden dolayı bu araçların hataya açık olduğu bilinmektedir [24].

III. VARLIK İSİMLERİNİN ÇIKARIMI

Varlık İsmi Tanıma (VİT)-(Named-Entity Recognition, NER), bir metin içerisinde geçen insan, coğrafi bölge ve organizasyon gibi özel isimlerinin, meblağ, saat, tarih, sıcaklık ve yüzdelik gibi sayısal değerlerin, sanatsal, kültürel ve doğal olayların metin içerisinde otomatik olarak tespit edilmesi problemidir. Bu problem ilk kez 1991 tarihinde Rau ve arkadaşları [27] finansal metinlerden şirket

isimlerinin otomatik olarak tespiti ile ortaya atılmış, 1996 yılında 6. Mesaj Anlama Konferansında NER terimi kullanılmış ve bu amaçla görev çağrısında bulunulmuştur. Bu problem halen popülaritesini korumakta, sadece VİT odaklı akademik toplantılar düzenlenmekte (CoNLL 2002, CoNLL 2003 gibi) ve bu problemle ilgili özel çağrılar yapılmaktadır [28-30].

VİT, iki alt problemden oluşur. Bunlardan ilki metin olarak adlandırılan varlık isimlerinin belirlenmesi, ikincisi ise metnin uygun tipe atanmasıdır. VİT sistemlerinin başarısını ölçmede tip ve metinlerin duyarlılık ve kesinlik ve değerlerden hesaplanan bazı diğer, F1-score gibi, ölçütler kullanılır. VİT sistemlerinde tipin doğru olması metnin içerdiği varlık isminin tipinin doğru tahmin edilmesidir. Metin doğruluğu ise tipin doğruluğuna bakılmaksızın varlık ismi olarak belirlenen metnin varlık ismini oluşturan kelimeler dışında kelime barındırmamasıdır.

VİT problemi için ilk geliştirilen sistemler kural tabanlı sistemlerdir [31]. Bu tür sistemler, metindeki varlık ismi olabilecek söz öbeklerinin belirlenmesi ve bu söz öbeklerinin önceden oluşturulan ve varlık isimlerinin saklandığı sözlüklerde aranmasına dayanır. Kural tabanlı VİT sistemlerinin temel problemi dile ve uygulama alanına özgü olmaları, güncel sözlüklere ihtiyaç duymaları ve varlık ismi olabilecek söz öbeklerini belirlemede kullanılan kuralların tanımlamasının zorluğudur. Bu tür sistemlerin en büyük avantajı ise işaretlenmiş veriye ihtiyaç duymamalarıdır. Bu tür sistemlerde kesinlik genellikle yüksek ancak duyarlılık düşüktür [3].

Gözetimli öğrenme algoritmaları VİT'da sıklıkla kullanılmıştır. Gözetimli öğrenme teknikleri, VİT problemini bir metni oluşturan kelimelerin ait oldukları sınıfı tahmin etme problemi olarak ele almaktadır. Bu amaçla, gözetimli öğrenme teknikleri bir dokümanı oluşturan tüm kelimeleri *1 basamaklı sayı*, *2 basamaklı sayı*, *sayı nokta içeriyor*, *ilk harfi büyük*, *içinde büyük harf var*, *tüm harfleri büyük* gibi çeşitli özellikler kullanarak ve ayrıca IOB2 [32], BILOU [33], IO [34] gibi standartlar kullanılarak işaretlenir. Bu standartlarda B(egin) kelimenin bir varlık isminin ilk kelimesi olduğunu, I(nterior/nside) kelimenin bir varlık ismi içinde geçtiğini, O(ther/utside) kelimenin varlık isminin parçası olmadığını, L(ast), kelimenin varlık isminin son kelimesi olduğunu, U(nit) varlık isminin bir söz öbeğinden oluştuğunu belirtir.

Destek vektör makinesi VİT'da sıklıkla kullanılan gözetimli öğrenme yöntemlerinden biridir [35]. Bu yaklaşımda örnekler $\{(x_1, y_1), (x_2, y_2) \ldots (x_n, y_n)\}$, x_i i. örneğin özellik vektörü, $y_i \in \{+1,-1\}$de i. örneğin sınıfını, şeklinde vektörlerle temsil edilir. SVM, pozitif ve negatif sınıfa ait olan örnekleri birbirinden ayıracak hiper düzlemi bulmaktadır.

HMM ve CRF, VİT için sıklıkla kullanılan yöntemlerdir [36-40]. Her iki öğrenme modeli de VİT problemini dizilim etiketleme olarak ele almaktadır. HMM, gözlenebilen çıktılardan gözlenemeyen durumları modellemede kullanılan bir stokastik modeldir. Bir HMM modeli bir 5liden oluşur, $H = <Q, A, O, B, P>$. Bu gösterimde Q durumları, A durumlar arası geçiş olasılıklarını, O gözlemleri, B her durum için görülebilecek gözlemlerin olasılıkları, P ise durumlar arası geçiş olasılıklarının başlangıç değerini belirtir. HMM, bir Q_j durumuna geçişin olasılığı sadece ve sadece Q_{j-1} durumuna bağlı olduğunu ve Qj durumunda gözlenebileceklerin sadece ve sadece Q_j durumunun belirlediğini kabul eder. Bir HMM modeli, $P = (A|B)$, ve bir gözlem dizilimi verildiğinde bu dizilimi en yüksek olasılıkla salımlayan gizli durumlar dizilimi Viterbi algoritması kullanılarak elde edilebilir. A ve B parametreleri ise Baum-Welch algoritması kullanılarak elde edilebilir. HMM tabanlı VİT tanımada, bir metni oluşturan kelimeler ve özellikleri gözlemler olarak, VİT tipleri de saklı durumlar olarak kabul edilir. İşaretlenmiş veri kümesi ile HMM eğitildikten sonra, Viterbi algoritması kullanılarak her kelime için en yüksek olasılıklı varlık ismi tipi ile işaretlenir.

CRF, sıralı verileri işaretlemek için kullanılan ayırt edici (discriminative) olasılıksal bir modeldir. Öznitelik kümesi kullanılarak koşullu olasılık dağıtımı modellenmektedir. Bu model için $P(y|x)$ yani y olarak belirtilen işaret için x olarak belirtilen öznitelikler kullanılarak aşağıdaki eşitlik üzerinden hesaplama yapılır.

$$P(y|x) = \frac{1}{z_\theta(x)} \exp\left\{\sum_{t=1}^{T}\sum_{k=1}^{K}\theta_k f_k(y_{t-1}, y_t, x_t)\right\} \quad (1)$$

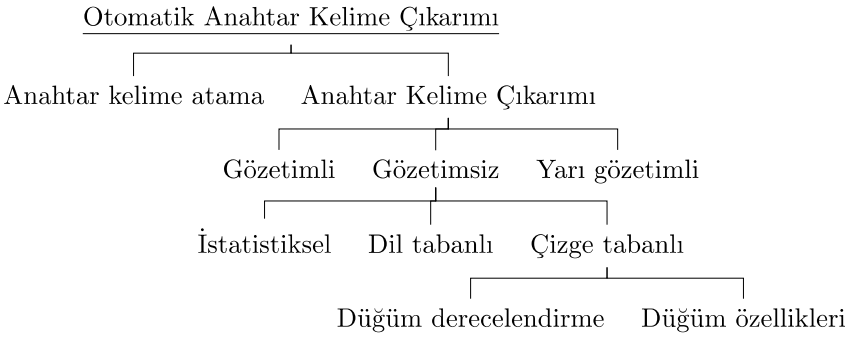
$f_k(y_{t-1}, y_t, x_t)$ fonksiyonu, verilen x_t özniteliklerine göre y_{t-1} durumundan y_t durumuna geçiş olasılığını hesaplamak için kullanılmaktadır. Model eğitilirken θ_k optimize edilirken, $z_\theta(x)$ normalizayson faktörü olarak kullanılmaktadır.

HMM ve CRF tabanlı VİT dayalı sistemler lineer modeller öğrenebilirken, derin öğrenmeye tabanlı VİT sistemleri lineer olmayan modelleri de öğrenebilir. Ayrıca CRF, HMM ve kural tabanlı sistemlerde alan uzmanlığı gerektiren öznitelik çıkarma işlemini, derin öğrenme ile otomatik olarak yapılabilir.

Son zamanlarda derin öğrenme metotları da VİT probleminde sıklıkla kullanılmıştır [40]. VİT problemi için en basit bir derin öğrenme modeli üç bileşenden oluşur: girdi katmanı, kodlayıcı katmanı (encoder) ve çözümleyici katmanı (decoder). Girdi katmanında, kelimeler üç farklı şekilde temsil edilir. Bunlardan ilki kelime-düzeyinde temsil olarak adlandırılan, GloVe ve fastText gibi önceden eğitilmiş temsiller kullanılarak kelime temsilleridir. Bu tür temsiller n-gram, skip-gram gibi yöntemler kullanılarak elde edilir. İkinci temsil şekli ise karakter-düzeyi temsildir. Bu yaklaşımda kelimeler yerine kelimeleri oluşturan karakterler için temsiller oluşturulur ve karakter temsillerinden kelime temsillerine geçilir. Bu şekilde kelimelerde geçen ön ve son ekler de işlenebilmektedir. Karakter düzeyinde temsiller için Konvolüsyonel sinir ağı (Convolutional Neural Network, CNN), uzun kısa süreli bellek (Long Short-Term Memory, LSTM) gibi yaklaşımlar kullanılmaktadır. Son kelime temsil yöntemi ise melez olarak adlandırılan ve kelime ya da karakter düzeyinde elde edilen temsillerin sözlükler, dil özellikleri gibi yapıların da temsile eklenmesi ile elde edilir. Kodlayıcı katmanda girdi olarak alınan kelime temsilleri gizli ortak bir temsile çevirir. Kodlayıcı katmanda konvolüsyonel sinir ağı, Tekrarlayan sinir ağı (Recurrent Neural Network, RNN) ve özyinelemeli sinir ağı gibi yapılar kullanılabilir. Çözümleyici katman ise gizli temsili girdi olarak alır ve girdiye denk gelen etiketleri çıktı olarak verir. Çözümleyici katmanda kullanılan yapılar ise Multi-layer perception + softmax, CRF, RNN ve Pointer Network'tür.

IV. ANAHTAR KELİME ÇIKARIMI

Anahtar kelime çıkarma bir belgeyi tanımlayan en önemli kelimelerin tespit edilmesi problemi olarak tanımlanabilir [4]. Literatürde anahtar kelime çıkarımına yönelik çalışmalar kullandıkları öğrenme algoritmaları, metin temsili, odaklandıkları metin türüne göre farklılıklar göstermektedir. Şekil 1'de anahtar kelime çıkarma yöntemleri kullandıkları öğrenme yöntemine göre sınıflandırılmıştır.



Şekil 1. *Otomatik Kelime Çıkarımı Sınıflandırması.*

Anahtar kelime atama yöntemlerinde, dokümanlara anahtar kelimeler önceden oluşturulmuş sözlükler kullanılarak yapılır [41]. Bu yaklaşımda elle oluşturulmuş kurallar ve ontolojiler kullanılmaktadır. Anahtar kelime atama yaklaşımın temel dezavantajı dil bağımlı olması, sözlüklerin güncel olması ve kural oluşturma zorluğudur. Avantajı ise dokümanlara dokümanda yer almayan kelimelerin anahtar kelime olarak atanmasıdır.

Anahtar kelime çıkarımı yöntemleri, bir dokümanı sadece o dokümanda geçen kelimelerle etiketlemeyi amaçlar. Bu yaklaşımın alt kırılımı olan gözetimli anahtar kelime çıkarımı, problemi ikili sınıflama problemi olarak ele alır, dokümanı oluşturan kelimeleri *anahtar kelime* / *anahtar kelime değil* sınıflarından birine atamayı amaçlar. Bu amaçla Naive Bayes [42], SVM [43], Random Forest [44] gibi yöntemler kullanılmaktadır. Kelimelerin öznitelikleri olarak, TF-IDF değeri, kelimenin doküman içinde ilk görüldüğü pozisyon, kelime türü gibi özellikler kullanılmaktadır. Gözetimli öğrenme yaklaşımları, eğitildikleri alana ait dokümanlardan anahtar kelime çıkarımında başarılı iken farklı bir alandaki dokümanda genellikle başarısızdır. Ayrıca bu yaklaşımın dil bağımlılığı diğer bir dezavantajıdır.

Gözetimsiz istatistiksel yaklaşımlar kelimelerin önemlerini TF-IDF değeri, entropisi, birlikte görünme sıklığı gibi istatistiksel kriterlere göre belirlemektedir [45]. Dil tabanlı sistemler, kelimelerin türleri, isim ve sıfat tamlamaları gibi söz öbekleri, anlamsal ilişkiler gibi dil özellikleri kullanarak anahtar kelimeleri dokümanlar içinden otomatik olarak çıkarırlar [46].

Çizge tabanlı yaklaşımlar anahtar kelime çıkarımında sıklıkla kullanılmaktadır. Bu yaklaşım, dokümanları kelime çizgesi olarak temsil eder ve anahtar kelime çıkarımını düğüm derecelendirme problemi olarak ele alır. Kelime çizgelerinde düğümler dokümandaki kelimeleri temsil eder, kenarlar ise genellikle birbirinden en fazla *n*-pozisyon uzaktaki kelimeleri bağlamaktadır. Düğüm derecelendirme PageRank [47] algoritması ve varyasyonları ile yapılmaktadır. PageRank algoritması, web sayfalarının önemini hesaplamak için tasarlanmıştır. PageRank algoritması, önemli bir web sayfasının diğer önemli web sayfalarından link aldığını kabul eder. Denklem 2, anahtar kelime çıkarımı için PageRank algoritmasının iteratif varyasyonunu göstermektedir. Burada, $S(V_i)$ i. iterasyon sonunda V_i düğümünün ağılığını, α bir düğümden diğer düğüme geçiş olasılığını, p_i i. düğüm ile temsil edilen kelimenin ilk ağırlıdır. $Adj(v_i)$, v_i düğümüne bağlı düğümleri, $S(v_j)$ de bu düğümlerin ağırlıklarını göstermektedir. w_{ji}, i ve j düğümleri arasındaki kenarın ağırlığını göstermektedir.

$$S(v_i) = (1-\alpha).\widetilde{p_i} + \alpha \sum_{v_j \in Adj(v_i)} \frac{w_{ji}}{O(v_j)} S(v_j) \tag{2}$$

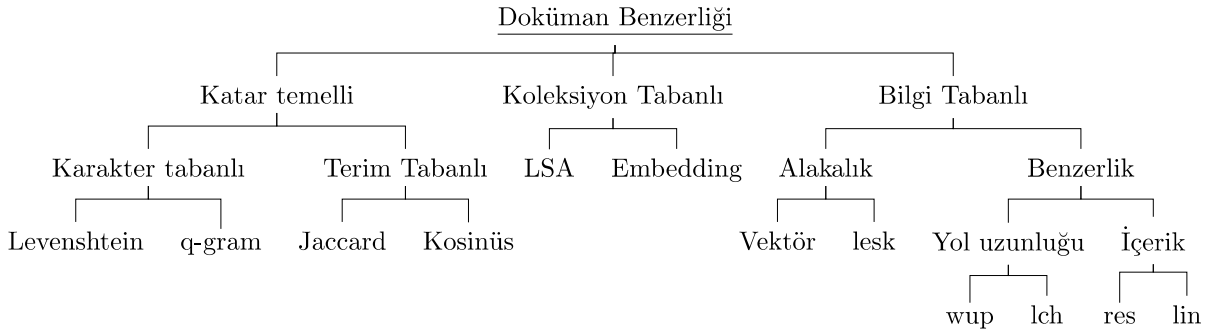
Çizge tabanlı anahtar kelime çıkarma algoritmaları genellikle düğümlere verdikleri ilk ağırlıklara göre farklılık göstermektedir. PositionRank [48], ilk düğümlerin ilk ağırlıklarını temsil ettikleri kelimenin doküman içindeki pozisyonuna göre belirlemektedir. TextRank [49], kelimelerin başlangıçta eş öneme sahip olduğunu kabul edip ilk ağırlık olarak tüm kelimelere 1 değerini vermektedir. YAKE [50], kelime pozisyonu, frekansı, büyük harfle başlaması, kelimenin dokümanla ilişkisi gibi birden çok değeri kullanarak kelimelerin ilk ağırlıklarını hesaplamaktadır. Düğüm özelliklerine dayalı çizge yaklaşımlarda ise düğümlerin merkezilik değerlerinden yararlanır. Bu merkezilik ölçütleri iç derece, dış derece, yoğunluk, yakınlık, arasındaklık ve özvektör gibi ölçütleri içermektedir. Bu ölçütler kelimelerin ilk ağırlıklarını oluşturmada kullanılabilmektedir [51]. Diğer yandan [52] gibi çalışmalar yine çizgelerin özelliklerinden faydalanarak anahtar kelimeleri elde etmektedir.

Yarı gözetimli öğrenmede az miktardaki etiketlenmiş veri ile fazla miktardaki etiketlenmemiş veri birleştirilerek model oluşturulur. Bu tür anahtar kelime çıkarma yaklaşımlarda etiketlenmiş veri ile modeller oluşturulur etiketlenmemiş veri ile de modellerin başarımının artırılması amaçlanmaktadır. [53]'te başarımını artırmak için etiketlenmemiş verideki aday anahtar kelimeler TF-IDF değerlerine göre sıralanır ve en yüksek skordan başlanarak adayların anahtar kelime olup olmayacakları belirlenir. [54]'te etiketlenmiş verideki anahtar kelimelerin öznitelikleri ve bu özniteliklerin olasılıksal dağılımları Bayes teoremine göre hesaplanır. Etiketlenmemiş bir dokümandaki her kelime, öznitelikleri dikkate alınarak daha sonra bu olasılıksal dağılımlara sınıflandırılır.

Anahtar kelime çıkarma problemi yoğun bir şekilde çalışılan bir problem olduğundan literatürde çeşitli çalışmalar bulunmaktadır. Bu bölümde anahtar kelime çıkarma problemine geniş bir bakış sağlanması amaçlanmıştır.

V. DOKÜMAN BENZERLİKLERİNİN HESAPLANMASI

Dokümanların benzerliği dokümanların sınıflandırılması, kümelenmesi ve indekslenmesi gibi dokümanı temel olarak birçok çalışmada önemli bir yer edinmektedir. Bu bağlamda doküman benzerliği literatürde sıklıkla çalışılmış ve birçok metriğin geliştirildiği bir alandır. Şekil 2'de metin benzerliği için geliştirilen yöntemlerin sınıflandırması ve her sınıf için bazı örnekler verilmiştir.



Şekil 2. *Metin benzerliği için geliştirilen yöntemlerin sınıflandırması.*

Kelime benzerliği doküman benzerliğinin hesaplanmasındaki ilk adımdır. Kelimeler birbirlerine iki şekilde benzeyebilir: sözlük anlamına göre benzerlik ya da dokümanda geçtikleri yere göre hesaplanan anlamsal benzerlik. Sözlük anlamına göre benzerlik katar temelli benzerlik altında, anlamsal benzerlik ise koleksiyon ve bilgi tabanlı benzerlik altında incelenmektedir.

Karakter tabanlı yaklaşımlar, dokümanları karakter dizisi olarak kabul eder ve bu karakter dizilerinin benzerliklerine bakılır. Levenshtein indeksi [55], bir kelimeyi diğer kelimeye dönüştürmek için kaç ekleme, silme ve değiştirme gerektiğine bakar. *q*-gram [56] yönteminde ise iki kelimenin sahip olduğu q uzunluklu alt dizilimlerin sayısı iki kelimenin benzerliğini hesaplamada kullanılır.

Terim tabanlı benzerlikte, dokümanların içerdikleri ortak kelimelere bakılmaktadır. Jaccard indeksinde [57], iki dokümanda geçen ortak kelimelerin sayısı, iki dokümanda geçen toplam tekil kelime sayısına bölünür. Jaccard indeksi 0 ile 1 arasında değer alır ve daha yüksek değer daha fazla benzerlik anlamına gelir. Kosinüs benzerliğinde dokümanlar vektör olarak temsil edilir ve iki dokümanın benzerliği iki doküman vektörünün kosinüs değeri olarak hesaplanır [58]. Bir dokümanın vektör temsilinde, her kelime bazı sayısal değerlerle ifade edilir. Bu sayısal değer, kelimenin TF değeri, TF-IDF değeri gibi tek bir sayı olabileceği gibi vektör de olabilir.

Koleksiyon tabanlı kelime benzerlik hesaplanmasında, kelimeler arası benzerlik kelimelerin geçtikleri içerikler de dikkate alınarak hesaplanır. Koleksiyon tabanlı bir yaklaşım olan Latent Semantic Analysis (LSA) [59], dokümanların sütunları dokümanlarda geçen kelimelerin ise satırları oluşturduğu bir matris tanımlanır. Matrisin hücreleri, kelimenin ilgili dokümanda bulunup bulunmadığına göre 0 / 1 ile veya kelimin TF-IDF değeri gibi değerlerle doldurulabilir. Matrisin her sütunu bir vektör gibi kullanılıp, vektör benzerliği yöntemleri ile iki doküman arasındaki benzerlik hesaplanabilir. Embedding [60], yakın zamanda ortaya çıkan bir kelime temsil modelidir. Çok büyük koleksiyonlar kullanılarak, bir kelimenin birlikte geçtiği kelimeler dikkate alınarak kelime temsil vektörleri oluşturulur. Bu vektörler 200 – 400 boyutlu kadar büyük olabilirler. Bu tür temsillerin oluşturulması büyük koleksiyonlar gerektirmektedir. GloVe, Word2Vec ve fastText gibi hazır modeller mevcuttur. Vektör benzerlik yöntemleri kullanılarak bu kelimelerin benzerlikleri hesaplanabilir.

Bilgi tabanlı yaklaşımlarda, kelimeler anlamlarına göre gruplanır ve gruplar arasında anlamsal ve yapısal hiyerarşiler oluşturulur [61]. İki kelimenin benzerliği bu hiyerarşiler kullanılarak hesaplanır. Bilgi tabanlı benzerlik ölçütlerine Resnik [62], Lin [63], Wup [64] birer örnek olarak verilebilirler. Bu

metriklerde iki kelime arasındaki benzerlik, kelimelerin ortak atalarına olan uzaklıklarına, ortak atanın ve kelimelerin hiyerarşideki yerlerine göre hesaplanır. Örnek olarak Wup benzerlik değeri Denklem 3'e göre hesaplanır. Denklem 3'de *lcs(s1,s2)*, *s1* ve *s2* kelimelerinin en yakın ortak atasını, *depth(lcs(s1,s2))* de bu ortak atanın hiyerarşideki derinliğini verir. *depth(s1)* ve *depth(s2)* de *s1* ve *s2* kelimelerinin hiyerarşideki derinliklerini ifade eder. Lesk benzerliğinde iki kelime tanımının örtüşmelerin uzunluğuna bakılmaktadır. Burada örtüşme, ardışık olarak her iki tanımda da geçen kelimeler olarak tanımlanır.

$$wup(s1, s2) = \frac{2xdepth(lcs(s1, s2))}{depth(s1) + depth(s2)} \quad (3)$$

IV. SONUÇ

Dijital kütüphaneler ve bunlara ait dokümanlardan veri geri kazanımı, birçok problemi barındırması nedeniyle yirmi yıldan fazla bir süredir üzerinde çalışılmaktadır ve gelişimi halen devam etmektedir. Bilgi geri kazanımı için üst-veri çıkarımı, varlık isimlerinin elde edilmesi, anahtar kelimelerin elde erilmesi ve doküman benzerliklerinin oluşturulması gibi adımların göz önüne alınması gereklidir. Buradaki hedeflenen sonuç dokümanlardan elde edilebilecek veri çıkarımının olabildiğince tüm kullanıcılara en fazla verimi sağlayabilecek sonuçların elde edilmesidir. Bu verim bazı kullanıcılar için üst-veri çıkarımı ile elde edilebilen tarih, bazı kullanıcılar için varlık ismi ile özel isimler, bazı kullanıcılar için anahtar kelimeler ile hazırlanmış iyi bir özet, bazı kullanıcılar için de en yakın dokümanın bulunması olabilmektedir. Bu derleme çalışmasında, literatürde var olan yöntem ve tekniklerin dört ana başlıkta irdelenmesi ve çeşitliliği verilmiştir. Her bir konu için ondan fazla teknik açıklanmıştır.

Ayrıca, gerek üst-veri gerek varlık ismi tanıma ve anahtar kelime çıkarımları için kural tabanlı yapılardan makine öğrenimi yaklaşımlarına doğru eğilimleri inceleyerek dijital kütüphaneleri geliştirmek için kullanılan tekniklere genel bir bakış sağlanmıştır. Kural tabanlı yapıların nispeten yüksek bir seviyede iyi performans göstermesine karşın doküman sayılarının artması ve farklılaşması durumunda muazzam bir iş yükü oluşturduğu verilmiştir. Makine öğrenmesi kullanıldığında açıklamalı verilerin geniş bir koleksiyonunun bulunması dışında ek bir yük getirmeden veri geri kazanımının iyi seviyede elde edilebildiği yöntemler de açıklanmıştır. Bu doğrultuda gelecek çalışmaların sözü edilen tüm konu başlıklarında hızlı dağıtım vaat eden yarı denetimli ve denetimsiz öğrenme tekniklerinin üzerine kurgulanacak olduğu bir sonuç olarak ortaya konulmuştur.

TEŞEKKÜR: Bu çalışma Türkiye Bilimsel Ve Teknolojik Araştırma Kurumu (TÜBİTAK) tarafından desteklenmiştir (Proje No: 5190074).

V. KAYNAKLAR

[1] M. Afzali, "Karma Kütüphane: Dijital ve Geleneksel Kütüphanelerin Odak Noktası," *Türk Kütüphaneciliği*, c. 22, s. 3, ss. 266-278, 2008.

[2] L. Masinter (1995). *Document management, digital libraries and the Web* [Online]. Available: http://www. cernet. edu. cn/HMP/PAPER/243/html/paper.htm

[3] V. Yadav, S. Bethard, "A Survey On Recent Advances In Named Entity Recognition From Deep Learning Models," *The 27th International Conference on Computational Linguistics (COLING)*, 2018, ss. 1-14.

[4] S. Beliga, "Keyword extraction: a review of methods and approaches," *University of Rijeka, Department of Informatics*, Rijeka, 2014, ss. 1-9.

[5] S. Beliga, A. Meštrović and S. Martinčić-Ipšić, "An overview of graph-based keyword extraction methods and approaches," *Journal of information and organizational sciences*, c. 39 s. 1, ss. 1-20, 2015.

[6] S. Chatvichienchai, "SEMEXSS - A Rule-Based Semantic Metadata Extraction System for Spreadsheets," *International Journal of Computer Theory and Engineering*, c. 8, s. 2, ss. 102–108, 2016.

[7] K. Hamad and M. Kaya, "A Detailed Analysis of Optical Character Recognition Technology," *International Journal of Applied Mathematics, Electronics and Computers*, c. 4, s. Special Issue-1, ss. 244–249, Dec. 2016.

[8] N. Sahu and M. Sonkusare, "A Study on Optical Character Recognition Techniques," *The International Journal of Computational Science, Information Technology and Control Engineering*, c. 4, s. 1, ss. 01–15, Jan. 2017.

[9] A. Chaudhuri, K. Mandaviya, P. Badelia and S. K. Ghosh, "Optical Character Recognition Systems," *In Optical Character Recognition Systems for Different Languages with Soft Computing*, c. 352, ss. 9–41, 2017.

[10] I. G. Councill, C. L. Giles, E. Di Iorio, M. Gori, M. Maggini, A. Pucci, "Towards Next Generation CiteSeer: A Flexible Architecture for Digital Library Deployment," *International Conference on Theory and Practice of Digital Libraries*, 2006, ss. 111–122.

[11] J. Zhao and H. Liu, "Metadata Extraction Approach of PDF Documents Based on Measurement Fusion," *Journal of Multimedia*, c. 8, s. 6, Nov. 2013.

[12] P. Flynn, L. Zhou, K. Maly, S. Zeil, M. Zubair, "Automated template-based metadata extraction architecture," *International Conference on Asian Digital Libraries*, 2007, ss. 327-336.

[13] L. Kovriguina, A. Shipilo, F. Kozlov, M. Kolchin, E. Cherny, "Metadata extraction from conference proceedings using template-based approach", *Semantic Web Evaluation Challenges*, 2015, ss. 153-164.

[14] Z. Huang, H. Jin, P. Yuan, Z. Han, "Header metadata extraction from semi-structured documents using template matching," *International Conferences On the Move to Meaningful Internet Systems*, 2006, ss. 1776-1785.

[15] H. Han, C. L. Giles, E. Manavoglu, H. Zha, Z. Zhang, E. A. Fox, "Automatic Document Metadata Extraction using Support Vector Machines," *Joint Conference on Digital Libraries (JCDL03)*, 2003, ss. 37-49.

[16] L. Shi, R. Khushaba, S. Kodagoda, G. Dissanayake, "Application of CRF and SVM based semi-supervised learning for semantic labeling of environments," *12th International Conference on Control Automation Robotics & Vision (ICARCV)*, 2012, ss. 835-840.

[17] H. Han, E. Manavoglu, H. Zha, K. Tsioutsiouliklis, C. L. Giles, X. Zhang, "Rule-based word clustering for document metadata extraction," *ACM symposium on Applied computing (SAC '05)*, 2005, ss. 1049-1053.

[18] M. Granitzer, M. Hristakeva, K. Jack, R. Knight, "A comparison of metadata extraction techniques for crowdsourced bibliographic metadata management," *27th Annual ACM Symposium on Applied Computing (SAC '12)*, 2012, ss. 962-964.

[19] D. Misra, S. Chen, G. R. Thoma, "A System for Automated Extraction of Metadata from Scanned Documents using Layout Recognition and String Pattern Search Models," *Archiving*, 2019, ss. 107–112.

[20] J. Azimjonov, J. Alikhanov, "Rule Based Metadata Extraction Framework from Academic Articles," *ArXiv*, 2018, ss. 1-10.

[21] L. Runtao, L. Gao, D. An, Z. Jiang, Z. Tang, "Automatic document metadata extraction based on deep networks," *National CCF Conference on Natural Language Processing and Chinese Computing*, 2017, ss. 305-317.

[22] I. Safder, S. Hassan, A. Visvizi, T. Noraset and R. Nawaz, "Deep Learning-based Extraction of Algorithmic Metadata in Full-Text Scholarly Documents," *Information Processing & Management*, c. 57, s. 6, 102269, 2020.

[23] J. Greenberg, W. Klas, *Metadata for Semantic and Social Applications*, Dublin Core Metadata Initative and Universitätsverlag, Göttingen, 2008.

[24] M. Lipinski, K. Yao, C. Breitinger, J. Beel, B. Gipp, "Evaluation of header metadata extraction approaches and tools for scientific PDF documents," *13th ACM/IEEE-CS joint conference on Digital libraries (JCDL '13)*, 2013, ss. 385-386.

[25] E. Mannens, R. Verborgh, S. Hooland, L. Hauttekeete, T. Evens, S. Coppens and R. Walle, "On the Origin of Metadata," *Information*, c. 3, s. 4, ss. 790–808, 2012.

[26] L. Kovriguina, A. Shipilo, F. Kozlov, M. Kolchin, and E. Cherny, *Metadata Extraction from Conference Proceedings Using Template-Based Approach*, in Semantic Web Evaluation Challenges, Springer International Publishing, 2015, ss. 153–164.

[27] Lisa F. Rau, "Extracting company names from text," *The Seventh IEEE Conference on Artificial Intelligence Application*, 1991, ss. 29-32.

[28] Ö. Uzuner, et al. "2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text," *Journal of the American Medical Informatics Association*, c.18, s. 5, ss. 552-556, 2011.

[29] I. Segura-Bedmar, P. Martínez, M. Herrero Zazo, "Semeval-2013 task 9: Extraction of drug-drug interactions from biomedical texts (ddiextraction 2013)," *Second Joint Conference on Lexical and Computational Semantics (SEM)*, 2013, ss. 341-350.

[30] Piskorski, Jakub, et al. "The first cross-lingual challenge on recognition, normalization and matching of named entities in Slavic languages," *6th Workshop on Balto-Slavic Natural Language Processing*, 2017, ss. 76–85.

[31] D. Farmakiotou, et al. "Rule-based named entity recognition for Greek financial texts," *The Workshop on Computational lexicography and Multimedia Dictionaries (COMLEX 2000)*, 2000, ss. 75-78.

[32] Sang, Erik F., and Sabine Buchholz. "Introduction to the CoNLL-2000 shared task: Chunking," *2nd workshop on Learning language in logic and the 4th conference on Computational natural language learning*, 2000, ss. 127-132.

[33] L. Ratinov, D. Roth. "Design challenges and misconceptions in named entity recognition," *Thirteenth Conference on Computational Natural Language Learning (CoNLL '09)*, 2009, ss. 147-155.

[34] E. F. Sang, F. D. Meulder. “Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition,” *The Seventh Conference on Natural Language Learning at HLT-NAACL*, 2003, ss. 142-147.

[35] Z. Ju, J. Wang, F. Zhu, “Named Entity Recognition from Biomedical Text Using SVM,” *5th International Conference on Bioinformatics and Biomedical Engineering*, 2011, ss. 1-4.

[36] A. Ekbal, R. Haque, S. Bandyopadhyay, “Named entity recognition in Bengali: A conditional random field approach,” *Third International Joint Conference on Natural Language Processing*, 2008.

[37] D. Zeng, C. Sun, L. Lin, and B. Liu, “LSTM-CRF for drug-named entity recognition,” *Entropy*, c. 19, s. 6, ss. 283, 2017.

[38] S. Morwal, N. Jahan, and D. Chopra, “Named entity recognition using hidden Markov model (HMM),” *International Journal on Natural Language Computing*, c. 4, ss. 15-23, 2012.

[39] G. T. Ngompé, S. Harispe, G. Zambrano, J. Montmain, and S. Mussard, “Detecting sections and entities in court decisions using HMM and CRF graphical models.” *Advances in Knowledge Discovery and Management*, ss. 61-86, 2019.

[40] J. Li, A. Sun, J. Han, and C. Li, “A survey on deep learning for named entity recognition.”, *IEEE Transactions on Knowledge and Data Engineering*, Early Access, 2021.

[41] C. Zhang, H. Xu, “Using Citation-KNN for automatic keyword assignment.” *International Conference on Electronic Commerce and Business Intelligence*, 2009, ss. 131-134.

[42] Witten, I. H., Paynter, G. W., Frank, E., Gutwin, C. and Nevill-Manning, C. G. “Kea: Practical automated keyphrase extraction,” *Fourth ACM conference on Digital Libraries*, 1999, ss. 129-152.

[43] K. Zhang, H. Xu, J. Tang, J. Li, “Keyword extraction using support vector machine,” *International conference on web-age information management*, 2016, ss. 85-96.

[44] A. K. John, L. Di Caro, G. Boella, “A supervised keyphrase extraction system,” *12th International Conference on Semantic Systems*, 2016, ss. 57-62.

[45] M. R. Murty, J. V. R. Murthy, P. P. Reddy, S. C. Satapathy, “Statistical approach based keyword extraction aid dimensionality reduction,” *International Conference on Information Systems Design and Intelligent Applications (INDIA)*, 2012, ss. 445-452.

[46] S. Beliga, A. Meštrović, and S. Martinčić-Ipšić, “An overview of graph-based keyword extraction methods and approaches,” *Journal of information and organizational sciences*, c. 39, s. 1, ss. 1-20, 2015.

[47] M. Shishigan, C. Ridings, “PageRank Uncovered,” *Technical report*, 2002, ss. 1-55.

[48] C. Florescu, C. Caragea, “An unsupervised approach to keyphrase extraction from scholarly documents,” *55th Annual Meeting of the Association for Computational Linguistics*, 2017, ss. 1105-1115.

[49] R. Mihalcea, P. Tarau, “Bringing order into text”, C*onference on Empirical Methods in Natural Language Processing*, 2004, ss. 404-411.

[50] R. Campos, V. Mangaravite, A. Pasquali, A. Jorge, C. Nunes, and A. Jatowt, “YAKE! Keyword extraction from single documents using multiple local features”, *Information Sciences*, c. 509, ss. 257-289, 2020.

[51] D. A.Vega-Oliveros, P. S. Gomes, E. E. Milios, L. Berton, "A multi-centrality index for graph-based keyword extraction," *Information Processing & Management*, c. 56, s. 6,102063, 2019.

[52] A. Tixier, F. Malliaros, M. Vazirgiannis, "A graph degeneracy-based approach to keyword extraction," *Conference on Empirical Methods in Natural Language Processing*, 2016, ss. 1860-1870.

[53] F. C. Jonathan,O. Karnalim, "Semi-supervised keyphrase extraction on scientific article using fact-based sentiment,"*Telkomnika*, c. 16, s. 4, ss.1771-1778, 2018.

[54] H. M. Lynn, C. Choi, J. Choi, J. Shin, P. Kim, "The method of semi-supervised automatic keyword extraction for web documents using transition probability distribution generator," *International Conference on Research in Adaptive and Convergent Systems,* 2016, ss. 1-6.

[55] Z. Runqiang, "Text Similarity Calculation Method Based on Levenshtein and TFRSF," *Computer and Modernization*, c. 4, 2018.

[56] N. Gali, R. Mariescu-Istodor, D. Hostettler, P. Fränti, "Framework for syntactic string similarity measures," *Expert Systems with Applications*, c. 129, ss. 169-185, 2019.

[57] S. Temma, M. Sugii, H. Matsuno, "The document similarity index based on the Jaccard distance for mail filtering," *34th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC)*, 2019, ss. 1-4.

[58] M. Alewiwi, C. Orencik, E. Savaş, "Efficient top-k similarity document search utilizing distributed file systems and cosine similarity," *Cluster Computing*, c. 19, s. 1, ss. 109-126, 2016.

[59] N. Niraula, R. Banjade, D. Ştefănescu, V. Rus, "Experiments with semantic similarity measures based on lda and lsa," *International conference on statistical language and speech processing*, 2013, ss. 188-199.

[60] M. Farouk, "Measuring Sentences Similarity: A Survey," *Indian Journal of Science. And Technology*, c. 12, s. 25, ss. 1–11, Jul. 2019.

[61] C. Fellbaum, P. Vossen, P., "The Challenge of Multilingual WordNets," *Lexical Resources and Evaluation*, c. 46, ss. 313-326, 2012.

[62] P. Resnik. (1995, Kasım). *Using Information Content to Evaluate Semantic Similarity in a Taxonomy* [Çevrimiçi]. Erişim: http://arxiv.org/abs/cmp-lg/9511007. Erişim Tarihi: 11 Eylül 2020.

[63] D. Lin, "Extracting collocations from text corpora", *First workshop on computational terminology*, 1998, ss. 57-63.

[64] Z. Wu and M. Palmer. (1994, Haziran). *Verb Semantics and Lexical Selection* [Çevrimiçi]. Erişim: http://arxiv.org/abs/cmp-lg/9406033. Erişim Tarihi: 11 Eylül 2020.