



## **Anlık Sorgu Geri-Getirimi Test Veri Kümesi Hazırlamak İçin Bir Web Aracı**

### **A Web Based Tool For Preparing Adhoc Retrieval Test Data**

**Tayfun Burak Aktürk<sup>1\*</sup>, İlker Kocabaş<sup>2</sup>**

<sup>1</sup> Manisa Celal Bayar Üniversitesi, Turgutlu Meslek Yüksekokulu, Bilgisayar Teknolojileri Bölümü, Turgutlu, Manisa, TÜRKİYE

<sup>2</sup> Ege Üniversitesi, Uluslararası Bilgisayar Enstitüsü, Bornova, İzmir, TÜRKİYE

Sorumlu Yazar / Corresponding Author \*: tbakturk@hotmail.com

Geliş Tarihi / Received: 13.03.2020

Kabul Tarihi / Accepted: 25.08.2020

Araştırma Makalesi/Research Article

DOI:10.21205/deufmd.2021236724

Atıf şekli/ How to cite: AKTÜRK, T.B., KOCABAŞ, İ.(2021). Anlık Sorgu Geri-Getirimi Test Veri Kümesi Hazırlamak İçin Bir Web Aracı. DEUFMD, 23(67), 277-284.

#### **Öz**

Bu çalışmanın amacı, bilgi erişimi alanında yazılı belge derlemleri kullanarak hazırlanan test veri kümeleri için sorgulara karşılık gelen belgelerin alakalı/alakasız yargısının yapılacağı bir web aracı geliştirmektir. Anlık sorgu geri-getirim için test veri kümesi bir belge topluluğu, belirli sayıda sorgular ve bu sorgulara karşılık gelen alakalı belgeleri içermektedir. Özellikle, belgelerin sorgu hedeflerine uygun olup olmadığının değerlendirilmesi işlemi oldukça karmaşık ve zaman alıcıdır. Sorguların alakalı olduğu belgeleri işaretleme işlemi dijital ortamda gerçekleştirecek bir uygulamaya ihtiyaç vardır. Çünkü çok sayıda belgenin alaka yargısı yapan kullanıcılar tarafından incelenmesi gerekmektedir. Bu çalışmada, TREC'te (Text Retrieval Conference) uygulanan yöntemi temel alarak geliştirilen araç ile bu süreç yönetilebilir hale gelmiştir.

**Anahtar Kelimeler:** Test veri kümesi, anlık sorgu geri-getirimi, TREC

#### **Abstract**

The main purpose of this study is to develop a web tool, which helps out on making relevant/non-relevant judgement of documents used in the test datasets prepared for text retrieval. The test dataset for the adhoc retrieval contains a collection of documents, a certain number of queries, and relevant documents corresponding to these queries. Especially, the process of judging whether documents fit query targets or not is quite complex and time consuming. There is a need for an application that will perform the process of marking the documents relevant to the queries in a digital environment. Because a large number of documents need to be examined by users who are concerned with relevance. In this study, this process has become manageable by the tool developed on the basis of the method applied in the TREC (Text Retrieval Conference).

**Keywords:** Test dataset, adhoc retrieval, TREC

#### **1. Giriş**

Bilgi ve iletişim teknolojilerinin hızla geliştiği günümüz dünyasında, insanların ihtiyaçları ve

bu ihtiyaçlara bulunan çözümler farklılık göstermektedir. Hemen her konuda ihtiyaç duyulan bilgi internet ortamında çeşitli

sitelerden veya bloglardan elde edilebilmektedir. Bilgi Erişimi (kıs. BE) araçları özellikle web-tabanlı arama motorları en popüler ve en sık kullanılan uygulamalardır.

Yazılı belgeler üzerinde bilgi erişimi işlemi kullanıcının sisteme bir sorgu girmesiyle başlamaktadır. BE sistemi ise ihtiyaç duyulan bilgiyi ifade eden bu sorguya karşılık gelen belgeleri belirli bir alaka ağırlığına göre sıralayarak getirmektedir. Sistemin geri-getirim başarımı erişilen belgelerin ihtiyacı ne ölçüde karşılayabildiğidir.

BE sistemlerinin başarımlarını ölçmek için literatürde birçok [1]-[4] ölçüt bulunmaktadır. Ancak başarı ölçütünden daha önemli temel konu, bir BE sisteminin nasıl test edileceğidir. Cranfield deneyleri [5] ile ortaya konulan bilgi erişimi başarımlarını ölçme test modeli, TREC (Text Retrieval Conference)[6] gibi büyük ölçekli sistemlerin değerlendirilmesi çalışmalarında kullanılmaktadır.

Bu çalışmada TREC test derlemlerinin alaka yargısı yapılma süreci bir web aracı olarak gerçekleştirilmiştir. İlerleyen bölümlerde ilgili süreç detaylandırılmış; aracın tasarımı, yapısı ve kullanım şekli anlatılmıştır. Türkçe yazılı belgeler üzerinde yapılan BE çalışmalarını geliştirmek için geçerli ve kapsamlı test veri kümeleri oluşturmak gerekmektedir. İlgili kişilerin bu araca web üzerinden <http://www.IR-lab.ege.edu.tr> adresinden erişimi mümkündür.

## 2. Motivasyon

Bilgi Erişimi (İng. information retrieval, kıs. BE) işi bir belge topluluğu içinden hedef duyulan bilgi ihtiyacını karşılayan, yani alakalı olan belgelere erişimi verimli hale getirmekle uğraşan bir bilim alanıdır. Bilgi ihtiyacı kullanıcılar tarafından belirli bir yapıda olmayacağı gibi önceden tanımlı bir yapıya uygun olarak da sorgu veya konu başlığı olarak ifade edilebilmektedir. Örnek olarak, sorgular herhangi bir doğal dilde veya mantıksal bir ifade kullanılarak oluşturulabilmektedirler.

BE alanındaki ilk çalışmalar doğal dilde yazılmış belge topluluklarında metin geri getirmeyi (İng. Text retrieval) üzerine yapılmıştır. Ancak; ses, müzik, resim ve video belgelerinde erişim gibi [7] alt problemlere de benzer kuramsal alt-yapı ile çözümler bulunmaya çalışılmaktadır. 'Bilgi erişimi' süreci uygulamalarda geri getirme görevi

(İng. retrieval tasks, kıs. GGG) olarak adlandırılan bazı işlevsel farklılıklar göstermektedir. Metin geri getirmeyi olarak adlandırılan geleneksel BE'nde temel olarak iki geri getirme görevi bulunmaktadır: bunlar anlık-sorgu geri getirmeyi (İng. adhoc retrieval) ve yönlendirme görevi (İng. routing task) olarak adlandırılmaktadırlar. Anlık-sorgu geri getirmesinde sabit belge topluluğu kullanılarak değişen sorgular ile erişim hedeflenirken, yönlendirme görevinde ise sabit sorgular ile değişen belge topluluklarına erişim gerçekleştirilmektedir. Her iki görevde de erişim işlemindeki ana odak sistemlerin başarımlarıdır.

Bu çalışmada temel olarak hedeflenen konu: Bilgi Erişimi (İng. Information Retrieval, kıs. BE) alanında anlık sorgu görevinde yapılan testler için bir veri ve süreç uygulaması geliştirmektir. BE çalışmalarının testlerinde sistemlerin hedeflerine uygun olan veri kümelerine - görevle uygun derlem, sorgu kümesi ve alakalı belge kümesi- ve değerlendirme ölçütlerine ihtiyaç duyulmaktadır[8]. Bu görevlerdeki farklılıklardan doğan yeni veri kümelerinin hazırlanması ve ilgili ölçütler ile değerlendirilmesi işlemlerinin yönetilmesi oldukça zor ve vakit alıcıdır. Özellikle sorguların alakalı olduğu belgeleri işaretleme işlemi dijital ortamda gerçekleştirecek bir uygulamaya ihtiyaç vardır. Çünkü çok sayıda belgenin hakemler -alaka yargısı yapan kullanıcılar- tarafından incelenmesi gerekmektedir. Bu amaçla, çalışmada geliştirilen web-tabanlı yazılım sürecin yönetimini kolaylaştırmakta ve hızlandırmaktadır.

## 3. TREC Anlık Sorgu Geri-Getirme Sınama Kümesi Hazırlanma Süreci

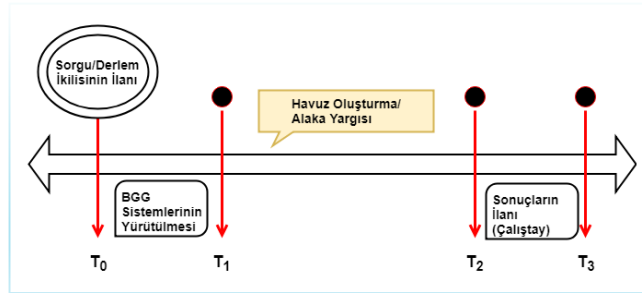
1992 yılından beri düzenlenen TREC (Text Retrieval Conference) çalışmaları A.B.D. Savunma Bakanlığı ve NIST (İng. National Institute of Standards and Technology) tarafından desteklenmektedir. Bu çalışmaların öncelikli hedefi BE yöntemlerinin metin geri getirme alanında büyük ölçekli değerlendirilmesine imkan sağlayacak bir sınav yapısı oluşturulmasıdır. Metin geri getirme sahasındaki araştırmalara katkı vermek üzere, büyük hacimli yazılı belge derlemleri ve BE başarımlarını değerlendirme yapısına ilişkin standartlar TREC kapsamında oluşturulmaya çalışılmaktadır [6]. Her yıl düzenlenen TREC kapsamındaki bilgi erişimi çalışmaları yeni

görevler ihtiyaçlarına ve kısıtlarına göre belirlenmektedir. Yönlendirme ve anlık-sorgu erişim görevlerinin alt-problemleri olan bu görevlerin her biri "iz" (İng. track) olarak isimlendirilir ve yıllar içinde şekillenerek geliştirilir. Yalnızca bu iki temel görevi içeren ilk iki TREC [9],[10]'ten sonraki yıllarda TREC-1 ve TREC-2'den oluşturulan alt-yapının kuramsal olarak farklı görevler için de uygun olacağı düşünülerek yeni görevler tanımlanmıştır. TREC-3'te [11] eklenen bu yeni iki erişim görevi etkileşimli görev (İng. interactive task) ve İspanyolca görevi (İng. Spanish task) olarak belirtilmiştir. 2007 yılına kadar toplam 27 farklı iz [12] TREC çalışmaları sırasında oluşturulmuştur. Güncel olarak ise, "TREC-Covid Challenge"[13] ve "TREC Deep Learning Track"[14] isimli görevler bulunmaktadır. Ancak; Hintçe[15], Çince[16] ve Rusça[17] gibi diller için TREC benzeri forumlar olmasına rağmen, hala Türkçe yazılı metinler için tanımlı bir görev bulunmamaktadır.

TREC izleri bir takım özellikleri önceden belirlenen bir işleyişe uygun olarak gerçekleştirilmektedir. İlk olarak, iz'e özel kullanılacak bir doğal dilde yazılmış belgeler topluluğu, sorgular ve erişim değerlendirme ölçütleri NIST tarafından belirlenmektedir. TREC'te yazılı belge toplulukları derlem, sorgular ile konular olarak adlandırılır ve belgeler ile oluşturulan konular SGML/XML standartlarına göre etiketlenmişlerdir. Ek olarak belgeler ve konular; yapısal (etiketli alanlar), niceliksel ve boyutsal olarak izlere göre

değişkenlik göstermektedir. Konu içinde birden fazla etiketli alan bulunmaktadır (anlık-sorgu izinde bunlar title, description, narrative olarak belirlenmiştir.) ve izlere göre bu alanlardan hangisinin kullanılacağı da değişebilmektedir. Başarım değerlendirme ölçütlerinin seçimi ise alaka yargısının işleyişi ile bağıntılı bir süreçtir. Anlık-sorgu görevine sahip izlerde katılımcılar genellikle sistem yürütüm (İng. run) sonuçlarından her sorguya karşılık gelen ilk 1000 (en fazla) belgeyi NIST'e göndermektedirler. Katılımcıların yürütüm sonuçlarındaki belgeler bir havuzda toplanarak hakemler tarafından belirlenen alaka yargılaması yöntemine göre alakalı/alakasız olarak işaretlenmektedir [18]. Daha sonra ise sistem başarımları iz'de belirlenen değerlendirme ölçütlerine göre hesaplanmakta ve katılımcılara ilan edilmektedir. Bu işlem BE test veri kümesi oluşturma ve değerlendirme sürecidir.

TREC organizasyonlarının süreci Şekil 1'de gösterilmiştir. Süreç katılımcılara sorgu ve derlem kümelerinin sağlanmasıyla başlamaktadır (T0 anı). Sonra, katılımcılar BE sistemlerini verilen sorguları kullanıp, belirlenen derlem üzerinde çalışmaktadırlar (T0-T1 arası). BE yürütümlerinden sonra, elde edilen çıktılar TREC tarafından toplanmakta ve bir havuz oluşturularak bütün sonuçlar için alaka yargısı yapılmaktadır (T1-T2 arası). Alaka yargısı yapılan çıktılar sonrasında, sonuçlar açıklanarak bir tartışma oturumu ile süreç sonlanmaktadır.



Şekil 1. TREC organizasyon süreci

Aslen, her bir TREC izi sonunda yeni bir test veri kümesi hazırlanmış olmaktadır. İlgili test materyalleri üç bileşenden oluşmaktadır.

- [1] Derlem, bir belge topluluğu.
- [2] Sorgu kümesi, bilgi ihtiyacını tanımlayan ifadeler.

- [3] Alaka yargıları kümesi, her sorgu belge çifti için alakalı veya alakasız yargısını gösteren değerlendirmeler.

Genellikle derlemler standart olmakla birlikte farklılık sorgularla sağlanmaktadır. Örnek olarak web izi için 2012'ye kadar ClueWeb09 derlemi

kullanılırken[19], sonraki izlerde daha büyük boyutlu yeni bir derlem olan, ClueWeb12 kullanılmıştır [20]. Yani, bugüne kadar gerçekleştirilen çok sayıdaki TREC izine karşılık kullanılan derlem çeşitliliği oldukça kısıtlıdır. Sorgu kümelerinin oluşturulması ise yönetsel karmaşıklık ve yoğun iş hacminden çok, gerçek sorgu uzayına yakın veya benzer olmasına yönelik uzmanlık isteyen bir süreçtir. Aksine alaka yargıları kümesinin hazırlanması doğru yönetilmesi gereken yoğun bir iş yükü oluşturmaktadır. Binlerce sorgu-belge çiftinin alakalı/alakasız olarak değerlendirilmesi ancak çok sayıda değerlendime uzmanı/hakem tarafından yapılabilir. Bu değerlendirmelerin gerçekleşmesi ve sonuçlarının toplanması için doğal olarak dijital ortamda bir yönetim otomasyonuna ihtiyaç vardır. Şekil 1’de gösterilen T1 ve T2 zaman aralıklarında yapılan havuz oluşturma ve alaka yargısı yapılması işlemleri çalışmada geliştirilen araç ile yapılabilir.

#### 4. Uygulama Tasarımı

##### 4.1. Geliştirme çatısı ve teknolojiler

Geliştirme çatısı olarak NODE.JS kullanılmıştır. NODE.JS açık kaynak kodlu olup, JavaScript tabanlıdır. Ayrıca paket yönetim sistemiyle, bağlantılı modülleri kolayca ekleyip çıkarabilmeyi sağlamaktadır. NODE.JS’in bir diğer avantajlarından biri de modülerliktir. Kolayca taşınabilir ve çalışır hale getirebilir bir geliştirme çatısıdır.

Oturum açma, oturum yönetimi gibi durumlar için yine bir NODE.JS modülü olan EXPRESS.JS kullanılmıştır. EXPRESS.JS oturum yönetimi için bir ara katman yazılımdır. EXPRESS.JS sunucuya gelen istekleri doğrulama, hata yazdırma gibi özelliklere sahiptir. Gösterim motoru olarak HANDLEBAR.JS kullanılmıştır. HANDLEBAR.JS JavaScript dilleri için geliştirilmiş bir şablon motorudur. Sunucu tarafı ve istemci tarafı kullanılabilir.

Veri tabanı olarak NoSQL teknolojisine uygun olan MongoDB seçilmiştir. Başlıca nedenlerinden biri açık kaynak kodlu olup ücretsiz olmasıdır. NoSQL teknolojisini geleneksel SQL teknolojisinden ayıran en temel unsur unrelational yani bağlantısız veri tabanları oluşturulabilmesi, böylece tablo sayısını daha az tutmaya olanak sağlamasıdır.

Projede kullanılan bir diğer teknoloji ise Sunucu Gönderimli Olay’dır (İng. Server Sent Event, kıs. SSE). SSE, sunucudan istemci tarafına, web socket mantığında mesaj göndermeyi sağlayan bir teknolojidir. Bunun için ek bir modül kullanılmamış, doğal JavaScript yöntemleriyle sağlanmıştır. Ayrıca proje versiyon kontrolü için GIT teknolojisinden yararlanılmıştır ve kaynak kodlar dışardan erişime açık şekilde github.com sunucularında saklanmıştır.

##### 4.2. Yazılım tasarımı

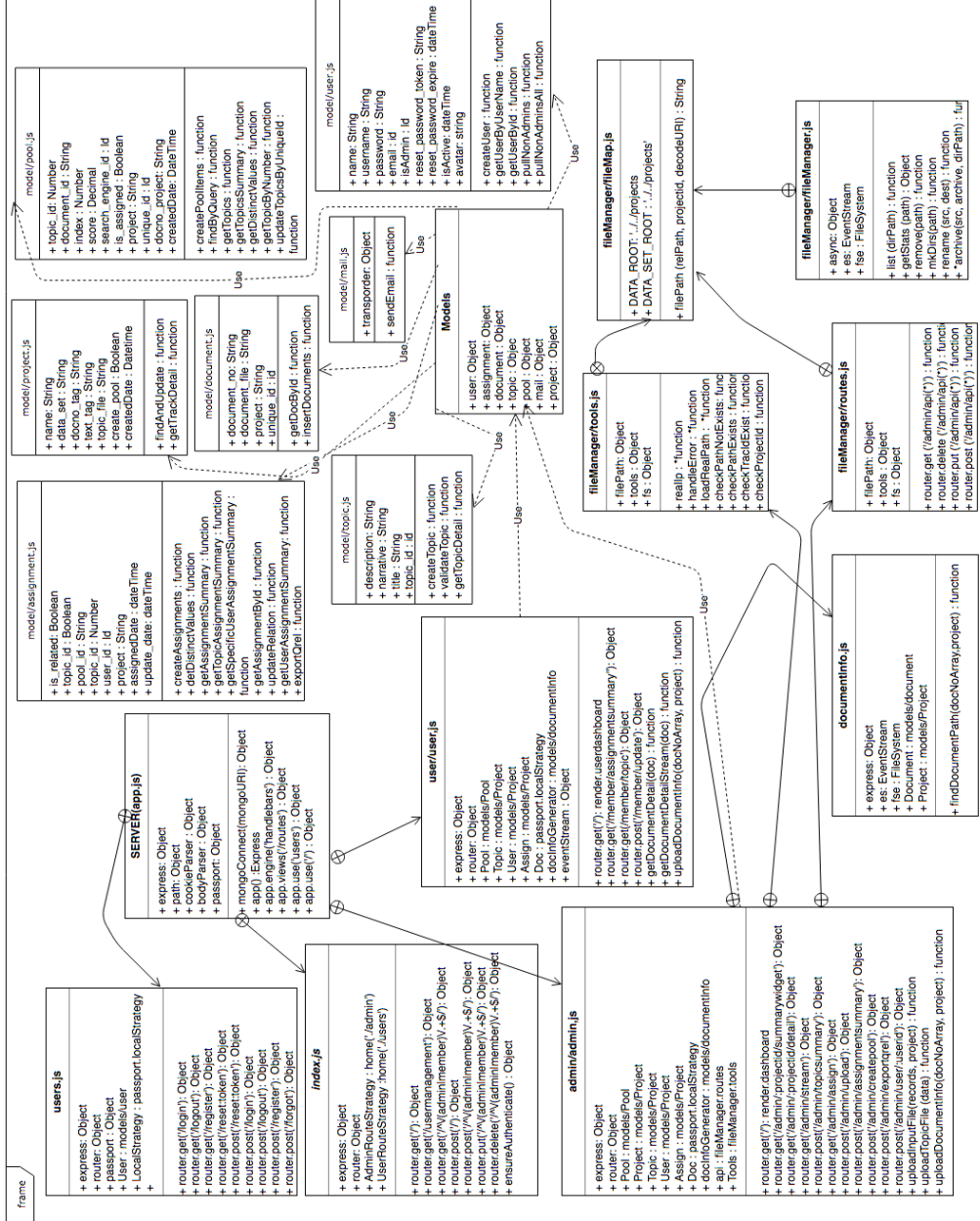
Yazılım çatısı olarak NODE.JS teknolojisini kullandığı sistemde, ana programlama dili JavaScript’tir. NODE.JS çatısının tercih edilmesindeki ana sebeplerden birkaçı ise, platform bağımsız çalışabilen, etkin veri akışı sağlayan, açık kaynak kodlu bir teknoloji olmasıdır. Açık kaynak kodlu oluşu, dünya genelinde milyonlarca geliştirici tarafından desteklenmesi ve herhangi bir sorun karşısında kolay bir şekilde çözüm bulunabilmesi gibi avantajları beraberinde getirmektedir. NODE.JS, diğer geleneksel internet uygulamalarının aksine olay-tabanlı (İng. event-driven) çalışabilmektedir. Geleneksel internet uygulamaları kullanıcının bir istek yapmasıyla tetiklenirken, NODE.JS de bu olay-tabanlı olup, sunucu tarafı isteklerle de tetiklenebilmektedir. Bu da daha etkileşimli (İng. interactive) internet uygulamaları tasarlamaya olanak sağlamaktadır.

NODE.JS yazılım çatısındaki temel özelliklerden biri değişik eylemler için farklı yönlendirmeler yapıp, böylece diğer betik (İng. script) tabanlı mimarilere göre daha modüler bir yapı sağlamasıdır. Tasarım mimarisinde ilk olarak ele alınan problem, sisteme giriş, çıkış ve kullanıcı tanımlamalarıdır. Uygulamaya giriş ve çıkış işlemleri, kullanıcı kayıtları ana dizin altındaki USER.JS tarafından yönetilmektedir. Şekil 2’de görüldüğü gibi kullanıcı doğrulamaları, şifre sıfırlamaları, sistemden giriş, çıkış gibi eylemler bu betik grubunda yönetilmiştir.

Mimarideki bir diğer ana yönlendirme ise, sistemdeki rol durumuna göre doğrulanmış kullanıcıların farklı dizinlere yönlendirilmesidir. Uygulamada iki temel rol vardır; yönetici ve normal kullanıcı. Sisteme başarıyla giriş yapan kullanıcılar, veri tabanındaki rol bilgisine göre ana dizin altındaki ADMIN ya da USER dizinine yönlendirilir. Bu aynı zamanda geliştirici tarafında da kodlama esnasından daha temiz bir okuma sağlamaktadır. ADMIN rolündeki

kullanıcıların; veri yürütümü yaratma, sorgu kümesi yükleme, daha önceden yaratılan yürütümler için atama gibi işlemleri bu dizin grubundaki ADMIN.JS betik dosyası tarafından yönetilmektedir. Aynı zamanda normal kullanıcı

rolündeki kullanıcıların; atanmış yürütüm detayını görüntüleme, atamalar için alakalı yargısını işaretleme gibi işlemleri de yine ADMIN.JS betik dosyası tarafından yönetilmektedir.



Şekil 2. Araç UML Şeması

Mimaride iki ana alt grup daha vardır. Bunlar FILEMANAGER ve MODEL dizinleridir. Bunlardan FILEMANAGER, dosya ve dizin

işlemlerinin ayrıştırılması, MODEL grubu ise, kayıtların veri tabanına kayıtlı olmadan önce bir ara yüz yaratılması amacıyla tasarlanmıştır.

FILEMANAGER dizini altındaki betik dosyaları, ADMIN rolündeki kullanıcıların, veri kümelerini görüntüleme, veri yürütümü için dosya açma silme gibi yerel dosyalama işlemlerini gerçekleştirmesi gibi işlemleri sağlamaktadır.

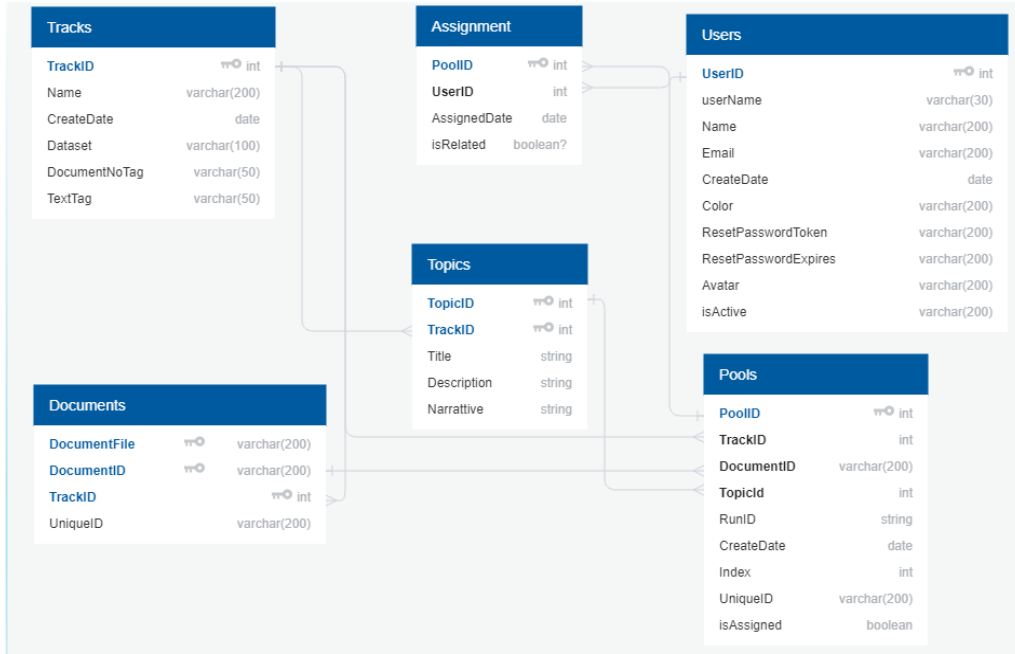
MODEL dizini ise, yukarıda belirttiği üzere veri tabanı kayıtlarına bir ara yüz sunmaktadır (DATA-LAYER). Bu sınıflar veri tabanı tablolarının bir yansıması gibi düşünülebilir. MODEL dizini altında; kullanıcı (USER), proje (PROJECT), havuz (POOL), döküman (DOCUMENTS), konu (TOPIC), atama (ASSIGNMENT) ve posta (MAIL) ara sınıflarını görmek mümkündür. Posta sınıfı hariç diğer bütün sınıflar veri tabanı tablolarının birer kopyasıdır. Bu sınıflarda verilen bilginin doğrulanması ve kayıt alanlarının hangi tipte tanımlamalar kabul ettiği gibi bilgiler tutularak, bilgi bütünlüğü ve doğruluğunu sağlamak amaçlanmıştır.

### 4.3. Veri tabanı tasarımı

Sistemde temel anlamda kullanıcılar dışında beş öge bulunmaktadır: izler, belgeler, konular, havuzlar ve atamalar; Şekil 3'te gösterilen veritabanı şemasında bunlar sırasıyla "tracks", "documents", "topics", "pools" ve "assignment"

tabloları ile belirtilmiştir. Bütün hiyerarşinin başında izler bulunmaktadır. Temel olarak bir izi oluşturan veriler onun numarası, adı, oluşturma tarihi, belge topluluğu, belge numaraları ile metnin yerini gösteren XML etiketleri ve sorgu kümesidir. Bunlardan "belge topluluğu" ve "sorgular" ayrı tablolar olarak tasarlanmıştır; TREC standartlarında sorgu "topic" olarak adlandırıldığından bu ifade kullanılmıştır. Belge topluluğunun kök dizin yolu ise izler'e eklenmiştir.

Havuzlarda ise her izde gerçekleştirilen yürütümlerdeki belge ve sorgu ikilileri tutulmaktadır. Bu ikililer birden fazla yürütüm sonucundan aynı gelebilecekleri için RunID ile yürütümler gösterilmektedir. Ayrıca isAssigned ile ilgili dokümanın bir değerlendiriciye atanıp atanmadığı belirtilmektedir. Atama'larda havuzdaki belgelerin kullanıcılara atanmasına ilişkin bilgiler mevcuttur. Kullanıcılar, aslen değerlendirici/hakem olan varlıklardır. Kendilerine atanan belgelerin değerlendirilmesi ise isRelated özelliğiyle sağlanmaktadır. Bu özellik null değeri alabilmektedir: Böylece değerlendirme işleminin gerçekleşip gerçekleşmediği anlaşılabilir.

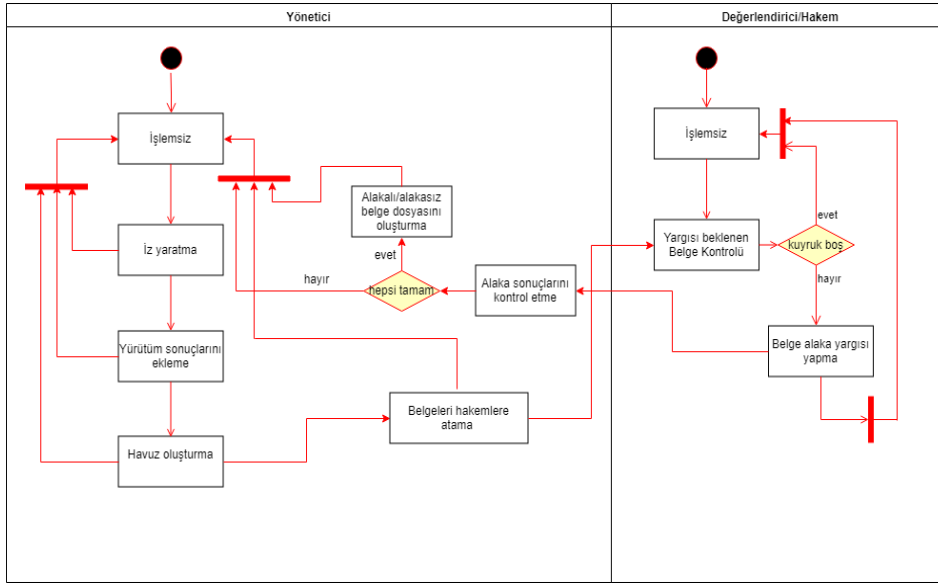


Şekil 3. Veritabanı şeması

### 5. Sistem Kullanım Akışı

Sistemde kayıtlı iki tip kullanıcıdan yönetici ve değerlendirici/hakem işlemleri açısından geliştirilen sistemin akış şeması Şekil 4'te verilmiştir. İlk olarak, yönetici haklarına sahip kullanıcı tarafından belge topluluğu ve sorgu kümesi seçilerek yeni bir iz oluşturulmaktadır. İlgili ize ait yürütüm sonuçları ancak bu

aşamadan sonra yüklenebilmektedir. Bu iki aşamanın başarıyla tamamlanmasından sonra ise değerlendirilecek belgelerin bulunduğu havuz oluşturulabilmektedir. İz yöneticisi, sistemde kayıtlı hakem haklarına sahip kullanıcılara bu havuzdan istediği belgeleri atayabilmekte veya atanmış belgeleri tekrar havuza alabilmektedir.



Şekil 4. Sistem akış şeması

Hakem ise kendine atanmış olan belgelerin olup olmadığını kontrol edebilmekte ve eğer alaka yargısını yapmadığı bir belge mevcut ise değerlendirme sürecini başlatabilmektedir. Belge değerlendirme sonucunu sisteme girerse, sistem tarafından tüm havuzun değerlendirilip değerlendirilmediği kontrol edilmektedir. Eğer havuzdaki belgelerin hepsi alakalı/alakasız olarak işaretlenmiş ise TREC standartlarında belirtilen sonuç dosyası otomatik olarak oluşturulmaktadır.

### 6. Tartışma ve Sonuç

Bu çalışmada geliştirilen araç ile anlık sorgu geri getiriminde kullanılacak test veri kümesinin oluşturulması hızlandırılarak süreç yönetilebilir hale getirilmiştir. Türkçe metinlerin geri getirmesi üzerine yapılan çalışmaların test edilebileceği standartlara uygun tek BE veri kümesi BilCol2005 deney koleksiyonudur[21]. Ancak BE çalışmalarının gelişmesi ve ilerlemesi için

yenilerine ihtiyaç vardır. Büyük ölçekli BE sistemlerini değerlendirmeyi hedefleyen TREC ve benzeri kuruluşlar bu çeşitliliği sağlayarak doğru sonuçlara ulaşmayı amaçlamaktadır.

Özetle, Türkçe yazılı derlemler için TREC benzeri bir çalıştayın sürdürülebilir halde gerçekleştirilmesi BE çalışmalarının artması için çok önemlidir. TREC'in temel aldığı havuzlama (ing. Pooling) mekanizması, farklı katılımcılardan/sistemlerden gelen yürütüm sonuçlarına ihtiyaç duymaktadır. Doğal olarak böyle bir süreçte sistem performanslarının katılımcılara geri bildirilmesi gerekmektedir. Gelecek çalışmalarda alaka yargısı yapmayı sağlayan bu araca ilgili izdeki sistem başarımlarını değerlendirmeye yönelik bir eklenti yapılması planlanmaktadır. Bu sayede, Türkçe BE için düzenlemeyi düşündüğümüz çalıştay teknik anlamda tam otomatik hale gelecektir; yani çalıştay ile ilgili tüm süreçler dijital ortamda gerçekleştirilecektir.

**Teşekkür**

Bu proje Ege Üniversitesi Bilimsel Araştırma Projeleri Koordinasyon Birimi tarafından 18-UBE-001 proje numarası ile desteklenmiştir.

**Kaynakça**

- [1] Chapelle, O., Metzler, D., Zhang, Y., Grinspan, P. 2009. Expected reciprocal rank for graded relevance. In Proceedings of the 18th ACM Conference on Information and Knowledge Management, 621-630.
- [2] Järvelin, K., Kekäläinen, J. 2002. Cumulated gain-based evaluation of IR techniques. ACM Transactions on Information Systems (TOIS), 20(4):422-446.
- [3] Clarke, C.L.A., Kolla, M., Cormack, G.V., Vechtomova, O., Ashkann, A., Butcher, S., MacKinnon, I. 2008. Novelty and diversity in information retrieval evaluation. In Proceedings of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 659-666.
- [4] Agrawal, R., Gollapudi, S., Halverson, A., Ieong, S. 2009. Diversifying search results. In Proceedings of the Second ACM International Conference on Web Search and Data Mining, 5-14.
- [5] Cleverdon, C. 1967. The Cranfield tests on index language devices. Aslib proceedings, 19(6), 173-194.
- [6] TREC web sayfası. <http://trec.nist.gov/> (Erişim Tarihi: 14/12/2018).
- [7] Lew, M. S., Sebe, N., Djeraba, C., Jain, R. 2006. Content-based multimedia information retrieval: State of the art and challenges. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 2(1), 1-19. <https://dl.acm.org/doi/abs/10.1145/1126004.1126005>
- [8] Sanderson, M., 2010. Test Collection Based Evaluation of Information Retrieval Systems. Foundations and Trends® in Information Retrieval, 4(4), 247-375. DOI: 10.1561/1500000009
- [9] Harman, D. 1992. Overview of the First Text REtrieval Conference (TREC-1). NIST Special Publication 500-207: The First Text Retrieval Conference Proceedings (TREC-1), 1-20.
- [10] Harman, D. 1993. Overview of the Second Text REtrieval Conference (TREC-2). NIST Special Publication 500-215: The Second Text Retrieval Conference Proceedings (TREC-2), 1-20.
- [11] Harman, D. 1994. Overview of the Third Text REtrieval Conference (TREC-3). NIST Special Publication 500-225: The Third Text Retrieval Conference Proceedings (TREC-3), 1-20.
- [12] Voorhees, E.M. 2007. Overview of TREC-2007. NIST Special Publication 500-274: The 16th Text Retrieval Conference Proceedings (TREC 2007), 1-17.
- [13] TREC-COVID Challenge web sayfası. <https://ir.nist.gov/covidSubmit/> (Erişim Tarihi: 02/07/2020).
- [14] TREC-2019 Deep Learning Track web sayfası. <https://trec.nist.gov/data/deep2019.html> (Erişim Tarihi: 02/07/2020).
- [15] Hintçe için bilgi erişim görevleri içeren forum. web sayfası. <http://fire.irs.res.in/fire/2019/home> (Erişim Tarihi: 02/07/2020).
- [16] Çince için bilgi erişim görevleri içeren web sayfası. <http://www.thuir.cn/ntcirwww2/> (Erişim Tarihi: 02/07/2020).
- [17] Rusya bilgi erişim değerlendirme semineri web sayfası. <http://romip.ru/en/> (Erişim Tarihi: 02/07/2020).
- [18] Voorhees, E.M. 2007. TREC: Continuing information retrieval's tradition of experimentation. Communications of the ACM. 50(11), 51-54. DOI:10.1145/1297797.1297822
- [19] Clarke, C.L.A., Craswell, N., Soboroff, I. 2012. Overview of the TREC 2012 Web track. 21<sup>th</sup> Text REtrieval Conference, Gaithersburg, Maryland.
- [20] Collins-Thompson, K., Macdonald, C., Bennett, P., Diaz, F., Voorhees, E.M. 2014. TREC 2014 web track overview. 23<sup>th</sup> Text REtrieval Conference, Maryland.
- [21] Can, F., Kocberber, S., Balcık, E., Kaynak, C., Ocalan, H.C., Vursavas, O.M. 2008. Information retrieval on Turkish texts. Journal of the American Society for Information Science and Technology 59(3), 407 - 421. DOI: 10.1002/asi.20750