TBV BBMD
TÜRKİYE BİLİŞİM VAKFI BİLGİSAYAR BİLİMLERİ VE MÜHENDİSLİĞİ DERGİSİ

TÜRKİYE BİLİŞİM VAKFI BİLGİSAYAR BİLİMLERİ VE MÜHENDİSLİĞİ DERGİSİ Haziran 2018 CİLT: 11 - SAYI: 2

11/2

11/2 2018

EDİTÖR KURULU

Adres:
TBV Bilgisayar Bilimleri ve Mühendisliği Dergisi
Kemankeş Karamustafa Paşa Mah.
Alipaşa Değirmeni Sok. No:3 34560 Karaköy / İstanbul
E-posta: info@tbv.org.tr

İÇİNDEKİLER

Sitopatolojik Değerlendirme Süreçleri için Optimum Aralığın Korunmasıyla Yüksek Çözünürlüklü Otomatik Panoramik Görüntüleme

High Resolution Automatic Panoramic Imaging by Maintaining Optimal Range for Cytopathological Analysis

Hülya DOĞAN, Elif BAYKAL, Murat EKİNCİ
Karadeniz Teknik Üniversitesi
Mühendislik Fakültesi
Bilgisayar Mühendisliği Bölümü
Trabzon, TÜRKİYE
{hülya, ebaykal, ekinci@ktu.edu.tr}

Mustafa Emre ERCİN, Şafak ERSÖZ
Karadeniz Teknik Üniversitesi
Tıp Fakültesi
Patoloji Bölümü
Trabzon, TÜRKİYE
{drmustafaemreercin, sersoz@ktu.edu.tr}

Öz

Mikroskop dar bir görüş alanına sahip olduğu için sitopatolojik değerlendirme süreçlerinde patologlar numunenin sadece belirli bir kısmını görebilmektedirler. Numunenin tüm alanını inceleyebilmek için mikroskop platformunu X-Y-Z yönünde hareket ettirerek numune üzerinde üç boyutlu tarama yapmaktadırlar. Yapılan çalışmada sitopatolojik değerlendirme süreçleri otomatikleştirilerek numunenin geniş görüş alanına sahip yüksek çözünürlüklü panoramik görüntüsünün elde edilmesi amaçlanmaktadır. Panoramik birleştirme sürecinin otomatikleştirilmesi için yapılan literatür çalışmalarında ortak alanlı görüntüler oluşturulurken mikroskopta var olan ve mikron cinsinden ölçülen odaklama derinliği dikkate alınmamaktadır. Bu yüzden ortak alanlı görüntüler arasında odaklama farklılıkları oluşmakta ve tarama anında bulanık görüntüler elde edilmektedir. Bu problemi çözmek için çalışmada odaklama derinliği artırılarak optimum odaklanmış ortak alanlı görüntüler oluşturulmaktadır. Önerilen yöntemin başarısının ispatı için literatürde önerilmiş 2 farklı tarama süreci kullanılarak panoramik görüntüler elde edilmiştir. Oluşturulmuş panoramik görüntüler referans görüntü gerektirmeyen metrikler kullanılarak karşılaştırılmış ve önerilen yöntemin başarısı hem sayısal hem de görsel sonuçlarla ispatlanmıştır.

Gönderme ve kabul tarihi: 30.01.2018-06.06.2018
Makale türü: Araştırma

Anahtar Sözcükler—Sitopatolojik Değerlendirme, Panoramik Görüntüleme, Odaklama Derinliği, Odaklama Derinliğinin Artırılması

Abstract

Since the microscope has a small field of view, pathologists can only see a certain part of the specimen during the cytopathological analysis process. In order to see the whole area of the sample, they scan the sample in three dimensions by moving the microscope platform in the X-Y-Z direction. The aim of the study is to obtain a high resolution panoramic image of the sample by automating the process of cytopathological analysis. Literature studies for the automation of the panoramic imaging process do not take into account the depth of focus measured in microns, which is present in the microscope, while creating images with the same field of view. This results in differences in focus between the images and during the scanning process blurred images are obtained. In order to solve this problem, the depth of focus is extended to produce optimum focused images. To evaluate the success of the proposed method, panoramic images were obtained using two different scanning processes suggested in the literature. The generated panoramic images are compared using the metrics without requiring reference image and the success of the proposed method is proved by both quantitative and visual results.

Keywords— Cytopathological Analysis, Panoramic Imaging, Depth of Focus, Extended Depth of Field

1. Giriş

Sitopatolojik değerlendirme süreci bireylerden çeşitli yöntemlerle alınan hücrelerin analizi esasına dayanmaktadır [1]. Mikroskop dar bir görüş alanına sahiptir. Bu yüzden süreç anında patologlar sadece numunenin belirli bir alanını görebilmektedirler. Numunenin tüm alanını görebilmek için mikroskop platformunu odaklamayı kaybetmeden X-Y-Z yönünde manuel olarak hareket ettirmektedirler [2]. El-göz koordinasyonu ile gerçekleştirilen sitopatolojik değerlendirme süreci oldukça zaman almaktadır. Bu süreç anında hastalık teşhisinin konması patoloğun tecrübesine bağlıdır. Bu yüzden patoloğun konsantrasyon bozukluğu sebebiyle numuneyi çok kısa sürede ya da dikkat etmeden incelemesi yanlış teşhis ve bulgulara sebep olabilmektedir. Numunenin geniş görüş alanına sahip panoramik görüntüsünün elde edilmesi patolağa olan bağımlığı azaltmakta ve bundan kaynaklanan eksiklikleri minimize etmektedir. Odaklama derinliği (depth of focus) nesnelerin tamamıyla net görülebildiği bir aralıktır. Mikroskobik sistemlerde odaklama derinliği, görüntüleme düzlemi sabit iken numunenin net görüntülenebildiği eksenel (Z) yöndeki mesafe olarak tanımlanmaktadır. Mikroskopta mikron cinsinden ölçülebilen kısıtlı odaklama derinliği bulunmaktadır. Mikroskop üzerinde incelenen numunenin eksenel yöndeki boyutu odaklama derinliğinden daha büyük olduğu durumlarda tüm alanı tamamıyla odaklanmış görüntü elde edilememekte ve odaklama derinliği dışında kalan bölgeler net gözükmemektedir. Literatürde ışıklı mikroskoplar için odaklama derinliği Eşitlik 1 kullanılarak hesaplanmaktadır.

$$d = \frac{\lambda}{n \sin \alpha^2} \tag{1}$$

Eşitlik 1'de d odaklama derinliğini, n objektifin ön merceği ile numune arasındaki ortamın kırılma indisini, αobjektife gelen ışığın kırılma açısını, λise ışığın dalga boyunu ifade etmektedir [3-5]. Işığın kırılma açısı (α) Eşitlik 2 kullanılarak hesaplanmaktadır. Eşitlik 2 ve 3'te NA objektifin kırılma açısını, M objektifin büyütme oranını ve const ise kullanılan mikroskop çeşidine göre seçilen bir değeri göstermektedir. Eşitliklerde görüldüğü gibi objektifin sayısal açıklığı (NA) ve mikroskopobjektifinin büyütme oranı (M) odaklama derinliği ile ters orantılıdır [6]. Bu yüzden mikroskop objektifinin büyütme oranı artıkça tüm alanı odaklanmış görüntü elde etmek mümkün olamamaktadır.

$$NA = n \sin \alpha \tag{2}$$

$$M = const x NA \tag{3}$$

Literatürde numunenin geniş görüş alanına sahip panoramik görüntüsünü elde etmeyi amaçlayan çalışmalar sıklıkla gerçekleştirilmiştir. Örneğin; Bin Ma çalışmasında panoramik birleştirme için geliştirilmiş bir yazılım olan Autostitch uygulamasını mikroskobik görüntüleri panoramik birleştirmek amacıyla kullanmıştır. Çalışmada manuel ve otomatik oluşturulan test setleri üzerinde panoramik birleştirme gerçekleştirilmiş ve karşılaştırmalar yapılmıştır [7]. Yang çalışmasında mikroskobik görüntü birleştirme sürecini hızlandırmayı amaçlamış ve klasik panorama adımlarına bir ön işlem eklemiştir. Önerdiği ön işlem kısmında faz korelasyon metodu kullanarak ortak alanları kabaca bulmuştur. Bulduğu bu ortak alanlar üzerinde diğer adımları gerçekleştirmiştir [8]. Appleton ve arkadaşları dinamik programlama tabanlı panoramik görüntü birleştirme algoritması önermişler ve var olan yöntemlerle karşılaştırmışlardır [9]. Sun çalışmasında panorama birleştirme adımlarından olan geometrik dönüşüm için parametre kestirimi gerçekleştirmiş ve doğruluğu yükseksonuç görüntü elde etmeyi sağlamıştır [10]. Loewke çalışmasında gerçek zamanlı çalışabilen etkin bir panoramik görüntü birleştirme algoritması sunmuştur. Çalışmasında görüntü hizalama hatalarını ve görüntü deformasyonunu minimize etmeye yönelik algoritmalar önermiş ve görüntü dizilerini konfokal mikroskop kullanarak elde etmiştir [11]. Wu çalışmasında mikroskobik sistemlerde panoramik görüntünün yapısını belirlemek için tarama kuralları ve görüntü birleştirme stratejileri belirlemiştir. Belirlediği tarama kurallarıyla tarama yönünde (X-Y) milimetrik ölçekli bir görüntüleme aralığı sağlamıştır[12]. Hsu çalışmasında mikroskobik görüntülerin panoramik birleştirilmesi için otomatik bir sistem geliştirmiştir. Klasik panoramik birleştirme adımlarına renk ayarı ve bozulma kompanzasyonu aşamalarını dahil etmiştir. Ek olarak çalışmada panoramik görüntü birleştirme sürecinin son aşaması olan piksel değerlerinin belirlenmesi kısmında dalgacık dönüşümü önerilmiştir [13]. Thevenaz klasik mikroskopların yanal alanını (X-Y yönünde) genişletme amacıyla yarı otomatikleştirilmiş bir yazılım geliştirmiştir. Geliştirdiği yazılım ortak alanlı görüntülerin kullanıcı tarafından kaba olarak hizalamasını gerektirmektedir[14]. Legesse çalışmasında lazer tarama mikroskobu kullanarak ortak alanlı görüntüleri dikişsiz olarak birleştirmeyi

amaçlamıştır [15]. Han ve arkadaşları patolojik dokulardan oluşan mikroskobik görüntüleri kullanarak geniş görüş alanına sahip tek bir görüntü elde etmeyi sağlayan otomatik bir görüntüleme sistemi geliştirmişlerdir. Çalışmada kenar detaylarının kaybolmasını önlemek için görüntüler arasında ortak alanlar oluşturulmuştur[16]. Yukarıda bahsedilen yayınlarda olduğu gibi literatür çalışmaları genelde sistemi hızlandırmaya, elde edilen panoramik görüntünün doğruluğunu artırmaya ve sistemin gerçek zamanlı çalışmasına yöneliktir. Yapılan çalışmalarda panoramik birleştirme için kullanılacak olan ortak alanlı görüntüler oluşturulurken mikroskopta var olan ve mikron cinsinden ölçülen odaklama derinliği dikkate alınmamıştır. Ortak alanlı görüntüler elde etmek için mikroskop üzerinde tarama yapılırken genelde Z ekseninde (optiksel yönde) hareket edilmemekte ve ortak alanlı görüntüler arasında odaklama farkının olup olmadığı kontrolü yapılmamaktadır. Oysaki mikroskopta var olan odaklama derinliğinden dolayı tarama anında ortak alanlı görüntüler arasında odaklama farklılıkları oluşmakta ve kontrol sağlanmadığı durumlarda bulanık görüntüler elde edilmektedir. Bu sebeplerden dolayı ortak alanlı görüntüler arasında eşleşen öznitelik nokta sayısı azalmakta ve sonuç olarak elde edilen panoramik görüntü bulanık olmaktadır. Gerçekleştirdiğimiz çalışmada bu problemi çözmek için tarama anında odaklama korunarak ortak alanlı görüntüler oluşturulmakta ve bu görüntüler kullanılarak panoramik birleştirme gerçekleştirilmektedir.

Bu çalışmada yüksek çözünürlüklü panoramik görüntü elde etmek için X-Y-Z ekseninde hareket ederek otomatik odaklama ve tarama yapabilen ışıklı mikroskobik görüntüleme sistemi kullanılmıştır. Çalışmada ilk olarak odaklama derinliği artırılarak optimum odaklanmış ortak alanlı 2 boyutlu görüntüler elde edilmiştir. Literatürde mikroskobik sistemlerde odaklama derinliğinin artırılması için Z ekseninde rastgele bir konumdan başlanmış ve belirli aralıkta hareket edilerek sabit sayıda görüntü alınmıştır. Sonuç olarak bu görüntülerdeki anlamlı bilgileri (odaklanmış bölgeleri) içeren tek bir görüntü elde edilmiştir [5-6, 17-19]. Literatürde yaptığımız çalışmada odaklama derinliğininartırılması sürecinde Z ekseninde taranan aralığın ve kullanılan görüntü sayısının etkili olduğu ve optimum alınmadıkları taktirde tüm alanı odaklanmış görüntünün elde edilemediği ispatlanmıştır [20]. Yapılan bu çalışmada da odaklama derinliğininartırılması süreci için Z ekseninde taranan aralık ve kullanılan görüntü sayısı rastgelelikten çıkarılmıştır. Z ekseninde numunenin tüm alanının taranması garanti edilmiş ve böylece numunenin tüm anlamlı verilerinin elde edilmesi sağlanmıştır. Çalışmanın ikinci aşamasında tarama (X-Y) yönünde hareket edilmiş ve optimum aralığın kontrolü sağlanmıştır. Son aşamada ise elde edilen optimum odaklamaya sahip ortak alanlı görüntüler birleştirilerek numunenin geniş görüş alanına sahip panoramik görüntüsü elde edilmiştir.

Çalışmanın 2. bölümünde önerilen yöntem ve kullanılan algoritmalar, 3. bölümünde ise deneysel sonuçlar ve karşılaştırmalar sunulmaktadır.

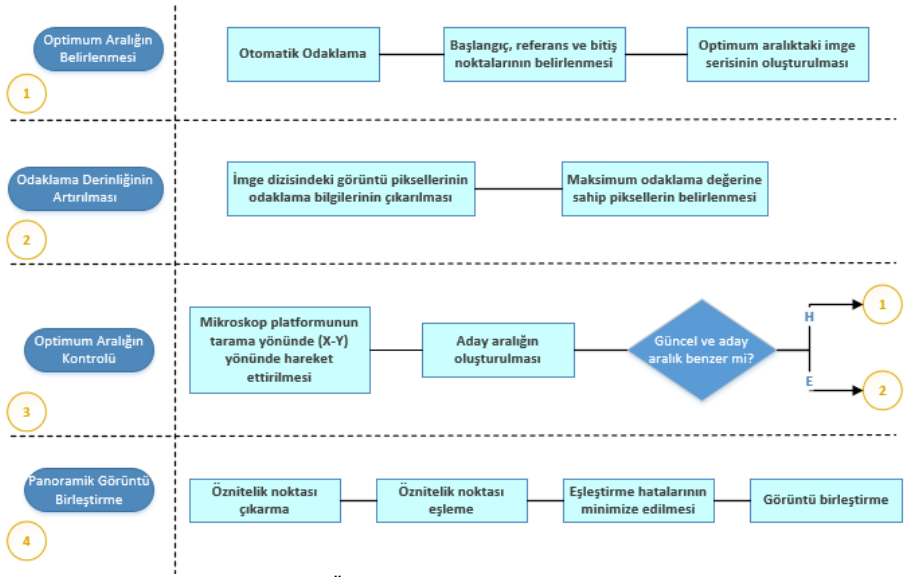
2. Önerilen Yöntem

Yapılan çalışmada optimum aralığın korunmasıyla sitopatolojik değerlendirme süreçleri için hazırlanmış numunenin yüksek çözünürlüklü panoramik görüntüsünün elde edilmesi amaçlanmaktadır. Önerilen çalışmanın akış diyagramı Şekil 1'de gösterilmiş olup aşamaları şu şekildedir:

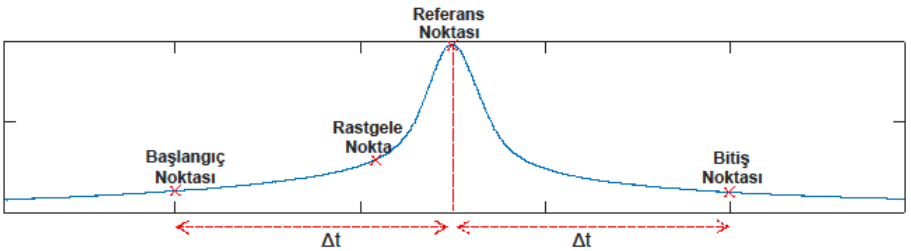
2.1 Optimum Aralığın Belirlenmesi

Yapılan çalışmanın ilk aşamasında odaklama derinliğinin artırılması süreci için Z ekseninde taranan aralık ve kullanılan görüntü sayısı rastgelelikten çıkarılmaktadır. Z ekseninde taranan aralık kullanılan numune ve objektife göre optimize edilebilmektedir.Şekil 2'de Z ekseninde hareket edilerek elde edilmiş görüntülerin odaklama değerleri mevcuttur. Şekilde görüldüğü gibi odaklama bakımından zengin ve odaklama derinliği artırma sürecinde etkili olan görüntüler odaklama değeri maksimum olan görüntüye yakın konumdadırlar. Numunenin Z ekseninde tüm alanının taranması ve odaklama bakımından tüm bilgilerin elde edilmesi için maksimum odaklama değerine sahip olan görüntü ve yakın konumdaki görüntülerin kestirilmesi garanti edilmelidir. Bu amaçla çalışmada en yüksek odaklama değerine sahip olan görüntü referans kabul edilmektedir. Referans görüntünün bulunması için herhangi bir rastgele noktadan başlanarak otomatik odaklama işlemi gerçekleştirilmektedir. Otomatik odaklama sürecinde gerçekleştirilen adımlar şu şekildedir [2]: (1) Adım motoru kontrolüyle mikroskop platformu Z ekseninde hareket ettirilerek farklı dikey konumlarda sabit sayıda görüntü dizisi elde edilir. (2) Dizideki her görüntünün odaklama değeri otomatik odaklama fonksiyonu kullanılarak hesaplanır. (3) Dizinin olasılık yoğunluk fonksiyonu (oyf) oluşturulur. (4) oyf'nin tepe noktası hesaplanır ve oyf'lerin tepe noktaları arasındaki değişim negatif oluncaya kadar 1-4 arası adımlar tekrarlanır. Otomatik

odaklama işlemi sonrasında oyf'nin tepe noktası referans noktası olarak atanmaktadır. Z ekseninde mikroskop platformu başlangıç noktasından bitiş noktasına kadar hareket ettirilmektedir.



Şekil 1. Önerilen Çalışmanın Akış Diyagramı.



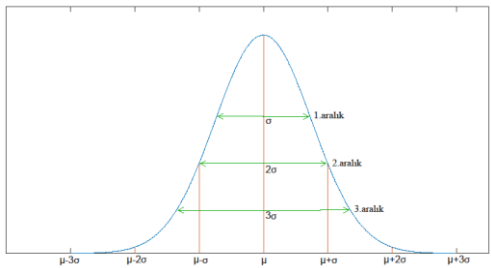
Şekil 2. Otomatik Odaklama ile başlangıç, referans ve bitiş noktalarının belirlenmesi.

rastgele konumdan başlanıldığı için odaklama derinliğinin artırılması sürecinde etkili olan tüm alanın tarandığı garanti edilemez. Bu yüzden başlangıç noktasını belirlemek için mikroskop platformu zıt yönde hareket ettirilmektedir. Başlangıç noktasını (x_i) hesaplamak için Eşitlik 4' te görüldüğü gibi noktalar arasındaki eğimlerin farkı kullanılmaktadır.

$$\left| \frac{f_i - f_{i+1}}{x_i - x_{i+1}} - \frac{f_{i+1} - f_{i+2}}{x_{i+1} - x_{i+2}} \right| \leq \varphi \qquad (4)$$

Eşitlik 4'te f_i, f_{i+1}ve f_{i+2}değerleri x_i, x_{i+1}ve x_{i+2}indisli görüntülerin odaklama değerlerini temsil etmektedir.

Bitiş noktası Şekil 2'de gösterildiği gibi başlangıç ve referans noktaları arasındaki uzaklık (Δt) kullanılarak hesaplanmakta ve imge dizisini tamamlamak için



Şekil 3. Farklı uzaklıklara sahip aralıklar.

Bundan sonraki aşamada referans nokta merkez alınarak Şekil 3'te görüldüğü gibi imge dizisi üzerinde farklı uzaklıklara sahip aralıklar tanımlanmaktadır. Aralıkları belirli bir matematiksel modele göre

tanımlamak için Eşitlik 5'te verilen Gauss (Normal) Dağılımı kullanılmakta ve önceki adımda oluşturulan imge dizisindeki görüntülerin odaklama değerlerine bu dağılım uydurulmaktadır.

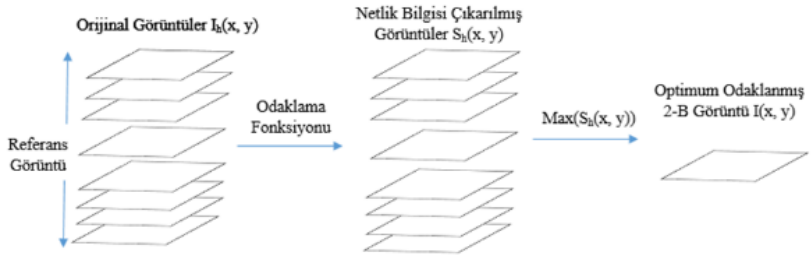
$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{(x-\mu)^2}{2\sigma^2}} \qquad (5)$$

$$\mu = \frac{\sum_{i=1} x_i y_i}{\sum_{i=1} y_i} \quad \sigma = \sqrt{\sum_{i=1} (x_i - \mu)^2 \frac{y_i}{\sum_{i=1} y_i}}$$

hesaplamak için çeşitli yaklaşımlar önerilmiştir [21]. Bunlar çoklu çözünürlük tabanlı (Laplacian [22], Gradient [23], Fourier [24], Cosinus [25], Wavelet [26], Curvelet [19] ve Contourlet dönüşümü [27]), seyrek gösterim tabanlı [28],komşu tabanlı [29-30] (varyans, gradyan, tenegrad)ve hibrit [21] olmak üzere üçe ayrılmaktadırlar.

- Maksimum Odaklama Değerine Sahip Pikselin Belirlenmesi:

Bu adımda imge dizisindeki maksimum odaklama değerine sahip olan pikseller alınarak optimum odaklanmış tek bir görüntü elde edilmektedir.



Şekil 4. Odaklama derinliğinin artırılması sürecinde gerçekleştirilen adımlar.

Eşitlik 5'te y_i, x_i indisli görüntünün odaklama değeridir. μ ve σ ise odaklama değerlerinin ortalama ve standart sapmasını temsil etmektedir.

Bu çalışmada önceki aşamada elde edilen imge dizisindeki görüntülerin odaklama değerleri, sahip oldukları ortalama ve standart sapmaya göre modellenmekte ve Şekil 3'te gösterildiği gibi farklı uzaklıklara (σ, 2σ, 3σ) sahip aralıklar tanımlanmaktadır. Kurduğumuz matematiksel model odaklama hakkında bilgi içeren değerler kullanılarak oluşturulduğundan tanımlanan aralıklar kullanılan numune tipine, mikroskop objektifine ve kamera tipine göre adapte edilebilmektedir.

2.2 Odaklama Derinliğinin Artırılması

Çalışmanın bu aşamasında optimum aralıktaki imge dizisi kullanılarak odaklama derinliği artırılmakta ve optimum odaklanmış 2 boyutlu görüntü elde edilmektedir. Şekil 4'te akışı verilen bu aşamada gerçekleştirilen adımlar şu şekildedir:

- İmge Dizisindeki Görüntü Piksellerinin Odaklama Bilgilerinin Çıkarılması:

İmge dizisi oluşturulduktan sonra görüntülerdeki tüm piksellerin odaklama bilgisi çıkarılmaktadır. Literatürde piksellerin odaklama değerlerini

2.3 Optimum Aralığın Kontrolü

Ortak alanlı optimum odaklanmış görüntüler elde etmek için tarama (X-Y) yönünde hareket edilmekte ve her hareket anında optimum aralığın kontrolü sağlanmaktadır. Bu işlemler için gerçekleştirilen adımlar şu şekildedir:

1. Mikroskop platformu adım motoru kontrolüyle tarama yönünde (X ya da Y) hareket ettirilir.
2. Platform Z yönünde önceki adımda belirlenen optimum aralık (güncel aralık) boyunca hareket edilerek aday aralıktaki imge dizisi elde edilir.
3. Aday aralıktaki görüntülerin odaklama değerleri hesaplanır.
4. Güncel ve aday aralıktaki görüntülerin odaklama değerleri arasında Bhattacharyya uzaklığı [31] hesaplanır ve uzaklık belirli bir eşik değeri ile karşılaştırılır.

$$BD = \left(\frac{1}{8}(\mu_g - \mu_a)^T \left(\frac{\Sigma_g + \Sigma_a}{2} \right)^{-1} (\mu_g - \mu_a) + \frac{1}{2} \ln \frac{|(\Sigma_g + \Sigma_a)/2|}{|\Sigma_g|^{1/2} + |\Sigma_a|^{1/2}} \right) \qquad (6)$$

Eşitlik 6'da *BD* güncel ve aday aralıktaki görüntülerin odaklama değerleri arasındaki Bhattacharyya uzaklığını, μ_g ve μ_a güncel ve aday aralıktaki görüntülerin odaklama değerlerininortalamalarını,$\Sigma_g$ve Σ_a ise güncel ve aday aralıktakigörüntülerinodaklama değerlerininkovaryansını ifade etmektedir.

5. Uzaklık eşik değerinden büyük ise güncel aralık aşama 1'e dönülerek yenilenmekte, büyük olmadığı durumda ise aşama 2'ye dönülerek X veya Y yönünde taramaya devam edilir.

2.4 Panoramik Görüntü Birleştirme

Çalışmanın bu aşamasında ortak alanlı 2 boyutlu optimum odaklanmış görüntüler panoramik olarak birleştirilmektedir. Panoramik birleştirme sürecinde gerçekleştirilen işlem adımları şu şekilde sıralanmaktadır:

2.4.1 Öznitelik Noktası Çıkarma

Bu adımda ortak alanlı görüntülerin belirgin öznitelik noktaları (kapalı sınır bölgeleri, kenarlar, köşeler) çıkarılmaktadır. Gerçekleştirilen çalışmada görüntülerin öznitelik noktalarını belirlemek için SIFT (Scale Invarient Feature Transform) algoritması kullanılmaktadır.

Scale Invariant Feature Transform (SIFT): David G. Lowe tarafından 1999 yılında önerilmiş SIFT algoritması görüntülerden ayırıcı öznitelik çıkarmak amacıyla literatürdesıklıkla kullanılmaktadır [32]. SIFT kullanılarak elde edilen öznitelikler ölçekleme, dönme ve ötelemeden bağımsızdır. Öznitelik elde etmek için gerçekleştirilen işlemler şu şekildedir:

- Ölçeksel Uzaydaki Ekstramum Noktaların Belirlenmesi:

Öznitelikleri çıkarılacak olan görüntünün ($I(x, y)$) ölçeksel uzayı ($L(x, y, \sigma)$) Eşitlik 7 kullanılarak oluşturulmaktadır. Eşitlikte $G(x, y, \sigma)$ farklı standart sapmalara sahip Gauss filtrelerini göstermektedir.

$$L(x,y,\sigma) = G(x,y,\sigma) * I(x,y) \quad (7)$$

Eşitlik 8'de görüldüğü gibi farklı standart sapmalara sahip Gauss filtreleri ile konvolüsyon edilmiş görüntülerin farkları alınarak Gauss uzayının farkları hesaplanmaktadır.

$$D(x,y,\sigma) = (G(x,y,k\sigma) - G(x,y,\sigma)) * I(x,y) \quad (8)$$

Görüntüdeki piksellerin yerel ekstremum nokta olup olmadığına aynı ölçek uzayındaki 8 komşu piksel, alt ve üst ölçek uzayındaki 9 komşu piksel ile karşılaştırılarak karar verilmektedir.

- Öznitelik Noktası Konumlarının Doğrulanması:

DoG operatörü gürültüye hassas olmasının yanında yoğun kenardan da etkilenmektedir. Bundan dolayı önceki adımda elde edilen ekstremum noktalardan düşük kontrasta sahip olanlar 2. dereceden Taylor serisi açılımıyla, kenar bölgelerinde olanlar ise Hessian matrisi kullanılarak elimine edilmektedir.

- Öznitelik Noktalarına Yön Atanması:

Öznitelik noktalarına dönmeden bağımsızlık özelliği kazandırmak için yön ataması yapılmaktadır. Bunun için her bir öznitelik noktası etrafında gradyan büyüklükleri ($m(x, y)$) ve yönleri ($\theta(x, y)$) sırasıyla Eşitlik (9) ve (10) kullanılarak hesaplanmakta ve bu bölgedeki en belirgin yön, öznitelik noktasının yönü olarak atanmaktadır.

$$m(x,y) = \sqrt{((L(x+1,y) - L(x-1,y))^2) + ((L(x,y+1) - L(x,y-1))^2)} \quad (9)$$

$$\theta(x,y) = \tan^{-1} \frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)} \quad (10)$$

- SIFT Öznitelik Tanımlayıcısı:

Öznitelik noktaları etrafında 16x16 boyutlu blok alınarak 4x4 boyutlu bloklara bölünmektedir. Her bir blok için bloktaki piksellerin gradyan büyüklükleri hesaplanarak 8 bölme içeren histogramı bulunmakta ve sonuç olarak 4x4x8=128 boyutlu bir öznitelik tanımlayıcısı oluşturulmaktadır.

2.4.2 Öznitelik Noktası Eşleme

Bu adımdaöznitelik tanımlayıcıları arasındaki benzerlik incelenmektedir. Çalışmada ortak alanlı görüntüler arasında eşleşmiş öznitelik noktalarını bulmak için Öklid uzaklığı kullanılmaktadır. Bu uzaklık ölçütü için R(r_1, r_2, r_3, …, r_n) ve T(t_1, t_2, t_3, …, t_n) noktaları arasındaki benzerlik Eşitlik 11 kullanılarak hesaplanmaktadır.

$$U_{RT} = \sqrt{\sum_{i=1}^{n} (r_i - t_i)^2} \quad (11)$$

2.4.3 Eşleştirme Hatalarının Minimize Edilmesi ve Model Oluşturma

Çalışmada eşleştirme hatalarını minimize etmek ve görüntüler arasında model oluşturmak için RANSAC kullanılmaktadır.

RANSAC (Random Sample Consensus): 1981 yılında Fischler tarafından geliştirilen RANSAC eşleşme hatalarını minimize etmek ve görüntüler arası model oluşturmak için homograf matrisi hesaplama aşamasında kullanılmaktadır [33]. Yöntemde gerçekleştirilen işlemler şu şekildedir:

(1) N çift öznitelik noktasından dört çift öznitelik noktası seçilir.

(2)Eşitlik 12 kullanılarak Homograf matrisin parametreleri hesaplanır.

$$P_b = H_{ab} P_a \quad (12)$$

Eşitlikte P_b ve P_a görüntü öznitelik noktalarını, H_{ab}ise homograf matrisi temsil etmektedir.

(3) Bulunan parametrelere göre kalan N - 4 öznitelik noktası çiftinin mesafeleri hesaplanır.

(4) Hesaplanan mesafeler belirlenen eşik değeri ile karşılaştırılarak aykırı durumda olup olmadıkları kontrol edilir.

(5) Minimum aykırı durumda olmayan öznitelik noktası içeren Homograf matrisi hesaplanana kadar adımlar tekrar edilir.

2.4.4 Görüntü Birleştirme

Bu adımda panoramik görüntü birleştirme süreci sonunda elde edilecek görüntüdeki ortak alanların hangi piksel değerlerini alacağı belirlenmektedir. Bu çalışmada sonuç görüntüdeki ortak alanların piksel değerleri, panoramik birleştirilen görüntülerdeki ortak alanların piksel değerlerinin maksimumlarının seçilmesiyle belirlenmektedir.

3 Deneysel Sonuçlar

3.1 Mikroskobik Görüntüleme Sistemi

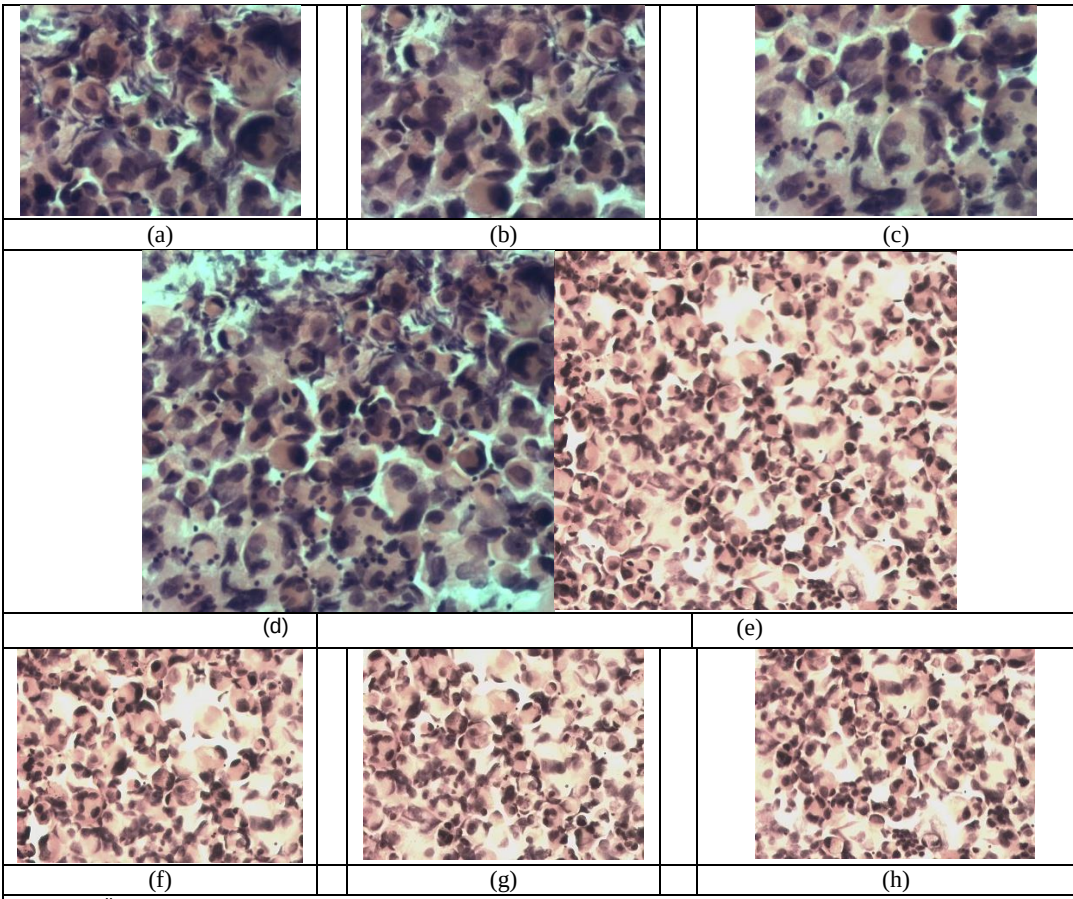
Çalışmada X-Y-Z eksenlerinde hareket ederek otomatik odaklama ve tarama yapabilen motorize mikroskop sistemi kurulmuştur. Sistem Intel Core i7 CPU, 8 GB RAM ve Windows 10 işletim sistemi özelliklerine sahip bir PC'den, Nikon Eclipse 80i marka bir mikroskoptan, 14 MP çözünürlüğe sahip bir CMOS kameradan, platformun X-Y-Z yönlerinde hareketinisağlayan 3 adet step motordan ve PC ile step motorlar arasındaki bilgi alışverişini gerçekleştiren kontrol devresinden oluşmaktadır.

3.2 Veritabanı

Çalışmada önerilen yöntemin performansını değerlendirmek için mikroskobik görüntülerden oluşan yeni bir veri tabanı oluşturulmuştur. Kullanılan veritabanı plevral efüzyon sıvısı sitopatolojik değerlendirmesi amacıyla Karadeniz Teknik Üniversitesi Tıp Fakültesi Patoloji Anabilim Dalında hazırlanmış 2 farklı numune (Numune 1 ve 2) üzerinden elde edilmiştir. Panoramik birleştirilecek ortak alanlı görüntüler 20X büyütme objektifi ile taranarak elde edilmiş ve 1280x960 çözünürlüğünde kaydedilmiştir.

Önerilen yöntem kısmında bahsedildiği gibi ilk aşamada Z ekseni boyunca taranan aralık ve görüntü sayısı rastgelelikten çıkarılmıştır. Optimum aralığın belirlenmesi aşamasının otomatik odaklama adımındakullanılan veritabanı içinoptimum olduğu ispatlanmış varyans odaklama fonksiyonu olarak kullanılmıştır [34]. Optimum aralık olarak 1. aralık (σ) kabul edilmiştir [20]. Odaklama derinliğinin artırılması aşamasında piksellerin odaklama bilgilerini çıkarmak için gradyan (5x5) fonksiyonu kullanılmış ve optimum odaklanmış 2 boyutlu görüntüler elde edilmiştir. Optimum aralığın kontrolü aşamasında güncel ve aday aralıklar arasındaki Bhattacharyya uzaklığı için eşik değeri 0.01 olarak belirlenmiştir.

Şekil 5'te Numune 1 ve 2 için önerilen yöntemin akışıyla elde edilmiş ortak alanlı görüntülerden örnekler (Şekil 5 a-b-c – Numune 1 ve Şekil 5 f-g-h – Numune 2) ve tüm ortak alanlı görüntülerin kullanılmasıyla oluşturulmuş panoramik görüntüler (Şekil 5d – Numune 1ve Şekil 5e – Numune 2) mevcuttur. Önerilen yöntemin başarısını ispatlamak için çalışma literatürdeki 2 farklı tarama süreci ile karşılaştırılmıştır. Literatürde klasik yöntem olan birinci tarama sürecinin akışı şu şekildedir: ilk olarak Z ekseninde optimum odaklama ile odaklanmış görüntü belirlenmiştir. Odaklanmış görüntü belirlendikten sonra Z ekseninde kontrol yapılmadan X-Y yönünde tarama yapılmış ve ortak alanlı görüntüler oluşturulmuştur. Son olarak elde edilen ortak alanlı görüntüler panoramik birleştirilmiştir. Şekil 6'da Numune 1 ve 2 için klasik yöntemin akışıyla elde edilmiş ortak alanlı görüntülerden örnekler (Şekil 6 a-b-c – Numune 1 ve Şekil 6 f-g-h – Numune 2) ve tüm ortak alanlı görüntülerin kullanılmasıyla oluşturulmuş panoramik görüntüler (Şekil 6d – Numune 1ve Şekil 6e – Numune 2) mevcuttur. Önerilen yöntemin karşılaştırıldığı ikinci yöntem ise



Şekil 5. Önerilen yöntemin akışıyla elde edilmiş Numune 1 için ortak alanlı görüntülerden örnekler (a) – (b) – (c) ve tüm ortak alanlı görüntülerin kullanılmasıyla oluşturulmuş panoramik görüntü (d), Numune 2 için ortak alanlı görüntülerden örnekler (f) – (g) – (h) ve tüm ortak alanlı görüntülerin kullanılmasıyla oluşturulmuş panoramik görüntü (e)

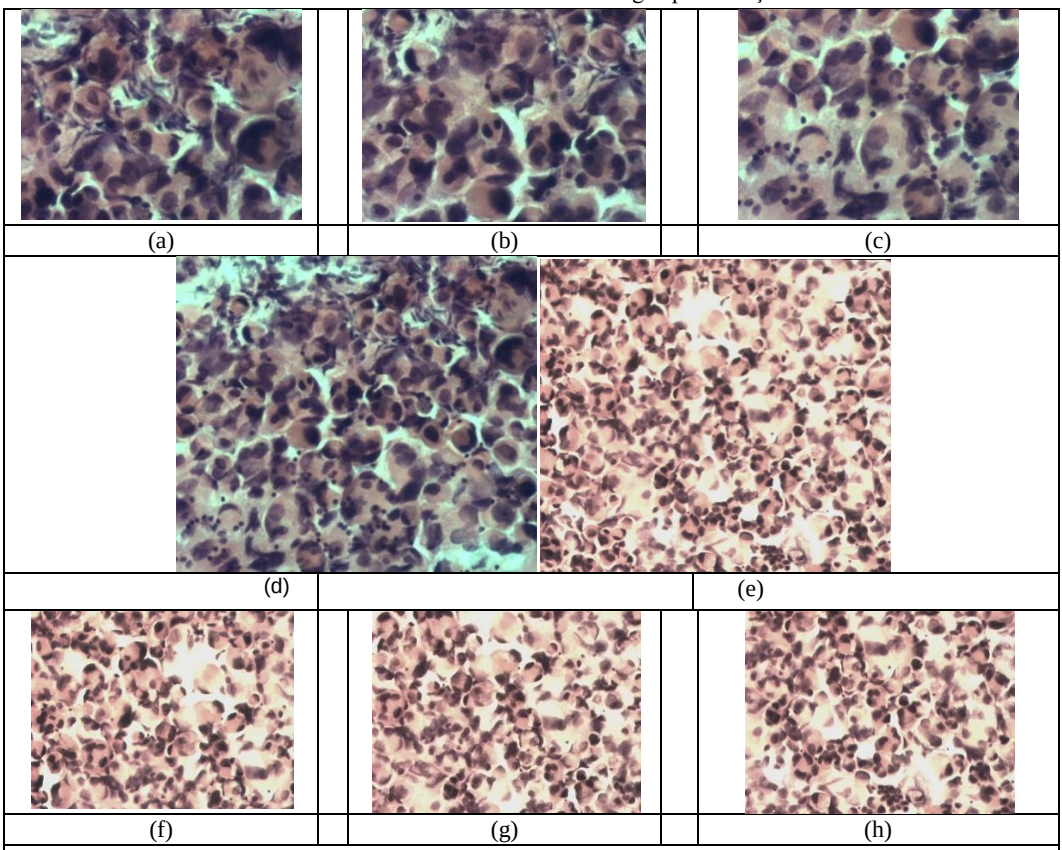
Doğan ve arkadaşları tarafından önerilmiştir [2]. Şekil 7'de Numune 1 ve 2 için Doğan ve arkadaşlarının önerdiği yöntemin akışıyla elde edilmiş ortak alanlı görüntülerden örnekler (Şekil 7 a-b-c – Numune 1 ve Şekil 7 f-g-h – Numune 2) ve tüm ortak alanlı görüntülerin kullanılmasıyla oluşturulmuş panoramik görüntüler (Şekil 7d – Numune 1ve Şekil 7e – Numune 2) mevcuttur. Her tarama süreci için ortak alanlıgörüntüleri oluştururken x yönünde 5 adım, Y yönünde ise 5 adım hareket edilmiştir.

Şekil 5-6-7'de farklı tarama süreçleri kullanılarak elde edilmiş panoramik görüntülerde gösterildiği gibi mikroskobik sistemler için panoramik birleştirme sürecinde odaklama derinliği etkilidir.Literatürdeki klasik süreçte olduğu gibi odaklama derinliği dikkate alınmadan oluşturulan ortak alanlı görüntüler tarama süresince bulanıklaşmakta ve sonuç olarak da bulanık bir panoramik görüntü elde edilmektedir. Odaklama kontrolü yapılmadan elde edilen ortak alanlı görüntülerin panoramik birleştirme sürecinde görüntülerdeki bulanıklıklardan dolayı zamanla öznitelik noktası çıkarılamamaktadır. Öznitelik noktası çıkarılamadığı için süreç durmakta ve ortak alanlı görüntüler arasında ilişki kurulamamaktadır. Doğan ve arkadaşların önerdiği süreçte Z ekseninde belirli sayıda görüntü alınarak odaklanmış görüntü elde edilmiş ve bu sayı tarama boyunca değiştirilmemiştir. Oysaki Z ekseninde alınan görüntü sayısı kullanılan numune ve objektife göre ayarlanmalıdır. Görüntü sayısının yeterli olmadığı ve Z ekseninde numunenin tam taranmadığı durumlarda bulanık görüntüler elde edilebilmektedir. Önerilen tarama süreci ile bu problemler ortadan kaldırılmaktadır

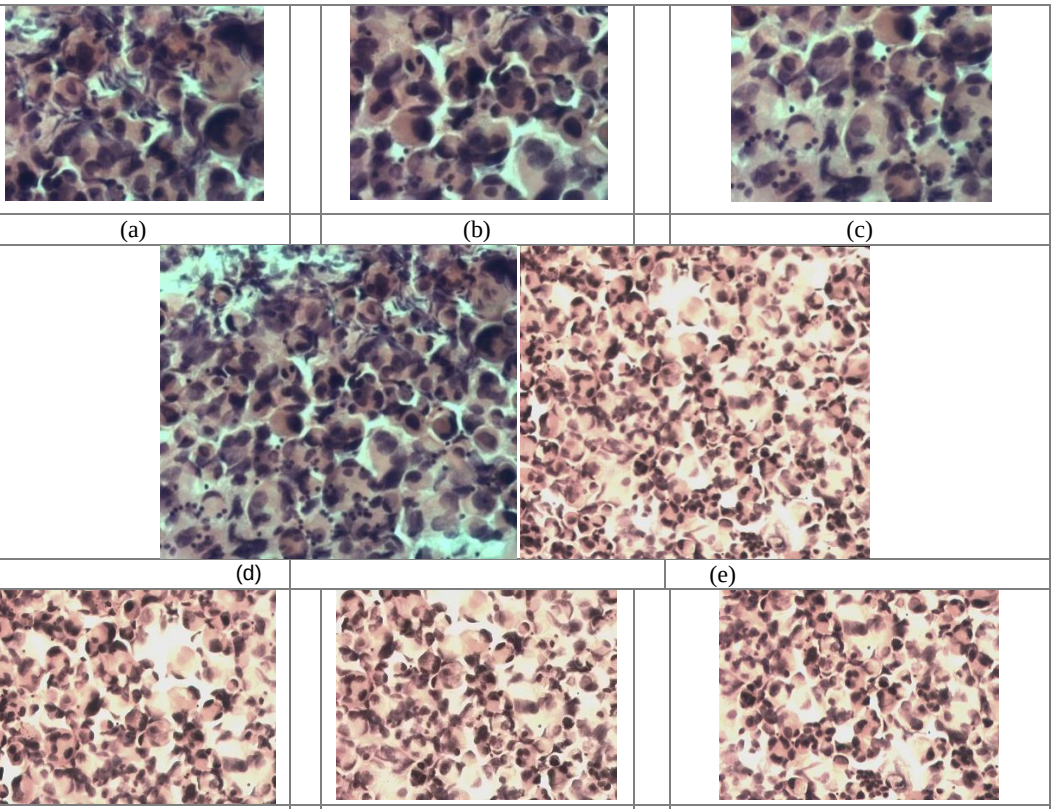
Tablo1. Numune 1 ve 2 için farklı tarama süreçleri ile elde edilmiş panoramik görüntülerin karşılaştırma metrikleri sonuçları

	Tenen	***SS***	***BM***
Şekil 5d	98.8642	56.0951	0.6338
Şekil 6d	96.5657	54.9042	0.7249
Şekil 7d	97.5204	55.0782	0.6355
Şekil 5e	134.9231	55.5699	0.5193
Şekil 6e	120.6450	52.5682	0.6743
Şekil 7e	125.7802	53.2309	0.5568

Çalışmada görsel sonuçlara ek olarak panoramik birleştirme süreçlerini kıyaslamak için referans görüntü gerektirmeyen karşılaştırma metrikleri kullanılmıştır. Panoramik birleştirme sonucunda oluşturulan görüntünün çözünürlük kalitesinin yüksek, bulanıklığının ise az olması beklenmektedir. Çalışmada panoramik görüntülerin çözünürlük kalitelerini ölçmek için Tenengrad (Tenen) [30] ve standart sapma (SS), bulanıklık bilgisini ölçmek için ise bulanıklık (BM) [35] metrikleri tercih edilmiştir. Tablo 1'de Numune 1 ve 2 için 3 farklı tarama süreci kullanılarak elde edilmiş panoramik görüntülerin karşılaştırma metrik sonuçları mevcuttur. Hesaplanan sonuçlarda görüldüğü gibi klasik süreçte odaklama kontrolü yapılmadığından en düşük değere sahip panoramik görüntüler oluşmuştur. Diğer iki tarama sürecinde ise yakın sonuçlar elde edilmiştir. Çözünürlük kalitesi bakımından ise önerilen yöntem ile oluşturulan panoramik görüntüler daha yüksek değerlere sahiptir. Sonuç olarak önerilen yöntem ile daha kaliteli görüntüleme sağlandığı ve bulanıklığın azaltıldığı ispatlanmıştır.



Şekil 6. Klasik yöntemin akışıyla elde edilmiş Numune 1 için ortak alanlı görüntülerden örnekler (a) – (b) – (c) ve tüm ortak alanlı görüntülerin kullanılmasıyla oluşturulmuş panoramik görüntü (d), Numune 2 için ortak alanlı görüntülerden örnekler (f) – (g) – (h) ve tüm ortak alanlı görüntülerin kullanılmasıyla oluşturulmuş panoramik görüntü (e).



Şekil 7. Doğan ve arkadaşları tarafından önerilen yöntemin akışıyla elde edilmiş Numune 1 için ortak alanlı görüntülerden örnekler (a) – (b) – (c)ve tüm ortak alanlı görüntülerin kullanılmasıyla oluşturulmuş panoramik görüntü (d), Numune 2 için ortak alanlı görüntülerden örnekler (f) – (g) – (h)ve tüm ortak alanlı görüntülerin kullanılmasıyla oluşturulmuş panoramik görüntü (e).

4 Sonuçlar

Yapılan çalışmada sitopatolojik inceleme amacıyla hazırlanan numunenin optimum aralığın korunmasıyla yüksek çözünürlüklü panoramik görüntüsü elde edilmiştir. Önerilen çalışmanın dört temel katkısı mevcuttur: (1) Z ekseninde numunenin tüm alanının taranmasının garanti edilmesi, (2) Odaklama derinliğinin artırılması sürecinde Z ekseninde taranan aralığın ve kullanılan görüntü sayısının rastgelelikten çıkarılması, (3) Optimum aralığın belirlenmesi ve X-Y yönünde tarama boyunca kontrolünün sağlanması, (4) X-Y-Z eksenlerinde hareket ederek otomatik odaklama ve tarama yapabilen yeni bir ışıklı mikroskobik görüntüleme sisteminin geliştirilmesi Literatürdeki çalışmaların aksine, çalışmada ortak alanlı görüntüler oluşturulurken odaklama derinliği dikkate alınmıştır. Bu sayede ortak alanlı görüntüler arasında oluşan odaklama farklılıkları engellenmiş ve yüksek kontrasta ve düşük bulanıklığa sahip panoramik görüntüler oluşturulmuştur. Önerilen yöntem literatürdeki 2 farklı tarama süreci ile karşılaştırılmış ve başarısı hem görsel hem de sayısal sonuçlarla ispatlanmıştır

Teşekkür

Bu çalışma, 117E961 nolu TUBİTAK projesinin desteği altında KTÜ Bilgisayarla Görü ve Örüntü Tanıma Laboratuvarınca yürütülmüştür.

Kaynakça

[1] Schneider T. E., Bell A. A., Meyer-Ebrecht D., Böcking A., Aach T. “Computer aided cytological cancer diagnosis: cell type classification as a step

towards fully automatic cancer diagnostics on cytopathological specimens of serous effusions", Medical Imaging, International Society for Optics and Photonics, vol.6514, pp. 6514-6524, 2007.
[2] Doğan H., Ekinci M., "Automatic panorama with auto-focusing based on image fusion for microscopic imaging system", Signal, Image and Video Processing, vol. 8, pp. 5-20, 2014.
[3] Born M., Wolf E., "Principles of Optics (7th Ed)", Cambridge University Press, 1999.
[4] Goldsmith N.T., "Deep focus; a digital image processing technique to produce improved focal depth in light microscopy", Image Analysis – Stereology, vol. 19, pp. 163-167, 2011.
[5] Piccinini F., Tesei A., Zoli W., Bevilacqua A., "Extended depth of focus in optical microscopy: Assessment of existing methods and a new proposal", Microscopy Research and Technique, vol. 75, pp. 1582-1592, 2012.
[6] Forster B., Van De Ville D., Berent J., Sage, D., Unser M., "Complex wavelets for extended depth-of-field: A new method for the fusion of multichannel microscopy images", Microscopy Research and Technique, vol. 65, pp. 33–42, 2004.
[7] Ma B., Zimmermann T., Rohde M., Winkelbach S., He F., Lindenmaier W., Dittmar K. E., "Use of autostitch for automatic stitching of microscope images", Micron, vol. 38, pp. 492-499, 2007.
[8] Yang F., Deng Z. S., Fan Q. H., "A method for fast automated microscope image stitching", Micron, vol. 48, pp. 17-25, 2013.
[9] Appleton B., Bradley A. P., Wildermoth M., "Towards Optimal Image Stitching for Virtual Microscopy", in Digital Image Computing: Techniques and Applications (DICTA'05), Queensland, Australia, 2005, pp. 44-44.
[10] Sun C., Beare R., Hilsenstein V., Jackway P., "Mosaicing of microscope images with global geometric and radiometric corrections", Journal of Microscopy, vol. 224, pp. 158-165, 2006.
[11] Loewke K. E., Camarillo D. B., Piyawattanametha W., Mandella M. J., Contag C. H., Thrun S., Salisbury J. K.,"In vivo micro-image mosaicing" IEEE Transactions on Biomedical Engineering, vol. 58, pp. 159-171, 2011.
[12] Wu Y., Fang Y., Liu X., Ren X., Guo J., Yuan X., "Millimeter scale global visual field construction for atomic force microscopy based on automatic image stitching", In Manipulation, Automation and Robotics at Small Scales (MARSS), 2017, pp. 1-5.
[13] Hsu W. Y., Poon W. F., Sun Y. N., "Automatic seamless mosaicing of microscopic images: enhancing appearance with colour degradation compensation and wavelet-based blending", Journal of Microscopy, vol. 231, pp. 408-418, 2008.
[14] Thévenaz P., Unser M., "User-friendly semiautomated assembly of accurate image mosaics in microscopy", Microscopy Research and Technique, vol. 70, pp. 135-146, 2007.
[15] Legesse F. B., Chernavskaia O., Heuke S., Bocklitz T., Meyer T., Popp J., Heintzmann R., "Seamless stitching of tile scan microscope images", Journal of Microscopy, vol. 258, pp. 223-232, 2015.
[16] Han S., Yang J., Wan H., "An automated wide-view imaging system of pathological tissue under optical microscopy", in Biomedical Image and Signal Processing (ICBISP), 2017, pp. 1-6.
[17] Forster B., Van De Ville D., Berent J., Sage D., Unser M., "Extended Depth-of-Focus for Multichannel Microscopy Images: A Complex Wavelet Approach", in International Symposium on Biomedical Imaging: Nano to Macro, 2004, pp. 660-663.
[18] Choi H., Cheng S., Wu Q., Castleman K. R., Bovik A. C., "Extended depth-of-field using adjacent plane deblurring and MPP wavelet fusion for microscope images", in 3rd IEEE International Symposium on Biomedical Imaging: Nano to Macro, 2006, pp. 774-777.
[19] Tessens L., Ledda A., Pizurica A., Philips W., "Extending the Depth of Field in Microscopy Through Curvelet-Based Frequency-Adaptive Image Fusion", in International Conference on Acoustics, Speech and Signal Processing - ICASSP '07, 2007, pp. 861-864.
[20] Doğan H., Baykal E., Ekinci M., Ercin M. E., Ersöz Ş., "Optimal focusing with extended depth of focus in microscopic systems", in 25th Signal Processing and Communications Applications Conference (SIU), Antalya, 2017, pp. 1-4.
[21] Li S., Kang X., Fang L., Hu J., Yin H., "Pixel-level image fusion: A survey of the state of the art", Information Fusion, vol. 33, pp. 100-112, 2017.
[22] Sahu A., Bhateja V., Krishn A., Himanshi, "Medical image fusion with laplacian pyramids", In 2014 International Conference on Medical Imaging, m-Health and Emerging Communication Systems (MedCom), 2014, pp. 448-453.
[23] Petrovic V.S., Xydeas C.S., "Gradient-based multiresolution image fusion", IEEE Transactions on Image Processing, vol. 13, pp. 228-237, 2004.
[24] Denipote J.G., Paiva M.S.V., "A fourier transform-based approach to fusion high spatial resolution remote sensing images", In 2008 Sixth Indian Conference on Computer Vision, Graphics Image Processing, 2008, pp. 179-186.
[25] Naidu V., "Discrete cosine transform-based image fusion", Defence Science Journal, vol. 60, pp. 48-54, 2010.
[26] Pajares G., de la Cruz J.M., "A wavelet-based image fusion tutorial", Pattern Recognition, vol. 37, pp. 1855-1872, 2004.
[27] Chai Y., Li H., Zhang X., "Multifocus image fusion based on features contrast of multiscale products in nonsubsampled contourlet transform domain", Optik – International Journal for Light and Electron Optics, vol. 123, pp. 569-581, 2012.
[28] Nejati M., Samavi S., Shirani S., "Multi-focus image fusion using dictionary-based sparse representation", Information Fusion, vol. 25, pp. 72-84, 2015.

[29] Xia X., Yao Y., Liang J., Fang S., Yang Z., Cui D., "Evaluation of focus measures for the autofocus of line scan cameras", Optik - International Journal for Light and Electron Optics, vol. 127, pp. 7762-7775, 2016.
[30] Krotov E.P.,"Active computer vision by cooperative focus and stereo", New York: Springer-Verlag, 1989.
[31] Kailath T., "The Divergence and Bhattacharyya Distance Measures in Signal Selection", IEEE Transactions on Communication Technology, Vol. 15, pp. 52-60, 1967.
[32] Lowe D.G., "Object recognition from local scale-invariant features", in International Conference on Computer Vision, 1999, pp. 1150-1157.
[33] Fischler M. A., Bolles R. C., "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography", Comm. of the ACM, vol. 24, pp. 381-395, 1981.
[34] Doğan H., Baykal E., Ekinci M., Ercin M. E., Ersöz Ş., "Determination of optimum auto focusing function for cytopathological assessment processes",in 2017 Medical Technologies National Congress (TIPTEKNO), Trabzon, Turkey, 2017, pp. 1-4.
[35] Crete-Roffet F., Dolmiere T., Ladret P., Nicolas M.,"TheBlur Effect: Perceptionand Estimation with a New No-Reference Perceptual Blur Metric", In SPIE ElectronicImaging Symposium Conf. Human Vision and Electronic Imaging, 2007, pp. 6492-16.

Kutulama Problemi için Geliştirilmiş Karınca Aslanı Optimizasyonu Algoritması

Improved Antlion Optimization Algorithm for Bin Packing Problem

Haydar KILIÇ
Bilecik Şeyh Edebali Üniversitesi
Mühendislik Fakültesi
Bilgisayar Mühendisliği Bölümü
Bilecik, TÜRKİYE
haydar.kilic@bilecik.edu.tr

Uğur YÜZGEÇ
Bilecik Şeyh Edebali Üniversitesi
Mühendislik Fakültesi
Bilgisayar Mühendisliği Bölümü
Bilecik, TÜRKİYE
ugur.yuzgec@bilecik.edu.tr

Öz

Bu çalışmada kutulama problemi içinbir geliştirilmiş karınca aslanı optimizasyonalgoritması (GKAO) önerilmiştir. Karınca aslanı optimizasyonalgoritması (KAO) temel olarak karınca aslanlarınınavlanma stratejilerini taklit eden bir meta-sezgisel optimizasyon algoritmasıdır. KAO algoritmasınınen büyük handikaplarından birisi uzun çalışma süresidir. KAO yapısında yer alan rastgele karınca yürüyüşü modelive seçim yöntemindeyapılan iyileştirmelerle ortaya çıkarılan GKAO bu handikabı ortadan kaldırmıştır. Önerilen GKAO algoritması kutulama problemi olarak adlandırılan optimizasyon problemine uyarlanarak test edilmiştir. Önerilen algoritma parçacık sürüsü optimizasyon algoritması (PSO), ateş böceği algoritması (FA), istilacı yabani ot optimizasyon algoritması (IWO) ve karınca aslanı optimizasyon algoritması(KAO) ile karşılaştırılmıştır. Sonuçlar önerilen GKAO algoritma performansınınkullanılan meta-sezgisel algoritma performanslarından daha başarılı olduğunu göstermiştir.

***Anahtar Sözcükler*— Karınca Aslanı Algoritması, Optimizasyon, Kutulama Problemi**

Abstract

In this study, an improved antlion optimization algorithm (IALO) was proposed for bin packing problem. Antlion optimization algorithm is a meta-heuristic optimization algorithm that basically imitates the hunting mechanism of antlions. The biggest disadvantage of antlion algorithm is its long running time. By the improvements on the ant random walking model and selection method in ALO algorithm, IALO algorithm eliminated this deficiency. The proposed IALO algorithm was tested by adapting to the optimization problem known as bin packing problem. The proposed IALO algorithm was compared with particle swarm optimization algorithm (PSO), firefly algorithm (FA), invasive weed optimization algorithm (IWO) and antlion optimization algorithm (ALO). The results show that the performance of IALO algorithm is more successful than the performances of used meta-heuristic algorithms.

***Keywords*— Antlion Algorithm, Optimization, Bin Packing Problem**

Gönderme ve Kabul tarihi: 21.04.2018-26.10.2018
Makale türü: Araştırma

1. Giriş

Günümüzün mühendislik bilimlerinde, meta-sezgisel algoritmalar, optimizasyon problemlerini çözmede büyük avantajlara sahiptir. Son yıllarda farklı optimizasyon problemlerinin çözümünde pek çok meta-sezgisel algoritmaların geliştirildiği görülmektedir. Bu algoritmalar evrimsel tabanlı, fiziksel tabanlı, sürü zekasına dayanan ve biyolojik ilhamlı algoritmalar olarak sınıflandırılabilirler [1-3].

Evrimsel tabanlı meta-sezgisel algoritmalardan en bilinenleri, genetik algoritma (GA) [4][5] vefarksal gelişim (DE) algoritmasıdır [6][7]. Bu algoritma sınıfında, rastgele bir populasyon ile çözüme başlanır ve bu populasyon çarpazlama ve mutasyon gibi çeşitli evrimsel mekanizmalarla güncellenir. Benzetilmiş tavlama algoritması (SA) [8], tabu arama algoritması (TS) [9], harmoni arama algoritması (HSA) [10][11] ve kurbağa sıçraması algoritması (SFLA) [12][13] fiziksel tabanlı meta-sezgisel algoritmaların en popüler olanlarındandır. Sürü zekasına dayanan algoritmalar kuş sürüleri, karınca kolonileri, balık sürüleri gibi kollektif zekayı taklit eden meta-sezgisel algoritmalar

grubundandır [1]. Bu algoritmalardan bazıları parçacık sürüsü optimizasyon algoritması (PSO) [14][15], yapay arı kolonisi algoritması (ABC) [16][17], karınca koloni algoritması (ACO) [18][19] olarak sayılabilir. Yapay bağışıklı algoritması (AI) [20][21] ve bakteriyel yiyecek arama algoritması [22][23] biyolojik ilhamlı meta-sezgisel algoritmalara önemli örneklerdendir.

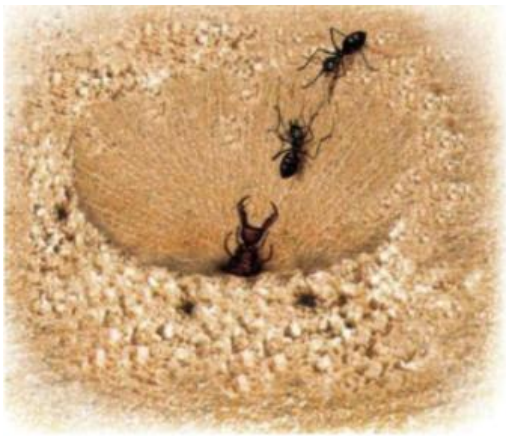
Bu çalışmada 2015 yılında Mirjalili [24] tarafından sunulan karınca aslanı optimizasyon (KAO) algoritması ele alınmıştır. Bu algoritma, karınca aslanı larvalarının kendilerine özgü avlanma tekniklerinden esinlenerek geliştirilmiş bir meta-sezgisel optimizasyon algoritmasıdır. Literatürde kontrolcü tasarımı [25][26], yük sevkiyat problemi [27][28], rota planlaması [29], esnek süreç planlaması [30], üretim çizelgeleme [31], optimizer tabanlı regülator [32], optimal topluluk tespiti [33], optimal filtre tasarımı [34] ve güç sistemleri optimizasyon problemleri [35] gibi mühendislik alanlarında KAO algoritmasına ait uygulamalara rastlanmaktadır.

Kutulama problemi lojistik ve üretim gibi alanlarda karşımıza çıkan bir optimizasyon problemidir. Bu problem bir kutunun içerisine en az boş alan kalacak şekilde eşyalarıyerleştirmeyi hedeflemektedir. Kutulama problemi boyutlarına göre bir, iki ve üç boyutlu kutulama problemleri olarak üçe ayrılmaktadır. Kamyon yükleme, konteynır yükleme ve şerit paketleme problemleri bu problemin özel halleridir [36][37].

Bu çalışma kapsamında karınca aslanı optimizasyon algoritması geliştirilerek, GKAO algoritması kutulama problemine uyarlanmıştır. Önerilen GKAO algoritmasının performansına test etmek için parçacık sürüsü optimizasyon algoritması (PSO), ateş böceği algoritması (FA), istilacı yabani ot optimizasyon algoritması (IWO) ve karınca aslanı optimizasyon algoritması (KAO) kullanılmıştır. Elde edilen sonuçlar önerilen GKAO algoritmasının diğer meta-sezgisel algoritmalara alternatif olabileceğini göstermektedir.

3. Karınca Aslanı Algoritması

Myrmeleontidae ailesinden olan karınca aslanları larva evresindeki son derece ilginç beslenme davranışlarından ismini alan yırtıcı bir böcek türüdür. Karınca aslanları karıncaların bulunduğu bölgelere tuzaklarını dairesel bir yol çizerek bir koni şeklinde oluştururlar ve bu tuzağın dibine yani koninin sivri ucuna kendilerini gömerek tuzağa düşecek karıncaları beklerler. Karıncalar tuzağa girdiğinde tuzaktan çıkmasını engellemek ve tuzağın dibine kaydırmak amacıyla karınca aslanları kum fırlatmaya başlarlar. Sonunda tuzağın dibine kaydırdıkları karıncaları büyük çeneleri ile yutarlar. Bu şekilde gelişen her bir avlanma işinden sonra karınca aslanlar tuzaklarını yeni bir av için hazır hale getirirler. Bu avlanma mekanizması Şekil 1'de gösterilmektedir.



Şekil 1. Karınca aslanı avlanma stratejisi.

Bu ilginç avlanma mekanizmasına ait matematiksel model rastgele yürüyüşlerle başlar:

$$X(t) = \begin{bmatrix} 0 \\ cumsum(2r(t_1) - 1) \\ cumsum(2r(t_2) - 1) \\ \vdots \\ cumsum(2r(t_n) - 1) \end{bmatrix} \quad (1)$$

Burada n maksimum iterasyon sayısı, $t$rastgele yürüyüş adımları, *cumsum*kümülatif toplam ve aşağıda tanımı verilen$r(t)$ bir stokastik fonksiyondur:

$$r(t) = \begin{cases} 1, & if rand > 0.5 \\ 0, & if\ rand \leq 0.5 \end{cases} \quad (2)$$

Rastgele yürüyüşe başlayan karıncaları arama uzayında tutmak için aşağıdaki formülle bu yürüyüşleri normalize etmek gerekmektedir:

$$X_i^t = (X_i^t - a_i)(d_i^t - c_i^t)(b_i - a_i)^{-1} + c_i^t \quad (3)$$

Buradaki i değişken sayısını, t iterasyon sayısını, a minimum rastgele yürüyüşünü, b maksimum rastgele yürüyüşünü, c ve d her bir iterasyonda güncellenen karınca aslanı pozisyonlarının sırasıyla minimum ve maksimum değerlerini göstermektedir.

Karıncaların yürüyüşleri karınca aslanlarından doğal olarak etkilenmektedir. Karınca tuzağa girdiğinde,

karınca aslanı onları tuzağın dibine çekmek için kum fırlatmaya başlar. Bu işleme ait matematik modellemesi olmak üzere aşağıdaki gibidir:

$$c_i^t = Antlion_i^t + c^t \quad (4)$$

$$d_i^t = Antlion_i^t + d^t \quad (5)$$

$$c^t = c^t . I^{-1} \quad (6)$$

$$d^t = d^t . I^{-1} \quad (7)$$

Burada *I*kaydırma oranını göstermektedir ve optimizasyon sırasında belirli oranlarda arttırılır. Bu mekanizmanın ayrıntıları Mirjalili'nin çalışmasında [24] bulunabilir. Eşitlik (3) yardımıyla karıncalar rulet tekerleği ile seçilen karınca aslanı ve elit karınca aslanı etrafında dolaşırlar. Bu şekilde karıncaların yeni pozisyonları aşağıdaki gibi güncellenir:

$$Ant_i^t = 0.5(R_A^t + R_E^t) \quad (8)$$

Burada $Ant_i^t t.$ iterasyondaki *i.* karıncayı ifade etmektedir. Karınca aslanı tuzağın dibine kaydırdığı karıncaları yediğinde kendi pozisyonunu Eşitlik (9)'a göre günceller:

$$if\ f(Ant_i^t) < f(Antlion_i^t), Antlion_i^t = Ant_i^t \quad (9)$$

Burada $Antlion_i^t t.$ iterasyondaki *i.* karınca aslanını, $Ant_i^t t.$ iterasyondaki *i.* karıncayı ifade eder.

4. Geliştirilmiş Karınca Aslanı Optimizasyon Algoritması (GKAO)

Karınca aslanı optimizasyon algoritması (KAO) Mirjalili [24] çalışmasında farklı özelliklere sahip optimizasyon test fonksiyonları için başarılı sonuçlar vermesine rağmen literatürde bu algoritma ile ilgili herhangi bir çalışma süresi analizine rastlanamamıştır. KAO algoritmasının en büyük dezavantajı çalışma süresinin uzun olmasıdır. Bunun nedeni karınca yürüyüş modeli için kullanılan rastgele yürüyüş modeli uzunluğudur. Bu nedenle geliştirilen KAO algoritmasında öncelikleriastgele yürüyüşlerde güncelleme yapılarak, var olan algoritmanın koşma süresi bakımından iyileştirilmesi sağlanmıştır. Buradarastgele yürüyüş modelinde maksimum iterasyon sayısı yerine bu sayının yüzde yirmisi alınarak daha kısa rastgele yürüyüşlerle daha optimal bir sonuç elde edilmeye çalışılmıştır.

$$X(t) = [0, \cdots, cumsum(2r(t_n - 1))], \quad n: [1, Max\_iter/5] \quad (10)$$

Meta-sezgisel optimizasyon algoritmalarında bir sonraki jenerasyon için birey seçiminde bir çok yöntem kullanılmaktadır. Bu yöntemlerden birisi de rulet tekerleği yöntemidir. Mirjalili [24] çalışmasında rastgele yürüyüş modelinde karıncaların seçilen bir karınca aslanı etrafında yürümesini formüle etmiş ve bu seçim işlemini de rulet tekerleği yöntemi ile yapmıştır. Ancak kullanılan optimizasyon problemine ait arama uzayı negatif değerler içeriyorsa, bu yöntem seçilen karınca aslanını sürekli ilk indeksten almaktadır. Bu hata rulet tekerleğine gönderilen değerin mutlak değeri alınarak Eşitlik (11)'deki gibi çözülmüştür.

$$\frac{|f(Antlion_i^{-1})|}{\sum_{j=1}^{n}|f(Antlion_j^{-1})|}, i = 1,2, \cdots, n \quad (11)$$

KAO algoritmasında, karıncalar *I* kaydırma oranında kaydırılarak karınca aslanına yem olmaktaydılar. Daha sonra karınca aslanıpozisyonu güncellenmekteydi. Önerilen GKAO yapısında bu güncelleme işlemi rastgele değişen bir parametreye bağlanarak yeni bir model oluşturuldu. Bu model karıncaların tuzakiçerisindeki durumlarına göre karınca aslanının pozisyonunu güncellemektedir.

$$\left.\begin{aligned} c_i^t &= Antlion_i^t + c^t \\ d_i^t &= Antlion_i^t + d^t \end{aligned}\right\} if\ 0.75 < opt < 1 \quad (12)$$

$$\left.\begin{aligned} c_i^t &= Antlion_i^t - c^t \\ d_i^t &= Antlion_i^t - d^t \end{aligned}\right\} if\ 0.5 < opt < 0.75 \quad (13)$$

$$\left.\begin{aligned} c_i^t &= -Antlion_i^t + c^t \\ d_i^t &= -Antlion_i^t + d^t \end{aligned}\right\} if\ 0.25 < opt < 0.5 \quad (14)$$

$$\left.\begin{aligned} c_i^t &= -Antlion_i^t - c^t \\ d_i^t &= -Antlion_i^t - d^t \end{aligned}\right\} if\ opt < 0.25 \quad (15)$$

Elit karınca aslanı ve rulet tekerleği yöntemi ile seçilen karınca aslanı etrafında yürüyen karıncaların normalize edilmiş yürüyüş modelleri karıncaların güncel pozisyonlarının hesaplanmasında aşağıdaki gibi kullanılmaktadır.

$$Ant_i^t = \frac{R_A^{r(t_n)} + R_E^{r(t_n)}}{2}, n = 1,2, \cdots, \frac{Max_{iter}}{5} \quad (16)$$

Burada $r(t_n)[0\ t_n]$aralığında değişen rastgele sayıyı göstermektedir.Karıncaların güncelleme sonrasında arama uzayı sınırları dışına çıkmasını engellemek için rastgele bir şekilde tekrar arama uzayında bir pozisyon belirlenir. Eşitlik (17)'de bu mekanizma verilmektedir.

$$Ant_i^t = b_{low} + rand \times (b_{up} - b_{low}) \quad if\ (Ant_i^t > b_{up})\ or\ (Ant_i^t < b_{low}) \quad (17)$$

Burada *rand*0-1 arasında rastgele bir sayıyı, b_{low}alt sınırı, b_{up}üst sınırı göstermektedir. Orijinal KAO algoritmasında, her iterasyon sonunda karıncalar ve karınca aslanlarıbirleştirilerek, maliyet değerlerine göre sıralanmakta vepopülasyon boyu kadar olan ilk kısımalınarak karınca aslanı olarak kabul edilmektedir. Önerilen algoritmada ise birleştirme ve sıralama işlemleri yerine iterasyon sonunda i. karınca ile i. karınca aslanı maliyetleri karşılaştırılarak, daha iyi olan bir sonraki iterasyondaki karınca aslanı pozisyonu olarak alınmıştır. Önerilen GKAO algoritmasına ait sözde kod yapısı Algoritma 1'de verilmektedir.

Algoritma1.Kutulama Problemi için GKAO algoritmasının sözde kodu.

```
Karınca aslanları başlangıç pozisyonu belirle
Karınca aslanları maliyet değerleri hesapla
Elit karınca aslanı ve pozisyonu sakla
  while (iterasyon < maksimum iterasyon)
for (her bir karınca aslanı)
       Karınca aslanı seçimi
       Rastgele yürüyen karıncaların bir tuzağa
       doğru kayması (Eşitlik 12-15)
       Elit karınca aslanı etrafında karıncanın
       rastgele yürüyüş modelinin çıkarılması
       Seçilen karınca aslanı etrafında
       karıncanın rastgele yürüyüş modelinin
       çıkarılması
       Rastgele yürüyüş modelinin normalize
       edilmesi
       Karıncanın pozisyonun belirleme
       (Eşitlik 8)
       if Karınca arama uzayı dışında ise,
          Rastgele arama uzayı içerisine at
       end if
end for
  Karınca maliyet değerleri hesapla
  for (her bir karınca aslanı)
     if karıncanın maliyeti daha iyi ise,
       karınca aslanı karıncayı yer ve
       pozisyonunu karıncanınki ile günceller.
     end if
  end for
  Elit karınca aslanı güncelle
end while
```

5. Kutulama Problemi

Kutulama problemi üzerinde çok çalışılan klasik ve zor bir optimizasyon problemidir. Bu problemde temel olarak bir kutunun içerisinde en az boş alan bırakılarak, eşyaların en optimum şekilde nasıl yerleştireceğine çözüm bulmaya çalışılmaktadır.Kutulama problemi boyutuna ve paketlenen nesneye göre sınıflandırılabilmektedir [36][37].

s_p boyutunda P adet ürün ve her birinin kapasitesi C olan $B = \{1, \cdots, P\}$ aday kutular kümesi için kutulama optimizasyon probleminin matematiksel tanımı aşağıda verilmiştir.

$$min \sum_{b \in B} y_b \tag{18}$$

$$s.t. \sum_{b \in B} x_{pb} = 1,\ p \in P$$

$$min \sum_{p \in P} s_p x_{pb} \leq C y_b, b \in B$$

$$x_{pb} \in \{0,1\}, p \in P, b \in B$$

$$y_b \in \{0,1\}, b \in B$$

Burada kullanılan kısıtlardaher bir ürünün tam olarak bir kutuyaatanmasısağlanmaktadır.$y_b$vex_{pb} ikili değişkenlerdir. Eğer $x_{pb} = 1$ ise, p ürününün b kutusuna atandığını göstermektedir. Aynı şekilde eğer $y_b = 1$ ise, b kutusunun kullanıldığı anlaşılmaktadır.

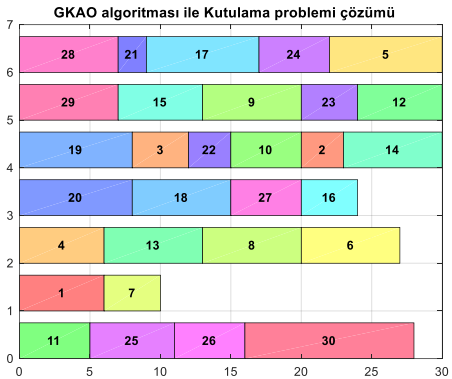
6. Deneysel Sonuçlar

Bu çalışma kapsamında önerilen GKAO algoritması kutulama problemi için uyarlanarak, bu problemin çözümü için elde edilen algoritma sonuçları literatürden alınan parçacık sürüsü optimizasyon algoritması (PSO), ateş böceği algoritması (FA), istilacı yabani ot optimizasyon algoritması (IWO) ve karınca aslanı optimizasyon algoritması (KAO) ile karşılaştırılmıştır. Kutulama problemi örneği www.yarpiz.com web sitesinden alınmıştır [38]. Bu problemde, farklı ebatlarda 30 ürünün maksimum kapasitesi 30 olan kutulardan en az kaç tanesine minimum maliyetle yerleştirilebileceği bulunmaya çalışılmıştır. Çözümlerde kutu kapasitesini aşma durumlarında maliyet fonksiyonuna ihlal cezası ($\alpha \cdot \overline{Vıol}$) ilave edilmiştir. Burada $\overline{Vıol}$ ortalama ihlali göstermektedir ve maliyet hesaplamalarında $\alpha = 60$ olarak kullanılmıştır. Bir boyutlu kutulama probleminde her bir ürün boyu $v[1 \times 30]$ isimli bir vektörde tutulmaktadır.

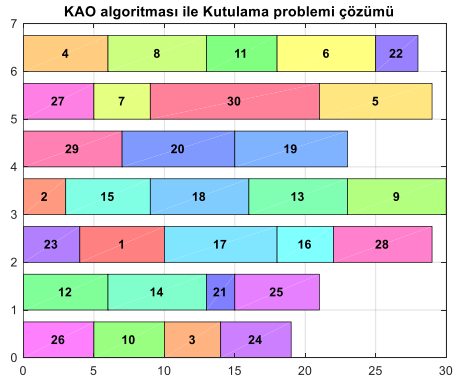
GKAO ve diğer meta-sezgisel algoritmalara ait kodlar, Intel(R) Core(TM) i5-3230M CPU@2.60GHz RAM/8 bir bilgisayarda koşturulmuştur. Her bir algoritma için popülasyon boyu 20 ve maksimum iterasyon sayısı ise 1000 olarak kullanılmıştır. Karşılaştırma için bu çalışmada kullanılan meta-sezgisel algoritmaların parametreleri Tablo 1'de sunulmaktadır. Şekil 2-6'da bu çalışmada önerilen GKAO ve diğer meta-sezgisel algoritmalar ile bulunan kutulama problemi çözümleri gösterilmektedir.

Tablo 1.Kutulama probleminde kullanılan metasezgisel algoritmaların parametre değerleri.

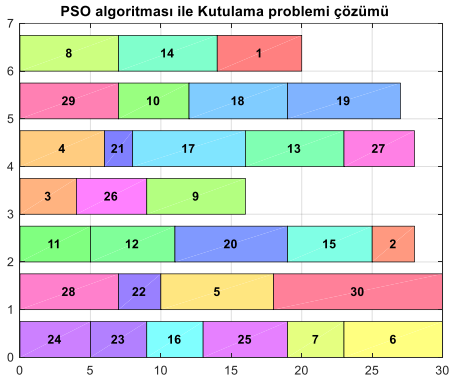
Algoritma	Parametreler
PSO	Ataletağırlığı : 1.0 Ataletağırlığısönümleme oranı : 0.99 Bireysel öğrenme katsayısı : 1.5 Global öğrenme katsayısı: 2.0
FA	Işık emme katsayısı: 1.0 Başlangıç çekim katsayısı: 2.0 Mutasyon katsayısı : 0.2 Mutasyon sönümleme oranı: 0.98
IWO	Minimum tohum sayısı : 0 Maksimum tohum sayısı : 5.0 Varyans azaltma bileşeni : 2.0 Standart sapmabaşlangıç değeri : 1.0 Standart sapma sondeğeri : 0.001
KAO	Karınca aslanı sayısı : 20
GKAO	Karınca aslanı sayısı : 20



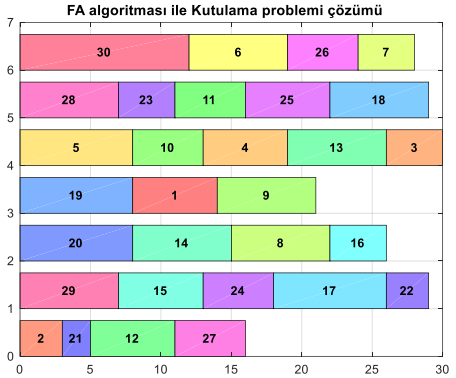
Şekil 2. GKAO algoritması ile bulunan kutulama problemi çözümü.



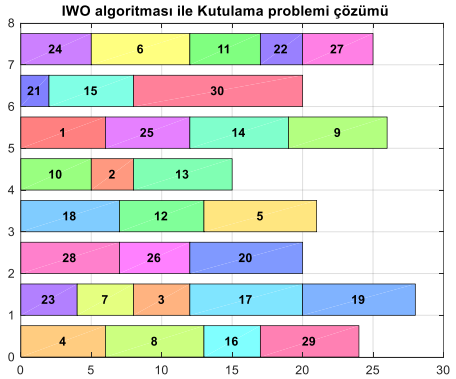
Şekil 3. KAO algoritması ile bulunan kutulama problemi çözümü.



Şekil 4. PSO algoritması ile bulunan kutulama problemi çözümü.



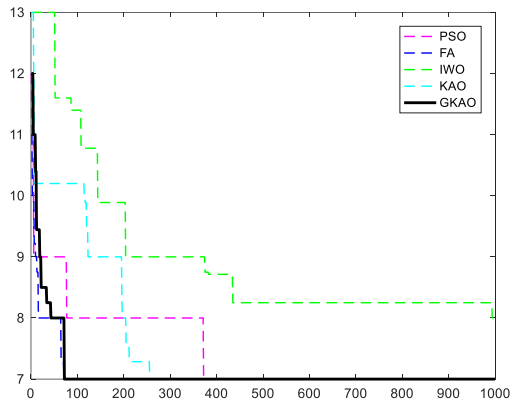
Şekil 5. FA algoritması ile bulunan kutulama problemi çözümü.



Şekil 6.IWO algoritması ile bulunan kutulama problemi çözümü.

Şekil 6'da gösterilen IWO algoritması ile bulunan kutulama problemi çözümü haricinde önerilen GKAO ve diğer algoritmaların çözümlerinde 1000 iterasyon

sonunda maliyet değeri 7 kutu olarak bulunmuştur. Algoritma çözümleri arasındaki fark ürünlerin eşleştirildiği kutu farklılıkları olarak öne çıkmaktadır. Şekil 7'de önerilen GKAO ve diğer meta-sezgisel algoritmalarının kutulama optimizasyon problemi çözümünde elde edilen yakınsama eğrileri bir arada gösterilmiştir. Şekilden de görüleceği gibi, GKAO ve FA algoritmaları performans olarak yaklaşık 80. iterasyonda en iyi maliyet değerini yakalamışlardır.



Şekil 7.Kutulama problemi için yakınsama eğrileri (PSO, FA, IWO, KAO, GKAO)

7. Sonuçlar ve Tartışma

Bu çalışmada meta-sezgisel algoritmalardan birisi olan ve karıca aslanı avlanma stratejisine dayanan karınca aslanı optimizasyon (KAO) algoritması ele alınmıştır. İlk olarak mevcut orijinal KAO algoritması geliştirilerek, algoritma hızlandırılmış ve önerilen GKAO algoritması zor optimizasyon problemlerinden birisi olan kutulama problemine uyarlanmıştır. GKAO algoritmasının kutulama problemi çözümü performansını değerlendirmek için literatürden PSO, FA, IWO ve orijinal KAO algoritması alınarak, bu algoritmaların karşılaştırılması yapılmıştır. Elde edilen sonuçlardan GKAO algoritmasının FA algoritması ile birlikte en iyi yakınsama eğrisine sahip olduğu görülmektedir. İleriki çalışmalarda GKAO algoritmasına farklı mekanizmalar eklemek suretiyle algoritmanın performansının daha da geliştirilmesi ve GKAO algoritmasının farklı zor optimizasyon problemlerine uygulanması hedeflenmektedir.

Kaynakça

[1] S. J. Nanda and G. Panda, "A survey on nature inspired metaheuristic algorithms for partitional clustering," Swarm and Evolutionary Computation, vol. 16. pp. 1–18, 2014.

[2] Z. Beheshti and S. M. H. Shamsuddin, "A review of population-based meta-heuristic algorithm," International Journal of Advances in Soft Computing and its Applications, vol. 5, no. 1, pp. 1-35, 2013.

[3] M. H. N. Tayarani, X. Yao, and H. Xu, "Meta-Heuristic Algorithms in Car Engine Design: A Literature Survey," IEEE Trans. Evol. Comput., vol. 19, no. 5, pp. 609–629, 2015.

[4] J. H. Holland, "Adaptation in Natural and Artificial Systems," Ann Arbor MI Univ. Michigan Press, vol. Ann Arbor, p. 183, 1975.

[5] D. E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, vol. Addison-Wesley, 1989.

[6] R. Storn and K. Price, "Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces," J. Glob. Optimization, vol. 11, no. 4, pp. 341–359, 1997.

[7] K. V Price, R. M. Storn, and J. A. Lampinen, Differential Evolution: A Practical Approach to Global Optimization, vol. 28. 2005.

[8] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by Simulated Annealing," Science (80-.)., vol. 220, no. 4598, pp. 671–680, 1983.

[9] F. Glover, "Future paths for integer programming and links to artificial intelligence," Comput. Oper. Res., vol. 13, no. 5, pp. 533–549, 1986.

[10] Zong Woo Geem, Joong Hoon Kim, and G. V. Loganathan, "A New Heuristic Optimization Algorithm: Harmony Search," Simulation, vol. 76, no. 2, pp. 60–68, 2001.

[11] X. S. Yang, "Harmony search as a metaheuristic algorithm," Studies in Computational Intelligence, vol. 191. pp. 1–14, 2009.

[12] M. Eusuff, K. Lansey, and F. Pasha, "Shuffled frog-leaping algorithm: A memetic meta-heuristic for discrete optimization," Eng. Optim., vol. 38, no. 2, pp. 129–154, 2006.

[13] K. K. Bhattacharjee and S. P. Sarmah, "Shuffled frog leaping algorithm and its application to 0/1 knapsack problem," Appl. Soft Comput. J., vol. 19, pp. 252–263, 2014.

[14] J. Kennedy and R. Eberhart, "Particle swarm optimization," Neural Networks, 1995. Proceedings., IEEE Int. Conf., vol. 4, pp. 1942–1948 vol.4, 1995.

[15] R.Eberhart, J.Kennedy, "A new optimizer using particle swarm theory", in:Sixth International Symposium on Micro Machine and Human Science, MHS, 1995,pp.39–43.

[16] D. Karaboga and C. Ozturk, "A novel clustering approach: Artificial Bee Colony (ABC) algorithm," Appl. Soft Comput. J., vol. 11, no. 1, pp. 652–657, 2011.

[17] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Technical Report-TR06, Erciyes University, Engineering Faculty, Computer Engineering Department, Tech. Rep., 2005.

[18] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: Optimization by a colony of cooperating agents," IEEE

Trans. Syst. Man, Cybern. Part B Cybern., vol. 26, no. 1, pp. 29–41, 1996.
[19] M. Dorigo, M. Birattari, and T. Stutzle, “Ant colony optimization,” IEEE Comput. Intell. Mag., vol. 1, no. 4, pp. 28–39, 2006.
[20] D. Dasgupta, “Artificial Immune Systems and their Applications”, Springer-Verlag, 1999, ISBN3540643907.
[21] L.N. de Charsto, J. Timmis, “An Introduction to Artificial Immune Systems: A New Computational Intelligence Paradigm”, Springer-Verlag, 2002.
[22] K. Passino, “Biomimicry of bacterial foraging for distributed optimization and control”, IEEE Control Syst. Mag., vol. 22, no.3, pp.52–67, 2002.
[23] S. Mishra, “A hybrid least square-fuzzy bacterial foraging strategy for harmonic estimation”, IEEE Trans. Evol. Comput., vol. 9, no.1, pp.61–73, 2005.
[24] S. Mirjalili, “The ant lion optimizer,” Adv. Eng. Softw., vol. 83, pp. 80–98, 2015.
[25] More Raju, Lalit Chandra Saikia, Nidul Sinha, "Automatic generation control of a multi-area system using ant lion optimizer algorithm based PID plus second order derivative controller", International Journal of Electrical Power and Energy Systems, Volume 80, September 2016, Pages 52-63, ISSN 0142-0615
[26] Satheeshkumar, R., Shivakumar, R. "Ant Lion Optimization Approach for Load Frequency Control of Multi-Area Interconnected Power Systems", Circuits and Systems, 7(09), 2357,2016.
[27] Kamboj, V. K., Bhadoria, A., Bath, S. K.,"Solution of non-convex economic load dispatch problem for small-scale power systems using ant lion optimizer", Neural Computing and Applications, 1-12, 2016.
[28] Nischal, M. M., Mehta, S.,"Optimal load dispatch using ant lion optimization", Int J Eng Res Appl, 5(8), 10-19, 2015.
[29] Yao, P., Wang, H., "Dynamic Adaptive Ant Lion Optimizer applied to route planning for unmanned aerial vehicle", Soft Computing, 1-14, 2016.
[30] Petrovic, M., Petronijevic, J., Mitic, M., Vukovic, N., Plemic, A., Miljkovic, Z., Babic, B., "The ant lion optimization algorithm for flexible process planning. JPE, 18(2), 65-68, 2015.
[31] N. Chopra and S. Mehta, "Multi-objective optimum generation scheduling using Ant Lion Optimization," 2015 Annual IEEE India Conference (INDICON), New Delhi, 2015, pp. 1-6.
[32] Gupta, E., Saxena, A., "Performance Evaluation of Antlion Optimizer Based Regulator in Automatic Generation Control of Interconnected Power System", Journal of Engineering, 2016.
[33] Babers, R., Ghali, N. I., Hassanien, A. E., Madbouly, N. M., "Optimal community detection approach based on Ant Lion Optimization", In Computer Engineering Conference (ICENCO), 2015, 11th International (pp. 284-289). IEEE.
[34] Nair, S. S., Rana, K. P. S., Kumar, V., Chawla, A., "Efficient Modeling of Linear Discrete Filters Using Ant Lion Optimizer", Circuits, Systems, and Signal Processing, 1-34,2016.
[35] Rebecca, N., Shin, M., MH, S., Zuriani, M., "Ant Lion Optimizer for Optimal Reactive Power Dispatch Solution", Journal of Electrical Systems, (3), 67-74,2015.
[36] Martinez-Sykora, A., Alvarez-Valdes, R., Bennell, J. A., Ruiz, R. and Tamarit, J. M., “Metaheuristics for the irregular bin packing problem with free rotations,” Eur. J. Oper. Res., vol. 258, no. 2, pp. 440–455, 2017.
[37] Christensen,H. I., Khan,A., Pokutta,S. and Tetali, P., “Approximation and online algorithms for multidimensional bin packing: A survey,” Computer Science Review, vol. 24. pp. 63–79, 2017.
[38] Yarpiz, Bin Packing Problem using GA, PSO, FA, and IWO.http://yarpiz.com/363/ypap105-bin-packing-problem, Accessed20April 2018.

Artırılmış Gerçekliğin Sanal Sınıf Ortamlarında Kullanılması Noktasında Öğrenci Görüşleri

Student Opinions on Using Augmented Reality in Virtual Classroom Environments

Arif KOYUN
Süleyman Demirel Üniversitesi
Mühendislik Fakültesi,
Bilgisayar Müh.Böl.,Isparta
arifkoyun@sdu.edu.tr

Handan BUDAK
Süleyman Demirel Üniversitesi
Mühendislik Fakültesi
Bilgisayar Müh.Böl YL.Öğr.,Isparta
handanbudak94@gmail.com

İbrahim Arda ÇANKAYA
Süleyman Demirel Üniversitesi
Mühendislik Fakültesi,
Bilgisayar Müh.Böl.,Isparta
ardacankaya@sdu.edu.tr

Öz

Teknolojide yaşanan gelişmelerle birlikte eğitim ortamında da güncel teknolojilerin kullanımına yönelik gereklilikler artmakta ve yeni teknolojiler kullanılmaya başlanmaktadır. Bu bağlamda gelişen teknolojilerden biri de Artırılmış Gerçeklik olmuştur. Artırılmış gerçeklik sanal nesnelerin gerçek dünya ile birleştirildiği ve birbirleriyle etkileşimde bulunduğu bir teknolojidir. Bu çalışmanın amacı, Artırılmış gerçekliğin Uzaktan Eğitim öğrencileri üzerinde öğrenme, ilgi, tutum ve davranışları açısından ne gibi faydaları olduğunu incelemekte ve bu araştırma sonucunda artırılmış gerçekliğin sanal sınıf ortamlarında kullanımının artırılması amaçlanmaktadır. Çalışma grubu, 2017-2018 öğretim yılında Süleyman Demirel Üniversitesi Uzaktan Eğitim Meslek Yüksekokulu'nda okumakta olan 40 Adet öğrenciden oluşmaktadır. Öncelikle çalışmada artırılmış gerçeklikle ilgili öğrencilere kısaca bilgi verilmiş ve artırılmış gerçeklik alanında var olan örnek bir uygulama gösterilmiştir.

Daha sonra öğrenciler üzerinde hem kişisel bilgileri hem de artırılmış gerçeklikle ilgili olmak üzere toplamda 12 adet sorudan oluşan bir anket düzenlenmiştir. Anket soruları 5'li Likert türünde oluşturulmuş ve katılım internet üzerinden sağlanmıştır. Elde edilen sonuçlaraSPSS 1.0.0.1012 programında Güvenilirlik Analizi, T Testi, Ki-Kare Testi uygulanarak sanal sınıf ortamlarında etkisi araştırılmış ve bulgular değerlendirilmiştir.

***Anahtar Sözcükler**:Artırılmış Gerçeklik, Sanal Sınıf Teknolojileri, Sanal Sınıf Ortamları, Eğitim Teknolojileri, Uzaktan Eğitim*

Abstract

Along with the developments in technology, requirements for using up-to-date technologies are also increasing in the educational environmentsand new technologies are being used. One of the developing technologies in this context is augmented reality. Augmented reality is a technology where virtual objects are combined with the real world and interact with each other. Because of its benefits to create highly interactive applications because of that, augmented reality based studies are often performed in the field of education. Considering the explanations so far, the purpose of this study is to examine the benefits of augmented reality in terms of learning, interest, attitudes and behaviors on Distance Education students. In the study, the effect of augmented reality on virtual classroom environments is examined and with the performed works, it is aimed to increase the awaraness of using augmented reality in virtual classroom environments. The study group consists of 40 students from Vocational School of Distance Education of Süleyman Demirel University, during2017-2018 academic year. Firstly, students

Gönderme ve kabul tarihi: 03.05.2018-26.10.2018
Makale türü: Araştırma

were briefly informed about the augmented reality in the study, and a sample application was presented in the field of augmented reality. Then a questionnaire consisting of 12 questions in total was prepared for the students regarding both personal information and the augmented reality. Questionnaires were created in the form of quinary Likert types and participation was provided over the internet. Reliability Analysis, T-Test, Chi-Square Test were applied to the results obtained in SPSS 1.0.0.1012 program and the effects were investigated in virtual classroom environments and the findings were evaluated.

Keywords:*Augmented Reality, Virtual Classroom Technologies, Virtual Classroom Environments, Educational Technologies, Distance Education*

1.Giriş

Günümüz çağdaş toplumlarının gelişmişlik düzeylerini, ortaya çıkardıkları bilim ve teknoloji belirlemektedir [1]. Hemen hemen insanoğlunun var olmasına dayanmakta olan teknoloji temellerini insanların mağarada yaptığı çizimler, yaptıkları süs eşyaları, çanak, çömlek gibi ilk insanların maddi kültürünü oluşturan ürünlerle atmıştır [2].

İnsanoğlunun 'Nasıl öğrenirim?' sorusuna farklı çözümler üretmeye başlamasıyla ortaya çıkan eğitim teknolojileri gün geçtikçe dış etkenlerle önemli değişimler geçirerek farklı durumlar kazanmaktadır. Dış etkenlerden sayılabilecek en önemli değişim ise teknoloji alanında yaşanan yeniliklerdir [3]. Teknoloji alanında gerçekleşen hızlı gelişmeler hayatın her alanında olduğu gibi öğrenme ortamlarında da yeniliklerin ortaya çıkmasını zorunlu kılmış ve bu teknolojilerin eğitimde ne denli etkili olduğu sorusunu gündeme getirmiştir [4].

Çağımızın ayırt edici başlıca özellikleri arasında büyük insan toplulukları, hareketlilik, bilimsellik, hızlı değişim ve teknolojinin ilerlemesi bulunmaktadır. Gelişen bu toplumlarda insanların her an değişen ve gelişen teknolojik düzene uyum sağlaması gerekmektedir. Bilgi çağı olarak adlandırılan, yaşadığımız bu yıllarda bilgisayar teknolojisi alanında özellikler bilgisayar ağları üzerinden yapılan iletişimde büyük adımlar atılmıştır. İletişim için geliştirilen donanım araçlarının görselliği de işin içine katılarak sanal adı verilen ortamlar oluşturulmaya çalışılmıştır. Sanal ortamlar öğrenme-öğretme açısından da büyük faydalar sağlamış ve sanal üniversiteler, sanal sınıf ortamları kurulmuştur[5]. Sanal eğitim ortamlarının kurulmasındaki en önemli nedenler ise özellikle yükseköğretimdeki kapasite sorunları ve çalışanların eğitim talebi olmuştur [1].

Eğitim üzerinde etkisi olup olmadığı tartışılan teknolojilerden birisi de son zamanlarda kullanımı gittikçe artan Artırılmış Gerçeklik (AG) teknolojisidir [4]. Artırılmış Gerçeklik her ne kadar sanal geçekliğin bir parçasıymış gibi görünse de karşılaştırıldığında sanal gerçeklik yaşadığımız hayata benzetilmesi ile zaten var olan gerçekliğin yerine geçer. Artırılmış Gerçeklikte ise sanal ve gerçeklik birlikte bulunmaktadır ve gerçek zamanla etkileşimdedir [6]. Başka bir deyişle Artırılmış Gerçeklik gerçek görüntü üzerine bilgisayar aracılığı ile resim, metin, ses gibi sanal nesnelerin eklendiği ve değiştirilmiş halinin görüntülendiği bir teknolojidir. Artırılmış Gerçeklik teknolojisi taşınabilir cihazların(mobil telefon, tablet) ortaya çıkması, yaygınlaşması ve sağladığı avantajlar sayesinde daha da önem kazanmış ve kullanım alanları gün geçtikçe daha da artmıştır. Yeni ortaya çıkan bir teknoloji olmamasına rağmen önceleri askeri tıp, turizm, mühendislik, reklamcılık, spor gibi alanlarda kullanılmış, öğrenme ve öğretme ortamlarındaki etkisi yeni yeni keşfedilmeye başlanmıştır [7].

Çalışmanın ikinci bölümünde Artırılmış Gerçeklik teknolojisinin eğitim ortamlarında kullanılmasının öneminden bahsedilmiştir. Bölüm 3'te Artırılmış Gerçeklik ile ilgili geçmiş çalışmalara yer verilmiştir. Bölüm 4'te çalışmanın yöntemi anlatılmış ve bölüm 5'te sonuçlar ve gelecek çalışmalardan bahsedilmiştir.

2.Artırılmış Gerçekliğin EğitimdeÖnemi

Araştırmacılar artırılmış gerçekliğin güçlü potansiyel etkileri sayesinde öğrenme-öğretme ortamlarını güçlendirme açısından birçok faydasının olduğunu savunmaktadırlar. Örneğin; işbirlikçi öğrenme yöntemini sağlamak(öğretmen-öğretmen, öğretmen-öğrenci), öğrencilerin görerek öğrenmelerin zor olduğu ve aynı zamanda maliyetli olan konularda(astronomi, fizik vb.) konuları kolay bir şekilde öğrenip kavramalarını sağlamak, onların düşünce ve hayal gücünü geliştirmek, öğrencilerin öğrenme hızlarını kendilerinin belirlemesini sağlamak [8].

3.Geçmiş Çalışmalar

AG her ne kadar dünya üzerinde çok yaygın bir teknoloji olsa da Türkiye'de henüz üzerinde çok fazla çalışma yapılmamıştır.

Küçük, Yılmaz ve Göktaş [4], İngilizce öğreniminde ortaokul öğrencilerinin başarı, tutum ve bilişsel yük düzeylerini inceleyen çalışmalarını Erzurum ilinde 5 farklı ortaokulda öğrenim görmekte olan 5. Sınıf düzeyinde toplam 122 öğrenci(56 kız, 66 erkek) üzerinde gerçekleştirmişlerdir. Çalışma İngilizce dersine yönelik artırılmış gerçeklik uygulamaları tasarlanarak uygulanmıştır. Çalışma sonucunda AG uygulamalarını kullanan öğrencilerin uygulamadan memnun kalarak kaygı düzeylerinin düşük olduğu, ileride bu uygulamaları kullanmaya istekli oldukları ve konuyu öğrenmede motivasyonlarının yüksek olduğu sonuçlarına varmışlardır.

Korucu, Gençtürk ve Sezer (**9**), çalışmalarında artırılmış gerçeklik uygulamalarının öğrencilerin başarı ve tutumlarına etkisini araştırmışlar ve öntest-sontest tekniğini kullandıkları bu çalışmalarını Konya'nın Çumra ilçesinde bulunan bir ortaokulda öğrenim görmekte olan 120 öğrenci üzerinde gerçekleştirmişlerdir. Çalışmada Artırılmış Gerçeklik tutum ölçeği puanlarını cinsiyet ve öğrenim gördükleri sınıf durumuna göre hesaplamışlar, sonuçların cinsiyete göre değil de öğrenim gördükleri sınıf durumlarına göre anlamlı bir farklılık gösterdiğini ortaya çıkarmışlardır.

Tülü ve Yılmaz [10], Unity sayesinde simule ettikleri görsel bir nesneyi gerekli kodlamaları yaptıktan sonra iPad üzerinde denemişlerdir. Uygulamalarında benzetimini yaptıkları bir nesnenin 3 boyutlu halini hedef resim üzerinde aslında yokken var gibi gösterebilmişler ve kamera ile nesneye yaklaşıp uzaklaşma işlemlerinde de sanki nesneye gerçekten yaklaşıp uzaklaşıyormuş hissini verebilmişlerdir. Çalışmaları sonucunda bu teknolojinin eğitim alanında kullanılmasının oldukça faydalı olacağı kanısına varmışlardır.

Köroğlu [11] çalışmasında incelemelerini artırılmış gerçeklik kavramı üzerinde gerçekleştirmiş ve kullanım alanları hakkında araştırma yapmıştır. Artırılmış Gerçekliğin geliştiricilerinden ve geliştirdikleri bazı uygulamalardan bahsetmiş, özellikle reklamcılık, iletişim gibi alanlar üzerinde durmuştur. Son olarak artırılmış gerçekliğin birey ve toplum üzerinde ne gibi etkileri olduğundan bahsetmiştir.

Artırılmış Gerçekliğin eğitici kart (flashcard)'larda kullanıldığı programlardan Letters Alive, anaokulu öncesi ve anaokulu öğrencilerinin okuma becerilerini geliştirmek için Alive Studios şirketi tarafından geliştirilmiştir. Programda 26 adet eğitici alfabe kartı bulunmaktadır. Bu program sayesinde öğrenciler çizgi film karakterleri ve hayvanlar gibi sanal nesnelerle etkileşime girebilmektedir. Bu uygulamanın en önemli avantajı ise özel gözlüklere gerek duymamasıdır [12].
FETCH! Lunch Rush uygulaması 6-8 yaş aralığında bulunan çocukların aritmetik problem çözme becerilerini geliştirmek, matematiksel problemleri görselleştirilmek [13] ve öğrenmeyi hızlandırmak için geliştirilmiş 3D bir oyundur [14].

Reitmary ve Schmalstieg [15] artırılmış gerçeklik teknolojisini kullanarak turistlerin gittikleri yerlerde bilgilendirilmesi amacıyla bir çalışma gerçekleştirmişlerdir. Turistler bu uygulama sayesinde gidecekleri konuma yönlendirilmekte, ilk kez gittikleri yerlerde rehbere ihtiyaç duymamakta ve ziyaret ettikleri yerde önceden belirlenmiş simgeler sayesinde ayrıntılı bilgilere ulaşabilmektedir. Bu sistemde kullanıcının kafasına yerleştirilmiş bir kasket ve bu kaskette de kamera ve bazı sensörler bulunmaktadır. Artırılmış gerçeklik kısmı için Schmalstieg tarafından geliştirilen Studierstube yazılımını kullanmışlar ve bu yazılım sayesinde gerçek görüntü üzerine resim, yazı, şekil ve 3 boyutlu nesneler eklemişlerdir.

4.Yöntem

Bu çalışmada öğrencilere Artırılmış Gerçeklik donanım ve yazılımları tanıtıldıktan sonra onlara hem farkındalık hem de başarımları ile ilgili 12 soruluk anket yapılmıştır. Anket, bünyesinde hem kişisel bilgileri hem de Artırılmış Gerçeklik ile ilgili soruları barındırmaktadır. Çalışma grubu Süleyman Demirel Üniversitesi Uzaktan Eğitim Meslek Yüksekokulu öğrencilerinden 40 kişilik bir gruptur. 40 kişiden oluşan bu çalışma grubunun 25 tanesi erkek, 15 tanesi kadın öğrenciden oluşmaktadır. Çalışma grubunda bulunan öğrencilerin cinsiyetlerine göre betimsel sonuçlar Çizelge 1'de verilmektedir.

Çizelge-1: Çalışma Grubunun Cinsiyete Göre Dağılımı

Cinsiyet	N	%
Erkek	25	62,5
Kadın	15	37,5
Toplam:	40	100,0

Araştırmaya katılan 40 öğrenciden 25'inin (% 62,5) erkek, 15'inin (% 37,5) kadın olduğu Çizelge 1'de gösterilmektedir.

Çalışma grubundaki öğrencilerin bölümlere göre dağılımı ise Çizelge 2'de verilmektedir.

Çizelge-2: Çalışma Grubunun Bölümlere Göre Dağılımı

Bölüm	N	%
Bilgisayar Programcılığı	20	50,0
Tıbbi Dokümantasyon ve Sekreterlik	20	50,0
Toplam:	40	100,0

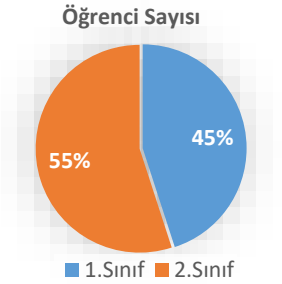
Araştırmaya katılan 40 öğrenciden 20'sinin (% 50,0) erkek, 15'inin (% 50,0) kadın olduğu Çizelge 2' de görülmektedir.

Araştırmada çalışma grubunda bulunan öğrencilerin öğrenim gördükleri sınıflara ait dağılımı Çizelge 3'de gösterilmektedir. Grafik 1'de sınıflara göre öğrenci sayıları gösterilmektedir.

Çizelge-3: Çalışma Grubunun Sınıflara Göre Dağılımı

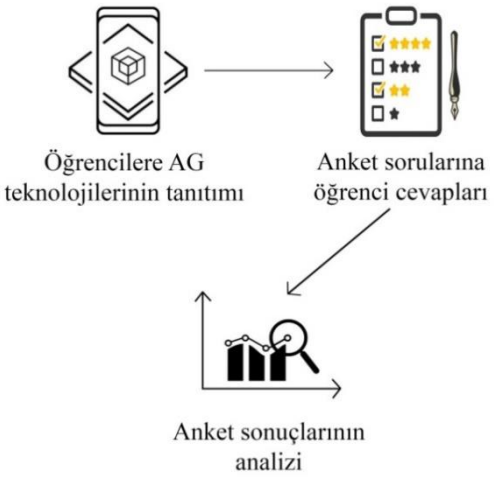
Sınıf	N	%
1.Sınıf	18	45,0
2.Sınıf	22	55,0
Toplam:	40	100,0

Grafik-1: Sınıflara Göre Öğrenci Sayıları



Çizelge 3'te görüldüğü gibi araştırmaya katılan öğrencilerin 18'i (%45) 1.sınıfta, 22'si (% 55) 2. sınıfta öğrenim görmektedir.

Anket, çalışma grubuna internet üzerinden yapılmış ve sonuçlar yine aynı şekilde internet ortamından elde edilmiştir. Yapılan işlemlerle ilgili Sistem Mimarisi Şekil 1'de gösterilmiştir.



Şekil-1: Sistem Mimarisi

4.1.Öğrencilere Artırılmış Gerçeklik Teknolojisinin Tanıtımı

Çalışmada artırılmış gerçekliğin sanal sınıf ortamlarına faydaları araştırılmaktadır. Bu yüzden çalışma grubu Uzaktan Eğitim Meslek Yüksekokulu'ndan seçilmiştir. Bu öğrencileri sınav zamanları dışında okul ortamında bulmak çok mümkün değildir. Bu yüzden öğrencilerin vize haftasında bir gün belirlenip 40 kişi bir sınıfta toplanmıştır. Bu çalışma grubuna artırılmış gerçeklik teknolojileri ile ilgili kısa bir bilgi verilmiştir. Bu teknolojinin kullanıldığı örnek Android uygulamalarından bir uygulama öğrencilere gösterilmiş ve giyilebilir gözlük öğrencilere gösterilerek incelemeleri sağlanmıştır.

4.2. Anket Sorularının Hazırlanması ve Sorulması

Öğrencilerin Artırılmış Gerçeklik ile ilgili görüşlerinin tespiti için 8'i Küçük, Yılmaz, Baydaş ve Göktaş(2014) tarafından geliştirilen 5'li likert (1:Kesinlikle Katılmıyorum,2:Katılmıyorum, 3: Kısmen Katılıyorum, 4:Katılıyorum, 5: Kesinlikle Katılıyorum) türünde olmaküzere toplam 12 soruluk anket oluşturulmuştur.

Anket soruları, araştırmacıların artırılmış gerçekliğin eğitimde önemli gördüğü temel noktalar üzerinden oluşturulmuştur. Ankette kullanılan Artırılmış Gerçeklik ile ilgili sorular Çizelge 4'te verilmektedir.

Çizelge-4: Artırılmış Gerçeklik İle İlgili Sorular

Aşağıdaki sorulara katılma düzeyinizi ilgili rakamı işaretleyerek belirtiniz.	**1**	**2**	**3**	**4**	**5**
1.Artırılmış Gerçekliğin derste faydasının olacağına inanıyorum.	7	7	6	3	17
2.Uzaktan Eğitim dersinde ek materyal kullanımının gerekli olduğunu düşün-müyorum.	7	7	10	9	7
3.Artırılmış gerçeklik için yapılan bir uygulama ödev olarak verildiğinde ödevi anlamam kolaylaştı.	1	7	11	12	9
4.Artırılmış gerçekliğin görsel derslerde katkısı daha çoktur.	4	4	10	15	7
5.Artırılmış gerçeklikle tasarlanmış donanımları gözlemlemek gerçek dünyada bu cihazları gözlemleme gereksinimini azaltmaktadır.	5	10	13	8	4
6.Mobil platformlarda kullanılan Artırılmış Ger-çeklik gözlüklerinden, uy-gulamalarından haberim var.	4	8	9	6	13
7.Mesleki eğitim ders-lerinde(Montaj, bilgisayar parçaları tanıma) Artırılmış Gerçeklik uygulamalarının öğrencinin gerçek hayata uyumluluğunu artıracağına inanıyorum.	1	7	11	11	10
8.Dersler Artırılmış Ger-çeklik teknolojisinin kul-lanımı Uzaktan Eğitim yön-temlerine ek olarak tek-noloji sunduğu için ders-lere katılım arttı.	5	7	11	11	6

Anket soruları internet ortamında oluşturulduktan sonra çalışma grubu vize haftalarında bir sınıfta toplanmış ve onlara kısaca artırılmış gerçeklik hakkında bilgi verilmiştir. Bu sürecin sonunda anket linki öğrencilere verilmiş ve bu anketi yapmaları için kendilerine 10 gün süre verilmiştir. Anketin son kısmında ise öğrencilere mail adresi verilmiş ve bu adrese anket soruları çerçevesinde ek olarak görüşlerini belirtmeleri istenmiştir.

Oluşturulan çalışma grubunun tamamından dönüt elde edildikten sonra SPSS ortamında güvenilirlik testi yapılmıştır. Güvenilirlik Testi sonuçları Çizelge 5'te verilmektedir.

Çizelge-5: Güvenilirlik Analizi

Cronbach's Alpha Değeri	Soru sayısı
,909	8

Çizelge 5'te görüldüğü gibi Cronbach's Alpha değeri 0,909 olarak elde edilmiştir.

4.3. Dönütlerin Analizi

Öğrencilere anketi cevaplamaları için verilen 10 günlük süre sonunda internet ortamından dönütler elde edilmiştir. Elde edilen dönütlerin analizi yapılmıştır.

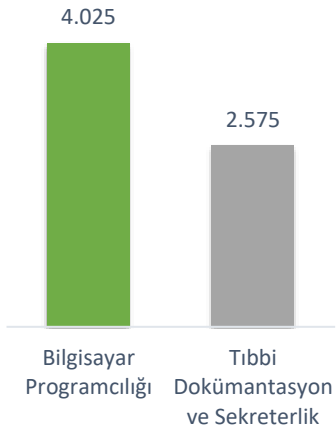
Çalışma grubundaki her bir öğrencinin aldıkları puanlar, cevapları doğrultusunda 5'li likert ölçeğindeki değerler göz önüne alınarak hesaplanmıştır. Değerlendirme öncelikle bölüm bazında yapılmıştır. Sonuçlar Çizelge 6'da gösterilmektedir.

Çizelge-6: Bölümlere Göre Standart Sapma ve Puan Ortalaması

Bölümler	Ortalama	Standart Sapma
Bilgisayar Programcılığı	4,025	5,25
Tıbbi Dokümantasyon	2,575	5,87

Her iki bölümün öğrencilerinin ayrı ayrı puan ortalamaları toplam soru sayısına bölünerek ortalama değerler elde edilmiştir. Tablo 6'da görüldüğü gibi Bilgisayar Programcılığı öğrencilerinin verdikleri cevapların likert ölçek ortalaması 4,025 olarak hesaplandığından 5'li likert ölçeğinde 4 değeri ile 5 değeri arasında yer aldığı gözlenmiştir. Grafik 2'de bölümlere göre likert ölçekortalaması gösterilmektedir. Tıbbi Dokümantasyon öğrencilerinde ise aynı hesaplama sonucunda 2,575 elde edilmiş ve bu da 5'li likert ölçeği göz önüne alındığında 2 değeri ile 3 değeri arasında yer aldığı söylenebilmektedir.

Grafik-2: Bölümlere Göre Likert Ölçek Ortalaması



Standart sapma açısından bakıldığında ise Bilgisayar Programcılığı bölümü öğrencilerinin standart sapması diğer bölümün standart sapmasından düşük çıkmıştır. Yani bu durum Bilgisayar Programcılığı öğrencilerinin öğrenme düzeylerinin birbirine yakın olduğunu ve puan farklılaşmalarının az olduğu göstermektedir. Bu grubun öğrencilerinin öğrenme düzeyi bakımından homojen özellik gösterdiği söylenebilmektedir.

Bazı sorulara verilen dönütler incelendiğinde bölümler bazında bazı yorumlar yapılmış ve bu yorumlar ki-kare testi ile desteklenmiştir. Bu yorumlar ve ki-kare testine ait tablolar aşağıda verilmiştir.

Çizelge 7: Bölüm-Soru 1'e Ait Ki-Kare Test Sonucu

Chi-Square Tests

	Value	df	Asymptotic Significance (2-sided)
Pearson Chi-Square	132,000[a]	4	,000
Likelihood Ratio	177,006	4	,000
Linear-by-Linear Association	111,390	1	,000
N of Valid Cases	132		

a. 1 cells (10,0%) have expected count less than 5. The minimum expected count is 3,55.

- Bilgisayar Programcılığı öğrencilerinin artırılmış gerçekliğin derste faydası olacağını düşündüğü, Tıbbi Dokümantasyon ve Sekreterlik bölümü öğrencilerinin ise ilgili soruya olumlu bakmadıkları gözlemlenmiştir. Çizelge 7'de ilgili ki-kare testi sonuçları verilmiştir. Burada Pearson Chi-Square satırına bakıldığında Asymptotic Significance değeri 0,000 < 0,05 olduğundan Soru 1'in Bölüm bilgisi ile ilişkisi olduğu sonucuna varılmaktadır.

Çizelge 8: Bölüm-Soru 4'e Ait Ki-Kare Test Sonucu

Chi-Square Tests

	Value	df	Asymptotic Significance (2-sided)
Pearson Chi-Square	34,579[a]	4	,000
Likelihood Ratio	41,090	4	,000
Linear-by-Linear Association	22,096	1	,000
N of Valid Cases	134		

a. 2 cells (20,0%) have expected count less than 5. The minimum expected count is 3,56.

- Artırılmış gerçekliğin görsel derslerde katkısının daha fazla olacağına Bilgisayar Programcılığı öğrencilerinin, Tıbbi Dokümantasyon ve Sekreterlik bölümü öğrencilerine göre daha olumlu cevaplar verdiği görülmüştür. Yine ilgili sorunun bölüm ile ilişkisini görmek için yapılan ki-kare testine ait sonuçlar Çizelge 8'de gösterilmektedir. Pearson Chi-Square satırına bakıldığında Asymptotic Significance değeri 0,000 < 0,05 olduğundan Soru 4'ün Bölüm bilgisi ile ilişkisi olduğu sonucuna varılmaktadır.

Çizelge 9: Sınıf- Soru 3'e Ait Ki-Kare Test Sonucu

Chi-Square Tests

	Value	df	Asymptotic Significance (2-sided)
Pearson Chi-Square	35,277[a]	4	,000
Likelihood Ratio	48,526	4	,000
Linear-by-Linear Association	26,830	1	,000
N of Valid Cases	134		

a. 2 cells (20,0%) have expected count less than 5. The minimum expected count is ,72.

- 2.sınıftaki öğrenciler derste artırılmış gerçeklik ile ilgili bir uygulama ödev olarak verildiğinde ödevi anlamalarının kolay olacağını düşünürken, 1. Sınıf öğrenciler bu düşüncede değillerdir. İlgili sorunun sınıf ile ilişkisinin olup olmadığını anlamak için yapılan ki-kare testine ait sonuçlar Çizelge 9'da verilmektedir. Asymptotic Significance değeri 0,000 < 0,05 olduğundan soru 3'ün sınıf bilgisi ile ilişkisi olduğu sonucuna varılmaktadır.

Çizelge-10: Çalışma Grubunun Sınıf Bölüm ve Cinsiyet Durumlarına Göre Anlamlılık Düzeyleri

Ölçütler		N	S	p
Sınıf	1.Sınıf	18	6,45	0,00000209
	2.Sınıf	22	5,67	
Bölüm	BP	20	5,25	0,00000009
	TDS	20	5,87	
Cinsiyet	Kadın	15	8,11	0,027
	Erkek	25	6,72	

*p<0.05

Çizelge10'da görüldüğü gibi *p<.05 anlamlılık düzeyi için sınıf ölçütünde .00000209< .05 olduğu için sonuç anlamlı çıkmıştır. Yani çalışma grubu öğrencilerinin anket sorularına verdikleri dönütlerden aldıkları puanlar sınıf durumlarına göre anlamlı bir farklılık göstermektedir.

Çalışma grubu öğrencilerinin anket sonucunda aldıkları puanların bölüm ölçütüne göre anlamlılık değeri Çizelge10'da gösterilmektedir. Bu değer

*p<.05 anlamlılık düzeyi için bölüm ölçütünde .00000009< .05 olduğu için çalışma grubu öğrencilerinin puanlarının bölüm durumlarına göre anlamlı bir farklılık gösterdiğini vermektedir.
Çizelge10'da belirtilen sonuçlara göre *p<.05 anlamlılık düzeyi için çalışma grubu öğrencilerinin aldıkları puanlar cinsiyet durumuna göre .027< .05 olduğundan öğrencilerin puanlarının cinsiyet durumuna göre anlamlı bir farklılık gösterdiği görülmektedir.

Çizelge-11: Bilgisayar Programcılığı Öğrencilerinin Sınıf ve Cinsiyet Durumlarına Göre Anlamlılık Düzeyleri

Ölçütler		N	S	p
Sınıf	1.Sınıf	6	4,71	0,007
	2.Sınıf	14	4,25	
Cinsiyet	Kadın	7	4,03	0,001
	Erkek	13	3,72	

*p<0.05
Çalışma grubu öğrencilerinden Bilgisayar Programcılığı'nda okumakta olan öğrenciler için anlamlılık düzeyi sınıf ve cinsiyet durumlarına göre hesaplanmış ve sonuçları Çizelge11'de verilmiştir. *p<.05 anlamlılık düzeyi için sınıf durumuna göre .007< .05 ve cinsiyet durumuna göre .001 < .05 olduğundan Bilgisayar Programcılığı öğrencilerinin sınıf ve cinsiyet durumlarına göre anlamlı farklılıklar gösterdiği gözlemlenmiştir.

Çizelge-12: Tıbbi Dokümantasyon ve Sekreterlik Öğrencilerinin Sınıf ve Cinsiyet Durumlarına Göre Anlamlılık Düzeyleri

Ölçütler		N	S	p
Sınıf	1.Sınıf	12	3,5	0,0000364
	2.Sınıf	8	4,02	
Cinsiyet	Kadın	8	6,22	0,295
	Erkek	12	5,59	

*p<0.05

Çizelge12'de Tıbbi Dokümantasyon ve Sekreterlik bölümünde okuyan öğrencilerin sınıf ve cinsiyet durumuna göre anlamlılık düzeyleri gösterilmektedir. Bu sonuçlara bakıldığında *p<.05 anlamlılık düzeyi için sınıf durumuna göre
.0000364 < .05 olduğu için anlamlı bir farklılık olduğu sonucuna varılmaktadır. Fakat cinsiyet durumuna göre anlamlılık düzeyi .295> .05 olarak elde edildiğinden Tıbbi Dokümantasyon ve Sekreterlik öğrencilerinin cinsiyet durumuna göre anlamlı bir farklılık göstermediği görülmektedir.
Verilen mail adresine öğrencilerin göndermiş oldukları bazı görüşler şu şekildedir:

- "Sevmediğim bir ders Artırılmış Gerçeklik ile işlenecek olursa derse severek gelirim."
- "Artırılmış gerçeklik ders dışında da ilgimi çektiği için derslerin artırılmış gerçeklikle işlenmesinin derslere ilgiyi daha da artıracağını düşünüyorum."

5.Sonuçlar ve Gelecek Çalışmalar

Yapılan bu çalışmada artırılmış gerçekliğin Uzaktan Eğitim öğrencileri üzerinde öğrenme, ilgi, tutum ve davranışları açısından ne gibi faydaları olduğu araştırılmıştır. Bu araştırma çerçevesinde Uzaktan Eğitim Meslek Yüksekokulu'nda okumakta olan 40 kişilik çalışma grubunun 25'ini (%52,5) erkek, 15'ini (%37,5) ini kadın öğrenciler oluşturmuştur. Bu çalışma grubunun 20'si(%50) Bilgisayar Programcılığı bölümünde, 20'si (%50) tıbbi Dokümantasyon ve Sekreterlik bölümünde öğrenim gören öğrencilerdir. Bu öğrencilerden 18'i 1.sınıf (%45), 22'si 2. Sınıfta(%55) okumaktadırlar. Dönütler alındıktan sonra yapılan güvenilirlik analizinin 0,909 olarak hesaplandığı Çizelge 5'te gösterilmektedir. Çalışma grubu öğrencilerinin Çizelge 6'da görüldüğü gibi 5'li Likert ölçeği ortalama puanı Bilgisayar Programcılığı öğrencilerinde 4,025 olarak, Tıbbi Dokümantasyon ve Sekreterlik bölümü öğrencilerinde ise 2,575 olarak elde edilmiştir. Standart sapma ise Bilgisayar Programcılığı öğrencilerinde 5,25 Tıbbi Dokümantasyon ve Sekreterlik öğrencilerinde ise 5,87 olarak elde edilmiştir. Bu sonuçlar anketteki sorular göz önüne alındığında Bilgisayar Programcılığı öğrencilerinin derslerin Artırılmış

Gerçeklik hakkında daha çok bilgi sahibi olduklarını ve sorulara daha yüksek puanlar verdiklerini göstermektedir.

Yapılan çalışma sonrasında çalışma grubunun anlamlılık düzeyi ile ilgili elde edilen sonuçlar:

- Çalışma grubu için sınıf ve bölüm durumunda sonuçlar anlamlıdır. Başka bir deyişle çalışma grubu öğrencileri aldıkları puanlar doğrultusunda sınıf, cinsiyet ve bölüm durumlarına göre anlamlı farklılıklar göstermektedir.
- Bilgisayar Programcılığı öğrencileri sınıf ve cinsiyet durumlarına göre anlamlı farklılıklar göstermektedir.
- Tıbbi Dokümantasyon ve Sekreterlik bölümünde okuyan öğrencilerin sınıf durumunda anlamlı farklılık gösterirken cinsiyet durumunda anlamlı bir farklılık göstermediği gözlemlenmiştir.

Gelecek çalışmalarımızda anket soru sayısı artırılıp soruların faktörlere bağlı hazırlanması düşünülmektedir. Mobil ortamda artırılmış gerçeklik ile ilgili bir uygulama gerçekleştirilip sonrasında anket cevaplanması istenecek ve böylece daha fazla dönüt alınması sağlanacaktır. Sonuçlar ise faktör analizine göre değerlendirilecektir.

Kaynakça

[1] Karasar, Ş. , *Eğitimde yeni iletişim teknolojileri-internet ve sanal yüksek eğitim,* The Turkish Online Journal of Educational Technology–TOJET, 3(4), pp. 117-125 , 2004.

[2] Aksoy, H. H. , *Eğitim Kurumlarında Teknoloji Kullanımı ve Etkilerine İlişkin Bir Çözümleme,* Eğitim Bilim Toplum Dergisi Cilt:1 , Sayı: 4, 2003.

[3] Göktaş, Y., Küçük, S., Aydemir, M., Telli, E., Arpacık, Ö., Yıldırım, G., & Reisoğlu, İ. ,*Türkiye'de eğitim teknolojileri araştırmalarındaki eğilimler: 2000-2009 dönemi makalelerinin içerik analizi,* Kuram ve Uygulamada Eğitim Bilimleri Dergisi, 12(1), pp. 177-199, 2012.

[4]Küçük, S., Yılmaz, R. M., & Göktaş, Y., *İngilizce öğreniminde artırılmış gerçeklik: öğrencilerin başarı, tutum ve bilişsel yük düzeyleri*, Eğitim ve Bilim, 39(176), pp. 393-404, 2014.

[5] Kayabaşı, Y.,*Sanal gerçeklik ve eğitim amaçlı kullanılması,* Turkish Online, 4(3), pp. 151-158., 2002

[6] Çankaya, İ.A., Yüksel,A.S., & Koyun, A. *iOS Platformunda Artırılmış Gerçeklik ile Yön Belirleme,* Akademik bilişim Konferansı Bildirileri, 2013.

[7] Sırakaya, M. (2015). *Artırılmış Gerçeklik Uygulamalarının Öğrencilerin Akademik Başarıları, Kavram Yanılgıları ve Derse Katılımlarına Etkisi*, Doktora Tezi, Gazi Üniversitesi, Ankara, 2015.

[8] Yuen, S., Yaoyuneyong, G., ve Johnson, E., *Augmented reality: An overview and five directions for AR in education*, Journal of Educational Technology Development and Exchange, 4(1), 2011.

[9] Korucu, A. T., Gençtürk, T., & Sezer, C., *Artırılmış gerçeklik uygulamalarının öğrenci başarı ve tutumlarına etkisi*, Akademik Bilişim Kongresi, 2016.

[10] Tülü, M., Yılmaz, M. , *İphone İle Arttırılmış Gerçeklik Uygulamalarının Eğitim Alanında Kullanılması*, Akademik Bilişim Kongresi,2013.

[11] Köroğlu, O. , *En yaygın iletişim ortamında artırılmış gerçeklik uygulamaları,* Türkiye'de 17. Internet Konferansı,2012.

[12] Greg Smedley-Warren(2017), *Augmented Realıty In The Classroom,* 22/11/2017 tarihinde https://thekindergartensmorgasboard.com/2017/04/augmented-reality-classroom-alive-studios.html adresinden erişilmiştir.

[13] Ogasawara, T.(2011), *PBS Kids, WGBH' New Augmented Reality Math App Gets Kids Moving*, 22/11/2017 tarihinde http://www.adweek.com/digital/pbs-

kidswgbh-app-dev-group-talks-about-augmented-reality-math-app-for-kids-fetch-lunch-rush/?red=im adresinden erişilmiştir.

[14]Melnick, C. (2011), *PBS KIDS Launches Its First Educational Augmented Reality App*, 22/11/2017 tarihinde http://www.pbs.org/about/blogs/news/pbs-kids-launches-its-first-educational-augmented-reality-app/ adresinden erişilmiştir..

[15] Reitmayr, G.,& Schmalstieg, D. , *Collaborative augmented reality for outdoor navigation and information browsing*, In Proc. Symposium Location Based Services and TeleCartography , pp. 31-41,2004.

Terim-Doküman Matrisleri için Sıralamaya Dayalı Bir Kayıpsız Sıkıştırma Şeması

Reordering Based Lossless Compression Scheme for Term-Document Matrices

Can ÖZBEY
IDEA Teknoloji Çözümleri
Ar-Ge Mühendisi
İstanbul, TÜRKİYE
can.ozbey@ideateknoloji.com.tr

Murat Cihan SORKUN
IDEA Teknoloji Çözümleri
Ar-Ge Mühendisi
Istanbul, TÜRKİYE
murat.sorkun@ideateknoloji.com.tr

Öz

Kayıpsız veri sıkıştırma, özellikle bellek içi veri tabanları ve önbellek kullanımlı bilgi geri kazanım sistemlerinde, harcanan disk alanını azaltmasının yanı sıra, etkin kod çözme algoritmaları aracılığıyla bilgiye erişimi hızlandırması sebebiyle önem arz etmektedir. Bu çalışmada, ters dizinlerin sıkıştırılması kapsamında yeni bir değişken sekiz ikili kodlama yöntemi geliştirilmiş ve terim-doküman matrisinin bant genişliğinin indirgenmesi amacıyla tepe tırmanmaya dayalı çift kutuplu dizilim şeması önerilmiştir. Bu şema, internet üzerinden toplanan haber metinlerine uygulanarak doküman dizmenin dizin sıkıştırma oranına olan etkisi incelenmiştir.

Anahtar Sözcükler— Terim-Doküman Matrisi, Kayıpsız Veri Sıkıştırma, Ters Dizin Sıkıştırma, Değişken Sekiz İkili Kodlama, Özyinelemeli Sekiz İkili Kodlama, Doküman Dizme, Matris Bant Genişliği İndirgeme, Çift Kutuplu Sıralama, Tepe Tırmanma

Abstract

Lossless data compression can have a great importance with regards to efficiency in data retrieval through effective decoding algorithms, especially for in-memory databases and information retrieval systems that use caching, not to mention the less required disk space for storage. In this work, we present a novel variable byte encoding technique to compress inverted indexes and a bipolar permutation scheme based on hill climbing to reduce the bandwidth of the term-document matrix.

Applying this scheme to a collection of news crawled from the Web, the effect of document reordering on index compression is investigated.

Keywords— Term-Document Matrix, Lossless Data Compression, Inverted Index Compression, Variable Byte Encoding, Recursive Byte Encoding, Document Reordering, Matrix Bandwidth Reduction, Bipolar Ordering, Hill Climbing

Gönderme ve kabul tarihi: 04.05.2018-17.07.2018
Makale türü: Araştırma

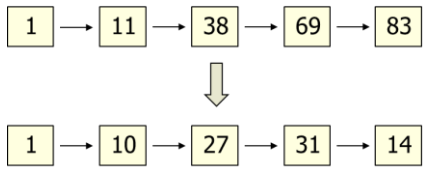
1. Giriş

Kayıpsız veri sıkıştırma tekniklerinden birçok veri tabanı ve bilgi geri kazanım (information retrieval) sistemlerinde faydalanılmaktadır. Sıkıştırılmış verinin disk üzerinde daha az yer kaplamasıyla depolama maliyetinin azaltılması ve bilgi yoğunluğunun artırılıp girdi-çıktı (I/O) işlemlerinin en aza indirgenmesiyle veri transfer hızının artırılması, veri sıkıştırma yöntemlerinin bu kapsamda araştırılmasını gerekli kılan en belirgin ihtiyaçlardandır [1]. Bunun yanında, sıkıştırılmış veri üzerinde sorgu işlemlerinin yapılmasını sağlayan sıkıştırma şemaları, özellikle C-Store [2] gibi sütun tabanlı (column-oriented) veri tabanı sistemlerinde, geleneksel satır tabanlı (row-oriented) veri tabanlarıyla kıyaslandığında yüksek verimde sorgu işlemeyi mümkün kılmaktadır [3]. Sütun tabanlı mimaride, ardışık kayıtların daha fazla benzerliğe sahip olması sıkıştırılabilirliği artırmakta, RLE (Run-Length Encoding) gibi temel kodlama yöntemlerinin veri üzerinde uygulanmasına olanak sağlamaktadır [4]. Benzer şekilde, bilgi geri kazanım sistemlerinde de veriyi önbellekte tutarak hızlı kod çözme teknikleriyle disk okuma maliyetini azaltma ve sıkıştırılmış veri üzerinde işlem yaparak verimliliği artırma imkanı, veri sıkıştırmayı cazip hale getirmektedir. Bu kapsamda yapılan veri sıkıştırma çalışmaları iki ayrı başlık altında incelenmektedir:

sözlük (dictionary) sıkıştırma ve dizin (index) sıkıştırma [5].

Bilgi geri kazanım sistemlerinde sözlük sıkıştırmanın amacı, dokümanlardan çıkarılan terimlerin (kelimeler, kökler, çok kelimeli ifadeler, vs.) ana bellekte alan açısından daha verimli bir şekilde tutulması ve tercihen kod çözmeye ihtiyaç duymadan terim indislerinin belirlenebilmesidir. Önkodlama (front-coding) şemaları [6], sıkıştırılmış son ek dizileri ve ağaçları [7], sıkıştırılmış komut tabloları (hash table) ve sıkıştırılmış Trie [8] bu amaçla kullanılan veri yapıları ve yöntemlere örnek olarak gösterilebilir. Özellikle biyoenformatik alanında, sözlük boyutlarının çok fazla olmasından ötürü sözlük sıkıştırma tekniklerinden yararlanılmaktadır.

Dizin sıkıştırma ise, terimlerin hangi dokümanlarda bulunduğu bilgisinin daha az yer kaplamasını ve mümkünse sorgu şemasında ihtiyaç duyulan temel mantıksal operatörlerin sıkıştırılmış vektörler üzerinde uygulanabilmesini amaçlamaktadır. Bu amacı gerçekleyen en bilindik yöntem, seyrek bir dağılıma sahip olan terim-doküman bit eşleminde yer alan bit vektörlerinin sadece '1' değerlerinin indis bilgilerinin tutulmasını sağlayan ters dizinleme (inverted indexing) tekniğidir [9]. Ters dizin veri yapısı, seyrek bit vektörlerini kayda değer bir oranda sıkıştırmakla kalmayıp üzerinde kesişim ve birleşim işlemlerinin kolayca yapılabilmesinden ötürü mantıksal operatörlerin kullanımını da mümkün hale getirmektedir. Literatürde, ters dizinlerde yer alan sayıların sıkıştırılması işlemi, dizin sıkıştırma (index compression) başlığı altında ele alınmakta olup evrensel (global) ve yerel (local) sıkıştırma şemaları uygulanarak gerçekleştirilmektedir. Söz konusu şemaların etkili olabilmesi için, öncelikle, dizinlere aralık kodlaması (difference coding) uygulanmaktadır. Ortaya çıkan doküman aralıkları (d-gaps) hem daha küçük sayılardan oluşmakta hem de daha az seyrek bir dağılım sergilemektedir. Böylece, entropi kodlaması ve evrensel sayı kodlaması (integer coding) teknikleriyle, dizinler, daha etkin bir şekilde sıkıştırılabilmektedir. Aralık kodlama uygulamasının bir örneği Şekil 1'de verilmiştir. Doküman indislerinin aralık değerlerinden geri çevrimi ise, dizinin ilk değerinden başlayarak özyinelemeli bir şekilde toplanmasıyla sağlanmaktadır. Denklem (1)'de aralık dizisi *A*'nın doküman indislerinin bulunduğu *D* dizisine özyinelemeli çevrimi formülize edilmiştir.



Şekil 1. Aralık kodlaması

$$D[i] = \begin{cases} A[i] & if\ i = 0 \\ A[i\text{-}1] + A[i] & if\ i > 0 \end{cases} \quad (1)$$

Aralık değerlerinin parametrik olmayan evrensel şemalar ile sıkıştırılması için kullanılan yöntemler arasında birli kodlama (unary coding), değişken sekiz ikili kodlama (vByte encoding) [10], Elias-γ ve Elias-δ kodlamaları [11], Simple9 kodlaması [12], Huffman kodlaması [13] ve aritmetik kodlama [14] gibi entropi temelli teknikler bulunmaktadır. Birbirinden farklı aralık değeri sayısının genelde yüksek olması, kod kelimelerinin bulunduğu arama tablolarının da boyutunu artırdığından Huffman kodlamasının aralıklara doğrudan uygulanmasını zorlaştırmaktadır. Fraenkel ve Klein (1985), *LLRUN* isimli bit eşlemi sıkıştırma şemalarında, Elias kodlamasındaki üssel bölümlendirmeyi (bucketing) genelleştirerek Fibonacci dizisi gibi farklı sayım (enumeration) sistemlerinin sayı kodlamaya nasıl uygulabileceğini göstermiş ve Huffman kodlamasını, kod kelimelerindeki bölümlendirme seviyelerini belirten ön ekleri (prefix) kodlamak için kullanıp aralık sıkıştırma oranını optimize etmişlerdir [15].

Evrensel şemaların sıkıştırma performansı, aralıkların istatistiksel dağılımlarına göre değişkenlik göstermektedir. Örneğin, birli kodlamanın optimum sıkıştırma oranını yakalaması için x uzunluğundaki bir aralığın $\left(\frac{1}{2}\right)^x$ olasılığı ile geometrik bir dağılıma sahip olması, Elias-γ kodlamasının optimum çalışması içinse aralıkların $\left(\frac{1}{2x^2}\right)$ olasılığı ile dağılması gerekmektedir. Dolayısıyla, dizinlenecek derlemin doküman aralıklarının bu dağılımlara uymaması, evrensel sayı kodlamalarının yetersiz kalmalarına neden olmaktadır. Bu noktada, parametrik yerel kodlama şemaları, aralık değerlerinin dağılım özelliklerini kullanarak evrensel şemalara göre daha iyi bir sıkıştırma performansı gösterebilmektedir. Bu bağlamda, Golomb kodlaması [16], parametrik kodlama şemaları arasında yaygın olarak kullanılan en eski yöntemlerden biridir. Bu

şemada, I sayısı, seçilen bir M böleniyle çarpan ve kalanlarına ayrılır:

$$I = \lfloor I/M \rfloor * M + (I \bmod M) \qquad (2)$$

Golomb kodu, $\lfloor I/M \rfloor$ çarpanının birli kodlaması ile $(I \bmod M)$ kalan değerinin azami $\lceil \log_2 M \rceil$ bitle temsil edilecek ikili kodlamasının birleşiminden oluşmaktadır. Kodlanacak sayının belirli bir p olasılığı ile geometrik dağılım sergilediği varsayımı, Golomb kodlamasının temelini oluşturmaktadır. Bu nedenle, dizin aralıklarının sıkıştırılması işleminde, rastgele seçilen bir terimin rastgele seçilen bir dokümanda bulunma ihtimali p ise, söz konusu dizinde X uzunluğunda bir doküman aralığının oluşma olasılığı aşağıdaki şekilde hesaplanmaktadır:

$$P(x) = (1-p)^{x-1} * p \qquad (3)$$

Geometrik dağılan sayı değerlerini optimum Golomb koduna dönüştüren M parametresi, p olasılığı kullanılarak tahminlenebilmektedir [17]. Dolayısıyla, dizin aralıkları sıkıştırılırken terim başına düşen ortalama doküman sayısına göre global bir M parametresi ile uygulanan Golomb kodlamasının, Elias kodlaması gibi parametrik olmayan tekniklere kıyasla daha iyi sıkıştırma oranları verdiği gözlenmiştir [10, 18]. Ayrıca, Golomb kodlamasının terim dizinlerine farklı parametrelerle uygulanması doküman aralıkları dağılımının daha iyi temsil edilmesini mümkün kılmaktadır. Böyle bir durumda, kazanılan alanın, bellekte tutulması gereken terim sayısı kadar parametrenin harcayacağı alandan büyük olması beklenmektedir.

Dizin sıkıştırma için kullanılan şemalardan bir diğeri de değişken sekiz ikili kodlama yöntemlerinden biri olan vByte kodlamasıdır. Bu yöntemde, sayı değerleri değişken sayıda sekiz ikili (byte) ile temsil edilmekte olup sekiz ikililerinin her birinin ilk bit değeri, yeni bir sayının kodlamasına başlanıp başlanmadığı bilgisini taşımaktadır. Bu sayede, kodlanan sayının kaç tane sekiz ikili ile temsil edileceği belirlenmiş olur. Sekiz ikili seviyesinde kodlama yapan vByte, bit seviyesinde kodlama yapan Elias ve Golomb kodlamalarıyla kıyaslandığında, sıkıştırma oranı açısından daha verimsizdir. Bunun nedeni, sayıyı kodlayan sekiz ikililerin sabit uzunlukta olmasının son sekiz ikilinin fazlalığına yol açmasıdır. Ancak, diğer taraftan, vByte kodlamasında 8-bit seviyesinde gerçekleşen kod çözme işlemi, bit seviyesinde işlem yapan yöntemlere göre daha hızlı tamamlanmaktadır [18, 19]. Bu nedenle, değişken sekiz ikili kodlama şemaları, alan kazancından feragat etmelerine karşın kod çözme verimliliklerinden ötürü bilgi geri kazanım sistemlerinde kullanılmaktadır [20].

Çalışmamızın ilk bölümünde, daha önce tanıtmış olduğumuz yeni bir değişken sekiz ikili kodlama yöntemi olan özyinelemeli sekiz ikili kodlama (recursive byte encoding) [21] şemasının daha detaylı bir incelemesi yapılmış ve sıkıştırma oranı açısından vByte kodlamasıyla karşılaştırılmıştır. İkinci bölümde ise, çift kutuplu sıralamaya dayalı olan ve çok sayıda doküman için ölçeklendirilebilir bir doküman dizme (document reordering) şeması geliştirilmiştir. Bu şemanın eniyilemesi için tepe tırmanma (hill climbing) algoritması uygulanmış ve çift kutuplu sıralamanın farklı sayı kodlama şemaları çerçevesinde, internetten toplanan 100000 adet haber dokümanı üzerinde dizin sıkıştırmaya olan katkısı ölçülmüştür.

2. Özyinelemeli Sekiz İkili Kodlama

Özyinelemeli sekiz ikili kodlama (RBE), Golomb kodlamasında olduğu gibi bir X sayısının bölen, çarpan ve kalan ile temsil edilmesine dayalıdır. Golomb kodlamasında isteğe bağlı olarak seçilen M parametresi, sekiz ikili kodlamada 256 olarak sabitlenmiştir. Böylece, X sayısı aşağıdaki şekilde temsil edilmiş olur:

$$X = 256 * C + R \qquad (4)$$

İfadedeki C çarpanı 8-bit ile temsil edilemeyecek kadar büyükse, bu çarpan da (4) ifadesiyle çarpan ve kalanına ayrıştırılarak kodlama işlemi devam eder. Bu işlem, en son üretilen çarpan 256'dan küçük oluncaya kadar tekrarlanır. İşlemin özyinelemeli ifadesi aşağıdaki gibidir:

$$f(x) = \begin{cases} x - 1 & if\ x < 256 \\ 256 * f\left(\left\lfloor \frac{x}{256} \right\rfloor\right) + (x \bmod 256) & if\ x \geq 256 \end{cases} \qquad (5)$$

Örneğin, 1000 sayısı (256*3 + 232) şeklinde çarpan ve kalanına ayrıldığında, 3 sayısı 256'dan küçük olduğu için sayının 1 eksiği çarpan bilgisi olarak tutulmaktadır. Böylece, 1000 sayısı [255, 2, 232] şeklinde 3 sekiz ikili, yani 24 bit ile temsil edilmiş olur. Diğer yandan, 158965 sayısı kodlanacak olursa, çarpan değeri $\lfloor 158965/256 \rfloor$, 256 değerinden küçük olmadığı için kodlamaya devam edilir ve 620 çarpanı [255, 1, 108] olarak kodlanır. Böylece, 158956 sayısı [255, 255, 1, 108, 245] şeklinde 5 sekiz ikili, yani 40

bit ile temsil edilmiş olur. Kodlama ve kod çözme işlemlerinin algoritmik adımları aşağıda sırasıyla verilmiştir.

Algoritma 1. Kodlama

```
1:  bytes ← {}, i ← 0
2:  ENCODE(N, bytes)
3:      IF N < 256 THEN
4:          bytes[i] ← N – 1
5:          i ← i + 1
6:          RETURN bytes
7:      ENDIF
8:      WHILE N ≥ 256
9:          unshift(bytes, 255)
10:         i ← i + 1
11:         R ← N mod 256
12:         N ← ⌊N/256⌋
13:         bytes← ENCODE(N, bytes)
14:         bytes[i] ← R
15:         i ← i + 1
16:         RETURN bytes
17:     ENDWHILE
18: ENDENCODE
```

Algoritma 2. Kod Çözme

```
1:  c ← 0, i ← 0
2:  DECODE(bytes)
3:      i ← i + 1
4:      IF bytes[i-1] < 255 THEN
5:          IF c == 0 THEN
6:              c = bytes[i-1] + 1
7:              RETURN c
8:          ELSE
9:              c = 256*c + bytes[i]
10:             RETURN c
11:         ENDIF
12:     ELSE
13:         x = 256*DECODE(bytes) + bytes[i]
14:         i ← i + 1
15:         RETURN x
16:     ENDIF
17: ENDDECODE
```

RBE'nin harcadığı bit sayısı vByte ile karşılaştırmalı şekilde Tablo 1'de gösterilmiştir. Tablodan görüldüğü üzere RBE, sıkıştırma performansı bakımından ortalaması daha küçük olan sayı dağılımlarında başarılıdır. Aralıkların geometrik dağılım gösterdiği varsayıldığında, RBE'nin aralıkları vByte'tan daha iyi sıkıştırması için p değerinin en az kaç olması gerektiği yaklaşık olarak tahminlenebilir. Bu amaç doğrultusunda, kümülatif geometrik dağılım fonksiyonu kullanılarak bir aralığın Tablo 1'de verilen sayı değerleri arasında kalma olasılıkları hesaplanmıştır. Rastgele seçilen bir terimin rastgele seçilen bir dokümanda bulunma olasılığı p ise, aralık değerlerinin N sayısından küçük olma olasılığı aşağıdaki şekilde hesaplanmaktadır:

$$P(x < N) = 1 - (1-p)^N \qquad (6)$$

Bu fonksiyon kullanılarak aralık değerlerinin M ve N sayıları arasında kalma olasılığı bulunur:

$$P(M \le x < N) = P(x < N) - P(x < M) = (1-p)^M - (1-p)^N \qquad (7)$$

Tablo 1. RBE ve vByte kodlamalarında sayı aralıkları ve gereken bit sayıları

RBE	vByte	Sayı Aralığı
8	8	$0 \le N < 2^7$
8	16	$2^7 \le N < 2^8$
24	16	$2^8 \le N < 2^{14}$
24	24	$2^{14} \le N < 2^{16}$
40	24	$2^{16} \le N < 2^{21}$
40	32	$2^{21} \le N < 2^{24}$
56	32	$2^{24} \le N < 2^{28}$
56	40	$2^{28} \le N < 2^{32}$

RBE'nin hangi koşulda vByte'a göre daha iyi sıkıştırma performansı gösterdiğinin bulunması için kazanç sağlayan ve kayba yol açan sayı aralıklarının (7) ile hesaplanan olasılık toplamlarının birbirlerine eşit olduğu p değerinin bulunması gerekmektedir. Aşağıda, RBE'nin daha iyi sıkıştırması için sağlanması gereken eşitsizlik verilmiştir:

$$P(2^7 \le x < 2^8) * 8 > P(2^8 \le x < 2^{14}) * 8 + P(2^{16} \le x < 2^{21}) * 16 + P(2^{21} \le x < 2^{24}) * 8 + \cdots \qquad (8)$$

Eşitsizliği basitleştirmek amacıyla, sağ tarafta yer alan olasılık değerlerinin, $P(2^8 \le x < 2^{14})$hariç, eşitliği sağlayan p değerinde yok sayılabilir oldukları varsayılmıştır. Böylece, yukarıdaki ifade $P(2^7 \le x < 2^8) > P(2^8 \le x < 2^{14})$ ifadesine indirgenmiş olur. Aşağıda bu ifade kullanılarak p değerinin çıkarım adımları gösterilmiştir.

$$(1-p)^{128} - (1-p)^{256} > (1-p)^{256} - (1-p)^{16384}$$

$$(1-p)^{16384} \approx 0$$

$$(1-p)^{128} > 2 * (1-p)^{256}$$

$$(1-p)^{128} < 0.5$$
$$(1-p) < 0.9946$$

$$p > 0.0054 \qquad (9)$$

Geometrik dağılan aralıkların beklenen değeri, aralık değerleri ve karşılık gelen olasılıkların çarpımlarının toplamıdır:

$$E(x) = \sum_{x=1}^{\infty} x * (1-p)^{x-1} * p$$

$$E(x) = (1-p)/p \qquad (10)$$

Beklenen değer, p olasılığı 0.0054 alındığında 184.19 olmaktadır. Dolayısıyla, RBE'nin vByte'tan daha iyi sıkıştırması için terim dizinlerinin ortalama doküman aralığı yaklaşık olarak 184 değerini geçmemelidir. Ancak, çok sayıda doküman içeren geniş derlemlerde ortalama aralık değeri büyük olasılıkla 184'ten fazla olacağından RBE yerine vByte kodlamasının kullanılması daha az sayıda sekiz ikili ile dizinlerin temsil edilmesini sağlayacaktır. Bu noktada doküman dizme işlemi, aralıkların yerelliğini (locality) artırarak ortalama aralık değerini küçülttüğünden önem arz etmektedir. Bu nedenle bir sonraki bölümde, çift kutuplu doküman dizme şeması gerçek veri üzerinde uygulanarak RBE'nin sıkıştırma performansına olan etkisi ölçülmüştür.

3. Doküman Dizme

Doküman dizme, terimlerin dizin aralıklarının toplamının minimum olmasını sağlayacak doküman sıralamasının bulunması işlemidir. Bu işlemin öncelikli amacı ortalama doküman aralığı değerini küçülterek dizinlerin sıkıştırılabilirliğini artırmaktır. Bunun yanı sıra, aralıkların küçülmesi sıkıştırılmış dizinlerin kod çözme verimliliğini de artırarak ortalama sorgu işleme süresini kısaltmaktadır [22]. Bu bakımdan doküman dizme işlemi, bilgi geri kazanım sistemlerinde hem alan hem de hız açısından fayda sağlaması nedeniyle öneme sahiptir.

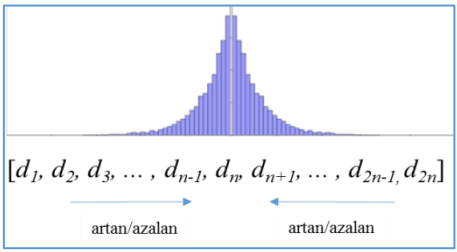
Doküman dizme, aslında, terim-doküman matrisi A'dan elde edilen simetrik doküman kovaryans matrisi A^TA'nın bant genişliği indirgeme problemidir. Bant genişliği indirgeme bir birleşimsel eniyileme (combinatorial optimization) problemi olup NP-tam karmaşıklığındadır [23]. Dolayısıyla, problemin çözümüne yönelik farklı yaklaşımlar içeren birçok akademik çalışma mevcuttur. Bu bağlamda, kullanılan metasezgisel yöntemler arasında genetik algoritma [24], benzetimli tavlama (simulated annealing) [25] ve tabu araması [26] örnek olarak gösterilebilir. Diğer yandan, Cuthill-McKee algoritması [27], GPS algoritması [28] ve Sloan algoritması [29] gibi çizge temelli yöntemler, PCA (Principal Component Analysis) [30] ve MDS (Multi-Dimensional Scaling) [31] gibi boyut indirgeme teknikleri ve ikili kümeleme (biclustering) [32] simetrik matrislerin bant genişliğini indirgeme amacıyla uygulanan farklı yaklaşımlardır. Ancak, bu yöntemler çok büyük matrislere uygulandıkları takdirde yüksek miktarda hafıza ve hesaplama kaynağına ihtiyaç duymaktadırlar. Bu nedenle çalışmamızda dokümanlar, matris veya çizge yapısı yerine bir dizide temsil edilmişlerdir. Bu dizi, bir sıralama ölçütüne göre çift kutuplu (bipolar) olacak şekilde sıralanmış ve ardından tepe tırmanma algoritması ile eniyilemesi yapılmıştır.

3.1 Çift Kutuplu Dizilim Şeması

Doküman dizme işleminin çok sayıda dokümandan oluşan derlemlere ölçeklenebilmesi için basit sıralama şemalarının önerildiği çalışmalar literatürde yer almaktadır. Silvestri [33], dokümanların indirildikleri URL'e göre alfabetik şekilde sıralanmasının dizin aralıklarını önemli ölçüde azalttığını göstermiştir. Benzer şekilde, Ramaswamy vd. [34], e-ticaret ürünlerine ait dokümanların ürün kategorisi gibi ontolojik özelliklere göre sıralanmasının dizin sıkıştırma oranı ve sorgu işleme hızına olan katkısını incelemiştir. Bu yöntemler, hızlı ve etkili olmalarına rağmen dokümanlarla ilgili harici bilgi gerektirdiklerinden ötürü her türlü derleme uygulanamıyor olup bilhassa dokümanların URL bilgilerinin tutulduğu büyük ölçekli arama motorlarının doküman sıralama optimizasyonu için tercih edilmektedirler.

Dizin aralıklarını küçülten diğer bir doküman dizme yöntemi, dokümanların içerdikleri ayrık terim sayısına göre sıralanmasıdır [35]. Bu yöntem, aslında, toplam ayrık terim sayıları benzer dokümanların paylaştıkları ortak terim sayısının, genel olarak daha yüksek olacağı beklentisine dayalıdır. Diğer taraftan, terim sayıları arasındaki fark yüksek olan iki doküman, doğal olarak daha az sayıda ortak terime sahip olacaktır. Bunun nedeni, ortak terim sayısının alabileceği azami değerin daha az sayıda terim içeren dokümanda bulunan terim

sayısıyla sınırlanmış olmasıdır. Çift kutuplu dizilimde ise dokümanlar, toplam terim sayısı gibi bir sıralama ölçütü ile sıralaması en yüksek dokümandan en aza doğru, dizinin ortasından başlayacak şekilde sağ ve sol kutuplara dağıtılır. Bu şekilde sıralaması yapılmış doküman dizisi Şekil 2'de resmedilmiştir.



Şekil 2. Çift Kutuplu Dizilim

Görselde görüldüğü üzere dokümanlar, belirlenen bir sıralama ölçütüne göre dizinin ortasından uçlara doğru dizilmektedirler. Dizme sürecinde sağ ve sol kutupta toplanan doküman sayıları eşitse, dokümanın ekleneceği taraf rastgele belirlenmektedir. Aksi halde, eksik sayıda olan tarafa sıradaki doküman eklenmektedir. Çift kutuplu dizilimin belirlenen bir$R(d)$ sıralama ölçütü ile oluşturulma adımları aşağıda verilmiştir.

Algoritma 3. Çift Kutuplu Dizme

```
SORT DocArray by R(d)
SET Left_Size to 0
SET Right_Size to 0
FOR each Doc in DocArray
    IF Left_Size == Right_Size
        IF Rand() <0.5
            REMOVE Doc from DocArray
            UNSHIFT DocArray with Doc
          SET Left_Size to Left_Size + 1
        ELSE
          SET Right_Size to Right_Size + 1
        ENDIF
    ELSE
        IF Left_Size < Right_Size
            REMOVE Doc from DocArray
            UNSHIFT DocArraywithDoc
          SET Left_Size to Left_Size + 1
        ELSE
          SET Right_Size to Right_Size + 1
          ENDIF
     ENDIF
ENDFOR
RETURN DocArray
```

Çalışmamızda iki farklı sıralama ölçütü kullanılarak çift kutuplu dizilim şemasının dizin sıkıştırmaya olan etkisi ölçülmüştür. Bunlardan ilki, dokümanda geçen ayrık terim sayısı; ikincisi de dokümanda geçen ayrık terimlerin IDF (Inverse Document Frequency) değerlerinin toplamıdır. Toplam doküman sayısının *N* olduğu bir derlemde *t* terimini içeren doküman sayısı *f(t)* ise, terimin IDF değeri aşağıdaki şekilde hesaplanmaktadır:

$$idf(t) = \log\left(\frac{N}{f(t)}\right) \qquad (11)$$

Bu durumda, dokümanlar için kullanılan sıralama ölçütü aşağıdaki şekilde elde edilir:

$$R(d) = \sum_i idf(t_i) \qquad (12)$$

Tanımlanan bu iki sıralama ölçütü kullanılarak çift kutuplu dizilimin terimlerin ortalama bant genişliğini azaltma oranının ölçülmesi için 100000 adet doküman üzerinde deneyler yapılmıştır. Bu dokümanlar, geliştirilen bir veri toplayıcı aracılığıyla internetten çekilen Türkçe haber metinlerinden oluşmaktadır. Terimlerin çıkarılması sürecinde herhangi bir köke indirgeme işlemi yapılmadığından 346003 adet terimden oluşan oldukça seyrek bir terim-doküman matrisi elde edilmiştir. Terim dizinlerinin başlangıç ve bitiş noktaları arasındaki uzaklıkların toplamının en aza indirgenmesi doğrultusunda oluşturulan hedef fonksiyonu aşağıdaki gibidir:

$$min\ F(x) = \sum(d_i - d_j) \qquad (13)$$

Hazırlanan veri setiyle oluşturulan dizinlerin ortalama bant genişliği, 5 farklı doküman dizme şeması kullanılarak (13)'te ifade edilen toplam genişliğin terim sayısına bölünmesiyle hesaplanmıştır. Bu şemalar sırasıyla rastgele dizilim, terim sayısına göre artan (ascending) dizilim, toplam IDF değerine göre artan dizilim, terim sayısına göre çift kutuplu dizilim ve son olarak toplam IDF değerine göre çift kutuplu dizilimdir. Tablo 2'de ortalama dizin genişliği değerleri karşılık gelen dizilim şemalarıyla birlikte verilmiştir.

Tablo 2. Dizilim şemalarına karşılık gelen ortalama bant genişlikleri

Dizilim Şeması	Ortalama Bant Genişliği
Rastgele	35545
Terim Sayısı (Artan)	15638
Toplam IDF (Artan)	15097
Terim Sayısı (Çift Kutuplu)	13100
Toplam IDF (Çift Kutuplu)	12888

Tablodan görüldüğü üzere çift kutuplu dizilim, tekdüze artan dizilime göre terim sayısı ölçütünde %16.2, toplam IDF ölçütünde de %14.6 oranında terim dizinlerinin ortalama bant genişliğini azaltmıştır. Toplam IDF ölçütünü kullanan çift kutuplu dizilim, rastgele dizilime göre %63.7 oranında bant genişliğini azaltarak en iyi sonucu veren dizilim olup terim sayısı ölçütünü kullanan dizilime göre %1.6 oranında iyileştirme sağlamıştır. Bu nedenle, bir sonraki bölümde değinilecek olan tepe tırmanma algoritması bu şema üzerinden uygulanmıştır.

3.2 Tepe Tırmanma

Çift kutuplu dizilim, bant genişliğini başlı başına önemli ölçüde azaltmasının yanı sıra, aynı zamanda dizilen dokümanları sağ ve sol kutup olmak üzere ikiye ayırarak ikili yer değiştirmeye (swap) dayalı tepe tırmanma ile eniyileme yapılmasını mümkün kılmaktadır. Burada hedeflenen, terim dizinlerinin başlangıç ve bitiş noktalarının olabildiğince tek bir kutupta yer almasını sağlayarak yerelliğin artırılmasıdır. Başka bir ifadeyle, ortak terim sayısı minimum düzeyde olan dokümanların zıt kutuplarda bulundurulması amaçlanmaktadır. Bu doğrultuda, çift kutuplu dizilimdeki toplam IDF ölçütüne dayalı sıralama asgari ölçüde bozulacak şekilde, simetrik bir yer değiştirme operatörü tanımlanmıştır. Bu operatör, bir dokümanı, önceden tanımlı bir tolerans aralığında, yalnızca toplam IDF değeri kendisine yakın olan zıt kutuptaki bir doküman ile değiştirebilmektedir. Böylece, N tane elemana sahip çift kutuplu dizilmiş bir D dizisinde yer alan $D[i]$ dokümanının yer değiştirebileceği dokümanlar, belirlenen bir A tolerans değeri ile $D[N - i - 1 \pm A]$ indis aralığında yer alan elemanlarla sınırlandırılmış olur. Bu bakımdan her doküman için, uçlarda bulunan istisnai durumlar hariç, zıt kutupta yer alan $2A$+1adet yer değiştirilebilecek aday bulunmaktadır.

İki doküman arasında yer değiştirme işlemi, bu dokümanlarda bulunan tüm ayrık terimlerin toplam bant genişliğini azalttığı takdirde gerçekleşmektedir. Tolerans aralığında bu şartı sağlayan birden fazla sayıda aday mevcut ise bant genişliğini en çok azaltan doküman seçilmektedir. Derlemde yalnızca bir kere geçen terimler bulundukları dokümanın yer değiştirmesi durumunda toplam bant genişliğini etkilememektedir. Diğer taraftan, derlemde birden fazla sayıda geçen bir terim için bulunduğu dokümanın bir başka dokümanla yer değiştirmesi halinde, dizin genişliğinin ne kadar değiştiğinin bulunabilmesi için üç farklı durumun ayrı ayrı incelenmesi gerekmektedir. Bir t teriminin D dizisinde ilk defa geçtiği doküman indisi i, son defa geçtiği doküman doküman indisi j ise, t terimini içeren k indisine sahip bir dokümanın m indisine sahip başka bir dokümanla yer değiştirmesi durumunda, bant genişliği değişimi aşağıdaki koşullara göre şekillenmektedir:

1. $i<k<j$: Eğer k değeri i ve j arasında kalıyorsa aşağıdaki koşullar incelenmelidir:
    a. $i \leq m \leq j$: Bu durumda t terimine ait dizinin genişliği etkilenmez ve $j - i$ olarak kalır.
    b. $m<i$:Bu durumda t terimine ait dizinin genişliği artarak $j - m$ olur.
    c. $j<m$:Bu durumda t terimine ait dizinin genişliği artarak $m - i$ olur.
2. $i == k$: Eğer k değeri dizinin başlangıç indisi olan i değerine eşitse, yani, yeri değiştirilecek olan doküman t terimini içeren ilk dokümansa, aşağıdaki koşullar incelenmelidir:
    a. $m<i$:Bu durumda t terimine ait dizinin genişliği artarak $j - m$ olur.
    b. $i<m \leq j$: Bu durumda m indisinde bulunan doküman t terimini içeriyorsa, t terimine ait dizinin genişliği etkilenmez ve $j - i$ olarak kalır. Aksi halde, t terimini içeren ilk dokümanın yeri değiştiğinden dizinin yeni başlangıç indisi bulunur ve $\bar{\imath}$ olarak belirlenir. Sonuç olarak dizinin genişliği azalarak $j - \bar{\imath}$olur.
    c. $j<m$: Bu durumda m indisinde bulunan doküman t terimini içeriyorsa, dizinin genişliği artarak $m - i$ olur. Aksi halde, t terimini içeren ilk dokümanın yeri değiştiğinden dizinin yeni başlangıç indisi bulunur ve $\bar{\imath}$ olarak belirlenir. Dizinin genişliği $m - \bar{\imath}$olur ve bu değerlere bağlı olarak eski değerine göre azalmış veya artmış olabilir.
3. $j == k$: Eğer k değeri dizinin bitiş indisi olan j değerine eşitse, yani, yeri değiştirilecek olan doküman t terimini içeren son dokümansa, aşağıdaki koşullar incelenmelidir:
    a. $j<m$: Bu durumda t terimine ait dizinin genişliği artarak $m - i$ olur.

b. $i \leq m < j$: Bu durumda m indisinde bulunan doküman t terimini içeriyorsa, t terimine ait dizinin genişliği etkilenmez ve $j - i$ olarak kalır. Aksi halde, t terimini içeren son dokümanın yeri değiştiğinden dizinin yeni bitiş indisi bulunur ve $\bar{j}$ olarak belirlenir. Sonuç olarak dizinin genişliği azalarak $\bar{j} - i$ olur.

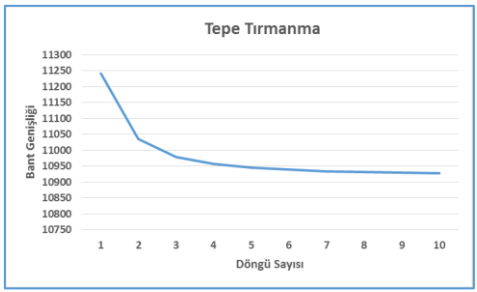
c. $m < i$: Bu durumda m indisinde bulunan doküman t terimini içeriyorsa, dizinin genişliği artarak $j - m$ olur. Aksi halde, t terimini içeren son dokümanın yeri değiştiğinden dizinin yeni bitiş indisi bulunur ve $\bar{j}$ olarak belirlenir. Dizinin genişliği $\bar{j} - m$ olur ve bu değerlere bağlı olarak eski değerine göre azalmış veya artmış olabilir.

Yukarıda belirtilen adımlar uygulandığında ikili yer değiştirmeye uğrayan bir dokümanda bulunan bir terime ait dizin genişliğinin ne kadar arttığı veya azaldığı bilgisi elde edilmektedir. İki doküman arasında gerçekleşen yer değiştirmenin tüm dizinlerin toplam bant genişliğine olan etkisi ise iki dokümanda bulunan tüm ayrık terimlerin dizin genişliği değişimlerinin toplanmasıyla hesaplanmaktadır. İki doküman ancak ve ancak ortaya çıkan safi değere göre toplam bant genişliğinin azalması durumunda yer değiştirebilmektedir. Bu işlem, dizideki her dokümanın belirlenen bir tolerans aralığıile yer değiştirebileceği adayların belirlenmesiyle uygulanır ve bir döngü tamamlanmış olur. Tepe tırmanma süreci, bu döngülerin toplam bant genişliği artık azalmayacak bir noktaya geldiğinde veya değişimi belirlenen bir eşik değerinin altında kaldığında sona erer. Tablo 3'te toplam IDF ölçütüyle çift kutuplu sıralaması yapılmış doküman dizisine farklı tolerans aralıklarıyla uygulanan tepe tırmanmanın ortalama bant genişliğine etkisi gösterilmiştir.

Tablo 3. Tepe tırmanmanın farklı tolerans aralıklarıyla ortalama bant genişliğine etkisi

A	Döngü Sayısı	Ortalama Bant Genişliği
1	10	11203
2	10	11017
3	10	10962
4	10	10927

Tablodan görüldüğü üzere tolerans aralığı arttıkça ortalama bant genişliği daha fazla oranda azalmaktadır. Ancak tepe tırmanma algoritmasının tamamlanma süresi $2A+1$ oranında arttığından A değeri en fazla 4'e kadar çıkarılmış ve 10 döngü sonrası 10927 ortalama bant genişliği elde edilmiştir. Şekil 3'te verilen grafikte, tepe tırmanmanın her döngüde bant genişliğini ne kadar azalttığı gösterilmiştir.



Şekil 3. Tepe tırmanma döngüleriyle elde edilen ortalama bant genişlikleri (A = 4)

Şekilde görüldüğü üzere tepe tırmanmanın, döngü sayısı arttıkça bant genişliğine olan etkisi azalmakta ve yeterli sayıda döngü gerçekleştiği takdirde belirli bir seviyeye yakınsaması beklenmektedir. Bu sebeple, yapılan deneylerde algoritmanın tamamlanma süresini makul seviyede tutmak amacıyla döngü sayısı 10 ile sınırlandırılmıştır. Tablo 4'te tepe tırmanma sonucunda, başlangıç ve bitiş indisleri yalnızca sol veya sağ kutupta yer alan dizinlerin sayısının %23.2 oranında çoğalıp her iki kutupta bulunan dizinlerin sayısının %25.9 oranında azalarak yerelliğin arttığı ve terim dizinlerinin ortalama bant genişliğinin çift kutuplu dizilime göre %15.2 oranında azaldığı gözlenmektedir. Sonuç olarak, çift kutuplu dizilim şemasının tepe tırmanma ile eniyilemesi yapıldığında ortalama bant genişliğinin rastgele dizilime göre %69.2 oranında azaldığı görülmüştür.

Tablo 4. Tepe tırmanmanın yerelliğe etkisi

Dizilim Şeması	Sol Kutup	Sağ Kutup	Her İki Kutup	Bant Genişliği
Toplam IDF (Çift Kutuplu)	90242	92329	163432	12888
Toplam IDF (Çift Kutuplu) + Tepe Tırmanma (A = 4)	115045	109805	121153	10927

4. Değerlendirme

Bu bölümde tepe tırmanma ile eniyilemesi yapılmış çift kutuplu dizilim şemasının dizin aralıklarının sıkıştırılabilirliğine katkısı RBE, vByte ve Elias-δ kodlamaları kullanılarak ölçülmüştür. Bunun yanı sıra, RBE ve vByte kodlama şemaları, ürettikleri sekiz ikililerin Huffman kodlamasıyla ne kadar oranda sıkıştırılabildiği üzerinden karşılaştırılmıştır. Tablo 5'te 100000 doküman ve 346003 terimden oluşan terim-doküman bit eşleminin ters dizinleme sonucu ne kadar alan harcadığı gösterilmiştir.

Tablo 5.Ters dizinlemenin sıkıştırma oranı

Bit Eşlemi	Ters Dizin (32-bit)	Ters Dizin (17-bit)
4.028 GB	26.13 MB	13.88 MB

Ters dizinlerin sabit uzunlukta kod kelimeleriyle temsil edilmesi durumunda 100000 dokümanı temsil edecek minimum uzunluğa sahip kod kelimesi uzunluğu 17'dir ($2^{16} < 100000 < 2^{17}$). Bu durumda, 4.028 GB yer kaplayan bit eşlemi %0.34'üne inerek 13.88 MB yer kaplamaktadır. 17-bit uzunlukta kod kelimeleriyle sayıların temsil edilmesi durumunda aralık kodlaması veya doküman dizmenin sıkıştırma oranına herhangi bir etkisinin olması mümkün değildir. Ancak, aralıkların minimum sabit uzunluğa sahip kodlarla temsil edildiğinde elde edilen sıkıştırma oranı, değişken bit ve sekiz ikili kodlama şemalarının karşılaştırmalı biçimde ne kadar sıkıştırdıklarının görülmesi bakımından önemlidir. Tablo 6'da, aralıkların RBE, vByte ve Elias-δ kodlamalarıyla kodlandıklarında, sırasıyla rastgele dizilim ve tepe tırmanma ile optimize edilmiş çift kutuplu dizilim uygulandığında ne kadar yer kapladıkları verilmiştir. Son satırda geliştirilen doküman dizme şemasının kullanılan kodlama tekniklerinin sıkıştırma oranına ne kadar katkı sağladığı görülmektedir.

Tablo 6.Doküman dizmenin aralık sıkıştırmaya katkısı

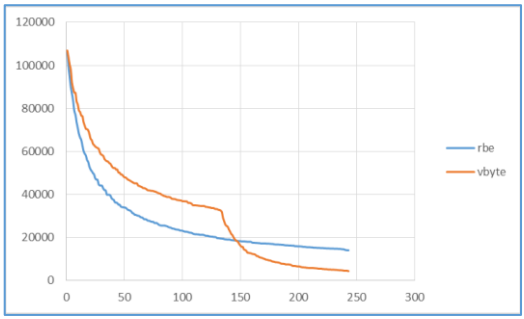
	17-bit	RBE	vByte	Elias-δ
Rastgele Dizilim	13.88 MB	12.26 MB	10.35 MB	10.52 MB
Çift Kutuplu + Tepe Tırmanma	13.88 MB	10.13 MB	9.17 MB	8.50 MB
Katkı	%0	%17.4	%11.4	%19.2

Terim dağılımının oldukça seyrek olduğu bir veri setinin kullanılmış olması, ortalama dizin aralığının, RBE'nin vByte'tan daha iyi sıkıştırması için gereken seviyeden fazla olmasına neden olduğundan vByte kodlaması RBE'ye göre aralıkları daha iyi oranda sıkıştırmıştır. Doküman dizme şemasının RBE'nin sıkıştırma oranına daha fazla katkıda bulunması ise ortalama dizin aralığının azalması nedeniyle beklenen bir durumdur. Bu noktada, RBE ve vByte kodlamaları sonucu ortaya çıkan sekiz ikililere Huffman kodlaması uygulandığında elde edilen sıkıştırma oranları Tablo 7'de verilmiştir.

Tablo 7.Sekiz ikili Huffman kodlaması sonuçları

	RBE + Huffman	vByte + Huffman
Rastgele Dizilim	9.63 MB	9.31 MB
Çift Kutuplu + Tepe Tırmanma	7.89 MB	7.80 MB
Katkı	%18.1	%16.2

RBE'den çıkan sekiz ikililere Huffman kodlaması uygulandığında %22.1, vByte'tan çıkanlara uygulandığında ise %14.9 oranında alan kazancı sağlanmaktadır. Bunun nedeni, RBE sonucu ortaya çıkan sekiz ikililerin, frekansları büyükten küçüğe sıralandığında vByte'a göre daha keskin bir düşüş seyri göstermesidir. Bu durum Şekil 4'te daha açık bir şekilde görselleştirilmiştir. Grafikte, vByte sekiz ikililerinin iki farklı dağılımın birleşimi görünümü sergilemesinin nedeni ilk bit değerinin 'sürdürme biti' (continuation bit) olarak kullanılmasıdır.



Şekil 4. RBE ve vByte sekiz ikili frekans dağılımı

5. Sonuç ve Gelecek Çalışmalar

Bu çalışmada, yeni bir sekiz ikili kodlama yöntemi olan özyinelemeli sekiz ikili kodlama (RBE) şeması incelenmiş ve vByte kodlamasıyla karşılaştırılmıştır. Bunun yanı sıra, doküman dizme problemi kapsamında çok sayıda dokümana ölçeklenebilen çift kutuplu dizilim şeması geliştirilmiş ve bu şemanın dizin aralıklarının sıkıştırılabilirliğine olan katkısı ölçülmüştür. Aynı zamanda, RBE'nin, vByte'a göre hangi koşullar altında daha iyi sıkıştırma performansı gösterdiği irdelenmiş; doküman dizme ve Huffman kodlamasının sıkıştırma oranına olan katkısı mukayeseli bir şekilde gösterilmiştir. Sonuçlar maddeler halinde sıralanacak olursa,

1. RBE, bu çalışmadaki gibi çok seyrek matrislerde vByte'a göre daha düşük sıkıştırma performansı sergilemektedir.
2. RBE, doküman dizme işlemi sonucu artan yerellikten ve sonrasında uygulanan Huffman kodlamasından vByte'a göre sıkıştırma performansı açısından daha çok fayda sağlamaktadır. Bu nedenle, geometrik dağıldığı varsayılan doküman aralık değerlerinin ortalama değeri teorik limit olan 184 değerini geçse bile söz konusu işlemlerin yapılması, seyrek terim-doküman matrislerinde dahi RBE'nin daha iyi sıkıştırma performansı göstermesini sağlayabilir.
3. Doküman dizme kapsamında tepe tırmanmayla eniyilemesi yapılan çift kutuplu dizilim şeması, dizin aralıklarının değişken bit seviyesinde kodlama yapan yöntemlerce sıkıştırılabilirliğini %19'a varan oranda artırmıştır. Yöntemin lineer karmaşıklığa sahip olup çok sayıda dokümana kolaylıkla ölçeklenebilir olması da diğer bir avantajıdır.

En son aşamada Huffman kodlaması uygulanmış sekiz ikililerin hızlı kod çözümü için komut tablolarının [36] kullanılması öngörülmektedir. Kodlanacak sembol sayısının yalnızca 256 adet olması, oluşturulacak komut tablolarının makul seviyede alan harcamasını sağlayacaktır. Huffman kodlaması uygulanmış dizin aralıklarının ters dizinlere geri çevrimi sürecinde sırasıyla gerçekleştirilecek olan komut tabloları aracılığıyla sekiz ikililerin geri edinimi, değişken uzunlukta sekiz ikililerin kod çözümü ve son olarak aralıkların doküman indislerine çevrimi işlemlerinin tamamlanma süresinin ölçülmesi gelecekte yapılacak çalışmalar arasındadır. Ayrıca, tepe tırmanmanın, yüksek tolerans değerleriyle test edilebilmesi için dinamik programlamayla daha verimli hale getirilmesi ve RBE şemasının kod çözme hızının vByte'ın hızıyla karşılaştırılması da bu araştırmanın devamı niteliğinde olacak çalışmalardır.

Kaynakça

[1] G.Graefe and L.Shapiro. Data compression and database performance. In ACM/IEEE-CS Symp. On Applied Computing pages 22 -27, April 1991.

[2] M. Stonebraker, D. J. Abadi, A. Batkin, X. Chen, M. Cherniack, M. Ferreira, E. Lau, A. Lin, S. Madden, E. J. O'Neil, P. E. O'Neil, A. Rasin, N. Tran, and S. B. Zdonik. C-Store: A column-oriented DBMS. In VLDB, pages 553–564, 2005.

[3] D. J. Abadi, S. Madden, and M. Ferreira. Integrating Compression and Execution in Column-Oriented Database Systems. In SIGMOD, 2006.

[4] C. Lin, J. Wang, and Y. Papakonstantinou. Data Compression for Analytics over Large-scale In-memory Column Databases, ACM 2016.

[5] H. Schütze, C. D. Manning, and P. Raghavan. Introduction to Information Retrieval. Vol. 39. Cambridge University Press, 2008.

[6] I. H. Witten, A. Moffat, T. C. Bell. Managing Gigabytes: Compressing and Indexing Documents and Images. San Francisco, CA, USA, 1999.

[7] R. Grossi and J. S. Vitter. Compressed Suffix Arrays and Suffix Trees with Applications to Text Indexing and String Matching. In 32nd ACM Symposium on Theory of Computing, pages 397–406, 2000.

[8] M. A. Martinez-Prieto, N. Brisaboa, R. Canovas, F. Claude, and G. Navarro. Practical compressed string dictionaries. Information Systems, 56:73–108, 2016.

[9] R. Baeza-Yates and B. Ribeiro-Neto. Modern Information Retrieval. Addison-Wesley, 1999.
[10] H. E. Williams,J. Zobel. Compressing Integersfor Fast File Access. Comput.J.42, pp. 193-201, 1999.
[11] P. Elias. Universal codeword sets and representations of the integers. IEEE Transactions on Information Theory, IT-21(2):194–203, 1975.
[12] V. Anh and A. Moffat. Index compression using fixed binary codewords. In Proc. of the 15th Int. Australasian Database Conference, pages 61–67, 2004.
[13] D. A. Huffman. A method for the construction of minimum-redundancy codes. *Proceedings of the IRE* 40.9, pages 1098-1101, 1952.
[14] I. H. Witten, M. N. Radford, and G. C. John. Arithmetic coding for data compression. *Communications of the ACM* 30.6, pages 520-540, 1987.
[15] A. S. Fraenkel and S. T. Klein. Novel compression of sparse bit-strings. Preliminary report. *Combinatorial Algorithms on Words, Volume 12.* NATO ASI Series F, pages 169-183, 1985.
[16] S. W. Golomb. Run-length encoding. IEEE Transactions on Information Theory, vol. IT-12, pp. 399-401, 1966.
[17] R. Gallager, and D. Van Voorhis. Optimal source codes for geometrically distributed integer alphabets. *IEEE Transactions on Information theory* 21.2,pages: 228-230, 1975.
[18] A. Trotman. Compressing inverted files. Information Retrieval, 6.1: 5-19, 2003.
[19] F. Scholer, H.E. Williams, J. Yiannis, J. Zobel. Compressionof Inverted Indexes for Fast Query Evaluation. In SIGIR 2002, pp. 222-229, 2002.
[20] J. Plaisance, N. Kurz, and D. Lemire. Vectorized vbyte decoding. CoRR, 2015.
[21] M. C. Sorkunand C. Özbey. Compression experiments on term-document index. Computer Science and Engineering (UBMK), 2017 International Conference on. IEEE, 2017.
[22] H. Yan, S. Ding, and T. Suel. Inverted index compression and query processing with optimized document ordering. In Proceedings of the 18th international conference on World wide web. ACM, 401–410, 2009.
[23] C. Papadimitriou. The NP-completeness of the bandwidth minimization problem.Computing, vol. 16, pp. 263–270, 1976.
[24] A. Lim, B. Rodrigues, F. Xiao. Integrated genetic algorithm with hill climbing forbandwidth minimization problem. GECCO 2003. LNCS, vol. 2724, pp. 1594–1595, 2003.
[25] E. Rodriguez-Tello, J. K. Hao, J. Torres-Jimenez. An improved simulated annealing algorithm for bandwidth minimization, European Journal of Operational Research 185, pp. 1319–1335, 2008.
[26] R. Martí, M. Laguna, F. Glover and V. Campos.Reducing the bandwidth ofa sparse matrix with tabu search. European Journal of Operational Research 135, pp. 450–459, 2001.
[27] E. Cuthill, J. McKee. Reducing the bandwidth of sparse symmetric matrices. In Proceedings of the 1969 24th National Conference (New York, NY, USA, 1969), ACM '69, ACM, pp. 157–172, 1969.
[28] N. E. Gibbs, W. G. Poole and P. K. Stockmeyer. An algorithm for reducing the bandwidth and profile of a sparse matrix. SIAM Journal on Numerical Analysis 13, pp. 236-250, 1976.
[29] S. W. Sloan. An algorithm for profile and wavefront reduction of sparse matrices. International Journal for Numerical Methods in Engineering 23, 2. 239–251, 1986.
[30] D. Harel and Y. Koren. Graph drawing by high-dimensionalembedding. In Revised Papers from the 10th International Symposium on Graph Drawing. GD '02, Springer-Verlag, pp. 207–219, 2002.
[31] I. Spence andJ. Graef. The determination of the underlying dimensionalityof an empirically obtained matrix of proximities. Multivariate Behavioral Research 9, pp. 331–341, 1974.
[32] Y. Cheng, G. M. Church. Biclustering of expression data. In Ismb, vol. 8, pp. 93–103, 2000.
[33] F. Silvestri. Sorting out the document identifier assignment problem. In Proc. of 29th European Conf. on Information Retrieval, pages 101–112, 2007.
[34] V. Ramaswamy, R. Konow, A. Trotman, J. Degenhardt, and N. Whyte. Document Reordering is Good, Especially for e-Commerce. In Proceedings of the SIGIR 2017 Workshop on eCommerce (ECOM 17), 2017.
[35] S. Büttcher, C. Clarke, and G. Cormack. Information Retrieval: Implementing and Evaluating Search Engines. MIT Press, p. 214, 2010.
[36] Y. Nekritch. Decoding of canonical Huffman codes with look-up tables. In Proc. Conf. Data Compression, p. 566, 2000.

Türkçenin Anlamsal Görev Çözümlemesi
Semantic Role Labeling of Turkish

Gözde Gül Şahin
İTÜ Bilgisayar ve Bilişim Fak.
isguderg@itu.edu.tr

Eşref Adalı
İTÜ Bilgisayar ve Bilişim Fak.
adali@itu.edu.tr

Öz

Bir oluş, sözdizimsel yapıları farklı tümceler ile ifade edilebilir (ör: Ekonomi %5 oranında büyümüştür ve Ekonomideki büyüme %5'tir). Bilgisayarlara, bu farklı biçimlerin aynı anlama denk geldiğini gösterebilmek için, ortak bir anlamsal gösterim dili gerekir. Bu çalışmada, tümce anlamlarını, eylem ve paydaş ikilisiyle göstermeye yarayan "Anlamsal Görev Çözümlemesi" işi Türkçe için gerçeklenmiştir. Bunun için, Türkçe Önerme Veri Tabanı oluşturulmuş, ağaç derlem üzerinde eylem anlamları ve sözcüklerin anlamsal görevleri imece topluluğu tarafından işaretlenmiş ve tüm imlemeler uzmanlar tarafından denetlenmiştir. Derlem imleme kalite ölçütleri kullanılarak etiketleme kalitesi ölçülmüş ve yüksek kaliteli bir Türkçe Önerme Derlemi oluşturulduğu pek çok farklı ölçütle gösterilmiştir. Oluşturulan derlem üzerinde Türkçeye özgü ikili ve ulamsal nitelikler ve Türkçe sözcük vektörlerine dayalı dağıtık niteliklerle lojistik regresyon modelleri eğitilmiş ve böylece yüksek başarımlı bir anlamsal görev çözümleyici gerçeklenmiştir.

***Anahtar Sözcükler**: Türkçe Doğal Dil Anlama, Anlam Bilimi, Anlamsal Görev Çözümleme, Önerme Veri Tabanı, Makine Öğrenmesi*

Abstract

An event can be expressed by sentences with different syntactic realizations (e.g. economy grew by 5% and the growth in the economy was 5%). Computers require a common semantic representation to understand that the different syntactic forms correspond to the same meaning. In this study, we perform Semantic Role Labeling (SRL) task which aims to dissolve the understanding problem into identifying action/event bearing units and their participants. To do SRL, we create the Turkish Proposition Bank and add a semantic annotation layer on top of the Turkish dependency treebank. We present our annotation workflow that harnesses crowd intelligence, and discuss the procedures for ensuring annotation consistency and quality control. We show that the final corpus is of high-quality with various annotator agreement scores. We train logistic regression models that use (1) binary and categorical linguistic features and (2) distributed features based on Turkish word vectors and report a high performing Turkish SRL system.

***Keywords**: Turkish Natural Language Understanding, Semantics, Semantic Role Labeling, Proposition Bank, Turkish PropBank, Machine Learning*

1. Giriş

Türkçenin Anlamsal Görev Çözümlemesine (AGÇ) (Semantic Role Labeling) odaklanılmış olan bu makale "*Türkçenin Önerme Veri Tabanının (Proposition Bank) oluşturulması ve Türkçenin Anlamsal Görev Çözümlemesini gerçekleştiren tez çalışması"* [24] ve ilgili yayınlarından [19,20,21,22,23] derlenerek hazırlanmıştır.

AGÇ, doğal dili anlama işini, tümcelerden eylem içeren birimleri ve bunların ögelerinin çıkarılmasına indirgemektedir. Böylece tümcenin yapısından bağımsız olarak, farklı yapılardaki tümceler için aynı anlamsal gösterim biçimi elde edilmektedir.

AGÇ'yi gerçekleştirebilmek için, makine öğrenmesi yöntemlerini yönlendirmek üzere eylem (predicate) içeren birimlerin (Türkçe için yüklemlerin) anlamlarını ve paydaşlarını diğer bir deyişle ögelerini içeren bir kaynak, diğer bir deyişle bir veri tabanı, gerekmektedir. Bu veri tabanına yayınlarda **Önerme Veri Tabanı** (ÖVT) ya da ***PropBank*** adı

Gönderme ve kabul tarihi: 09.05.2018-11.10.2018
Makale türü: Araştırma

verilmektedir. Böyle bir veri tabanını oluşturmak uzun zaman, büyük bütçe ve çok sayıda dil uzmanı gerektirmektedir. Bu nedenle Türkçe için ÖVT henüz oluşturulamamıştır. Bu makalede, yukarıda değinilen sorun, imece topluluğunu ÖVT'nin oluşturulması sürecine katarak çözülmüştür. Uzman sayısı en az olacak şekilde tasarımı yapılan yeni iş modeli, uzmanlardan yalnızca şu durumlarda yararlanmaktadır:

- ÖVT'nin ilk ve önemli adımı olan anlamsal görev çerçevelerinin oluşturulması,
- Kalite denetim sürecinde belli miktarda soru ve yanıtın elle imlenmesi ve
- İmleyicilerin üzerinde anlaşamadıkları yanıtların doğru olanlarına karar verme aşamasında.

ÖVT'nin oluşturulmasında karşılaşılan diğer zorluklar ise şunlardır:

- Türkçenin eklemeli dil olması, Türkçedeki eklerin çok sayıda olması ve Türkçe sözcüklerin peş peşe çok sayıda ek alması nedeniyle,
- Türkçenin kuramsal olarak sonsuz sayıda eylem içeren sözcük üretebilmesidir.

Bunun için tüm eylem içeren türetilmiş sözcüklerin, kök çerçevesi kullanılarak karşılanmasına karar verilmiştir. Bu yaklaşımla etiketlenen ÖVT'nin yüksek nitelikli olduğu çeşitli imleyici uzlaşması ölçme yöntemleri kullanılarak kanıtlanmıştır.

Çalışmada Türkçe AGÇ'ye uygun makine öğrenmesi yöntemlerinin geliştirilmesi üzerinde durulmuştur. Bu amaçla bir makine öğrenme modeli olan lojistik regresyon sınıflandırıcısı kullanılmıştır. İlk olarak, diğer dillerin anlamsal görev çözümlenmesi için tasarlanmış öznitelikler kullanılmış, ancak başarımlarının yetersiz olduğu gözlemlenmiştir. Bunun nedenleri şöyle açıklanabilir:

- Derlem dışı sözcüklerin çokluğu,
- Eğitim kümesinin küçük olması,
- Eylem ve ögelerinin söz dizimsel farklılıklarının yüksek olması.

Bu özellikler, çıkarılan özniteliklerin seyrek olması nedeniyle istatistiksel sistemin anlamsal görevler hakkındaki kalıpları öğrenememesine neden olmaktadır. Bu sorunları azaltmak amacıyla,

- Türkçeye daha uygun olan biçim bilimine dayalı öznitelikler (özellikle adın durumları),
- Büyük etiketsiz veri kümesinde eğitilmiş sözcük vektörlerine dayalı öznitelikler kullanılmış ve bu özniteliklerin AGÇ'nin başarımını artırdığı gözlemlenmiştir.

Böylece ilk yüksek başarımlı (79.84 F1 puanlı) Türkçe AGÇ sistemi geliştirilmiştir. Deneylerimiz;

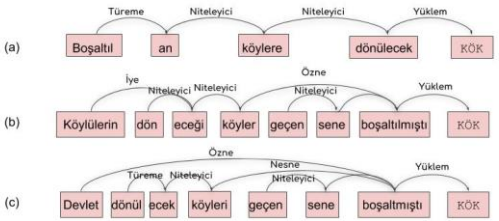
- Biçim anlamsal özniteliklerin Türkçe AGÇ için önemini;
- Tasarlanan sistemin eğitim verisinin yalnızca %60'ını kullanarak, anlamlı sonuçlar üretilebileceğini;
- Bağlılık ağacı ve söz dizimsel sınıf bilgisine dayalı özniteliklerin yokluğunda başarımın azımsanmayacak şekilde düştüğünü ve
- Sürekli özniteliklerin bilgi düzeyleri arasındaki etkileşimi modelleyerek başarıyı artırdığını gösterilmiştir.

İstatistiksel sistemin, sözcüklerin gerçek etiketlerinin bilindiği durumda başarılı olduğu gösterilmişse de, bu etiketlerin bilinmediği durumlarda peş peşe kullanılan doğal dil araçlarının her birinden kaynaklanan hataların birikmesi dolayısıyla başarımı düşmektedir. Bu nedenle, araçlara en alt düzeyde gerek duyan, çift yönlü LSTM birimlerinin alt sözcükleri işlemesine dayanan bir yapay sinir ağı yöntemi önerilmiştir. Eğitilmiş sözcük vektörleri kullanan önceki yöntemlerin tersine, önerilen yöntem alt sözcükleri çeşitli fonksiyonlarla birleştirerek sözcük vektörü oluşturmaktadır. Var olan birleştirme yöntemleri biçimbirimsel farklılıkları göz önüne almamaktadır. Bu nedenle yapım ve çekim eklerinin ayrı ayrı birleştirildiği farklı bir yöntem sunulmuştur. Alt sözcük birimleri ve birleştirme fonksiyonları sistematik olarak analiz edilerek, etkileri ölçülmüştür.

- Yalnızca karakter bilgisi kullanan modellerin, zayıf üretme yetenekli diller için biçim bilimsel bilgi kullanan modellerle benzer sonuçlar verdiği fakat üretim bakımından zengin dillerde biçim bilimsel bilginin başarımı en az %3 F1 puan artırdığı,
- Önerilen birleştirme yönteminin öncekilerden daha başarılı olduğu gösterilmiştir.

Alt sözcüklerin AGÇ için tamamlayıcı özellikleri öğrenip öğrenmediğinin sınanması için birden çok alt sözcük sınıfı çeşitli tekniklerle birleştirilmiştir. Karakter ve karakter üçlülerinin birleştirilmesinin her durumda başarımı artırdığı gözlemlenmiş, ancak biçim bilimsel bilginin karakterle birleştirilmesinin, üretken dillere katkı sağlamadığı görülmüştür. Bu bulgu, karakter modellerinin, söz konusu diller için, zaten biçim bilimsel modellerde olmayan herhangi bir

bilgiyi yakalayamadığını düşündürmektedir. Son



Şekil 2: "köyleri boşaltma" ve "köylere dönme" olaylarının farklı bağlılık ağaçlarıyla gösterimi

olarak, çalışmanın tüm kaynakları araştırmacıların Türkçe dili üzerinde çalışmasını özendirmek amacıyla erişilir biçimde tüm araştırmacılara sunulmuştur.

2.Anlam Sunum Dilleri

Bir oluşu veya olayı açıklamanın birden fazla yolu vardır. Örneğin Şekil-1'de köyleri boşaltma ve köylere dönme ana olaylarının dört farklı şekilde anlatımı olduğu gösterilmiştir. Bu dört tümce, aslında aynı şeyi anlatmaktadır. Bilgisayarlara, bu farklı biçimlerin aslında aynı anlama denk geldiğini gösterebilmek için, bu biçimleri simgelerle göstermek gerekir. En gelişmiş dil araçlarından biri olan bağlılık ayrıştırıcılar bile, Şekil-1'de görüldüğü üzere tüm tümceler için çok farklı ağaçlar üretmektedir. Dolayısıyla yalnızca bağlılık ağaçlarına bakarak, anlamların benzerliğine karar vermek olanaklı değildir. Bu nedenle Anlam Sunum Dilleri (Meaning Representation Languages) oluşturulmuştur. Bu diller ana yapıları açısından birinci dereceden mantık (first order logic) [1], anlamsal ağ (semantic network) ve anlamsal çerçeve (semantic frames) tabanlı olmak üzere üçe ayrılmaktadır. Bu çalışmada, söz diziminden aktarımı, diğerlerine oranla daha kolay olan anlamsal çerçeve tabanlı diller kullanılmıştır.

Çalışmada sıkça kullanılan bazı terimler aşağıda tanımlamıştır:

Baş sözcük: Sözcüğün görüldüğü biçiminin, anlamsal açıklamasına çevrilmesi için kullanılır. Ad ve eylem soylu sözcükler için kullanılır. Örneğin buldum, bulundu, bulunan sözcüklerinin baş sözcüğü bul'dur.

Anlamsal Çerçeve: Her baş sözcük için tanımlıdır ve birden fazla olabilir. Genellikle, baş sözcüğün dildeki tüm farklı anlamları için ayrı bir çerçeve oluşturulur. Her çerçeve bir eylem veya oluşu karşılar. Çerçevede eylemin tanımı verilir ve eylemin sıklıkla görülen paydaşları/ögeleri, anlamsal görevleriyle beraber tanımlanır.

Sözce: Baş sözcüklerin sıralı listesinden oluşan eser. Her baş sözcüğün anlamsal çerçevelerini içerir.

Anlamsal Görev Çözümlemesi (AGÇ): Anlamsal görev çözümlemesi, sırasıyla;

1- Tümcedeki eylem nitelikli sözcüklerin saptanması,
2- Bu sözcüklerin sözce içerisindeki uygun anlamsal çerçeveyle eşleştirilmesi,
3- Eylemin paydaşlarının saptanması ve
4- Paydaşlara anlamsal görev atanması

işlerinin toplamına denir.

Yüklem: Bu bölüm boyunca eylem içeren tüm sözcüklere yüklem denilecektir. Yüklemler, yüklem (bul, yaptım, edildi vb.), yüklemden türemiş ad soylu sözcük (bulunmuş, yapmak, edilen vb.) ya da ad soylu sözcüklerden türemiş eylemler (havalandırdı, kalabalıklaştı vb.) olabilir.

Farklı anlamsal çerçeve kuramları bulunmaktadır. Anlamsal görevler ve çerçevenin içeriği, kullanılan kurama göre farklılık gösterir. Bunların başlıcaları FrameNet (FN) [2], VerbNet (VN) [3], Abstract Meaning Representation (AMR) [4] ve PropBank (PB) [5]'tir. PropBank ya da diğer adıyla Önerme Veri Tabanı, aşağıdaki üstünlükleri nedeniyle bu çalışma için seçilmiştir:

Genelleştirilmiş anlamsal görev tanımlarına sahiptir. FN ve VN ise neredeyse her eyleme özel anlamsal görev tanımlar. Bu durum kaynağın oluşturulmasını ve ileride görevlerin çözümlenmesini zorlaştırır.

Esnekliği nedeniyle, yayınlardaki en çok kullanılan kuramdır. Öyle ki, birçok farklı dil ailesinden dil için (İngilizce, Hintçe, Çince, Arapça, Fince, Portekizce) oluşturulmuş [6,7,8,9,10]; AGÇ yarışmalarının yapıldığı konferansların tercihi olmuş ve yeni oluşturulan (AMR gibi) kuramların da altyapısını oluşturmuştur [11].

Şekil-1'e geri dönecek olursak, ÖVT anlamsal çerçeveleri kullanarak şekildeki dört farklı tümce için Çizelge.1'de gösterilen iki anlamsal çerçeveyle gösterilecektir. Böylelikle aynı anlamdaki tümcelerin,

farklı söz dizimi yapılarına karşın, aynı anlam gösterimleri olacaktır.

Çizelge-1: Ortak Paydaşlar ve Anlamsal Görevleri

Boşal.01: Boş duruma gelmek	Dön.02: Geri gelmek
Boşalan:köyler	Dönülen yer: köyler

Ortak paydaşlara ek olarak, fazladan bilgi içeren tümceler için de bilgiler yine paydaşlar ve anlamsal görevleriyle tanımlanacaktır. Örneğin Şekil-1 içindeki (b) tümlecinde boşalma yükleminin paydaşı olan geçen sene, eylemin olduğu zaman anlamsal göreviyle etiketlenecektir.

2.1 Sorunun Tanımı

Bu çalışmadan önce Türkçe için anlam sunum dilleri kullanılarak hazırlanmış herhangi bir kaynak yoktur. Dolayısıyla AGÇ'nin gerçekleştirilebileceği bir derlem de mevcut değildir. Yayınlarda bu kaynakların oluşturulması için kullanılan genel yaklaşım şu şekildedir. Önce dil uzmanları elle baş sözcükler için anlamsal çerçeveleri tanımlarlar. Sonra, derlemdeki tüm eylemler ve paydaşları yine dil uzmanları tarafından elle tek tek saptanıp, anlamsal görevleriyle etiketlenirler. Tüm bu işlemler uzmanlar tarafından elle yapıldığı için büyük bütçe, uzun zaman ve çok dil uzmanı gerektirir. Diğer bir seçenek yöntem de, kaynakları bol olan dillerden, kaynakları kısıtlı dillere etiket aktarmaktır [12]. Fakat aktarma sırasında, çeviri yanlışları, koşut derlem eksikliği, yüklem uyuşmazlığı ve hizalama sorunları nedeniyle zorluklar yaşanmaktadır. Dolayısıyla iki yöntem de Türkçe için uygun değildir [13].

Yukarıdaki sorunlar göz önüne alındığında, uzmanların en az sayıda kullanılacağı, ana dili Türkçe olan fakat dil uzmanlığı olmayan ve imeci olarak adlandırdığımız kişilerden ise en yüksek katkıyı sağlayabileceğimiz bir iş modeline gerek olduğu görülmektedir. Böylelikle, küçük bütçe, kısa zaman ve az sayıda dil uzmanıyla, kaliteli bir veri tabanı oluşturulabileceği değerlendirilmiştir. Çalışmada izlenen iş modeli Şekil-2'de gösterilmiştir. Bu şekilde, uzmanlardan yalnızca yüksek düzeyde dil bilgisi gerektiren çerçeveleme ve imecilerin yanlışlarını düzeltme aşamalarında yararlanılacaktır. Eylem anlamını ve sözcüklerin anlamsal görevlerini seçenekler arasından seçme işlemleri için imecilerden yararlanılacaktır.

Son olarak, Şekil-2'de oluşturulan etiketli derlemden makine öğrenmesi yöntemlerinin eğitilmesi gerekmektedir. Yayınlarda sıklıkla kullanılan makine öğrenmesi yöntemleri genellikle İngilizce, Çince gibi biçim bilimsel açıdan fakir fakat eğitim veri boyutu bakımından zengin diller için geliştirildiğinden Türkçe AGÇ'ye uygun değildir. Bu nedenle çalışmada

- Türkçeye uygun kullanışlı, yüksek kaliteli ve geniş kapsamlı bir ÖVT oluşturulmuştur.
- Türkçe tümcelerin anlamsal görev çözümlemesini yapan yüksek başarımlı bir yöntem geliştirilmiştir.

3 Türkçe Önerme Veri Tabanının Oluşturulması

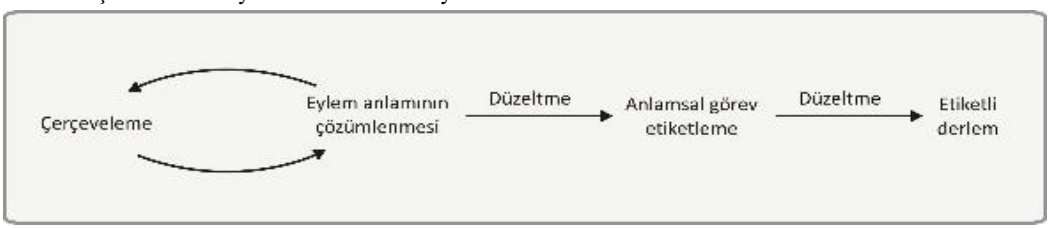
ÖVT temel olarak üç aşamadan oluşmaktadır:

1- Çerçeveleme (Framing),
2- Derlemin, eylem anlamları ve anlamsal görevlerle etiketlenmesi,
3- Anlamsal Görev Çözümlemesi

3.1 Çerçeveleme

Eylemlerin çerçevelenmesi, eylemlerin seçilen anlamsal çerçeve kuramına uygun olarak, tüm anlamlarının tanımlanması ve her anlam için gerekli anlamsal rollerin eklenmesinden oluşmaktadır. Anlamsal çerçevelerin oluşturulması, kullanılan anlamsal çerçeve kuramına bağlı olarak az da olsa değişiklik gösterebilir. Çerçevelenecek eylemlere karar verdikten sonra, ÖVT kuramına göre uzmanlar tarafından izlenen adımlar aşağıdaki gibi özetlenebilir:

1. Çerçevelenmek istenen eylemi içeren çok sayıda tümcenin incelenmesi,



Şekil-3: Önerme Veri Tabanını oluşturma iş akışı

2. Tümcelerde sıkça karşılaşılan farklı anlamların saptanması,
3. Her anlam için, sık karşılaşılan paydaşların belirlenmesi; bu ortak paydaşların sıfırdan başlayarak önerme veri tabanı etiketleme kılavuzuna göre sırayla belirlenmesi.
4. Her anlamsal çerçeve için etiketli örnek tümcelerin oluşturulması.

Çalışma eylemi için oluşturulan üç farklı çerçeve, örnek olarak Şekil-3'te gösterilmiştir.

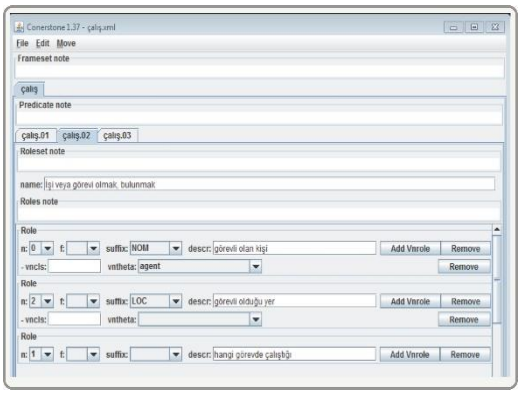
Roleset id:	çalış.01	emek harcamak
Roles:		
Arg0:	*emek harcayan kişi*	NOM
Example:		
	Çalışan ilerler, yerinde kalmaz	
Roleset id:	çalış.02	İşi veya görevi olmak, bulunmak
Roles:		
Arg0:	*görevli olan kişi*	NOM
Arg1:	*hangi görevde çalıştığı*	NOM
Arg2:	*görevli olduğu yer*	LOC
Example:		
	Artık diğer otellerde kaç kişi çalışıyor hesaplayın.	
	Arg0: kaç kişi	
	Arg2: diğer otellerde	
	ArgM-DIS: Artık	
Roleset id:	çalış.03	Bir şeyi öğrenmek ya da yapmak için uğraşmak
Roles:		
Arg0:	*emek harcayan kişi*	NOM
Arg2:	*emek harcadığı şey*	DAT
Example:		
	Üç senedir piyano çalmaya çalışıyor.	
	Arg2: piyano çalmaya	
	ArgM-TMP: Üç senedir	

Şekil-4: Çalış eylemi için oluşturulan örnek çerçeveler

Çerçeveleme işlemi sırasında ele alınan konular ve gerçekleştirilen çalışmalar aşağıda açıklanmıştır:

Araç: Türkçede anlamsal görevler, adın durumlarıyla yakından ilişkilidir. Bu nedenle çerçevelerin oluşturulması için açık kaynak kodlu bir araç [14], adın durumları da çerçeveye eklenecek biçimde zenginleştirilerek kullanılmıştır. Bu araç Şekil-4'te gösterilmiştir.

Eylem ve Anlamlarına Karar Verilmesi: Diğer dillere ilişkin önerme veri tabanlarında herhangi bir eylemin çerçevelenmesi gerektiğine karar vermek gerekli değildir. Bunun nedeni derlemlerin yeteri kadar büyük olması ve derlemdeki eylemlerin kapsamının geniş olmasıdır. Türkçe derlem boyut olarak küçük olduğundan kapsamlı bir veri tabanı için TDK tarafından sağlanan tüm eylem kökleri çerçevelenme adayı olarak belirlenmiştir. TDK'nın listesinde bulunan 759 eylem köklerinin kullanımları, TNC üzerinde sorgulanarak, az kullanılan eylemler saptanmış ve elenmiştir. Sonuç



Şekil-5: Çerçeveleme aracı

olarak ilk aşamada yalnızca 385 eylem kökü çerçevelenmek üzere seçilmiştir. Eylem için kaç farklı anlam çerçevesi tanımlanacağına karar vermek için kullanılan ilke şöyledir:

- Farklı anlamsal çerçevelerin farklı sayıda paydaşları olmalıdır,
- Aynı sayıda paydaşları olduğu durumda da paydaşların anlamsal görevleri birbirinden farklı olmalıdır.

Bu nedenle bir anlamsal çerçeve genellikle birden fazla TDK anlamının birleştirilmiş biçimi gibidir. Yabancı dillerden alınmış eylemler ve deyimler için daha ayrıntılı bir karar aşaması gerekmektedir.

Paydaşlar ve Anlamsal Görevleri: Anlamsal çerçevelerdeki paydaşlara karar verilirken de benzer şekilde, uzmanlar tarafından belli bir sıklığın üzerinde ortak olarak kullanılan paydaşlar belirlenmiş ve bunlar sıfırdan başlayarak çerçeveye eklenmiştir. Paydaşlar argüman olarak da anılmaktadır. İki tür paydaş tanımlanır:

- İlki eyleme özel, eylemin numaralı paydaşlarıdır;
- İkinci ise, eylemler tarafından paylaşılan geçici görevlerde kullanılanlardır.

Paydaş numaraları her zaman aynı anlamsal görevi göstermek için kullanılmasa da genellikle Çizelge-2'deki gibi kullanılırlar. Tanımlanan geçici anlamsal görev listesi de Çizelge-3'te verilmiştir.

Çizelge-2: Numaralı Paydaşların Genellikle Eşleştirildiği Konusal Görevler

Paydaş	Konusal Görev
Arg0 – P0	yapıcı, kılıcı, deneyimleyen, hisseden
Arg1 – P1	eylemden birebir etkilenen nesne/kişi, konu
Arg2 – P2	yararlanan, enstrüman/araç-gereç, alıcı
Arg3 – P3	kaynak, eylemden yararlanan, enstrüman/araç-gereç
Arg4 – P4	hedef, varış yeri

Çizelge-3: Geçici Anlamsal Görev Kodları ve Açıklamaları

Kodu	Açıklaması
A	Ettirgen eylemlerdeki yaptıran, ettiren vb
COM	Kiminle (Kardeşimle, NATOyla, onlarla)
LOC	Nerede (mahallede, konuşmasında, hayalinde)
DIR	İzlediği yol (patikadan)
GOL	Amacı, bitiş noktası (Eve, odaya vb.) ya da faydalanan (annem için, arkadaşıma vb)
MNR	Nasıl (hızlıca, güzel, yavaş, yapıp, koşup vb)
EXT	Miktarı (yüzde elli), (az, çok, biraz), (benden fazla) vb.
PRD	Yan cümlecik (..olarak, ... olmak üzere vb...)
CAU	Nedeni ya da kaynağı (yüzünden, onun için, dağdan vb)
DIS	Tümce başındaki Bağlaç (Ayrıca, Fakat, Buna rağmen vb.) ya da Seslenme (Allahım, duy sesimi)
NEG	Olumsuzluk anlamı ekleyici (Hiçbir zaman, asla, değil, yok, hiç)
LVB	Yardımcı eylem elemanı (mezun olmak'taki mezun, hayal etmek'teki hayal vb...)
TMP	Ne Zaman (Eylül, Pazartesi), Ne Sıklıkla (her zaman, bazen), Kaçıncı (ilk, son) ya da Ne kadarlığına (bir aylığına)
ADV	Tüm tümceyi etkileyen, diğer tanımlara uymayan zarflar (Mutlaka, muhtemelen vb...)
TWO	Eylem ikilemesi (yapıp yapıp, bakıla bakıla, ister istemez, olursan ol, vb...)
INS	Ne ile (uçakla, gözleriyle, çekiçle vb...)

Ad soylu sözcüklerden türeyen eylemler: Bu sözcüklerin kökü ad soyludur ve sayılarının çok olması nedeniyle adların baş sözcük olarak kullanılması uygun değildir. Bu nedenle, bu sözcükleri göstermek için üç farklı ek-baş sözcüğü tanımlanmıştır: xlA, xlAş ve xlAn. Burada x, ad soylu sözcüğe karşılık gelirken, A sembolü a veya e harfi için kullanılır. Örneğin sessizleşmek eyleminde x, "sessiz"'i sembolize ederken, lAş, "leş" ekini belirtir. xlA ve xlAn ile temsil edilen eylemlere örnek olarak da "yara-la" ve "yara-lan" eylemleri verilebilir.

İstatistikler: Çizelge-4'te kaç baş sözcüğün, kaç farklı anlamsal çerçeveye sahip olduğuna ilişkin istatistik değerler verilmiştir. Çizelge-4'e göre toplam 773 baş sözcük için, 1285 farklı anlamsal çerçeve oluşturulmuşur. Baş sözcüklerin yaklaşık %75'i (583/773) tek çerçeveye sahiptir. Baş sözcükler için ortalama çerçeve sayısı ise 1,66'dır. Çizelge-5'te ise, numaralı paydaş sayıları ve baş sözcük sayıları istatistiği verilmiştir. Buna göre, bir çerçevedeki ortalama numaralı paydaş sayısı 2,11'dir.

Çizelge-4: Çerçeve sayısına karşılıkgelen baş sözcük sayısı

Çerçeve sayısı	Baş sözcük sayısı	Çerçeve sayısı	Başsözcük sayısı
1	583	9	2
2	113	11	1
3	40	12	1
4	10	13	2
5	6	18	2
6	4	26	2
7	2	61	1
8	4		

Çizelge-5: Numaralı paydaş–Baş sözcük sayıları

1 numaralı	2 numaralı	3 numaralı	4 numaralı	5 numaralı	6 numaralı
210	750	303	17	4	1

3.2 Derlemin, Eylem Anlamları ve Anlamsal Görevlerle Etiketlenmesi

Tümce içerisinde geçen eylemlerin hangi anlamda kullanıldığını belirlemek ya da sözcüğün eylemle ilişkisini belirlemek için dil uzmanlarına gerek yoktur. Bu nedenle, bu iş büyük oranda ana dili Türkçe olan fakat özel bir dil uzmanlığı olmayan insanlar tarafından gerçekleştirilmiştir. Bunun için imece topluluklarına belli bir ücret karşılığında etiketleme gibi basit işler yaptırılabilen, "crowdsourcing" platformları kullanılmıştır.

İmece yöntemiyle imleme yapılırken dikkat edilmesi gereken bazı noktalar vardır. Bunların en önemlisi imleme kalitesini belli bir seviyenin üzerinde tutabilmektir. Bu çalışmada kalite kontrolü şu üç ilkeyle sağlanmıştır:

- İmeciler, gerçek imlemeye başlamadan önce kısa sınava çekilmişlerdir. Sınavı geçemeyenler imece topluluğuna alınmazlar.
- Gerçek işe başladıktan sonra da imecilerin başarımları sürekli olarak ölçülür. Başarımları belli bir eşik değerinin altındaysa topluluktan çıkarılırlar.
- Yanıldıkları sorulardan sonra, doğru yanıt ve açıklamasını görürler. Böylelikle aynı yanlışı yinelememeleri beklenir.

Bu ilkeler, işi imece topluluklarına teslim etmeden önce, toplam soruların %10'luk kısmının sınama sorusu olarak hazırlanması ile sınanmıştır.

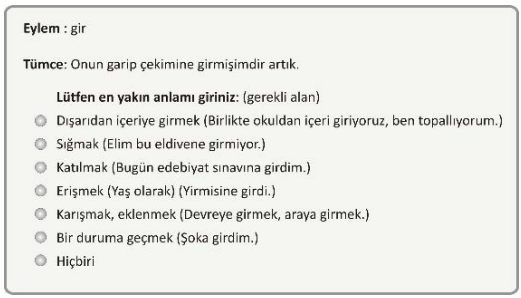
Hem eylem anlamlarının etiketlenmesi hem de anlamsal görevlerin etiketlenmesi işlerinde benzer kurallar kullanılmıştır. Bu kurallar şöyledir:

- Her soru, en az üç farklı kişi tarafından cevaplanmıştır.
- Her sayfada biri sınama sorusu olmak üzere, beş soru gösterilmiştir.
- Sayfa başına 5 USD senti ödenmiştir.
- İmecilerin, göreve katılabilmeleri için ana dillerinin Türkçe olması gerekmektedir
- Başarı eşik düzeyi %70 olarak belirlenmiştir. Katılımcıların başarı (güvenilirlik) düzeyleri doğru ve yanlış yanıtladıkları sınama sorularının sayılarına göre ölçülmektedir.

Eylemin anlamını imlemek için katılımcılara gösterilen arayüz Şekil-5'te gösterilmiştir. Önce imlenmesi istenen eylem ve kullanıldığı tümce verilir ve sonra imleyicilerden eylemin tümcedeki anlamına en yakın anlam tanımını seçmesi istenir. Bir önceki kesimde tanımlanan önerme veri tabanından eyleme karşılık gelen baş sözcük için hazırlanmış tüm anlam çerçeveleri ve örnek tümceler, kullanıcının anlayacağı biçime çevrilir ve seçenek olarak gösterilir. Son olarak, anlam çerçevesinin bulunmadığı durumlar için önlem olarak "Hiçbiri" seçeneği eklenir.

Eylem anlamını imleme işlemi sonucunda, 68 saatte toplam 5855 eylem etiketlenmiş (yalnızca bir anlamsal çerçeveye sahip eylemler işe katılmamıştır); 39 ilden 100'den fazla katılımcı işe katılmış; toplamda 277 USD ödenmiştir.

İş bitiminde, katılımcıların üzerinde uzlaşamadığı veya "Hiçbiri" seçeneğini seçtiği sorular uzmanlar tarafından incelenmiş ve nedenleri şöyle açıklanmıştır:



Şekil-5: Eylem anlamı işaretleme arayüzü

- Biçim bilimsel çözümleme yanlışından kaynaklanabilir. Örneğin sokulma eyleminin baş sözcüğünün sok olarak saptanması ve katılımcılara yanlış anlamsal çerçeveler gösterilmesi,
- Önerme veri tabanında olmayan anlamsal çerçeveye sahip eylemlerin varlığından kaynaklanabilir,
  - Değişmece kullanımların neden olduğu karışıklıklardan olabilir.

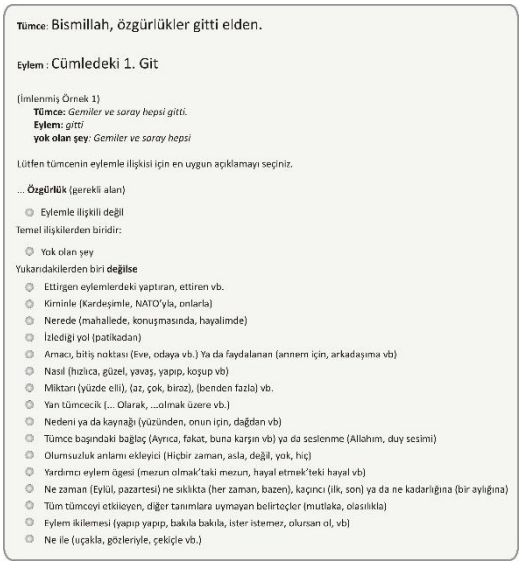
Bu yanlışlar uzmanlar tarafından düzeltilmiş, gerekli anlamsal çerçeveler önerme veri tabanına eklenmiştir.

Sözcüklerin anlamsal görevleri ile etiketlenmesi için hazırlanan arayüz de Şekil-6'da gösterilmiştir.

Katılımcılara öncelikle etiketlenmesi istenen tümce, önceden anlamını imlediğimiz eylem ve imlenmiş bir örnek tümce gösterilir. Sonra da anlamsal görevi bulunmak istenen sözcük gösterilir. Katılımcılar öncelikle sözcüğün eylemle ilişkili olup olmadığına karar verir. Daha sonraki aşamada, öncelik eyleme özgü temel numaralı paydaşlardan biri olup olmadığına karar vermektedir. Eğer temel görevlerden birinde değilse, eylemlerin paylaştığı geçici görevlerden en uygununu seçmeleri beklenir.

Çalışmanın sonucunda, 276 saatte, 351 USD ödenerek 20060 adet paydaş, 400'den fazla katılımcı tarafından anlamsal görevleriyle etiketlenmiştir.

Bir önceki işe benzer şekilde, katılımcıların uzlaşamadıkları soruların yanıtları uzmanlar tarafından düzeltilmiştir. Son olarak derlemin etiketleme kalitesini ölçmek için çeşitli uzlaşma ölçütleri kullanılarak, uzlaşma oranları ölçülmüştür.



Şekil-6: Sözcüklerin anlamsal görevlerinin işaretlenmesi örneği

Uzman-İmleyici Uzlaşması: Bu uzlaşma değeri, uzmanlar tarafından hazırlanan sınama soruları üzerinde Cohen'in Kappa Ölçütü kullanılarak 0,936 olarak ölçülmüştür.

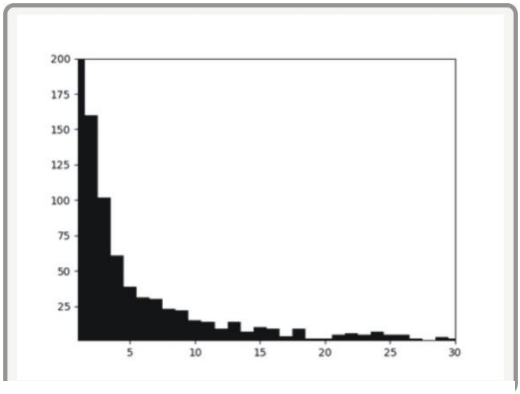
İmleyicilerin kendi aralarındaki uzlaşma: Bu uzlaşma, sınama soruları dışında kalan 18000 soru için Fleiss'ın Kappa Ölçütü ile 0,65 olarak ölçülmüştür. Bunun dışında tüm etiketli derlem üzerinde bulma, tutturma ve F1 değerleri sırasıyla 0,906, 0,908 ve 0,907 olarak hesaplanmıştır. Anlamsal görevler için ayrı ayrı uzlaşma değerleri ölçülmüş ve eyleme özgü numaralı paydaşların etiketlenmesinde uzlaşma düzeyinin daha yüksek olduğu saptanmıştır. Daha sonra anlamsal görevlerin hata matrisi hesaplanmış ve en çok karıştırılan görevlerin numaralı ve geçici görevler olduğu gözlenmiştir. Bunun nedeni, bazı numaralı görevlerin tanımının (git.01 - Arg2: gidilen yer), geçici görevlerle (AM-GOL: hedef) benzer tanımlara sahip olmasıdır. Katılımcılara, önceliğin temel görevler olduğu söylenmişse de bazı durumlarda yönergelere uymadıkları görülmüştür.

Derlem İstatistikleri: Tüm anlamsal katman etiketlemeleri, İTÜ-ODTÜ-SABANCI Ağaç Yapılı Derlemi (IMST) üzerinde yapılmıştır [15,16,17,18]. Eğitim, geliştirme ve sınama parçaları için tüm değerler Çizelge-6'da verilmiştir.

Çizelge-6: IMST Üzerindeki Anlamsal Katman İşaretlemeleri İstatistikleri

	Eğitim	Geliştirme	Sınama	Toplam
Tümce sayısı	3947	844	842	5633
Sözcük sayısı	39444	8627	8330	56401
Andaç sayısı	44034	9687	9337	63058
Eylem sayısı	8151	1834	1757	11742
Başsözcük türü sayısı	634	356	333	685
Çerçeve türü sayısı	960	504	506	1052
Paydaş sayısı	14778	3241	3180	21199

Şekil-7'de verilen anlamsal çerçeve histogramına göre, çerçevelerin büyük çoğunluğu eğitim verisinde 5'ten az defa görülmüştür.



Şekil-7: Eğitim seti üzerinden çıkarılan anlamsal çerçeve histogramı

3.3 Anlamsal Görev Çözümlemesi

Anlamsal Görev Çözümlemesi (AGÇ) için izlenen yöntem genellikle AGÇ'yi önceki kesimde anlatılan 4 bölüme ayırıp, her biri için birbirinden bağımsız yerel sınıflandırıcılar eğitmektir. Bu bölümler sırasıyla

- Eylemi Saptama (ES),
- Eylem Anlamı Atama (EA),
- Paydaş Tespiti (PT),
- Anlamsal Görev Atama (AGA)'dır.

Geleneksel olarak derlemle beraber eylemler de verildiğinden, ES parçası genellikle es geçilir. Diğer alt işler için ise, alt işe ve dile özgü nitelikler çıkarılıp, eğitim verisi üzerinde lojistik regresyon sınıflandırıcı

eğitilir. Dolayısıyla geleneksel yöntemler kullanıldığında, başarılı bir AGÇ için en önemli aşama, etkin niteliklerin belirlenmesidir.

Kullanılan Öznitelikler

Çalışmada kullanılan öznitelikler, Çizelge-7 ve Çizelge-8'de kabaca verilmiştir. Çizelge-8'deki sözcük vektörleri, AGÇ'den önce büyük bir Türkçe derlem üzerinden öğrenilmiştir.

Çizelge-7: Kullanılan ayrık nitelikler

Sözcesel	Sözcüğün görülen formu (sözcük), sözcük kökü
Konumsal	Sözcüğün yeri (eylemle arasındaki mesafe gibi)
Biçimbirimsel	Biçimbirimsel analiz sonuçları, çatı ekleri, sözcüğün aldığı ismin halleri ekleri vb.
Sözdizimsel	Sözcük tipi (POS), Bağlılık etiketi, Bağlılık ağacı patikası
Anlamsal	Tahmin edilen eylem anlamsal çerçeve numarası

Çizelge-8: AGÇ tarafından kullanılan Dağıtık/Vektörel nitelikler

Sözcük Vektörü-Paydaş	$\vec{a}$
Sözcük Vektörü-Eylem	$\vec{p}$
Sözcük Vektörü-Bileşim	$\vec{a} + \vec{p}$
Sözcük Vektörü-Ortalama	$\Sigma_j \vec{w_j}$
Sözcük Vektörü-Bağlılık Ağacı Patikası	$\Sigma_{w \in \text{bağlılık ağacı patikası}(a,p)} \vec{w}$

4.Sonuçlar

LR algoritması ve farklı öznitelikler kullanılarak, AGÇ sonuçları, CoNLL-09 "Shared Task" (Yarışma) tarafından katılımcılara dağıtılan F1 değerini hesaplayan eval09.perl betiğiyle hesaplanmıştır. Özet sonuçlar Çizelge-9'da verilmiştir.

Çizelge-9: AGÇ F1 Değerleri

Kullanılan Nitelik Kümesi	Adımlar	EA	PT+AGA	Genel
Temel		81,10	70,32	74,21
+Adın durumları	EA,PT,AGA	82,34	74,99	77,67
+Çatı Ekleri	EA,PT,AGA	82,34	75,68	78,11
+Dağıtık Nitelikler	AGT	82,34	77,36	79,19

Çizelge-9'un ilk satırda gösterilen temel nitelikler, çoğu dil tarafından ortak olarak kullanılan, dile özgü olmayan nitelikleri göstermektedir. Yalnıza bu nitelikler kullanıldığında bile, 75 F1 değerine yakın bir sonuç elde edilmiştir. Başarım üzerindeki en etkili niteliklerden biri adın durumları niteliği olmuştur. Bu da Türkçenin yapısı göz önüne alındığında beklenen bir sonuçtur. Sözcük vektörlerine dayalı nitelikler kullanmak, eğitim verisinin azlığından kaynaklanan, seyrek nitelik verisi sorununun çözümüne katkı sağlamış ve genel sonuçları artırmıştır. Bunlara ek olarak, nitelik takımları ve eğitim verisi boyutlarıyla farklı deneyler tasarlanmış ve aşağıdaki bulgular elde edilmiştir:

- Biçim anlamsal nitelikler (ör: ismin halleri), Türkçe AGÇ için oldukça önemlidir,
- Eylem anlamı atama (EA) başarımı, eğitim verisinin boyutuna PT ve AGT işlerinden daha bağımlıdır. Diğer bir deyişle, daha fazla eğitim verisi sağlandığında, EA başarımının diğerlerine oranla daha hızlı artması beklenmektedir,
- Eğitim verisinin %60'ı ve iyi tasarlanmış niteliklerle, paydaş tespiti ve bunlara anlamsal görev ataması işi kabul edilebilir derecede başarılı bir şekilde gerçeklenebilir,
- Başarılı bir sistem için en azından söz dizimsel seviyeden (özellikle bağlılık ağacından) niteliklere gerek vardır.

Hata Analizi

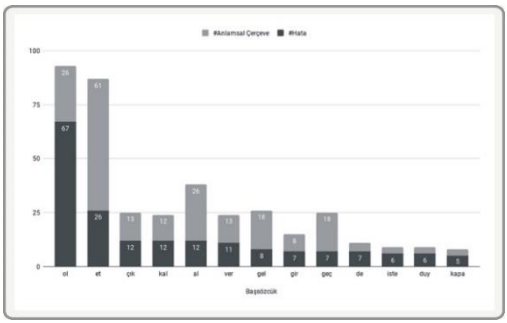
Eylemlere anlam atama için sözcük sınıfları ve yapılan hata oranları hesaplanmıştır. Sözcük türü ile sözcüğün türeyerek aldığı en son tür kastedilmektedir. Örneğin eylem kökünden, ad, önad ve sonra ad türüyorsa, sözcük türü ad olacaktır. Sonuçlar Çizelge-10'da verilmiştir.

Çizelge-10: EA Sözcük Sınıfına Göre Hata İncelemesi

Sözcük Türü	Yanlış	Doğru	Toplam	Hata Oranı
Sıfat	58	201	259	0,22
İsim	68	285	353	0,19
Zarf	16	88	104	0,15
Fiil	161	880	1041	0,15
Toplam	303	1454	1757	0,17

Çizelge-10'a göre, en sık hata yapılan sınıflar önad ve ad türleridir. Baş sözcük türlerine göre daha ayrıntılı inceleme yapıldığında, hataların özellikle ol baş sözcüğü için yapıldığı görülmektedir, ve ol baş

sözcüğü sıklıkla ad ve önad olarak görülmektedir. Ayrıntılı inceleme Şekil-9'da verilmiştir.



Şekil-9: Eylemlere anlam atama için sözcük sınıfları ve yapılan hata oranları hesapları

Paydaşlara anlamsal görev atanması işinin hata incelemesi için, her anlamsal görevin F1 başarısı ayrı ayrı ölçülmüş, sonuçlar Çizelge-11'de verilmiştir. Buradan da görüldüğü üzere, F1 değerleri, bir önceki kesimde anlatılan etiketleme uzlaşma değerleriyle paralellik göstermektedir. Örneğin yardımcı eylem ve eyleme özgü numaralı anlamsal roller (Arg0, Arg1,..,Arg4) makine tarafından yüksek başarıyla bulunurken, uzlaşma değeri düşük olan geçici/paylaşımlı anlamsal görevler (AM-DIS, AM-COM) başarılı bir şekilde bulunamamış ve tutturulamamıştır.

Çizelge-11: Anlamsal Görev Ulamlarına Göre Başarım
B: Bulma, T: Tutturma, n: Paydaş sayısı

Anlamsal Görev	*n*	*B*	*T*	*F1*
AM-LVB	100	0,93	0,85	0,89
A1	1209	0,85	0,89	0,87
A0	567	0,81	0,79	0,80
A4	74	0,76	0,81	0,78
AM-NEG	11	1,00	0,64	0,78
AM-TMP	218	0,77	0,78	0,77
AM-LOC	138	0,71	0,81	0,76
AM-PRD	34	0,91	0,62	0,74
AM-MNR	208	0,70	0,75	0,73
A3	46	0,71	0,70	0,70
A2	201	0,72	0,66	0,69
A-A	23	0,71	0,65	0,68
AM-EXT	67	0,69	0,60	0,64
AM-TWO	13	0,58	0,54	0,56
AM-ADV	38	0,62	0,47	0,54
AM-GOL	57	0,59	0,47	0,52
AM-INS	22	0,53	0,45	0,49
AM-CAU	62	0,47	0,40	0,43
AM-COM	13	0,67	0,31	0,42
AM-DIS	27	0,53	0,30	0,38
AM-DIR	12	0,36	0,33	0,35
C-A1	20	0,00	0,00	0,00

Kaynakça

[1] W.A. Woods, *Semantics for a Question-Answering System*,Ph.D. thesis, Harvard University, (1967)

[2] C.F. Baker, C.J. Fillmore, J.B. Lowe, *The Berkeley Framenet Project, Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*-Volume 1, Association for Computational Linguistics, pp.86–90. (1998)

[3] K.K. Schuler,Verbnet: *a Broad-Coverage, Comprehensive Verb Lexicon.* (2005).

[4] L. Banarescu, C. Bonial, S. Cai, M. Georgescu, K. Griffitt, U. Hermjakob, K. Knight, P. Koehn, M. Palmer, N. Schneider, *Abstract Meaning Representation (Amr) 1.0 Specification, Parsing on Freebase From Question-Answer Pairs.* In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. Seattle: ACL, pp.1533–1544. (2012).

[5] M. Palmer, D. Gildea, P. Kingsbury, *The Proposition Bank: an Annotated Corpus of Semantic Roles*, Computational linguistics, 31(1), 71–106.(2005).

[6] M. Palmer, R. Bhatt, B. Narasimhan, O. Rambow, D. M. Sharma, F. Xia, *Hindi Syntax: Annotating Dependency, Lexical Predicate-Argument Structure, and Phrase Structure*, Proceedings of the 7th International Conference on Natural Language Processing, ICON'09, 261—-268. (2009).

[7] N. Xue, M. Palmer, M. *Adding Semantic Roles to The Chinese Treebank*, Natural Language Engineering, 15(1), 143.(2008).

[8] W. Zaghouani, M. Diab, A. Mansouri, S. Pradhan, M. Palmer, *The revised Arabic PropBank*, 10 Proceedings of the Fourth Linguistic Annotation Workshop, 222–226.(2010).

[9] K. Haverinen, J. Kanerva, S. Kohonen, A. Missila, S. Ojala, T. Viljanen, V. Laippala, F. Ginter, *The Finnish Proposition Bank*, Language Resources and Evaluation, 49(4), 907–926.(2015).

[10] M.S. Duran, S.M. Aluísio, *Propbank-Br: a Brazilian Treebank Annotated With Semantic Role Labels*, LREC. (2012).

[11] J. May, SemEval-2016 *Task 8: Meaning Representation Parsing*, Proceedings of SemEval, 1063–1073. (2016).

[12] A. Akbik, I. chiticariu, M. Danilevsky, Y. Li, S. Vaithyanathan, H. Zhu, *Generating High Quality Proposition Banks for Multilingual Semantic Role Labeling*, Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics,

Association for Computational, Linguistics, Beijing, China, pp.397–407. 111(2015).

[13] K. Oflazer, I.D. El-Kahlout, *Exploring Different Representational Units in English-to-Turkish Statistical Machine Translation,* Proceedings of the Second Workshop on Statistical Machine, (2007).

[14] J.D. Choi, C. Bonial, M. Palmer, *Propbank Frameset Annotation Guidelines Using a Dedicated Editor,* Cornerstone., LREC.(2010).

[15] N. B. Atalay, K. Oflazer, B. Say *The Annotation Process in the Turkish Treebank.* In Proceedings of 4th International Workshop on Linguistically Interpreted Corpora, LINC at EACL 2003, Budapest, Hungary, April 13-14, 2003.

[16] U. Sulubacak, G. Eryiğit. *Implementing Universal Dependency, Morphology and Multiword Expression Annotation Standards for Turkish Language Processing.* Turkish Journal of Electrical Engineering Computer Sciences pages 1–23. 2018

[17] U. Sulubacak, T. Pamay, G. Eryiğit. *IMST: A Revisited Turkish Dependency Treebank.* In Proceedings of the 1st International Conference on Turkic Computational Linguistics (TurCLing) at CICLing, Konya, Turkey, 2016.

[18] K. Oflazer, B. Say, D. Z. Hakkani-Tür, G. Tür. *Building a Turkish treebank. In Treebanks,* Springer, pages 261–277. 2003.

[19] G. G. Şahin, M. Steedman. *Character-Level Models versus Morphology in Semantic Role Labeling.* In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15 - July 20. Long Papers. 2018

[20] G. G. Şahin, E. Adalı. *Annotation of semantic roles for the Turkish Proposition Bank*. Language Resources and Evaluation pages 1–34. 2017

[21] G. G. Şahin, E. Adalı. *Verb Sense Annotation for Turkish PropBank via Crowdsourcing.* In Computational Linguistics and Intelligent Text Processing - 17th International Conference, CICLing 2016, Konya, Turkey, April 3-9, 2016, Revised Selected Papers, Part I. pages 496–506. 2016.

[22] G. G. Şahin. *Framing of Verbs for Turkish PropBank.* In In Proceedings of 1st International Conference on Turkic Computational Linguistics, TurCLing. 2016.

[23] G. G. İşgüder, E. Adalı *Using morphosemantic information in construction of a pilot lexical semantic resource for Turkish.* Proceedings of Workshop on Lexical and Grammatical Resources for Language Processing. 2014.

[24] G. G. İşgüder, *Building of Turkish Propbank and Semantic Role Labeling of Turkish,* Doktora Tezi, İTÜ Fen Bilimleri Ens. 2018

Türkçe Ders Metinleri İçin Özelleştirilmiş Bir Varlık İsmi Tanıma Yapısı

A Named Entity Recognition Structure Specialized for Turkish Lecture Notes

Önder Can SARI[1], Özlem AKTAŞ[2]
[1]Dokuz Eylül Üniversitesi, Bilgisayar Mühendisliği Anabilim Dalı, İzmir, Türkiye,
onder.sari@ceng.deu.edu.tr
[2]Dokuz Eylül Üniversitesi, Bilgisayar Mühendisliği Bölümü, İzmir, Türkiye
ozlem@cs.deu.edu.tr

Öz

Varlık ismi tanıma (VİT); doğal dil işleme ve metin madenciliği alanlarının kapsamında yer alan bir bilgi çıkarımı görevidir. Kapsam ve kullanılan metotlar açısından, çalışmalar arasında farklılıklar görülse de temel olarak, bir metin içerisindeki kişi, yer, kurum-kuruluş vb. belirten ifadelerin doğru şekilde tespit edilmesini hedefler. Bu çalışmada, Türkçe yazılmış ders metinleri (tarih ve coğrafya alanlarında) için bir VİT yapısı geliştirilmiştir. Tek başına ele aldığımızda bu yapı, bir bilgi çıkarımı görevi doğrultusunda özelleştirilmiş bir projedir. Bunun yanı sıra çalışmanın eğitimsel bir değeri de vardır; çünkü sistemden beklenen sonuç, verilen ders metninin içeriğinden anlamlı kelime ya da kelime grupları bulunmasıdır ki; bu da farklı dersler ya da ders konuları için terimler sözlüğü yapıları oluşturmak için kullanılabilir. Oluşturulan sözlüklerin, bir ders metninin içeriğindeki soru değeri taşıyabilecek ifadelerin tespitine ve sınav hazırlama sürecine yardımcı olması hedeflenmektedir. Bu makalede, VİT görevi ve görevin kapsamı hakkında genel bilgi verilmiş; alanda yapılmış önceki çalışmalardan bahsedilmiş; bu çalışma doğrultusunda geliştirilen sistem tanıtılmış; sistemin başarısı, yapılan deney sonuçları üzerinden değerlendirilmiş ve geliştirme-iyileştirme olanakları hakkında yorumlar paylaşılmıştır.

Gönderme ve kabul tarihi: 22.10.2018-08.11.2018
Makale türü: Araştırma

Anahtar Kelimeler: *Bilişimsel dilbilim, varlık ismi tanıma, doğal dil işleme, bilgi çıkarımı, eğitimsel teknoloji*

Abstract

Named entity recognition (NER) is an information extraction (IE) task that is in the scope of natural language processing (NLP) and text mining. Its extent and methods may differ between studies, but basically, it aims to detect expressions that indicates a person, location, organization etc. In this study, a NER structure is developed for Turkish lecture notes (for history and geography courses). Separately, this structure is a project that is specialized for an IE task. Besides, it also has an educational value, as the projected outcome from its execution is meaningful words or word groups from the content of input lecture notes, which can be used to construct glossary of terms structures for individual courses or course subjects. With these glossary of terms structures, it is aimed to detect expressions in the content of a lecture note that can be used for questions and support a test preparation process. In this document, general information about NER task and its scope is given; previous studies on the field are mentioned; the system developed in line with this study is introduced; success of the system is evaluated through experiment results and some thoughts for enhancement are shared.

Keywords: *Computational linguistics, named entity recognition, natural language processing, information extraction, educational technology.*

1. Giriş

Varlık ismi (*named entity)*, bir özel isim kullanılarak atıfta bulunulan tüm varlıkları kapsayan bir ifadedir. Bir bilgi çıkarımı görevi olan VİTise, bir metin içerisindeki varlık isimlerinin tespit edilmesini ve önceden tanımlı kategoriler göz önüne alınarak sınıflandırılmasını hedefler. VİT birleşik bir görev olarak ele alınmalıdır; çünkü sırayla yerine getirmesi gereken iki gereksinim içerir: İlki, bir özel isim ifade eden metin parçalarının sınırlarını doğru belirlemek; ikincisi ise bu ifadeleri doğru kategoriler altında sınıflamaktır.

Haber metinleri üzerinde çalıştırılan genel yapılı VİT sistemleri, kişi, yer ve kurum isimlerinin tespitine yoğunlaşmıştır. Daha özelleşmiş uygulamalarda ise ticari ürün, sanat eseri, bazı biyolojik terim (protein, gen türleri gibi) isimleri gibi farklı kategoriler karşımıza çıkabilir [1]. Çizelge 1'de örnek varlık ismi türleri ve kapsamları gösterilmiştir. VİT sistemlerinin çoğunda, varlık ismi kavramının özel isimlerle sınırlı tutulmadığı, metin içerisindeki karakteristik anlama sahip ifadelerin (özel isim olmasa dahi) de bu kapsamda değerlendirildiği gözlenir. Bu durum tarih, saat gibi zamansal ifadelerin ya da ölçüm, sayım, fiyat gibi sayısal ifadelerin de varlık ismi kategorilerine (etiket olarak da adlandırılır) dahil edilmesine yol açabilir. Bu makalede detaylandırılan sistem, tarih ve coğrafya alanlarında yazılmış Türkçe ders metinleri üzerine özelleştirilmiştir. Tespit edilen karakteristik varlık isimlerinin, tarih ve coğrafya alanlarına özel bir terimler sözlüğü yapısının altyapısını oluşturması hedeflenmiştir.

Çizelge 1. Varlık ismi türleri ve işaret ettikleri varlıklara örnekler

Tür	Etiket	Örnek Kategoriler
Kişi	PER	Şahıslar, hayali karakterler, küçük topluluklar
Kurum	ORG	Şirketler, acenteler, siyasi partiler, spor kulüpleri
Yer	LOC	Fiziki alanlar, dağlar, göller, denizler
Jeopolitik İfade	GPE	Ülkeler, eyaletler, şehirler, ilçeler
Tesis	FAC	Köprüler, havayolları, binalar
Taşıt	VEH	Uçaklar, trenler, arabalar

VİT sistemlerinde genel işleyiş, girdi olarak işaretlenmemiş (*unannotated*) bir metin bloğu alıp, çıktı olarak tespit edilen varlık isimlerini gösteren, yani işaretlenmiş (*annotated*) bir metin bloğu döndürmektir. Örneğin işaretlenmemiş "Mustafa Kemal Atatürk 1881 yılında Selanik'te doğdu." cümlesi için beklenen çıktı "[Mustafa Kemal Atatürk]$_{\text{PER}}$ [1881]$_{\text{DATE}}$ yılında [Selanik]$_{\text{LOC}}$'te doğdu." şeklindedir.

Çok anlamlılık ya da anlam bulanıklığı (*ambiguation*), birçok doğal dil işleme görevinde olduğu gibi, VİT sistemleri için de önemli bir sorundur. Örneğin "Washington" kelimesi, bulunduğu içeriğe göre bir kişi, bir yer, bir kurum (spor kulübü) ya da bir taşıt (gemi) ismi belirtiyor olabilir. Ya da "Ural" kelimesi bir yer (nehir ismi) ya da kişiden bahsediyor olabilir. VİT sistemleri, başarı oranlarının buna benzer durumlardan olumsuz etkilenmesini önlemek adına farklı yaklaşımlardan faydalanabilir.

Sözcük birimleştirme (*tokenization*), kelime düzeyinde yapılan bir metin bölümlendirme operasyonudur ve VİT işlemi için yaygın bir başlama noktasıdır. İstatistiksel metotlar tercih ediliyorsa, sonrasında sekans etiketleme (*sequence labeling*) ile devam edilir. Bu işlemde, sınıflandırıcılar (*classifier*), sözcük birimleri (*token*) sistemde tanımlı belirli bir türe ait bir varlık ismi işaret edip etmediğine göre etiketlemek üzerine eğitilirler. Bir varlık isminin başlangıcını (B →*Beginning*) ya da devamını (I →*Inside*) içeren, ya da hiç varlık ismi içermeyen (O →*Outside*) kelimeleri ayrıştırmayı hedefleyen IOB, etiketleme için sık tercih edilen bir formattır. Çizelge 2'de, sözcük birimlerin etiketlenmesi için kullanılan IO ve IOB formatları arasındaki farklar, bir cümle üzerinde gösterilmiştir.

Çizelge 2. IO ve IOB kodlama farkına örnek bir sekans etiketleme

	IO Kodlama	IOB Kodlama
Mehmet	PER	B-PER
Edvard	PER	B-PER
Munch	PER	I-PER
'un	O	O
resmini	O	O
Ahmet	PER	B-PER
'e	O	O
gösterdi	O	O

Öznitelik seçimi (*feature selection*), VİT başarısını etkileyebilecek başka bir faktördür. Sözcük birimler ile ilgili elde edilen bulguların tutulduğu öznitelikler, sistemin daha isabetli tahminler yapması için

kullanılan yapılardır. Sistem eğitilirken birden fazla öznitelik (*feature*) kullanılabilir. Örneğin yaygın özniteliklerden biri olan şekil (*shape*), sözcük birimlerin yazımı hakkında karakter düzeyinde bilgi tutar (tamamı küçük harf, tamamı büyük harf, sadece ilk harf büyük, tire karakteri içeriyor vb.). Dilbilgisi kurallarına uygun yazılmış metinlerde, büyük harf kullanımı ve noktalama işaretleri, varlık isimlerini tespit etmek için önemli ipuçları verebilirler. Çoğu VİT sistemi, yer adları dizini (*gazetteer*) yapılarından faydalanır. Kurum-kuruluş isimleri ve biyolojik terimler için de benzer yapılar mevcuttur. Saygı ifadeleri ya da unvanları barındıran, kestirimci kelime (*predictive words)* listeleri de kullanılabilir. Liste kullanımı ile, sözcük birimlerin ilgili dizin ya da listede bulunup bulunmadığı bilgisini içeren öznitelikler elde edilebilir. Sözcük türü etiketi (*part-of-speech tag),* kök bulma (*stemming)* sonrası ifade, söz öbeği etiketi (*syntactic chunk label)* diğer yararlanılabilir özniteliklerden birkaçıdır.

VİT algoritmaları temel olarak üç kategoride incelenir: İstatistiksel, kural tabanlı ve karma (*hybrid)* modeller. İstatistiksel modellerdeki temel yaklaşım, varlık isimlerine ait kuralların ve örüntülerin, önceden işaretlenmiş eğitim verisi (*training data)* yardımıyla öğrenilmesidir. Eğitim verisi, eğer varsa kullanılacak öznitelikler hakkında da bilgi sağlayacak şekilde etiketlenmelidir. Saklı markov modelleri (*hidden markov models),* maksimum entropi ve koşullu rassal alanlar (*conditional random fields)*, en yaygın istatistiksel modellerdir. Kural tabanlı modeller, öznitelik kümelerinden elde edilen bulguların, kullanılan dil ile ilgili önceden tanımlanmış dilbilgisi ve örüntü kurallarına göre değerlendirilmesi esasına dayanır.

Günümüzde başarı oranı yüksek bir VİT sistemi, haber içeriklerinin sınıflandırılması, tavsiye sistemleri, müşteri destek sistemleri, sosyal medya analizi, duygu analizi, istenmeyen (*spam)* e-posta tespiti, literatür taraması, ya da bu çalışmada önerilen modelin geliştirilme sebebi olan eğitimsel amaçlar gibi birçok farklı kullanım senaryosunda fayda sağlayacaktır.

2. İlgili Bilimsel Çalışmalar

Mesaj Anlama Konferansları (*MUC – Message Understanding Conferences)* ile bilgi çıkarımı çalışmaları teşvik edilerek alanda ilerleme sağlamak hedeflenmiştir. Başlıca iki değerlendirme kriterikesinlik (*precision)* ve hassasiyet (*recall),* MUC-2'de detaylandırılmış ve verilen bilgi çıkarımı görevlerinde kullanılmıştır. 1996'da düzenlenen altıncı konferansta İngilizce için VİT, verilen görevlerden biri olmuş ve en başarılı sistemde %97 kesinlik ve %96 hassasiyet değerlerine ulaşılmıştır. Eğitim verisi olarak Wall Street Journal makaleleri işaretlenmiştir. ENAMEX (kişi, kurum ve yer adları için) ve NUMEX (saat, para ifadesi, yüzde için) etiketleri bu konferansta tanıtılmıştır [2].

Cucerzan ve Yarowksy (1999) [3], Türkçe üzerine yayımlanmış ilk VİT araştırmasıdır. Sistem dillerden bağımsız olup, karakter düzeyinde oluşturulan bir ağaç yapısı üzerinde yinelemeli öğrenme (*iterative learning)* temelli bir önyükleme (*bootstrapping)* algoritmasından faydalanır. Sistem eğitimi bir kelimenin, bir belge içinde genellikle belirli tek bir anlam ifade ettiği ön kabulü üzerinden ilerler. Kaynak dil ile ilgili küçük bir varlık ismi listesi ile süreç başlatılıp, metinlerden morfolojik ve bağlamsal (*contextual)* ipuçları elde etmeye çalışılır. Örneğin, "-escu" ifadesi Rumence için neredeyse hatasız bir soy ismi göstergesi olarak bulunmuştur. Sistemin Türkçe için başarısı %60 kesinlik, %47 hassasiyet ve %53 f-ölçütü olarak hesaplanmıştır.

Alfonseca ve Manandhar (2002) [4], WordNet ontolojisinden yararlanarak, sınıflandırılmamış bir kavram için en isabetli kapsayıcı (*hypernym)*terimi bulmayı hedeflemişlerdir. Sistem sınıflandırma için, arama motorları üzerinde çalıştırdığı sorgular ile, aday kelimeler için benzerlik skorları elde ederek çalışır. Buradaki yaklaşım, anlamsal (*semantic)* olarak birbiriyle bağlantılı kelimelerin bir arada bulunmasının muhtemel olduğudur.

Tür vd. (2003) [5], n-gram dil modellerini saklı markov modelleri içine entegre ederek bir yapı kurmuşlardır. Çalışma dört farklı model içerir: Sözcüksel (*lexical*) modelde, sözcük birimler arası boşluklar, varlık isimlerinin sınırlarını ifade eden *yes, no* ve *mid* işaretleri (*boundary flag)* ile etiketlenir. Bağlamsal model, varlık ismi türü bilinmeyen bir sözcük birimin (*unk* etiketli), metin içerisinde önünde ve arkasında yer alan diğer sözcük birimler yardımıyla sınıflandırmayı hedefler. Çizelge 3'te bağlamsal modelin bilinmeyen bir kelime için kullanımına örnek verilmiştir. Morfolojik model, sözcük birimler karakter düzeyinde ele alır (tamamı büyük harf, sadece ilk harfi büyük gibi); ayrıca Türkçe için bir kişi, yer ve kurum isimleri sözlüğünden yararlanır. Etiket modeli (*tag model)* ise varlık ismi türü belirten

etiketler ile (kişi, yer, kurum, diğer) sınır belirten etiketler (yes, no, mid) arasındaki üçlükombinasyonların (*trigram)* olasılıkları ile ilgilenir. Deneyler için gazete makaleleri kullanılmıştır. Tüm modeller birleştirildiğinde varlık ismi metinleri için %90.4, varlık ismi türleri için %92.7 doğruluk (*accuracy)* elde edilmiştir.

Çizelge 3. Bağlamsal modelin bilinmeyen bir kelime için kullanımı (Tür vd. [2003])

Trigram Sekans	Olasılık
Dr./*diğer* boşluk/*yes* unk/*kişi*	0.990119
Dr./*diğer* boşluk/*yes* unk/*yer*	0.000690
Dr./*diğer* boşluk/*yes* unk/*kurum*	0.000880
Dr./*diğer* boşluk/*yes* unk/*diğer*	0.002688

CoNLL konferansları katılımcılara bilişimsel dilbilim görevleri veren başka bir etkinliktir. 2003'teki konferans dillerden bağımsız bir VİT yapısı oluşturma üzerinedir. Katılımcılardan ek görev olarak da işaretlenmemiş veriyi eğitim sürecine katmaları istenmiştir. Bu veriden büyük harf kullanımı ile ilgili bilgi edinmenin sonuçlara etkisinin, verinin yeni dizin terimleri bulmak için kullanılmasından daha olumlu olduğu gözlemlenmiştir [6].

Wentland vd. (2008) [7], HeiNER isimli çokdilli bir varlık ismi kaynağı oluşturmuştur. Wikipedia, varlık isimlerini elde etmek için temel kaynak olarak kullanılmıştır. Kelimelere ait belirsizlikleri giderecek bir sözlük oluşturmak için, Wikipedia'daki adlandırma (*disambiguation)* ve yönlendirme (*redirect)* sayfalarından faydalanılmıştır. Wikipedia makale başlıklarının bir varlık ismi belirtme olasılığının yüksek olduğu; bu durumun morfolojik normalleştirme ya da varlık ismi sınırlarının tespiti gibi bazı görevleri kolaylaştırdığı gözlemlenmiştir.
Küçük ve Yazıcı (2009a) [8], Türkçe için kural tabanlı bir VİT sistemi oluşturup, başarısını farklı alanlardaki (haber metinleri, masallar, tarihi metinler) metinlerde test etmişlerdir. Sistem Türkçe kişi isimleri, tanınmış siyasetçiler, iyi bilinen kurum-kuruluş isimleri ve muhtemel örüntüler gibi farklı sözlüksel kaynaklardan faydalanmıştır. Sonuçlara göre haber metinlerinde görülen %78 f-ölçütü, alan değişiminden olumsuz etkilenerek masallarda %69, tarihi metinlerde %55 olarak ölçülmüştür. Yabancı kişi isimleri ve tarihi kişi-kurum isimlerinin sözlüksel kaynaklarda olmamasının bu performans düşüşünün temel nedenlerinden olduğu gözlemlenmiştir.

Küçük ve Yazıcı (2009b) [9], çalışmalarını videolardan elde edilen metinler üzerinde de test etmişlerdir. Bunun için TRT arşivinden seçilen 16 haber videosu metne dökülmüştür. Çalışmanın yapıldığı dönemde Türkçe için bir konuşma tanıma (*speech recognition*) sistemi olmadığı için bu işlem elle yapılmıştır. Başarı %73 kesinlik, %77 hassasiyet ve %75 f-ölçütü olarak ölçülmüştür.

Tatar ve Çiçekli (2011) [10], VİT sistemlerinin alan değişimlerinden olumsuz etkilenmesini önlemeyi amaçlayarak, gözetimli öğrenme (*supervised learning)* ile otomatik kural tanımlama üzerine çalışmışlardır. Sistem yazımsal (*orthographical)*, bağlamsal, sözcüksel ve morfolojik özniteliklerden yararlanmış, ayrıca 2 seviyeli sözlüksel kaynaklar kullanmıştır. Türkçe haber metinleri içeren TurkIE veri kümesinde yapılan testlerde %91.7 kesinlik, %90 hassasiyet ve %91 f-ölçütü değerlerine ulaşılmıştır.

Küçük ve Yazıcı (2012) [11], kural tabanlı sistemleri üzerinden devam ederek karma bir model kurmuşlardır. n (bir kelimenin görülme sayısı) ve p (aynı kelimenin varlık ismi olarak işaretlenmiş şekilde görülme sayısı) şeklinde iki istatistiksel öznitelik tanımlayarak; p/n değerini o kelime için güven değeri olarak kullanmışlardır. Sistem üç farklı alanda yeniden test edildiğinde haber metinleri için %85.9, masallar için %85 ve tarihi metinler için %66.9 f-ölçütü değerlerine ulaşıldığı görülmüştür.

Şeker ve Eryiğit (2012) [12], koşullu rassal alanlar (CRF) ile çalışarak bir istatistiksel model kurmuşlardır. CRF modelinde öznitelikler için pencere genişliği {-3,+3} şeklinde tanımlanmıştır. Dizinler kullanılmış; ilaveten normal kelimelerden sonra veya önce gelerek varlık ismi oluşturabilecek üretici kelimeler listesi (22 kişi, 44 yer, 60 kurum ismi için) de kullanılmıştır. Üç kategoriye ayrılmış (morfolojik, sözcüksel, dizin tarama) toplam 14 öznitelik tanımlanmış; bunlar sisteme birer birer eklenerek başarıya katkıları ölçülmüştür. Deney sonuçları cümle başını belirten SS (*start of sentence*) dışında tüm özniteliklerin sistem başarısını olumlu etkilediğini göstermiştir. Sistem son durumda MUC kriterlerine göre %94.6, CoNLL kriterlerine göre %91.9 f-ölçütü değerlerine erişmiştir.

Küçük vd. (2014) [13], Türkçe üretilmiş Twitter gönderileri üzerinde VİT deneyleri gerçekleştirmiştir. Klasik kategorilerden (çalışmada PLO olarak isimlendirilmiş) ayrı olarak bir "*misc*" (ticari ürün, televizyon programı, müzik grubu isimleri gibi) türü

eklenmiştir. Kişi isimleri için aranan isim – soy isim ikilisi şeklinde olma şartının ihmal edildiği durumlara sık rastlandığı için; Avrupa Basın Takip (*EMM: Europe Media Monitor)* veri tabanı taranarak, tek kelime şeklinde sık kullanılan (En az 30 kere geçme şartı aranmıştır) kişi ve kurum isimleri bulunmuş ve iki liste elde edilmiştir. Deney sonuçları, veri setindeki PLO kullanımlarında %25 oranında büyük harf kullanımı hatası olduğunu, kişi isimlerinin isim – soy isim ikilisi şeklinde yer alma oranının sadece %32 olduğunu ve PLO metinlerinde %10 oranında Türkçe karakter problemi olduğunu göstermiştir. Konu etiketi (*hashtag*) metninde bulunan, birden çok kelimeden oluşan varlık isimlerinin tespiti de boşluk kullanımı olmadığı için başka bir sorun olarak ortaya konmuştur. Sistem %66 kesinlik, %31.5 hassasiyet ve %42.6 f-ölçütü değerlerine ulaşmıştır.

Küçük ve Arıcı (2016) [14], ODTÜ Türkçe Derlem içerisinden seçtikleri 10 gazete makalesi üzerinde varlık isimlerini işaretleyerek, Türkçe için 1425 varlık ismi (398 kişi, 567 yer, 460 kurum ismi) içeren bir veri seti ortaya çıkarmışlardır.

Şeker ve Eryiğit (2016) [15], önceki çalışmaları üzerinden ilerleyerek, kullanıcı tarafından oluşturulmuş içerik (*UGC: User generated content*) üzerinde çalışmışlardır. Büyük harf kullanımının eksikliği kaynaklı performans düşüklüğünü önlemek için, çok anlamlı kelimeler arasından cins isim olarak kullanılma ihtimalleri düşük olanları belirleyerek "ilk harfi otomatik olarak büyük harfe çevirme listesi" (*CAP: Auto capitalization gazetteer)* elde etmişlerdir. Önceki çalışmanın aksine, öznitelikler sistemden birer birer çıkararak sisteme katkıları ölçülmüştür. SS özniteliğinin, bu ölçümde sisteme %2.11 olumlu katkı verdiği görülmüştür. UGC veri setindeki deneylerde %67.9 başarıya ulaşılmıştır. Ayrıca CAP özniteliğinin sistemden çıkarılmasının, %20 üzeri bir performans kaybına yol açtığı gözlemlenmiştir.

Ertopçu vd. (2017) [16], parametrelerin değiştirildiği farklı metotlar ile testler yaparak, en yüksek başarı oranını sağlayan parametrelere ulaşmaya çalışmışlardır. En başarılı sonuçların, sınıflandırma algoritması olarak çok katmanlı algılayıcı (*multilayer perceptron)* seçilip öğrenme hızı *(learning rate)*0.1, pencere genişliği 1 verildiğinde ve 7 öznitelik kullanıldığında (büyük harf kontrolü, tarih ifadesi kontrolü, saat ifadesi kontrolü, kesir ifadesi kontrolü, sözcük türü, kök formu, gövde formu) elde edildiği (%7.64 hata oranı ile) gözlemlenmiştir.

Şahin vd. (2017) [17], Türkçe Vikipedi sayfalarının otomatik olarak sınıflandırılmasıyla VİT için bir Türkçe derlem oluşturmuşlardır. Derlem 300 bine yakın terim içermektedir. Terimler 4 ana kategoriye (kişi, yer, kurum, diğer) bağlı toplam 77 alt kategoriyle ilişkilendirilmiştir. Kelime belirsizliklerinin önüne geçmek için Freebase bilgi tabanından yararlanılmıştır. Sistem VİT için %84 f-ölçütü değerlerine ulaşmıştır.

Güneş ve Tantuğ (2018) [18], iki yönlü uzun-kısa süreli bellek (*bidirectional long-short term memory*) yapay sinir ağı yapısını 5 farklı modelde kullanmışlardır. CoNLL ölçümleri baz alınarak yapılan deneylerde, temel veri kümesi için %91.59 f-ölçütü değerine ulaşılmıştır. Sözcük vektörlerini kullanan temel veri kümesinin, yazım özellikleri ve biçimbilim özellikleriyle desteklendiği katmanlı bir yapay sinir ağı modeli önerilmiştir. Son modelin sistem başarısı %93.69 seviyesine yükselmiştir.

Güngör vd. (2018) [19] önerdikleri yapay sinir ağı modelinde, cümledeki sözcükleri baştan ve sondan işleyerek konum bilgisinin tutulduğu karakter tabanlı sözcük vektörleri oluşturmuşlardır. Bu vektörlerin, dağılımsal sözcük vektörleri ile yakalanamayan sözcük içi ilişkilerini yakalamak için kullanılması hedeflenmiştir. Sadece dağılımsal sözcük vektörleri kullanıldığında %90.96 f-ölçütü olarak hesaplanan sistem başarısının, karakter tabanlı sözcük vektörlerinin eklenmesiyle %93.37 seviyesine yükseldiği gözlemlenmiştir.

3. Uygulama

Bu çalışmada önerilen VİT sistemi, eğitimsel amaçlar doğrultusunda geliştirilmiş bir bilgi çıkarımı yazılımıdır. Tarih ve coğrafya alanlarında yazılmış Türkçe ders metinleri için özelleştirilmiştir. Çalışmanın birincil hedefi, sisteme girdi olarak verilen metin belgelerinin içeriğindeki varlık isimlerinin yüksek başarıyla tespit edilmesi olup; bu varlık isimleri ile nitelikli ve beklentilere cevap veren terimler sözlüğü yapılarının temelini oluşturmak bir sonraki hedef olarak belirlenmiştir. Araştırmanın uzun vadeli hedefi ise, oluşturulacak terimler sözlüklerine, sınav hazırlama süreçlerine destek sağlayabilecek bir yapı kazandırmaktır.

3.1. Önerilen Sistem

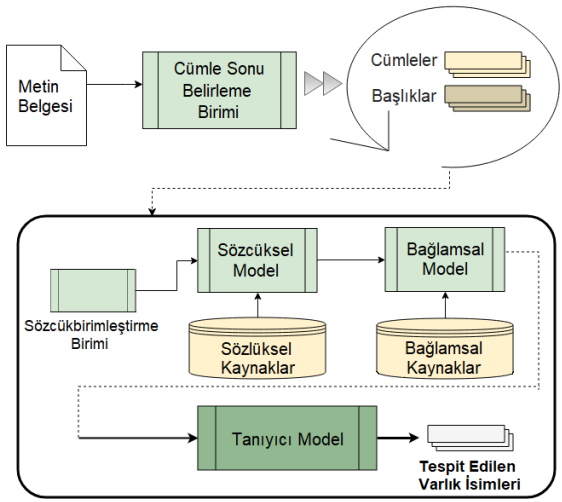
Önerilen VİT sistemi, kural tabanlı bir model kullanılarak inşa edilmiştir. Girdi olarak bir metin belgesi alıp, tespit edilen varlık isimlerini ve türlerini çıktı olarak vermektedir.

Sistem cümleler üzerinde çalışacak şekilde geliştirilmiştir. Bu nedenle girdi olarak alınan metin belgesi ilk olarak cümle sonu belirleme (*SBD*: *sentence boundary detection*) birimine gönderilir. Bu birim, aldığı metin belgesini ilk olarak bir ön işleme (*pre-processing)* sürecinden geçirir. Bu süreç gereksiz karakter, sembol ve boşlukların çıkarılması, maddelerle ayrılmış metin parçalarının birleştirilmesi gibi operasyonları kapsamaktadır. Ön işlemeden sonra başlıklar ve cümle sonları tespit edilir ve kullanıcıya cümleler ve başlıklar olmak üzere 2 sıralı liste döndürülür. Bu işlemler için cümle sonu koşullarının tanımlandığı bir liste oluşturulmuş; sonrasında bu koşullar arka uçta kurallı ifadelere (*regular expression*) dönüştürülerek SBD biriminin kullanımına sunulmuştur. Metin içerisindeki kısaltma kullanımlarının yol açabileceği hatalı sonuçların önüne geçebilmek adına, bu aşamada 204 elemandan oluşan bir Türkçe kısaltma listesi kullanılarak bir kısaltma denetim operasyonu da gerçekleştirilir.
Çizelge 4'te, SBD birimi çalıştırılırken kullanılan cümle sonu koşullarından bazıları gösterilmiştir (KH → küçük harf, BH → büyük harf, B → boşluk karakteri, R → rakam; Doğru → cümle sonu koşulunu işaret eder, Yanlış → cümle sonu olmama koşulunu işaret eder).

Çizelge 4. Örnek cümle sonu koşulları

Koşul	Çıktı
KH . BH	Doğru
KH . KH	Yanlış
KH . R	Doğru
BH. KH	Yanlış
KH .B . BH	Doğru
KH .B . R	Doğru

SBD biriminin çalıştırılmasının ardından, metinden cümleler elde edilerek VİT sisteminin kullanımına sunulur. Dolayısıyla, VİT modelinin başarısı, SBD biriminin başarısına da bağlıdır. Cümleler VİT yapısına birer birer verilir ve sırasıyla sözcük birimleştirme birimi (*tokenizer*), sözcüksel model ve bağlamsal modelde işlenir. Bu üç birim, verilen cümleyi etiketleyerek, son birim olan tanıyıcı modele (*recogniser model)* bilgi sağlar. Tanıyıcı model, varlık isimlerini tespit etmek için, etiketlenmiş sözcük birimlerden oluşan cümleyi analiz eder. Şekil 1'de, önerilen sistem çatısına dair bir akış şemasına yer verilmiştir.



Şekil 1. VİT yapısı için önerilen sistem çatısı

3.2. Sözcük birimleştirme Birimi ve Sözcük birimler

Sisteme verilen metin belgesinden elde edilen cümleler ilk olarak sözcük birimleştirme biriminde işlenir. Bu birim, girdi cümleyi tarayarak kelime sonlarını ve noktalama işaretlerini tespit ederek cümleyi bir sözcük birim listesine dönüştürür. Sözcük birim yapılarının kapsamı, bir kelime ile sınırlı olmayabilir; noktalama işareti ya da noktalama işareti sonrası bir morfem de ifade edebilir. Dolayısıyla bu birimde yapılan işlemi, cümleyi kelimelerine ayırmak şeklinde ifade etmek hatalı olacaktır.

Program tarafında sözcük birimler, Token sınıfı nesnelerine atanır. Bu sınıfa ait nesneler, kendisinden önceki ve sonraki Token nesnesinin bilgisini de tutar. Bu tasarım, cümleden elde edilen sözcük birimlerin çift yönlü bağlı liste (*double linked list)* yapısında tutulmasını sağlamış olur. Token sınıfı nesneleri aynı zamanda, etiketleme işlemleri ile değer atamalarının yapılacağı öznitelik bilgilerini taşıyan bir dizi mantıksal değişken de içermektedir. Bir sözcük birimin etiketlenmesi ile, varlık isimlerinin tespiti

sırasında kullanılabilecek önemli bilgilerin sağlanması hedeflenmiştir.

Sözcük birimler üzerinde ilk etiketleme sözcük birimleştirme biriminde yapılır. Bu etapta yapılan etiketleme ile sisteme yazımsal, numerik, noktalama ile ilgili ve sözcük birim pozisyonu ile ilgili bilgiler sağlanmış olur. Çizelge 5'te, bu birimde kullanılan etiketler gösterilmiştir.

Çizelge 5. Sözcük birimleştirme birimi etiketleri

Yazımsal Bilgi	Numerik Bilgi	Noktalama Bilgisi	Pozisyon Bilgisi
SW_CAPITAL	NUM	PUNCT_APOSTR	BEFORE_APOST
ALL_CAPITAL	ROMAN_NUM	PUNCT_OTHER_MID	AFTER_APOST
EW_DOT	ORD_NUM	PUNCT_OTHER_END	
	DAY_NUM	PERCT	
	MONTH_NUM		
	YEAR_NUM		

- SW_CAPITAL: Sözcük birim metninin büyük harf ile başlayıp başlamadığı bilgisini tutar.
- ALL_CAPITAL: Sözcük birim metninin tamamen büyük harflerden oluşup oluşmadığı bilgisini tutar.
- EW_DOT: Sözcük birim metninin son karakterinin nokta olup olmadığı bilgisini tutar.
- NUM: Doğru (*true*) değeri atanması, sözcük birim metninin sayısal bir ifade belirttiğini gösterir.
- ROMAN_NUM: Doğru değeri atanması, sözcük birim metninin bir Romen rakamı belirttiğini gösterir.
- ORD_NUM: Doğru değeri atanması, sözcük birim metninin bir sıra sayısı belirttiğini gösterir.
- DAY_NUM: Doğru değeri atanması, sözcük birimin [1,31] aralığında sayısal bir ifade belirttiğini gösterir.
- MONTH_NUM: Doğru değeri atanması, sözcük birimin [1,12] aralığında sayısal bir ifade belirttiğini gösterir.
- YEAR_NUM: Doğru değeri atanması, sözcük birimin [100, 5500] aralığında sayısal bir ifade belirttiğini gösterir.
- PUNCT_APOSTR: Sözcük birim metninin bir kesme işareti olup olmadığı bilgisini tutar.
- PUNCT_OTHER_MID: Sözcük birim metninin, cümle ortasında kullanılan bir noktalama işareti (virgül, noktalı virgül, parantez vb.) olup olmadığı bilgisini tutar.
- PUNCT_OTHER_END: Sözcük birim metninin, cümle sonunda kullanılan bir noktalama işareti (nokta hariç) olup olmadığı bilgisini tutar.
- PERCT: Sözcük birim metninin yüzde işareti olup olmadığı bilgisini tutar.
- BEFORE_APOST: Doğru değeri atanması, bir sonraki sözcük birimin bir kesme işareti olduğunu gösterir.
- AFTER_APOST: Doğru değeri atanması, bir önceki sözcük birimin bir kesme işareti olduğunu gösterir.

3.3. Sözcüksel Model Kaynakları

Sözcüksel ve bağlamsal modeller, sözlüksel kaynaklar (*lexicon*) yardımıyla sözcük birimlerin etiketlenmesi ve son hâllerinin verilmesini amaçlar. Bağlaçların tutulduğu yardımcı liste haricinde, sözcüksel modelin kullandığı kaynaklar muhtemel özel isimlerin (kişi, yer ya da bölge belirtmek üzere) tutulduğu yapılardır.

- **TR_FirstNames:** Türkçe kişi isimlerinin tutulduğu kaynak listedir. TDK Kişi Adları Sözlüğü elemanlarını içeren bir veri tabanı baz alınarak hazırlanmıştır. Listenin ilk hâli 9699 eleman içerirken, sayı 3.3.1 alt başlığı altında detaylandırılacak elemelerden sonra 9619 elemana düşürülmüştür.
- **TR_CommonSurnames:** Sık karşılaşılan Türkçe soy isimlerinin tutulduğu kaynak listedir. Elemanlar, Wikipedia üzerindeki Türk sinema oyuncuları, Türk siyasetçiler (20. ve 21. yüzyıl), Türk yazarlar ve Türk Kurtuluş Savaşı'na katılan üst düzey subaylar listelerinden elde edilmiştir. Aynı kişinin isminin birden fazla yer aldığı durumlar (örneğin hem 20., hem 21. yüzyılda görev almış siyasetçiler) ve sık karşılaşılan soy isimlerinin tekrar ettiği durumlar elendiğinde, listenin son hâli 3039 eleman barındırmaktadır.
- **FRGN_FirstNames:** Sık karşılaşılan yabancı kişi isimlerinin tutulduğu kaynak listedir. Elemanlar, *ranker.com* üzerindeki "gelmiş

geçmiş en etkili insanlar" (*the most influential people of all time*) listesinden elde edilmiştir. Bu listede farklı ülkelerden, aralarında bilim adamlarının, politikacıların, sanatçıların, sporcuların, filozofların bulunduğu 2762 kişi bulunmaktadır. Liste verisi XML dosyası şeklinde çekilerek, bir normalleştirme işlemine tabi tutulmuş; sonucunda isimler, soy isimler ve ikinci isimler (göbek adları) elde edilmiştir. Normalleştirme fazı, İngilizce yazılmış kişi isimlerinde karşımıza çıkabilen "*of, the*" gibi tanımlık (*article*) ifadelerinin, sıra sayılarının, Romen rakamlarının ve unvan ya da lakap belirten (Aziz, Deli, Kral, Kraliçe, Baron, Prens, Prenses gibi) ifadelerin elenmesini kapsar. Tekrarlanan isimler de aynı şekilde listeden çıkarılır. Son durumda listede 1489 eleman bulunmaktadır.

- **FRGN_CommonSurnames:** Sık karşılaşılan yabancı soy isimlerinin tutulduğu kaynak listedir. Elemanlar yine *ranker.com* üzerindeki kaynak listeden elde edilmiştir. Son durumda listede 1864 eleman bulunmaktadır.
- **FRGN_MidNames:** Yabancı kişi adları ve soyadları arasında karşılaşılabilen "de, von, bin" gibi ifadelerin ya da bir büyük harf ve noktadan oluşan kısaltılmış ikincil adların tutulduğu kaynak listedir. Elemanlar yine *ranker.com* üzerindeki kaynak listeden elde edilmiştir. Son durumda listede 34 eleman bulunmaktadır.
- **Countries:** Ülke isimlerinin tutulduğu kaynak listedir. Günümüzde Birleşmiş Milletler üyesi olan 193 ülkenin, üye ülkelerin kapsamında değerlendirilen ülkelerin (İngiltere, Galler, İskoçya gibi) ya da onlara bağlı özerk ülkelerin (Porto Riko, Virjin Adaları) isimleri listeye eklenmiştir. Filistin, Tayvan ve KKTC de listeye eklenen diğer ülke isimleri olmuştur. İlaveten, tarihi metinlerde geçme ihtimali bulunan bazı yakın dönem ülkelerinin (Yugoslavya, SSCB gibi) isimleri de dahil edilmiştir. Son durumda listede 257 eleman bulunmaktadır.
- **TR_Cities:** Türkiye'nin 81 şehrinin isminin ve yaygın kullanılan farklı isimlendirmelerinin (Afyonkarahisar için Afyon gibi) tutulduğu kaynak listedir. Liste 86 eleman içermektedir.
- **TR_Districts:** Türkiye'nin ilçelerinin isimlerini tutan kaynak listedir. Listenin ilk hâli 984 elemandan oluşmaktayken, aynı ismi taşıyan ilçelerin ve bulunduğu şehrin ismiyle anılan merkez ilçelerin elenmesiyle beraber, son durumda listede 897 eleman bulunmaktadır.
- **FRGN_StatesCities:** Tüm ülkelerin başkentleri başta olmak üzere, yüksek nüfuslu ya da tarihi-turistik önemi yüksek dünya şehirlerinin isimlerinin tutulduğu kaynak listedir. Listeden ülkesiyle aynı adı taşıyan şehirler (Tunus, Cezayir, Singapur gibi) çıkarıldığında, son durumda listede 380 eleman bulunmaktadır.
- **GeographicRegions:** Kıta ve önemli coğrafi bölgelerin isimlerinin tutulduğu kaynak listedir. Liste 22 elemandan oluşmaktadır.
- **Conjunctions:** Türkçe'de kullanılan bağlaçların tutulduğu yardımcı listedir. Bu liste, cümle başında bulunduğu için büyük harfle başlayan bağlaçların tespit edilerek hatalı varlık ismi sonuçlarının önlenmesi açısından önemlidir.

3.3.1. Kaynaklardan Çıkarılan Elemanlar

Yabancı kişi isimleri elde etmek için kullanılan *ranker.com* üzerinden alınan listede Mustafa Kemal Atatürk, Orhan Veli Kanık, Yunus Emre, Halide Edip Adıvar gibi Türk kişiler de bulunuyor. Bu durumun da etkisiyle, TR_FirstNames ve FRGN_FirstNames listeleri arasında 29, TR_CommonSurnames ile FRGN_CommonSurnames listeleri arasında 13 ortak eleman olduğu gözlemlendi. Bu durum göz önüne alınarak listeler üzerinde güncellemeler yapıldı:

- "Abdullah, Selma, Selman, Zakir" gibi kelimeler iki listede de bırakıldı.
- "Edip, Evliya, Halide, Hamdi, Kemal, Mustafa, Orhan, Yunus, Ziya" gibi kelimeler FRGN_FirstNames listesinden çıkarıldı.
- "Adam, Alan, Boy, Sun, San" gibi kelimeler TR_FirstNames listesinden çıkarıldı.
- "Adıvar, Çelebi, Emre, Kanık, Pamuk, Atatürk, Tanpınar" gibi Türkçe soyadı belirten kelimeler FRGN_CommonSurnames listesinden çıkarıldı.
- "Bradley, Reynaud, Spence" gibi kelimeler TR_CommonSurnames listesinden çıkarıldı. Bu kelimelerin, yabancı kökenli olan ya da yabancı biriyle evli kişiler dolayısıyla ilk etapta bu listeye girdikleri görüldü.
- Bağlamsal model kaynaklarında bulunan bazı elemanlarla da kesişim görüldü. Bu elemanların

bir bölümü sözlüksel kaynaklardan çıkarıldı ve listelere son durumları verilmiş oldu.

3.4. Bağlamsal Model Kaynakları

Bağlamsal model tarafından kullanılan kaynaklar, metin içerisinde özel isimlere komşu olması muhtemel ifadelerin tutulduğu listelerdir. Bu ifadelerin varlık ismi metnine dahil olup olmamasına dair kesin bir kural olmamakla birlikte, büyük harf ile başlayıp başlamaması bu karar verilirken kullanılan başlıca kriterdir.

- **Kişi Öncesi:** Bir kişi isminden önce gelebilecek kelime veya kelime gruplarının tutulduğu dört farklı liste kullanılır. Mesleki unvanlar ("Lord, Gazi" gibi), saygı ifadeleri ("Bay, Bayan" gibi), kısaltma şeklinde mesleki unvanlar ("Dr., Prof." gibi) ve ara ifadeler ("komutanı, padişahı" gibi), bu listelerde tutulan elemanlardır.
- **Kişi Sonrası:** Bir kişi isminden sonra gelebilecek mesleki unvan ya da lakapların ("Efendi, Hatun, Han, Paşa" gibi) tutulduğu bir liste kullanılır.
- **Ülke - Devlet Sonrası:** Bir ülke ya da devlet isminden sonra gelebilecek kelime veya kelime gruplarının tutulduğu iki farklı liste kullanılır. Biri "Krallığı, Cumhuriyeti" gibi bitiş ifadeleri, diğeri "başbakanı, imparatoru" gibi ara ifadeleri bünyesinde barındırır.
- **Yer Sonrası:** Bir yer – bölge isminden sonra gelebilecek "belediye başkanı, Bölgesi, valisi" gibi bitiş ifadelerinin tutulduğu bir liste kullanılır.
- **Kurum Sonrası:** Bir kurum – kuruluş isminin sonunda yer alabilecek "Derneği, Meclisi, Kurumu" gibi bitiş ifadelerinin tutulduğu bir liste kullanılır.
- **Coğrafi Oluşum Sonrası:** Bir coğrafi oluşum isminin sonunda yer alabilecek "Gölü, Dağı, Irmağı" gibi bitiş ifadelerinin tutulduğu bir liste kullanılır. Ayrı bir kelime olarak değil, bir kelimenin sonuna eklemlenmiş şekilde karşımıza çıkabilecek "-ırmak, -dağlar" gibi ifadeler için de ilave bir liste vardır.
- **Coğrafi Olay Sonrası:** "Depremi, Yangını" gibi bitiş ifadelerinin tutulduğu bir liste kullanılır.
- **Tarihi Olay Sonrası:** "Savaşı, Devrimi, İsyanı" gibi bitiş ifadelerinin tutulduğu bir liste kullanılır.
- **Tarihi Yapı Sonrası:** "Sarayı, Köprüsü, Heykeli" gibi bitiş ifadelerinin tutulduğu bir liste kullanılır.
- **Aylar:** Yılın aylarının isimlerinin tutulduğu bir liste kullanılır.

3.5. Sözcüksel ve Bağlamsal Model ile Etiketleme

Sözcük birimleştirme birimindeki etiketleme işleminin aksine, sözcüksel ve bağlamsal modelde etiketleme yaparken sözcük birimler birer birer değil, n-gram yapıları şeklinde ele alınır. Bunun nedeni kullanılan kaynak listelerinde, çok sözcüklü (*multiword)* ifadelerin de bulunmasıdır. Pencere genişliği için başlangıç değeri 4 olarak belirlenmiştir ve sıfıra ulaşıncaya kadar her döngü adımında bir azaltılır. Bu yaklaşımla çok sözcüklü ifadeler ıskalanmamış ve doğru etiketlenmiş olur. Çizelge 6'da, 7 sözcük birim içeren bir cümle üzerinde n-gram tarama operasyonu için oluşturulan arama modelleri gösterilmiştir.

Çizelge 6. N-gram tarama için oluşturulan arama modelleri (7 sözcük birim içeren cümle için)

N Değeri	Arama Modelleri
4	1234 – 2345 – 3456 – 4567
3	123 – 234 – 345 – 456 – 567
2	12 – 23 – 34 – 45 – 56 – 67
1	1 – 2 – 3 – 4 – 5 – 6 – 7

Çizelge 7. Sözcüksel (S) ve Bağlamsal (B) model etiketleri

Model	Etiket İsmi	Açıklama
S	LEX_TR_FN	Türkçe kişi ismi
S	LEX_TR_LN	Türkçe soy isim
S	LEX_FRGN_FN	Yabancı kişi ismi
S	LEX_FRGN_MN	Yabancı ikincil isim
S	LEX_FRGN_LN	Yabancı soy isim
S	LEX_CTRY	Ülke ismi
S	LEX_TR_CITY	Türkiye şehri ismi
S	LEX_TR_DIST	Türkiye ilçesi ismi
S	LEX_FRGN_CITY	Yabancı şehir ismi
S	CONJ_SWC	Büyük harfle başlayan bağlaç
S	NOT_LEX_SWC	Büyük harfle başlayan, sözlüksel olmayan ifade
B	B_PERSON	Kişi öncesi ifade

B	A_PERSON	Kişi sonrası ifade
B	A_LOC_CTRY	Ülke – devlet sonrası ifade
B	A_LOC_OTH	Yer sonrası ifade
B	A_ORG	Kurum sonrası ifade
B	A_HIST_BLDG	Tarihi yapı sonrası ifade
B	A_HIST_EVNT	Tarihi olay sonrası ifade
B	A_GEO_FORM	Coğrafi oluşum sonrası ifade
B	A_GEO_EVNT	Coğrafi olay sonrası ifade
B	EW_GEO_FORM	Bitişi coğrafi oluşum belirten ifade
B	MONTH_NAME	Ay ismi

Sözcüksel ve bağlamsal modeller, yararlanılan kaynak listeleri üzerinde n-gram taramalar (*lexicon lookup*) gerçekleştirerek sözcük birimleri etiketler. Kullanılan etiketler Çizelge 7 üzerinde gösterilmiştir. Şekil 2'de ise örnek bir kullanım senaryosu olarak, sırasıyla üç birimden geçerek sözcük birimleştirme ve etiketleme operasyonları tamamlanmış bir cümle gösterilmiştir. Sözcük birim etiketleri, değer atamasını yapan modele göre farklı renklerle gösterilmiştir. Etiketlemeler tamamlandığında sözcük birimlerin son hâlleri verilmiş olur ve tanıyıcı modele aktarılırlar.

3.6. Varlık İsimleri ve Tanıyıcı Model

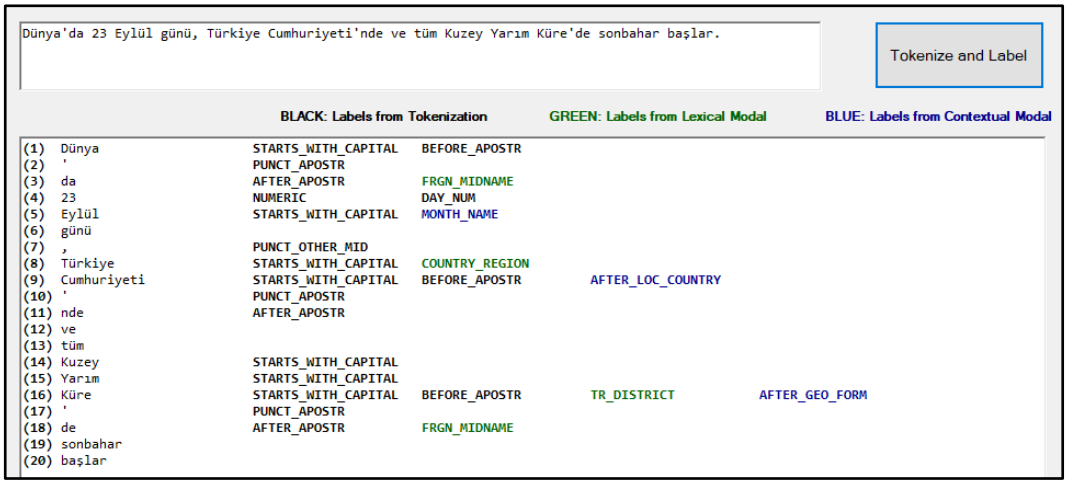
Geliştirilen VİT sistemi, tarih ve coğrafya alanlarında yazılmış ders metinleri için özelleştirildiği için, varlık ismi kavramının kapsamı da ihtiyaçlar doğrultusunda genişletilmiştir. Çizelge 8'de, sistem üzerinde tanımlanan 13 varlık ismi türü gösterilmiştir.

Çizelge 8. Sistemde tanımlanan varlık ismi türleri

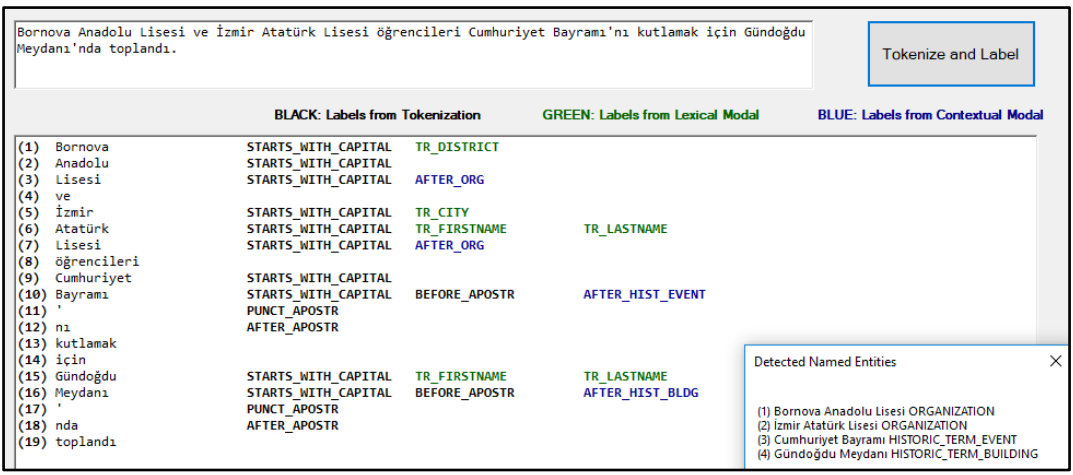
Orijinal İsim (İngilizce)	Türkçe Açıklama
Person_Turkish	Kişi İsmi (Türk)
Person_Foreign	Kişi İsmi (Yabancı)
Location_State_Country	Yer İsmi (Ülke - Devlet)
Location_Other	Yer İsmi (Diğer)
Historic_Term_Building	Tarihi Terim (Yapı İsmi)
Historic_Term_Event	Tarihi Terim (Olay İsmi)
Geographic_Term_Formation	Coğrafi Terim (Oluşum İsmi)
Geographic_Term_Event	Coğrafi Terim (Olay İsmi)
Organization	Kurum – Kuruluş İsmi
Percentage	Yüzde İfadesi
Date	Tarih
Date_or_Number	Tarih veya Sayı
Other	Diğer

Tanıyıcı model, varlık isimlerini tespit etmek için etiketlemesi tamamlanmış sözcük birimler üzerinde çalıştırılır. Sistem tek bir cümle ile test edilebildiği gibi, bütün hâlinde bir metin dosyası ile de çalıştırılabilir.

Şekil 3'te, sistemin "Bornova Anadolu Lisesi ve İzmir Atatürk Lisesi öğrencileri, Cumhuriyet Bayramı'nı kutlamak için Gündoğdu Meydanı'nda toplandı." cümlesi ile test edildiği örnek bir kullanım senaryosu gösterilmiştir. İşlemin sonucu olarak geriye dört adet varlık ismi döndürülmüştür.



Şekil 2. Sözcükbirimleştirme ve etiketleme işlemlerinden geçmiş örnek bir cümle



Şekil 3. Cümle üzerinde tespit edilen varlık isimlerinin gösterimine örnek bir sistem çıktısı

4. Araştırma Bulguları

4.1. Veri Kümesi ve Deney Sonuçları

Sistemin başarısı, gerçek ders metinleri üzerinde yapılan deneyler ile ölçülmüştür. Bu görev için 30 tarih ve 30 coğrafya metni seçilmiştir. Değerlendirmeler, kesinlik ve hassasiyet kriterleri ile METİN (*TEXT*) ve TÜR (*TYPE*) nitelikleri üzerinden yapılmıştır. METİN, varlık isminin sınırlarını doğru belirlemeyi; TÜR ise varlık ismi türünün doğru bulunmasını ifade etmektedir. Kıyaslama yapabilmek adına, coğrafya ve tarih alanları için deneyler ayrı olarak yapılmış; son raddede nihai sonuçlara iki deney kümesinin birleştirilmesiyle ulaşılmıştır. TÜR değeri doğru tahmin edilen varlık isimlerinin sınıflandırıldıkları tür bilgileri de sayılarak; alanlar arasındaki dağılım gözlemlenmiştir. Kesinlik, değerleri doğru tahminlerin toplam tespit edilen

Çizelge 9. Tarih ders metinleri üzerinde yapılan deney sonuçları (İlk 5 metin gösterilmiştir)

BELGE İSMİ	Gerçek Vİ (#)	Bulunan Vİ (#)	Doğru METİN (#)	Doğru TÜR (#)	Bulunamayan Vİ (#)	Kesinlik METİN (%)	Kesinlik TÜR (%)	Hassasiyet METİN (%)	Hassasiyet TÜR (%)
1. Bayezid Dönemi	71	72	70	68	1	97,22	94,44	98,59	95,77
1. Dünya Savaşı Öncesi Gelişmeler	69	66	64	63	5	96,97	95,45	92,75	91,30
1. Dünya Savaşı	50	48	47	46	3	97,92	95,83	94,00	92,00
1. Meşrutiyet	34	34	33	31	1	97,06	91,18	97,06	91,18
2. Dünya Savaşı'nın Nedenleri, Gelişimi	47	48	46	45	1	95,83	93,75	97,87	95,74
TOPLAM	**1654**	**1650**	**1585**	**1529**	**69**	**96,06**	**92,67**	**95,83**	**92,44**
ORTALAMA	**55,13**	**55,00**	**52,83**	**50,97**	**2,30**				

Çizelge 10. Coğrafya ders metinleri üzerinde yapılan deney sonuçları (İlk 5 metin gösterilmiştir)

BELGE İSMİ	Gerçek Vİ (#)	Bulunan Vİ (#)	Doğru METİN (#)	Doğru TÜR (#)	Bulunamayan Vİ (#)	Kesinlik METİN (%)	Kesinlik TÜR (%)	Hassasiyet METİN (%)	Hassasiyet TÜR (%)
Akarsu Havzalarımız	34	33	32	31	2	96,97	93,94	94,12	91,18
Aktif Nüfusun Ekonomik Faaliyet Gruplarına Göre Dağılımı	14	14	14	14	0	100,00	100,00	100,00	100,00
Basınç Çeşitleri ve Özellikleri	36	33	33	33	3	100,00	100,00	91,67	91,67
Başlıca Kıyı Tipleri	29	29	28	27	1	96,55	93,10	96,55	931,10
Bölgeler Coğrafyası – Akdeniz Bölgesi	21	22	20	20	1	90,91	90,91	95,24	95,24
TOPLAM	**991**	**996**	**962**	**930**	**25**	**96,59**	**93,37**	**97,07**	**94,84**
ORTALAMA	**33,03**	**33,20**	**32,07**	**31,00**	**0,83**				

varlık ismi sayısına bölünmesiyle; hassasiyet, değerleri doğru tahminlerin gerçek varlık ismi sayısına bölünmesiyle elde edilmiştir. Çizelge 9 ve 10, tarih ve coğrafya alanlarındaki ders metinleri için elde edilen deney sonuçlarını, Çizelge 11 ise birleştirilmiş sonuçları göstermektedir. (Çizelgelerde "varlık ismi" ifadesi "Vİ" şeklinde kısaltılmıştır.)

Deneylerde kullanılan veri kümesindeki 30 tarih metninde toplam 1654, 30 coğrafya metninde toplam 991, genel toplamda ise 60 metin için 2645 varlık ismi bulunmaktadır. Bu da bir tarih metninde ortalama 55.13, bir coğrafya metninde ortalama 33.03 ve genel ortalamada bir ders metninde 44.08 varlık ismi bulunduğunu ifade etmektedir.

Varlık ismi türlerinin metinler arasındaki dağılımını incelediğimizde, 30 tarih metninde toplam 133 Kişi

Çizelge 11.Birleştirilmiş deney sonuçları (Toplam 60 ders metni için)

METİN BELGESİ	Gerçek Vİ (#)	Bulunan Vİ (#)	Doğru METİN (#)	Doğru TÜR (#)	Bulunamayan Vİ (#)
TARİH Metinleri (30)	1654	1650	1585	1529	69
Coğrafya Metinleri (30)	991	996	962	930	25
TOPLAM	**2645**	**2646**	**2547**	**2459**	**94**
ORTALAMA	**44,08**	**44,10**	**42,45**	**40,98**	**1,57**

METİN BELGESİ	Kesinlik METİN (%)	Kesinlik TÜR (%)	Hassasiyet METİN (%)	Hassasiyet TÜR (%)	F-Ölçütü METİN (%)	F-Ölçütü TÜR (%)
TARİH Metinleri (30)	96,06	92,67	95,83	92,44	95,94	92,55
Coğrafya Metinleri (30)	96,59	93,37	97,07	93,84	96,83	93,61
TOPLAM	**96,26**	**92,93**	**96,29**	**92,97**	**96,28**	**92,95**

Çizelge 12. Doğru tahmin edilen varlık ismi türlerinin tarih ve coğrafya metinlerindeki dağılımı

METİN BELGESİ	Kişi İsmi (Türk)	Kişi İsmi (Yabancı)	Yer İsmi (Ülke – Devlet)	Yer İsmi (Diğer)	Kurum – Kuruluş İsmi	Tarihi Terim (Yapı İsmi)
TARİH Metinleri (30)	121	43	259	114	90	8
Coğrafya Metinleri (30)	0	7	223	185	4	3
TOPLAM	**121**	**50**	**482**	**299**	**94**	**11**
ORTALAMA	**2,02**	**0,83**	**8,03**	**4,98**	**1,57**	**0,18**

METİN BELGESİ	Tarihi Terim (Olay İsmi)	Coğrafi Terim (Oluşum İsmi)	Coğrafi Terim (Olay İsmi)	Tarih	Tarih veya Sayı	Yüzde İfadesi	Diğer	TOPLAM
TARİH Metinleri (30)	119	37	0	218	25	5	502	**1541**
Coğrafya Metinleri (30)	3	185	24	47	55	20	175	**931**
TOPLAM	**122**	**222**	**24**	**265**	**80**	**25**	**677**	
ORTALAMA	**2,03**	**3,70**	**0,40**	**4,42**	**1,33**	**0,42**	**11,28**	

İsmi (Türk), 48 Kişi İsmi (Yabancı), 273 Yer İsmi (Ülke – Devlet), 126 Yer İsmi (Diğer), 101 Kurum – Kuruluş İsmi, 9 Tarihi Terim (Yapı İsmi), 127 Tarihi Terim (Olay İsmi), 39 Coğrafi Terim (Oluşum İsmi), 221 Tarih, 26 Tarih veya Sayı, 5 Yüzde İfadesi ve 546 Diğer etiketli varlık ismi olduğu tespit edilmiştir. Bu metinlerde Coğrafi Terim (Olay İsmi) etiketi alan bir ifade olmadığı görülmüştür.

30 coğrafya metninde ise toplam 8 Kişi İsmi (Yabancı), 225 Yer İsmi (Ülke – Devlet), 200 Yer İsmi (Diğer), 4 Kurum – Kuruluş İsmi, 3 Tarihi Terim (Yapı İsmi), 3 Tarihi Terim (Olay İsmi), 209 Coğrafi Terim (Oluşum İsmi), 27 Coğrafi Terim (Olay İsmi), 47 Tarih, 62 Tarih veya Sayı, 20 Yüzde İfadesi ve 183 Diğer etiketli varlık ismi olduğu tespit edilmiştir. Bu metinlerde Kişi İsmi (Türk) etiketi alan bir ifade olmadığı görülmüştür.

Tarih alanı için yapılan deneylerde, METİN için %96.06 kesinlik, %95.83 hassasiyet; TÜR için %92.67 kesinlik, %92.44 hassasiyet değerlerine ulaşılmıştır. Coğrafya alanı için yapılan deneylerde, METİN için %96.59 kesinlik, %97.07 hassasiyet; TÜR için %93.37 kesinlik, %93.84 hassasiyet değerlerine ulaşılmıştır.

Sonuçlar birleştirildiğinde sistemin başarısı METİN için **%96.26** kesinlik, **%96.29** hassasiyet; TÜR için **%92.93** kesinlik, **%92.97** hassasiyet olarak ölçülmüştür. F-ölçütü değerleri ise (elde edilen kesinlik ve hassasiyet değerlerinin harmonik ortalaması alınarak) METİN için **%96.28**, TÜR için **%92.95** olarak ölçülmüştür.

Sonuçlar, coğrafya alanı için başarı oranının nispeten daha yüksek olduğunu ortaya çıkarmıştır. Ama bir tarih metninde bulunan ortalama varlık ismi sayısının, bir coğrafya metnindeki ortalama sayıdan yaklaşık 22 daha fazla olduğu da göz ardı edilmemelidir. İki alan için de METİN sonuçlarındaki doğruluğun TÜR'den yüksek olduğu (hem kesinlik hem hassasiyet değerleri için) görülmüştür. Bunun başlıca nedeni, varlık ismi sınırlarının doğru belirlenmemesi durumunda, türünün tahmin edilmesinin gerçekleştirilemez bir göreve dönüşmesidir. Kaynak listelerde yer alan çokanlamlı kelimeler ve cins isim olarak da kullanılabilen kişi isimleri, hatalı tahminlere yol açan iki diğer sebep olarak gözlemlenmiştir.

Çizelge 12, doğru tahmin edilen varlık ismi türlerinin iki alan için dağılımını göstermektedir. Diğer, Yer İsmi (Ülke – Devlet), Tarih, Kişi İsmi (Türk) ve Tarihi Terim (Olay İsmi), tarih metinlerinde en sık karşılaşılan beş varlık ismi türü olarak belirlendi. Yer İsmi (Ülke – Devlet), Yer İsmi (Diğer), Coğrafi Terim (Oluşum İsmi), Diğer ve Tarih veya Sayı ise coğrafya metinlerinde en sık karşılaşılan beş varlık ismi türü olarak belirlendi. Coğrafya metinlerinde hiç Kişi İsmi (Türk), tarih metinlerinde ise hiç Coğrafi Terim (Olay İsmi) etiketli varlık ismi olmaması dikkat çekici bir diğer sonuç olarak gözlemlendi. Yer İsmi (Ülke – Devlet), tüm deney kümesi içerisinde en homojen dağılmış varlık ismi türü olarak belirlendi.

4.2. Milliyet Veri Kümesi Deneyleri

Tarih ve coğrafya alanındaki ders metinleri için tasarlanmış sistemin, alan değişiminden ne düzeyde etkileneceğini görmek için, Türkçe VİT çalışmalarında sıklıkla kullanılan Milliyet veri kümesi üzerinde de deneyler yapılmıştır. Bu deneylerde, önerilen sistemin aksine varlık ismi türleri kişi, yer, kurum ve zamansal ifade ile sınırlandırılmıştır. Test kümesi ekonomi, siyaset, magazin, spor, eğlence gibi farklı kategorilere ait gazete metinleri içermektedir. Toplam 17215 kelime ve 1648 varlık ismi (623 kişi, 396 yer, 447 kurum ve 182 zamansal ifade olmak üzere) bulundurmaktadır. Çizelge 13, bu test kümesi üzerinde gerçekleştirilen deney sonuçlarını göstermektedir.

Çizelge 13. Milliyet Veri Kümesi Deney Sonuçları

	Kesinlik (%)	Hassasiyet (%)	F-Ölçütü (%)
METİN	91.46	94.17	92.80
TÜR	84.38	86.89	85.62

Alan değişikliğinin sistem başarısını bir miktar düşürdüğü gözlemlenmiştir. Sistemin haber metinleri üzerindeki başarısı f-ölçütü ile METİN için **%92.80**, TÜR için **%85.62** olarak ölçülmüştür. METİN sonuçlarındaki başarının TÜR'den (hem kesinlik, hem hassasiyet değerleri için), hassasiyet sonuçlarındaki başarının da kesinlikten yüksek olduğu belirlenmiştir. Varlık ismi türlerine göre başarı incelendiğinde ise, hem METİN hem TÜR değerinin doğru tespit edilme oranı kişi isimleri için %**78.01**, yer isimleri için **%93.18**, kurum isimleri için **%79.64,** zamansal ifadeler için **%98.35** olarak ölçülmüştür.

A. 4.3. Karşılaşılan Güçlükler

Sistemin geliştirilmesi süresince, Türkçe dilinin yapısından ya da girdi metni dosyalarında ihlallerden kaynaklı bazı güçlük ve kısıtlamalar ile karşılaşıldı.

Şekil 3'te verilen kullanım senaryosu örneğinde, "Bornova, İzmir, Atatürk, Gündoğdu" sözcük birimleri sözlüksel ifadeler olarak etiketlenmiştir ve farklı bir metin içeriğinde kendi başlarına da birer varlık ismi ifade edebilirler. Sistem bu gibi durumlarda kapsayıcı terimi dikkate alacak şekilde tasarlanmıştır. Şekilde görüldüğü üzere, bu terimler isabetli bir şekilde, daha uzun varlık isimlerinin parçası olarak görülmüşlerdir.

Türkçe kişi isimlerini tespit etmek için geniş bir sözlük yapısı kullanmak, hassasiyet kriterini olumlu etkilese de kesinlik kriterine olumsuz etkisi olması muhtemeldir. Bunun sebebi, bazı Türkçe kişi isimlerinin, ders metinlerinde sıkça karşılaşılan cins

isimleri işaret edebiliyor oluşudur. Bu duruma "Savaş, Barış, Nehir, Irmak" gibi kelimeler örnek olarak verilebilir. Bu ifadeler cümlenin başında olduğunda, komşu kelimelerin kontrol edilmesi büyük oranda sorunu çözmektedir; ancak tek başına yeterli olmadığı durumlar görülebilir. Örneğin, CONJ_SWC listesi cümlenin bir bağlaç ile başlayıp, bir varlık ismi ile devam ettiği durumlarda hataların önüne geçmek için oldukça önemli bir kaynaktır.

"Sultan, Şah" gibi bağlamsal modelde kullanılan bazı ifadeler, bir kişi isminden önce de sonra da karşımıza çıkabilir; hatta bazen "Kanuni Sultan Süleyman" gibi ifadelerde iki durum aynı anda gerçekleşebilir. Sistem iyileştirmeler yapılmadan önce, bu gibi durumlarda çıktı olarak iki farklı varlık ismi ("Kanuni Sultan" ve "Sultan Süleyman" şeklinde) vermeye meyilliyken; bu durum düzeltilerek yarım ifadelerin birleştirilip tek ve doğru varlık isminin verilmesi sağlanmıştır.
Başlıklar, varlık ismi barındırma ihtimali yüksek olduğu için sistem tarafından değerlendirmeye alınır. Ama genel kullanımda, başlık metinlerindeki bütün kelimelerin (eğer bağlaç değillerse) ilk harfleri, özel isim olmasa dahi büyük yazılmaktadır. Bu durum hatalı varlık ismi tespitlerine yol açabilmektedir. Bunu önlemek adına sisteme, üzerinde çalıştığı metnin bir başlık mı yoksa bir cümle mi olduğu bilgisi verilerek; eğer başlık ise tanıyıcı modelde "Diğer" etiketi için yapılan kontroller çalıştırılmamaktadır. Yine de, kesme işareti kontrolleri yapılmaya devam edildiği için "Diğer" etiketli bir varlık isminin tespit edilmesi ihtimali ortadan kalkmış değildir.

Genelde literatürde "Kişi İsmi" şeklinde kullanılan varlık ismi türünün "Kişi İsmi (Türk)" ve "Kişi İsmi (Yabancı)" şeklinde ikiye ayrılmasının bazı durumlarda, iki etiket birleştirilse gerçekleşmeyecek hatalı TÜR sonuçlarına yol açtığı gözlemlendi. Bunun sebeplerinden en önemlisi "Musa, Enver, Zeynel, Süleyman" gibi Türkçede kullanılan bazı kişi adlarının, Arap ülkelerinde de kullanılıyor olmasıdır. Yine de deney sonuçları ve yapılan ayrımın gelecekte sisteme sağlayabileceği faydalar baz alındığında, bu durumdan kaynaklı performans kaybı kabul edilebilir düzeydedir.

Girdi olarak sisteme verilen metinlerdeki önemli noktalama işareti eksiklikleri (kesme işareti ve virgül gibi) ve yazım hataları sistem başarısını aşağıya çekecektir. Aynı zamanda bulunan varlık isimlerinin kalitesini düşürecek, "Diğer" etiketi alan varlık ismi sayısını arttıracaktır. Bu nedenle metin dosyalarının sistem kullanımına sunulmadan önce yazım denetiminden geçirilmesi tavsiye edilir.

5. Sonuçlar ve Öneriler

Bu çalışmada, kapsamı tarih ve coğrafya alanları olarak belirlenen Türkçe ders metinleri için kural tabanlı bir VİT modeli geliştirilmiştir. Sistem girdi olarak bir metin dosyası alıp, metin içeriğini tarayarak varlık isimlerini tespit edecek ve bulguları çıktı olarak sunacak şekilde tasarlanmıştır. Geliştirilen model ve modelin kullanımı için oluşturulan sözlüksel kaynaklar, Dokuz Eylül Üniversitesi Doğal Dil İşleme (NLP) sunucusu bünyesinde tutulmaktadır.

Sistemin başarısı, Tarih ve Coğrafya ders metinleri üzerinde yapılan deneyler ile ölçülmüştür. 30 tarih ve 30 coğrafya metni rastgele seçilmiş, değerlendirmeler kesinlik ve hassasiyet kriterleri ile METİN ve TÜR nitelikleri üzerinden yapılmıştır. Sonuç olarak sistemin başarısı METİN için %96.26 kesinlik, %96.29 hassasiyet; TÜR için %92.93 kesinlik, %92.97 hassasiyet olarak ölçülmüştür.

5.1. Çalışmanın Eğitimsel Değeri

Çalışma doğal dil işleme, bilgi çıkarımı, metin madenciliği ve bilişimsel dilbilim alanlarının kapsamında olmakla beraber; bilgisayar destekli bir eğitim yazılımı olarak değerlendirmek de mümkündür.

Araştırmaya başlarken konulan birincil hedefler değerlendirildiğinde, deneylerden büyük oranda tatmin edici sonuçlar elde edilmiştir. Bu durum, geliştirilen VİT modelinin, uzun vadeli eğitimsel hedef olan tarih ve coğrafya alanlarında kullanılabilecek nitelikli ve esnek yapılı terimler sözlükleri elde etmek için uygun bir yardımcı araç olabileceğini ortaya koymuştur.

Çalışmada 13 farklı varlık ismi türü tanımlanmıştır. Bu sayede sistem, oldukça geniş bir "tarihi terim" ve "coğrafi terim" sınıflandırması yerine, terimler için daha anlamlı bir tasnif modeli önermektedir. Terimler sözlüğü yapıları, soru niteliği taşıyan varlık isimlerinden meydana geleceği için; bu yapıların sınav hazırlama süreçlerine yardımcı olabileceği öngörülmektedir.

5.2. İyileştirme Olanakları

Sonuçların kalitesini arttırmak adına, "Diğer" etiketli varlık isimlerinin sayısını azaltmak hedeflenebilir. Bunun için de ilave varlık ismi türleri tanımlanabilir. Örneğin, "Diğer" etiketi alan tarih metinlerindeki varlık isimlerinin büyük bir bölümünün millet, milliyet anlamı taşıyan ifadeler olduğu görülmüştür. Bu ifadelerin yeni ve daha anlamlı bir türün kapsamına alınmasının üzerinde durulabilir. Sözlüksel kaynaklar, antik çağ yer ve kişi isimlerini de barındıracak şekilde genişletilebilir. Noktalama işareti eksikliğinden kaynaklı olumsuz etkiyi azaltmak için, bir yazım denetimi biriminin sisteme dahil edilmesi düşünülebilir.

Teşekkür

Bu makale Dokuz Eylül Üniversitesi Bilimsel Araştırma Projeleri Koordinasyon Birimi (DEÜBAP) tarafından 2018.KB.FEN.015 numarasıyla desteklenen proje çalışması kapsamında hazırlanmıştır.

Kaynakça

[1] Jurafsky, D., Martin, J.H. "*Speech and language processing (2nd Edition)"*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA. (2009)

[2] Grishman, A., Sundheim, B. "Message Understanding Conference-6: a brief history". In Proceedings of the 16th conference on Computational linguistics - Volume 1 (COLING '96), Vol. 1. Association for Computational Linguistics, Stroudsburg, PA, USA, 466-471. (1996)

[3] Cucerzan, S., Yarowsky, D. "Language independent named entity recognition combining morphological and contextual evidence". Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora. New Brunswick, NJ: Association for Computational Linguistics. (1999)

[4] Alfonseca, E., Manandhar S. "An unsupervised method for general named entity recognition and automated concept discovery". In 1st International Conference on General WordNet. (2002)

[5] Tür, G., Hakkani-Tür G., Oflazer K. "A statistical information extraction system for Turkish". Natural Language Engineering, vol. 9 (2), pp. 181-210 (2003)

[6] Sang, E., Meulder F. "Introduction to the CoNLL-2003 shared task: language-independent named entity recognition". In Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003 - Volume 4 (CONLL '03), Vol. 4. Association for Computational Linguistics, Stroudsburg, PA, USA, 142-147 (2003)

[7] Wentland, W., Knopp, J., Silberer, C., Hartung, M. "Building a multilingual lexical resource for named entity disambiguation, translation and transliteration". in Proceedings of the 6th International Conference on Language Resources and Evaluation, Marrakech, Morocco. (2008)

[8] Küçük, D., Yazıcı, A. "Rule-based named entity recognition from Turkish texts". In Proceedings of the International Symposium on Innovations in Intelligent Systems and Applications, Trabzon, Turkey. pages 456–460. (2009)

[9] Küçük, D., Yazıcı, A. "Named entity recognition experiments on Turkish texts". In Proceedings of the 8th International Conference on Flexible Query Answering Systems, FQAS '09, pages 524–535, Berlin, Heidelberg. Springer-Verlag. (2009)

[10] Tatar, S., Çiçekli, İ. "Automatic rule learning exploiting morphological features for named entity recognition in Turkish". Journal of Information Science, 37 (2), 137-151. (2011)

[11] Küçük, D., Yazıcı, A. "A hybrid named entity recognizer for Turkish with applications to different text genres". In: Gelenbe E., Lent R., Sakellari G., Sacan A., Toroslu H., Yazici A. (eds) Computer and Information Sciences. Lecture Notes in Electrical Engineering, vol 62. Springer, Dordrecht. (2012)

[12] Şeker, G. A., Eryiğit, G. "Initial explorations on using CRFs for Turkish named entity recognition". In Proceedings of COLING 2012, Mumbai, India. (2012)

[13] Küçük, D., Jacquet, G., Steinberger, R. "Named entity recognition on Turkish tweets". In: Language Resources and Evaluation Conference. (2014)

[14] Küçük, D., Küçük, D., Arıcı, N. "A named entity recognition dataset for Turkish". In: 24th Signal Processing and Communications Applications Conference (SIU), Zonguldak, Turkey. (2016)

[15] Şeker, G., Eryiğit, G. "State of the art in Turkish named entity recognition". https://pdfs.semanticscholar.org/7e7f/ed9d21a3e3a36c4eb3c7df1ee8116e8ec2ce.pdf (2016)

[16] Ertopçu, B., Kanburoğlu, A., Topsakal, O., Açıkgöz, O., Gürkan, A., Özenç, B., Çam, İ., Avar, B., Ercan, G.,

Yıldız, O. “A new approach for named entity recognition”. In: International Conference on Computer Science and Engineering (UBMK), Antalya, Turkey. (2017)

[17] H. B. Şahin, C. Tirkaz, E. Yildiz, M. T. Eren, and O. Sonmez, “Automatically annotated turkish corpus for named entity recognition and text categorization using large-scale gazetteers”.arXiv preprint arXiv:1702.02363, (2017)

[18] Güneş, A., Tantuğ, A. C., “Turkish named entity recognition with deep learning”. 26th Signal Processing and Communications Applications Conference (SIU). doi:10.1109/siu.2018.8404500 (2018)

[19] Güngor, O., Üsküdarlı, S., Güngor, T., “Recurrent neural networks for Turkish named entity recognition”. 26th Signal Processing and Communications Applications Conference (SIU). doi:10.1109/siu.2018.8404788 (2018)