

# Doğal Dil İşleme Uygulamaları İçin Türkçe Veri Seti Oluşturulması

## Turkish Data Set for Natural Language Processing Applications

Tarık ŞAHİN<sup>1</sup> , Önder DEMİR<sup>2</sup> , Kazım YILDIZ<sup>3</sup> 

<sup>1</sup>Marmara Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği, 34722, Göztepe Kampüsü, Kadıköy, İstanbul,

<sup>2,3</sup>Marmara Üniversitesi, Teknoloji Fakültesi, Bilgisayar Mühendisliği, 34722, Göztepe Kampüsü, Kadıköy, İstanbul,

### Öz

Bu çalışma cümleden anlam çıkarılması ve Türkçe metinlerin çizge veri yapısında temsil edilmesi ile ilgili yaklaşımları ele almaktadır. İnsanların makinelerle etkileşiminin giderek artmasıyla, makinelerin de insanlar gibi davranışlar göstermesi giderek önem kazanmaktadır. Böylece makineler insanlara daha fazla yardımcı olabilecektir. İnsana ait en karakteristik davranışlardan birisi de konuşabilmektir. Konuşmanın en temel yapı taşlarından birisi de cümlelerdir. Makineler insanların konuştuğu cümleleri anlamlandırabilirse tahminleme, makinelerin kişisel asistanlığı gibi konularda büyük ilerleme kaydedilecektir.

Metinlerin SQL veri tabanı yerine çizge veri tabanı üzerinde temsili ise, metinlerin analizinde kullanıcılara hız kazandıracaktır. SQL veri tabanında kütle halinde kaydedilmiş içerikler yerine, Neo4j veri tabanında çizge veriyapısında kaydedilmiş içerikleri analiz etmek daha hızlı ve daha basittir. Ayrıca bu yaklaşım arama motoru altyapısı ve konuşma robotu altyapısı gibi konulara da öncülük edebilecektir.

**Anahtar Kelimeler :** Doğal dil işleme, kişisel asistan programı, cümleden anlam çıkarımı, Neo4j, çizge veri tabanı, çizge veri yapısı

### Abstract

This study deals with approaches to extract meaning from sentences and represent text based Turkish contents on graph data structure. With the increasing interaction of people with machines, it is becoming more and more important for machines to behave like humans. So machines can be more helpful to people. One of the most characteristic abilities of humanity is to talk each other. Sentences are one the most basic parts of talking. If the machines can make sense of the sentences that people speak, great progress will be made on issues such as prediction or personal assistance of machines.

Presantation of texts on graph databases instead of SQL databases provides faster environment for users. It is faster and simpler to analyze content saved in the graph data structure in the Neo4j database, rather than saved as bulk content in the SQL database. Also this approach can lead the studies such as search engine or speach robots.

The importance of interaction between machines and people is also seen through films. The war between machines and people, or films where people communicate with people by voice, attracts a lot of attention. In the future, perhaps for people, machines can pass in place of a friend. In order for this to happen, it is necessary to find the emotion expressed by the sentence.

The processing of the sentences will also help to form decision support systems. Systems that can process the texts consisting of sentences according to the meaning extracted from the sentence will both provide analytical results in the desired field and allow for predictions about the future.

Graphical database is a structure suitable for social network analysis. Since there are relations such as friendship and follow-up among people in social networks, their representation on the graph is quite logical and easy. Although social networks have many different expressions and models, the most widely accepted notation in the literature is the representation of individuals or entities as nodes and relationships as sides. In this article, the sentence analysis will be discussed on the suitability of graph data structure, just like in social networks, and the studies done and how the sentences are represented on the graph database will be discussed. In order for machines to learn a spoken language such as people, it must be modeled correctly on the machine. The most appropriate data structure should be selected and implemented.

When looking at the order of the words that make up the sentence, the relationship between each other in terms of meaning, it can be seen that the sentences represent a graph of words. The words are linked to each other in terms of meaning. Considering the order of arrival of words before or after each other, it can be seen that the sentences are in directional graph structure.

In order to analyze the sentences in terms of meaning, there must be certain pattern examples. The sentences that match these patterns can be loaded with various meanings (positive, negative, etc.) so that the language is taught to the machine from the sample data set. This method is an example of supervised learning.

Choosing a specific subject and preparing the data set to form the patterns and the fact that the sentences to be evaluated belong to this field will give healthier results. These fields may be subjects such as economy, sports, health.

If the experts of the selected subject and technical experts work together to create a sample data set, more efficient results will be achieved. Countries should create national, open source data sets according to their official languages. For this purpose, both an expert technical team and experts related to the subject of data set should be gathered. One of the most important points here will be to build the infrastructure where the data will be stored as the data set will be national and public. Since the data will be accessible to everyone, the most optimal solutions should be found. Such a study in the countries will be a big step in natural language processing. It will also prepare the ground for new perspectives and approaches to extracting meaning from the sentence.

In this study, it is tried to develop a different point of view in terms of extracting meaning from natural sentence processing. Various examples were performed using Neo4j from graph databases. Although these studies seem to be applicable in relational-based databases, they will bring many difficulties. However, how flexible and easy to query the sentence patterns represented in graph databases was demonstrated in this study.

Graph databases are also very effective in speed. Therefore, it is predicted that real-time performances can be achieved when sentence analysis is performed with the methods mentioned in this article.

**Keywords :** natural language processing, personel assistant program, extract meaning from sentence, Neo4j, graph database, graph data structure

## 1. GİRİŞ

Ülkemizde ve dünyada doğal dil işleme konusu giderek popüler hale gelmektedir. Özellikle akıllı telefonların da hayatımıza girmesiyle, kişisel asistanlık yapan, insanlardan aldığı sesli komutları yerine getiren uygulamalar bu konuya ilgiyi artırmaktadır. Google tarafından geliştirilen Google Asistan ürünü bu uygulamalara güzel bir örnektir[1].

Ülkemizde bu alanda akademik çalışmalar özellikle İstanbul Teknik Üniversitesi tarafından ciddi biçimde ele

alınmıştır. Üniversitenin ITU Türkçe Doğal Dil İşleme Yazılım Zinciri adlı uygulaması yapılan çalışmaları görebilmek açısından güzel bir ortam sunmaktadır.[2]

Türkçe yazım yanlışlarının bulunması ile ilgili olarak Aynur DELİBAŞ yaptığı çalışmada kelimelerin köklerinin ve eklerinin bulunması ile ilgili çeşitli çalışmalar yapmıştır[3]. Burada kullandığı algoritmalarda hem ağaç hem de çizge veri yapısından bahsetmektedir. Ayrıca dokümanda dikkat edilmesi gereken bir diğer husus, Türkçe ile ilgili gramer yapısı ve kurallar hakkında oldukça detaylı bilgiler verilmektedir. Eğer doğal dil işleme ile ilgili bir çalışma yapılacaksa bu dilin özelliklerinin de çok iyi bilinmesi gerekmektedir.

Bir diğer yüksek lisans tezi çalışmasında Gülşen CEBİROĞLU Sözlüksüz Köke Ulaşma Yönteminden bahsetmiştir[4]. Bu çalışma özellikle ilgi çekicidir. Çünkü doğal dil işlemeden bahsediliyorsa bir kelimenin köküne ulaşmak için neler yapılacağı önemli bir unsur olacaktır. Bu çalışmada sonlu durum makinelerinden bahsedilmiştir(Finite automa-tion).

Metin BİLGİN ve Mehmet Fatih AMASYALI'nın yazdığı "bileşik cümlelerde yan cümleciklerin otomatik etiketlenmesi"[5] makalesinde, Lafferty ve arkadaşları tarafından önerilen istatistiksel dizilim sınıflandırmasına dayanan bir makine öğrenmesi yöntemi olan Şartlı Rastgele Alanlar (Condition Random FieldsCRF)'dan[6] bahsedilmekte ve bu yöntem kullanılarak Türkçe'de cümle içindeki yan cümleyi ayırt edebilmek için yöntemler sunulmaktadır. Olasılık temelli bu yöntem ile en olası etiket seçilerek yan cümle bulunmaya çalışılır.

Veriler çizge veri tabanında tutulacağı için buna uygun teknolojiler araştırılmıştır. Bunun için web tabanlı bir analiz programından faydalanılmıştır[16]. Bu program ile çeşitli teknolojiler karşılaştırmalı olarak ele alınmıştır. Fernandes ve Bernardino tarafından yapılan araştırma da önemli bilgiler edinilmesini sağlamıştır[17]. Ayrıca çizge teoremi üzerine de araştırmaları incelemenin faydalı olacağı düşünülmüştür. SEKER tarafından hazırlanan çalışmada bu teoriden bahsedilmektedir. Bu teoriye göre problemler düğümler ve kenarlar kümesi şeklinde temsil edilerek modellenilebilir[18].

Çizge veri tabanı ile geliştirilen BioGraph uygulaması da fikir vermesi açısından faydalıdır. Bu çalışmada bir çok farklı kaynaktan beslenen biyolojik verilerin analizine dair bir uygulama çizge veri tabanı ile desteklenerek geliştirilmiştir[19].

## 2. MATERYAL VE METOT

### 2.1 Cümleden Anlam Çıkarabilmenin Önemi

Giriş bölümünde kelime işlemeyle ilgili bahsedilen çalışmaların yanında, cümleden anlam ve duygu çıkarılabilmesine yönelik çalışmalar da günümüzde ihtiyaç haline gelmiştir. İnsanlar giderek makinelerle daha fazla etkileşime girmektedir. Alpaslan Burak Eliaçık ve Nadia Erdoğan bu makalede anlatılan çalışmaya benzer duygu analizi çalışmasından bahsetmişlerdir[7]. Bu çalışmaya göre duygu sınıflandırması yöntemi iki ana dala ayrılmaktadır; makine öğrenmesi tabanlı yaklaşım ve sözlük tabanlı yaklaşım. Makine öğrenme tabanlı yaklaşım sınıflandırma sırasında makine öğrenme algoritmalarını ve dilbilimsel özellikleri kullanmaktadır. Sözlük tabanlı yaklaşım ise sınıflandırma sırasında önceden hazırlanmış duygu kavramlarından oluşan sözlüklerden yararlanmaktadır. Bazı çalışmalarda ise bu iki yönelimin melez bir yaklaşımı kullanılmaktadır.

Makinelerle insanların etkileşiminin önemi filmler aracılığıyla da görülmektedir. Makinelerle insanlar arasındaki savaş veya makinelerle insanların sesli iletişim kurduğu filmler oldukça ilgi çekmektedir. İlerleyen zamanlarda belki de insanlar için makineler birer arkadaş yerinde geçebilecektir. Bunun gerçekleşebilmesi için cümlenin ifade ettiği duyguyu bulabilmek gerekir.

Karar destek sistemleri, günümüzde önemli bir konu olmuştur. Tıp alanından örnekler verilebilir. “Tıbbi Karar Destek Sistemlerinin Yöntemsel Olarak Değerlendirilmesi Üzerine Bir Çalışma[8]” adlı makalede değişik hastalık tanılarına ilişkin olarak geliştirilmiş tıbbi karar destek sistemlerinin, içerdiği teknikler açısından incelemesi yapılmış ve bu tekniklerin performansları değerlendirilmiştir.

Cümlelerin işlenebilmesi karar destek sistemleri oluşturma konusuna da yardımcı olacaktır. Cümlelerden oluşan metinleri, cümleden çıkarılan anlama göre işleyebilen sistemler, istenilen alanda hem analitik sonuçlar elde etmeye yarayacak hem de gelecekle ilgili yapılacak tahminlere olanak sağlayacaktır.

### 2.2 Cümle Analizi İçin Çizge Veri Yapısının Kullanılması

Çizge veri tabanı yapısal olarak bakıldığında sosyal ağ analizi yapmaya uygun bir yapıdadır. Sosyal ağlarda insanlar arasında arkadaşlık, takip gibi ilişkiler olduğu için bunların çizge üzerinde temsili oldukça mantıklı ve kolay olmaktadır. Sosyal ağların bilimsel olarak çok farklı ifade ve modelleri olmakla birlikte, literatürde en çok kabul görmüş gösterim şekli çizge teoremi (graph theory) kullanılarak bireylerin veya varlıkların birer düğüm (node) ve ilişkilerin birer kenar (edge) olarak tasvir edildiği gösterimdir[9].

Bu makalede cümle analizinde de tıpkı sosyal ağlarda olduğu gibi çizge veri yapısının uygunluğu üzerine tartışılacak, yapılan çalışmalardan ve cümlelerin çizge veri tabanı üzerinde nasıl temsil edildiğinden bahsedilecektir.

Makinelerin insanlar gibi bir konuşma dili öğrenebilmesi için, o dili makine üzerinde doğru modellemek gerekir. En uygun veri yapısı seçilmeli ve uygulanmalıdır.

Cümle; “Bir düşüncüyü, bir duyguyu, bir olayı, bir hareketi, bir isteği, bir yargı biçiminde anlatan kelimeler topluluğudur.” şeklinde tanımlanabilir[10]. Cümleleri kelimeler oluşturur. Bu kelimelerin arasında ise dil bazında bağlantılar vardır. Bazıları tek başlarına cümle oluşturabilir. Edat gibi yapıların sözlük anlamları yoktur. Bazı kelimeler ise tek başlarına bir anlam ifade etmezler. Ancak başka kelimelerle birlikte kullanıldığında anlamlı bir bütün olurlar. Örneğin “etmek” kelimesi tek başına anlamsızdır. Fakat “tebrik etmek” denildiğinde bu iki kelime birbirini tamamlar ve ortaya anlamlı bir fiil çıkar. Bunun gibi örnekler çoğaltılabilir.

Cümleyi oluşturan kelimelerin diziliş sırası, birbirleriyle olan anlam bakımından ilişkileri gibi unsurlara bakıldığında, cümlelerin kelimelerden oluşan bir çizge(graph) ifade ettiği görülebilir. Kelimeler birbirlerine anlam bakımından bağlı durumdadır. Kelimelerin birbirlerinden öce veya sonra geliş sıralaması dikkate alındığında, cümlelerin yönlü çizge yapısında olduğu görülebilir. Şekil 1 de cümlenin çizge veri yapısıyla temsiline ilişkin örnek sunulmuştur.



Şekil 1. Cümlenin yönlü çizge şeklinde temsili

Günümüzde giderek gelişen No-SQL veritabanları, çizge veri yapısının hem hızlı, hem güvenli bir şekilde temsil edilebilmesine olanak sağlamaktadır. Bu gelişmeler de dikkate alındığında cümleleri temsil edebilmek için en uygun veri yapılarından birinin çizge olduğu söylenebilir.

### 2.3 Cümle Analiz Yöntemleri

Cümleleri anlam bakımından analiz edebilmek için elde belli başlı örüntü(pattern) örneklerinin olması gerekir. Bu örüntülerle eşleşen cümlelere çeşitli anlamlar yüklenebilir(olumlu, olumsuz vs.). Burada örnek veri setinden hareketle makineye dil öğretilmiş olunur. Bu yöntem gözetimli(-supervised) öğrenmeye örnek gösterilebilir.

Bu yöntem bazı örnekler ile anlatılabilir. “Kendisine verilen işi kolayca yaptı.” Bu cümlede altı çizili kelime grubu bir örüntü olarak kabul edilebilir. Bu örüntü cümleye olumlu

bir anlam katmaktadır. Cümledeki bu tarz örüntüleri bulabilmek için zengin bir örnek veri gereklidir. "kolay" ve "yap" kelime köklerinden türetilebilecek örüntülere örnek veri setinden erişebilmek mümkün olmalıdır. Örnek veri setinde kelime kökleri ve bu köklere gelen eklerle elde edilebilecek örüntüler bulunmalıdır. Aşağıda birkaç örnek örüntü sunulmaktadır.

Kolay – ca yap – tı. à anlam bakımından olumlu  
 Kolay – lık – layap – tı. à anlam bakımından olumlu  
 Kolay – ca yap – abil – di. à anlam bakımından olumlu  
 Zor – luk – la yap – tı. à anlam bakımından olumsuz

Örnekte görüldüğü üzere cümlelerin içinden bulunacak örüntülerden yola çıkılarak cümlenin anlam bakımından olumlu veya olumsuz olmasına karar verilebilecek mekanizmalar oluşturulabilir. Ayrıca bu örüntüler duygu tespiti gibi amaçlar için de kullanılabilir.

## 2. 4 Neo4j Çizge Veri Tabanı Kullanarak Veriyi Modelleme

### 2. 4. 1 Neo4j Hakkında

Neo4j, No-SQL sınıfına giren çizge veri yapısını temsil eden bir veri tabanıdır[11]. Sorgu diline Cypher denmektedir. Bu sorgu dilinin en önemli özelliği, kullanılan notasyon ve yapı itibarıyla, bir metinden çok şekli andırmasıdır. Bu sebeple çizgeler üzerinde sorgulama yapmak ve bu sorguları anlamak SQL tabanlı veri tabanlarına göre daha kolay ve anlaşılır olmaktadır. Şekil 2 de bazı örnek sorgular gösterilmiştir.

```
//data stored with this direction
CREATE (p:Person)-[:LIKES]->(t:Technology)

//query relationship backwards will not return results
MATCH (p:Person)<[:LIKES]-(t:Technology)

//better to query with undirected relationship unless sure of direction
MATCH (p:Person)-[:LIKES]-(t:Technology)
```

Şekil 2. Cypher diline ait bazı sorgu örnekleri[12]

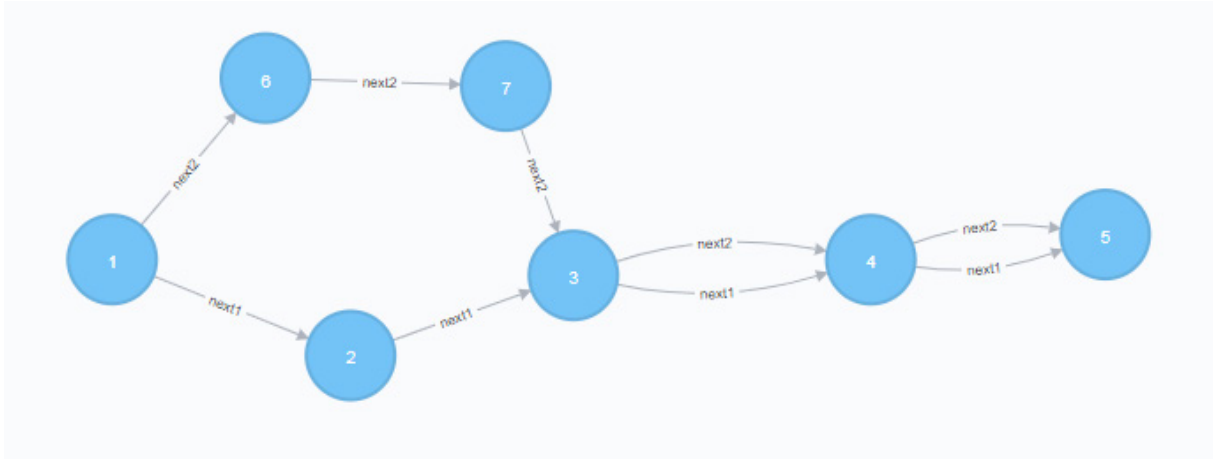
Şekil 2'deki sorgu örnekleri incelenirse, şekil 1'de gösterilen temsili çizge yapısının Neo4j veri tabanında kolaylıkla sağlanabileceği görülecektir.

Dil biliminde, büyük miktarda metin içeren yapılandırılmış doküman kümelerine derlem denir. Derlemler üzerinde söz dizimsel sorgular yapılabilmesi için derlemlerdeki cümlelerin ilişki çizgeleri bir çizge veri tabanına kaydedilmiştir. Bunun için yüksek oranda bağlantılı verileri saklamakta kullanılan bir çizge veri tabanı olan Neo4j kullanılmıştır. Neo4j, verileri düğümler ve bağlantılar şeklinde saklayan açık kaynaklı bir çizge veri tabanı olup, düğümlere ve bağlantılara etiket ve özellikler verilebilmektedir[13].

Çizge veri tabanları, verinin dinamik olarak saklanıp işlenmesine olanak sağlayan veri tabanı sistemleridir. Bu doğrultuda, sistemde esneklik gerektiren, aralarında çok sayıda ilişki bulunduran ve değişen yapıya sahip verilerin çizge veri tabanında tutulup işlenmesi fayda sağlamaktadır[14].

### 2. 4. 2 Örnek Veri Seti Üzerinden Örüntü Bulma

Cümle analiz yöntemleri başlığı altındaki örnek cümle, örnek veri tabanı üzerinde analiz edilebilir. Veriyi Neo4j üzerinde temsil ederken sonraki gelecek olan kelime veya ek bilgisi için "next" sözcüğüyle birlikte, gidilen yolu(path) ifade eden bir numara kullanılmıştır. Örüntünün sonuna ise anlamca olumlu veya olumsuz olduğunu belirten "positive" veya "negative" düğümü(node) eklenmiştir. Böylece sorgulamak daha kolay olacak ve örüntüler birbirine karışmayacaktır. Şekil 3 de "kolayca yaptı" ve "kolaylıkla yaptı" örüntülerinin örnek modellemeleri gösterilmiştir. Şekil 4 de ise metin tabanlı gösterimi sunulmuştur.



Şekil 3. Örnek modelleme (Neo4j görselleştirme ile)

<code>{"sequenceValue": "0", "key": "maxSequence"}</code>
<code>{"value": "kolay"}</code>
<code>{"value": "ca"}</code>
<code>{"value": "yap"}</code>
<code>{"value": "tı"}</code>
<code>{"value": "positive"}</code>
<code>{"value": "lık"}</code>
<code>{"value": "la"}</code>

Şekil 4. Örnek modelleme (Neo4j text seçeneği ile)

Doğru modelleme yapıldıktan sonra geriye doğru sorgularla örüntüleri bulup cümle içinde aramak kalmaktadır. Şekil 5 ve Şekil 6 da örnek bazı sorgular ve sonuçları göstermiştir.

```
$ match (a)-[n:next1*]->(b{value:'positive'}) return a.value as value
```

value
tı
yap
ca
kolay

Şekil 5. Anlamca pozitif örüntü bulduran sorgu ve sonucu

```
$ match (a)-[n:next2*]->(b{value:'positive'}) return a.value as value
```

value
tı
yap
la
lık
kolay

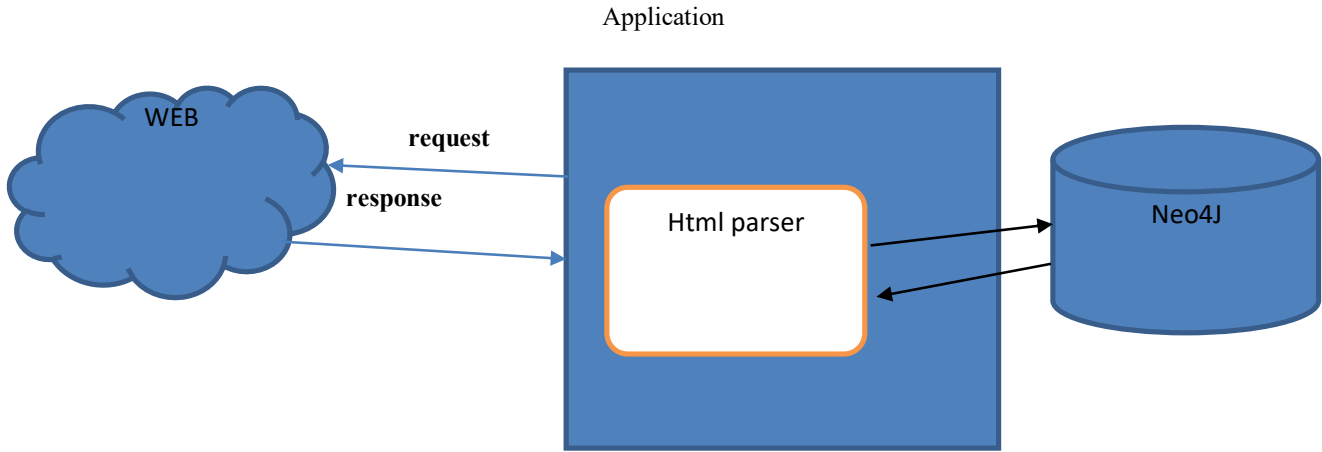
Şekil 6. Anlamca pozitif farklı yola (path) sahip başka bir örüntüyü bulduran sorgu ve sonucu.

Şekillerden anlaşılacağı gibi doğru bir modelleme ile istenen örüntüler hem esnek olarak saklanabilecek hem de basit sorgular kullanılarak kolayca erişilebilecektir.

## 2.5 Örnek Uygulama

Neo4J veri tabanını kullanarak web üzerinden haber çekip, bu veri tabanı üzerindeki örüntülerle cümleleri karşılaştıran bir uygulama geliştirilmiştir. Şekil 7'de uygulama mimarisinden bahsedilmiştir. Uygulama, web üzerinden bir haber adresine yapılan isteğin html ağacını parse ederek, analiz edilmek istenen içeriği saf metin haline getirir. Parse işlemi için javascript tabanlı jquery kütüphanesi gibi, html ağacını server tarafında parse etmeye yarayan kütüphaneler kullanılmıştır. Elde edilen saf metin Neo4j içerisindeki örüntülerle karşılaştırılıp sonuçlar ekrana yazdırılmıştır.





Şekil 7. Uygulama mimarisi

Örnek olarak bir futbol müsabakasında bir oyuncunun attığı gol ile ilgili cümle analiz edilmiştir. "meşin yuvarlağı filelere gönderdi" örüntüsü bir haber adresi içinde aratılmıştır. Bulunan sonuçlar Şekil8'de paylaşılmıştır.

#### POZİTİF EŞLEŞMELER

ESLEŞEN CÜMLE : Canterosun attığı topla defansın arkasında bulunan Boyd, düzgün bir vuruşla meşin yuvarlağı filelere gönderdi

ESLEŞEN PATTERN : meşin yuvarlağı fileler e gönder di

Şekil 8. Uygulama örnek ekran görüntüsü

### 3. BULGULAR VE TARTIŞMA

Örüntüleri oluşturacak veri seti hazırlanırken özel bir alan seçilmesi ve değerlendirilecek cümlelerin bu alana ait olması daha sağlıklı sonuçlar verecektir. Bu alanlar ekonomi, spor, sağlık gibi konular olabilir.

Örnek veri setini oluşturmak için seçilen alanın uzmanlarıyla, teknik uzmanlar bir araya gelip beraber çalışırsa daha verimli sonuçlar çıkacaktır. Ülkeler kendi resmî dillerine göre ulusal çapta, açık kaynak kodlu veri setleri oluşturmalıdır. Bunun için hem uzman bir teknik ekip hem de veri seti oluşturulacak alanla ilgili uzman kişiler bir araya gelmelidir. Burada en önemli noktalardan birisi de, oluşturulacak veri seti ulusal çapta olacağından verinin saklanacağı altyapıyı iyi kurgulamak olacaktır. Veriler herkes tarafından erişilebilir olacağı için en optimum çözümler bulunmalıdır. Ülkemizde yapılacak böyle bir çalışma doğal dil işleme konusunda büyük bir adım olacaktır. Cümleden anlam çıkarma konusunda yeni bakış açıları ve yaklaşımlara da zemin hazırlayacaktır.

### 4. SONUÇ VE ÖNERİLER

Bu çalışmada doğal dil işleme ile ilgili olarak cümleden anlam çıkarma konusunda farklı bir bakış açısı geliştirilmeye çalışılmıştır. Çizge veritabanlarından Neo4j kullanılarak

çeşitli örnekler gerçekleştirilmiştir. Bu yapılan çalışmalar ilişkisel tabanlı veritabanlarında da uygulanabilir gibi görünse de aslında bir çok zorluğu beraberinde getirecektir. Oysa çizge veritabanlarında temsil edilen cümle örüntülerinin ne kadar esnek ve kolay sorgulanabilir olduğu bu çalışmayla ortaya konulmuştur.

Çizge veritabanları hız konusunda da hayli efektiflerdir[15]. Bu nedenle cümle analizleri bu makalede bahsedilen yöntemlerle yapıldığında, gerçek zamanlı olarak da yapılabilecektir. Apache Spark veya Apache Kafka gibi büyük veri işlemeye yardımcı üçüncü parti yazılımlar kişilerin kendi oluşturduğu projelerine entegre edilerek daha etkili sonuçlar alınabilir. Bu tarz yardımcı uygulamalar anlık haber analizi veya sosyal medya analizi gibi uygulamalar geliştirmek mümkün olacaktır.

Cümlelerin çizge veri yapısıyla temsil edilmesiyle hem kelimelerin, hem cümlelerin birbirleriyle olan ilişkileri daha iyi anlaşılabilir. Veri tabanındaki cümleler alt çizgeler halinde birbirleriyle ilişkilendirilebilirse, birbirini takip eden ardışık cümle örüntüleri bulunabilir. Bu sayede gelişmiş konuşma robotları tasarlanabilir. Öğrenme veri seti oluşturulurken sorulan sorulara ne kadar çok alt çizge örüntüsü eklenirse o kadar alternatifli bir soru-cevap simülasyonu hazırlanmış olur.

Çizge üzerinde hızlı örüntü tarama algoritmaları sayesinde yüksek performans sağlayan arama motorları yapmak da mümkündür. SQL veritabanlarında kütle halinde tutulan metinler üzerinde performanslı arama yapabilmek pek mümkün değildir. Bu yüzden bu veriler üzerinde arama yapma ihtiyacı olduğunda veriler ayrıca bir başka ortamda saklanıp üzerinde arama yapabilecek uygulamalar kullanılmaktadır. Bu da fazladan maliyet getirmektedir.

Kelimelerin düğüm şeklinde tutulmasının bir avantajı da üzerlerine etiket şeklinde bilgiler eklenebilmesidir. Örneğin bir kelimenin sıfat, zarf veya zamir gibi özelliklere sahip olduğu belirtilebilir. Bu sayede cümleler sadece anlam bakımında değil, gramer bakımından da analiz edilebilir. Bu da Türkçe alanında faydalı olabilecek çalışmalara ışık tutabilir.

Yapılan bu çalışmada verinin saklanmasıyla ilgili bir bakış açısı sunulmaya çalışılmıştır. Bu bakış açısı sayesinde geliştirilebilecek başka uygulamalar konusunda fikirler verilmiştir. İhtiyaca göre başka fikirler de geliştirilebilir. Sunulan veri saklama yönteminin bir çok farklı ihtiyaca cevap verebileceği düşünülmektedir.

## Kaynaklar

- [1] Google assistant, [https://assistant.google.com/intl/tr\\_tr/platforms/phones/](https://assistant.google.com/intl/tr_tr/platforms/phones/) , Son erişim haziran 2019
- [2] Cebiroğlu G, 2013 , <http://tools.nlp.itu.edu.tr/Tokenizer> , Son erişim Haziran 2019
- [3] Delibaş, A. (2008, haziran). Doğal Dil İşleme İle Türkçe Yazım Hatalarının Denetlenmesi. Yüksek Lisans Tezi. İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, Türkiye, 35-36
- [4] Cebirgölu, G. (2002). Sözlüksüz Köke Ulaşma Yöntemi. İstanbul Teknik Üniversitesi. Yüksek Lisans Tezi. İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, Türkiye, 1-2
- [5] Bilgin M., Amasyali M.F. (2016). Bileşik Cümlelerde Yan Cümleciklerin Otomatik Etiketlenmesi. 18. Akademik Bilişim Konferansı, AYDIN, TÜRKİYE, 3-5 Şubat 2016, cilt.1, ss.1-4
- [6] Lafferty, J., McCallum, A. ve Pereira, F. (2001). Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. International Conference on Machine Learning (ICML), 28 June-1 July 2001, Massachusetts.
- [7] Eliaçık, A., Erdoğan, N. (2015). Mikro Bloglardaki Finans Toplulukları için Kullanıcı Ağırlıklandırılmış Duygu Analizi Yöntemi .UYMS, 782-793
- [8] Yurtay, Y. , Ak G., Bacıoğlu ,N,Z. (2013) . Tıbbi Karar Destek Sistemlerinin Yöntemsel Olarak Değerlendirilmesi Üzerine Bir Çalışma. ISITES2013 International Symposium On Innovate Technologies In Engineering Science,901-910
- [9] Seker, S,E.(2015). Sosyal Ağlarda Veri Madenciliği (Data Mining on Social Networks). YBS Ansiklopedi, 2(2), 30-39
- [10] A, Akçataş. (2007). Türkiye Türkçesinde yapı, işlev ve anlam ilişkileri açısından cümle grupları ve cümle türleri üzerinde bir deneme II. Türk Dili Araştırmaları Yıllığı – Bellekten,55(2007/2), 7-14
- [11] neo4j, <https://neo4j.com/> ,Son erişim haziran 2019
- [12] cypher, <https://neo4j.com/developer/cypher-query-language/> , Son erişim haziran 2019
- [13] Agan, C., Diri, B.(2016). Türkçe Derlemler İçinSöz Dizimsel Görselleştirme ve Sorgulama Aracı . Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi, 9(1) , 1-10
- [14] Karagöz G,N. , Komesli M.(2017). Çizge Veri Tabanı Kullanılarak Geliştirilen Yazılım Lisans Yönetimi Amaçlı Veri Görselleştirme Uygulaması: BigLogVis. Dokuz Eylül Üniversitesi-Mühendislik Fakültesi Fen ve Mühendislik Dergisi, 19(57), 779-789
- [15] neo4j, <https://neo4j.com/news/how-much-faster-is-a-graph-database-really/> , Son erişim haziran 2019
- [16] predictiveanalyticstoday , <https://www.predictiveanalyticstoday.com/top-graph-databases/> , son erişim temmuz 2019
- [17] Fernandes, D., Bernardino, J. (2018). Graph Databases Comparison: AllegroGraph, ArangoDB, InfiniteGraph, Neo4J and OrientDB. 7th International Conference on Data Science, Technology and Applications , 373-380
- [18] Seker, S.E. (2015) . Çizge Teorisi (Graph Theory) .YBS ansiklopedi, 2(2), 17-29
- [19] Messina, A., Fiannaca, A., Paglia, L., Rosa, M., Urso, A. (2018). BioGraph: a web application and a graph database for querying and analyzing bioinformatics resources. BMC Systems Biology,12(5),75-89