



**SIRALAMAYI ÖĞRENME
ALGORİTMALARINDA KULLANILMAK
ÜZERE SEO-TABANLI ÖZNİTELİKLERİN
TÜRETİLMESİ VE DEĞERLENDİRİLMESİ**

Doktora Tezi

Ahmet AYDIN

Eskişehir 2022

**SIRALAMAYI ÖĞRENME ALGORİTMALARINDA KULLANILMAK ÜZERE
SEO-TABANLI ÖZNİTELİKLERİN TÜRETİLMESİ VE
DEĞERLENDİRİLMESİ**

Ahmet AYDIN

Doktora Tezi

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Bilimleri Bilim Dalı

Danışman: Doç. Dr. Ahmet ARSLAN

Eskişehir

Eskişehir Teknik Üniversitesi

Lisansüstü Eğitim Enstitüsü

Kasım 2022

Bu tez çalışması BAP Komisyonu tarafından kabul edilen 20DRP018 no.lu proje kapsamında desteklenmiştir.

JÜRİ VE ENSTİTÜ ONAYI

Ahmet AYDIN'ın SIRALAMAYI ÖĞRENME ALGORİTMALARINDA KULLANILMAK ÜZERE SEO-TABANLI ÖZNİTELİKLERİN TÜRETİLMESİ VE DEĞERLENDİRİLMESİ başlıklı çalışması 25/11/2022 tarihinde aşağıdaki jüri tarafından değerlendirilerek “Eskişehir Teknik Üniversitesi Lisansüstü Eğitim-Öğretim ve Sınav Yönetmeliği”nin ilgili maddeleri uyarınca, Bilgisayar Mühendisliği Anabilim dalında Doktora Tezi olarak kabul edilmiştir.

Jüri Üyeleri

Unvan Adı Soyadı

İmza

Üye	: Prof. Dr. Yaşar HOŞCAN
Üye	: Prof. Dr. Bekir Taner DİNÇER
Üye	: Dr. Öğr. Üyesi Zühal KARTAL
Üye	: Dr. Öğr. Üyesi Özgür KILIÇ
Üye	: Dr. Öğr. Üyesi Ali YÜREKLİ

Prof. Dr. Murat TANIŞLI

Lisansüstü Eğitim Enstitüsü Müdürü

25/11/2022

DANIřMAN ONAYI

Daniřmanlıęını yrttęm Doktora ęrencisi Ahmet AYDIN, SIRALAMAYI ĖRENME ALGORİTMALARINDA KULLANILMAK ZERE SEO-TABANLI ZNİTELİKLERİN TRETİLMESİ VE DEęERLENDİRİLMESİ bařlıklı tez alıřmasını tamamlamıřtır. Hazırlamıř olduęu tez tarafımca incelenmiř ve ęrencinin tez savunma sınavına alınması bilimsel ve etik aıdan uygun grlmřtr.

Tez Daniřmanı

Do. Dr. Ahmet ARSLAN

ÖZET

SIRALAMAYI ÖĞRENME ALGORİTMALARINDA KULLANILMAK ÜZERE SEO-TABANLI ÖZNİTELİKLERİN TÜRETİLMESİ VE DEĞERLENDİRİLMESİ

Ahmet AYDIN

Bilgisayar Mühendisliği Anabilim Dalı

Bilgisayar Bilimleri Bilim Dalı

Eskişehir Teknik Üniversitesi, Lisansüstü Eğitim Enstitüsü, Kasım 2022

Danışman: Doç. Dr. Ahmet ARSLAN

Web belgesi koleksiyonlarını hedefleyen bilgi erişimi üzerine yapılan geçmiş çalışmalar, web belgelerinin kalitesini ölçen ve sadece belgelere dayalı (sorgu-bağımsız) ölçütlerin veya ön belge özelliklerinin (PageRank vb.) bir bilgi erişim sistemine dahil edilmesinin, sistem etkinliğini artırdığını göstermektedir. Bu çalışmada Arama Motoru Optimizasyonu tekniklerinden ilham alınarak yeni ön belge özellikleri tanıtılmaktadır. Tanıtılan bu özellikler sıralamayı öğrenme algoritmalarında öznitelik olarak kullanılarak etkinlikleri test edilmektedir. Deneyler iki standart web belge koleksiyonunda gerçekleştirilmektedir: 500 milyon belge içeren ClueWeb09 ve 733 milyon belge içeren ClueWeb12 koleksiyonları. Deneylerde sorgu kümesi olarak bu koleksiyonları hedef alan ve toplam 1,204 sorgu içeren TREC ve NTCIR sorgu kümeleri kullanılmaktadır. Önceki çalışmalarda tanıtılan standart öznitelikler kullanılarak güçlü bir referans öznitelik kümesi oluşturulmuş ve bu çalışmada tanıtılan özniteliklerin etkisi referans öznitelikler ile deneysel olarak karşılaştırılarak incelenmiştir. Deney sonuçları, bu çalışmada tanıtılan özniteliklerin bilgi erişim sistemi başarımını kullanılan test koleksiyonlarında istatistiksel olarak anlamlı derecede artırdığını göstermiştir (örneğin, ClueWeb09 veri setinde %18'lik ortalama nDCG@10 puan artışı). Ayrıca tanıtılan öznitelikler fonksiyonel özelliklerine göre 5 farklı gruba ayrılmış ve grupların sistem başarımına katkısı detaylı olarak analiz edilmiştir.

Anahtar Sözcükler: Bilgi erişimi, Web arama, Sıralamayı öğrenme, Makine öğrenmesi, Arama motorları, Ön belge özellikleri

ABSTRACT

CONSTRUCTION AND EVALUATION OF SEO-BASED FEATURES FOR TO USE IN LEARNING TO RANK ALGORITHMS

Ahmet AYDIN

Department of Computer Engineering

Programme in Computer Sciences

Eskişehir Technical University, Institute of Graduate Programs, November 2022

Supervisor: Assoc. Prof. Dr. Ahmet ARSLAN

The past work on information retrieval targeting web document collections shows that incorporating a measure that is solely based on documents (query-independent) and measures the quality of web documents, or rather the document prior (e.g., PageRank), into an information retrieval system improves the retrieval effectiveness. In this study, we introduce new document priors, inspired by Search Engine Optimization techniques. We also empirically investigate their effect by employing them as features in a learning to rank deployment. The experiments are performed on the two standard Web Information Retrieval test collections: the ClueWeb09 and the ClueWeb12 datasets, which include 500 and 733 million web documents, respectively. TREC and NTCIR query sets, which target those collections and contain a total of 1,204 queries, are used as query sets in the experiments. A strong baseline is formed by using standard features introduced in the previous works, with respect to which the effect of newly introduced features in this study is empirically compared. The experimental results reveal that the features introduced in this work led to statistically significant improvements in retrieval performance on the test collections in use (e.g., for the ClueWeb09 dataset, 18% improvement on average nDCG@10 score). The introduced features are classified into 5 groups with respect to functional properties and the contribution of each group to retrieval performance is also analyzed in detail.

Keywords: Information retrieval, Web search, Learning to rank, Machine learning, Search engines, Document priors

TEŞEKKÜR

Tez çalışmam boyunca tüm aşamalarda özverili bir biçimde desteğini ve bilgisini paylaşan danışmanım Sayın Doç. Dr. Ahmet ARSLAN hocama teşekkürü bir borç bilirim.

Yine tez çalışmam boyunca tüm aşamalarda desteğini ve bilgisini paylaşan tez izleme jüri üyesi Prof. Dr. Bekir Taner DİNÇER hocama teşekkürlerimi sunarım. Diğer tez izleme jüri üyeleri Dr. Öğr. Üyesi İsmail SAN ve Dr. Öğr. Üyesi Zühal KARTAL hocalarıma katkılarından dolayı teşekkür ederim.

Bu süreçte desteklerini esirgemeyen başta eşim Müjde SUSURLUK AYDIN olmak üzere, annem Nazmiye AYDIN, teyzem Yıldız SARGIN, ablam Ayşe AYDIN ÖZBAY ve tüm aileme teşekkürlerimi sunarım.

Ahmet AYDIN

ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Bu tezin bana ait, özgün bir çalışma olduğunu; çalışmamın hazırlık, veri toplama, analiz ve bilgilerin sunumu olmak üzere tüm aşamalarında bilimsel etik ve kurallara uygun davrandığımı; bu çalışma kapsamında elde edilen tüm veri ve bilgiler için kaynak gösterdiğimi ve bu kaynaklara kaynakçada yer verdiğimi; bu çalışmanın Eskişehir Teknik Üniversitesi tarafından kullanılan “bilimsel intihal tespit programı”yla tarandığını ve hiçbir şekilde “intihal içermediğini” beyan ederim. Herhangi bir zamanda, çalışmamla ilgili yaptığım bu beyana aykırı bir durumun saptanması durumunda, ortaya çıkacak tüm ahlaki ve hukuki sonuçları kabul ettiğimi bildiririm.

Ahmet AYDIN

İÇİNDEKİLER

Sayfa

BAŞLIK SAYFASI	I
JÜRİ VE ENSTİTÜ ONAYI.....	II
DANIŞMAN ONAYI	III
ÖZET	IV
ABSTRACT.....	V
TEŞEKKÜR	VI
ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ.....	VII
İÇİNDEKİLER	VIII
TABLolar DİZİNİ	XI
ŞEKİLLER DİZİNİ	XII
SİMGELER VE KISALTMALAR DİZİNİ	XV
1. GİRİŞ	1
1.1. Motivasyon.....	2
1.2. Amaçlar ve Katkılar	3
1.3. Tez Organizasyonu	4
2. BİLGİ ERİŞİMİ.....	5
2.1. Bilgi Erişimi Tarihi	5
2.2. Bilgi Erişim Sistemi Süreçleri	6
2.2.1. Belge içeriği temsili (Dizinleme).....	7
2.2.1.1. Tokenizasyon.....	8
2.2.1.2. Durak kelimelerin elenmesi	9
2.2.1.3. Normalizasyon	9
2.2.1.4. Gövdeleme	9
2.2.2. Bilgi ihtiyacı temsili	10
2.2.3. Sorgu-belge eşleşmesi	10
2.2.3.1. Boolean model	11
2.2.3.2. Vektör model	11

2.2.3.3. Olasılıksal modeller	13
2.2.4. Yeniden sıralama	16
2.3. Bilgi Erişim Sistemi Değerlendirilmesi	17
2.3.1. Bilgi erişim sistemi değerlendirme ölçütleri	19
2.3.1.1. Mean Average Precision (MAP)	20
2.3.1.2. Normalized Discounted Cumulative Gain (nDCG).....	21
2.3.1.3. Expected Reciprocal Rank (ERR).....	21
3. İLGİLİ ÇALIŞMALAR.....	23
3.1. Ön Belge Özellikleri	23
3.1.1. Bağlantı analizi tabanlı ön belge özellikleri	24
3.1.2. İçerik tabanlı ön belge özellikleri.....	25
3.1.3. Diğer ön belge özellikleri	26
3.2. LTR Deneylerinde Öznitelik Etkinliği	27
4. SIRALAMAYI ÖĞRENME	29
4.1. Noktasal Yaklaşım (Pointwise Approach)	30
4.2. İkili Yaklaşım (Pairwise Approach)	31
4.3. Listesel Yaklaşım (Listwise Approach).....	32
5. SEO-TABANLI ÖZNİTELİKLER.....	33
5.1. Genel Web Sayfası Yapısı.....	33
5.2. Arama Motorları ve SEO	34
5.3. SEO Teknikleri ile Türetilen Öznitelikler	36
5.3.1. Meta etiket öznitelikleri	37
5.3.2. İçerik öznitelikleri	39
5.3.3. Bağlantı ve resim öznitelikleri.....	40
5.3.4. URL öznitelikleri	41
5.3.5. Benzerlik öznitelikleri	42
6. DENEYSEL YÖNTEMLER.....	43
6.1. Kullanılan Belge Koleksiyonları	43
6.1.1. ClueWeb09	43
6.1.2. ClueWeb12	44
6.2. Kullanılan Sorgu Setleri	44

6.2.1. Web Track.....	44
6.2.2. Million Query Track	45
6.2.3. Tasks Track.....	45
6.2.4. We Want Web.....	46
6.3. Referans Öznitelikler	46
6.3.1. Sorgu öznitelikleri	47
6.3.2. Sorgu-belge öznitelikleri	49
6.3.3. Belge öznitelikleri	50
6.4. Deney Tasarımı	52
7. DENEY SONUÇLARI	56
7.1. Ortalama Sistem Başarımı Analizi	56
7.2. Sorgu Başına (Per-Query) Sistem Başarımı Analizi.....	61
7.3. SEO-tabanlı Özniteliklerin Ortalama Sistem Başarımına Katkısı	70
8. SONUÇLAR VE GELECEK ÇALIŞMALAR	79
8.1. Çalışma Kısıtları	80
8.2. Gelecek Çalışmalar	81
KAYNAKÇA.....	83
ÖZGEÇMİŞ	

TABLÖLAR DİZİNİ

Sayfa

Tablo 2.1. Durumsallık tablosu	20
Tablo 5.1. SEO teknikleriyle türetilen öznitelikler	36
Tablo 5.2. Web sayfası alan içeriklerinden SEO teknikleriyle türetilen benzerlik öznitelikleri	42
Tablo 6.1. Çalışmada kullanılan referans öznitelikler	47
Tablo 7.1. Her bir sorgu seti için NoLtr, Ltr ve Ltr* çalışmalarının ortalama nDCG@10 puanları	57
Tablo 7.2. Her bir sorgu seti için NoLtr, Ltr ve Ltr* çalışmalarının ortalama nDCG@20 puanları	57
Tablo 7.3. Her bir sorgu seti için NoLtr, Ltr ve Ltr* çalışmalarının ortalama ERR@10 puanları	58
Tablo 7.4. Her bir sorgu seti için NoLtr, Ltr ve Ltr* çalışmalarının ortalama ERR@20 puanları	58
Tablo 7.5. Her bir sorgu seti için NoLtr, Ltr ve Ltr* çalışmalarının ortalama MAP puanları	59

ŞEKİLLER DİZİNİ

Sayfa

Şekil 1.1. Web sayfası sayısının yıllara göre değişimi	1
Şekil 2.1. Bilgi erişimi sisteminin süreçleri.....	7
Şekil 2.2. Ters dizin yapısı	8
Şekil 2.3. Terim sözlüğü oluşturma aşamaları	10
Şekil 2.4. Örnek bir bilgi gereksinimi	18
Şekil 2.5. Örnek alakalılık yargıları.....	18
Şekil 2.6. Örnek bir çalışma kesiti.....	19
Şekil 4.1. Sıralamayı öğrenme çatısı	30
Şekil 5.1. Basit bir HTML belgesi yapısı	33
Şekil 5.2. Google arama motoru SERP listesi ilk sayfadaki web sayfalarının tıklanma oranları	35
Şekil 6.1. Çalışmanın deney tasarımı	53
Şekil 7.1. LambdaMART ile üretilen çalışmaların her bir sorgu için nDCG@10 puan farkları	62
Şekil 7.2. LambdaMART ile üretilen çalışmaların her bir sorgu için nDCG@20 puan farkları	63
Şekil 7.3. LambdaMART ile üretilen çalışmaların her bir sorgu için ERR@10 puan farkları	64
Şekil 7.4. LambdaMART ile üretilen çalışmaların her bir sorgu için ERR@20 puan farkları	64
Şekil 7.5. LambdaMART ile üretilen çalışmaların her bir sorgu için MAP puan farkları.....	65
Şekil 7.6. RankBoost ile üretilen çalışmaların her bir sorgu için nDCG@10 puan farkları.....	65
Şekil 7.7. RankBoost ile üretilen çalışmaların her bir sorgu için nDCG@20 puan farkları.....	66
Şekil 7.8. RankBoost ile üretilen çalışmaların her bir sorgu için ERR@10 puan farkları.....	66
Şekil 7.9. RankBoost ile üretilen çalışmaların her bir sorgu için ERR@20 puan farkları.....	67

Şekil 7.10. RankBoost ile üretilen çalışmaların her bir sorgu için MAP puan farkları.....	67
Şekil 7.11. SVM ^{rank} ile üretilen çalışmaların her bir sorgu için nDCG@10 puan farkları.....	68
Şekil 7.12. SVM ^{rank} ile üretilen çalışmaların her bir sorgu için nDCG@20 puan farkları.....	68
Şekil 7.13. SVM ^{rank} ile üretilen çalışmaların her bir sorgu için ERR@10 puan farkları.....	69
Şekil 7.14. SVM ^{rank} ile üretilen çalışmaların her bir sorgu için ERR@20 puan farkları.....	69
Şekil 7.15. SVM ^{rank} ile üretilen çalışmaların her bir sorgu için MAP puan farkları	70
Şekil 7.16. LambdaMART ile üretilen çalışmaların nDCG@10 sonuçları için tek yönlü varyans analizleri	71
Şekil 7.17. LambdaMART ile üretilen çalışmaların nDCG@20 sonuçları için tek yönlü varyans analizleri	72
Şekil 7.18. LambdaMART ile üretilen çalışmaların ERR@10 sonuçları için tek yönlü varyans analizleri	72
Şekil 7.19. LambdaMART ile üretilen çalışmaların ERR@20 sonuçları için tek yönlü varyans analizleri	73
Şekil 7.20. LambdaMART ile üretilen çalışmaların MAP sonuçları için tek yönlü varyans analizleri	73
Şekil 7.21. RankBoost ile üretilen çalışmaların nDCG@10 sonuçları için tek yönlü varyans analizleri	74
Şekil 7.22. RankBoost ile üretilen çalışmaların nDCG@20 sonuçları için tek yönlü varyans analizleri	74
Şekil 7.23. RankBoost ile üretilen çalışmaların ERR@10 sonuçları için tek yönlü varyans analizleri	75
Şekil 7.24. RankBoost ile üretilen çalışmaların ERR@20 sonuçları için tek yönlü varyans analizleri	75
Şekil 7.25. RankBoost ile üretilen çalışmaların MAP sonuçları için tek yönlü varyans analizleri	76
Şekil 7.26. SVM ^{rank} ile üretilen çalışmaların nDCG@10 sonuçları için tek yönlü varyans analizleri	76

Şekil 7.27. SVM ^{rank} ile üretilen çalışmaların nDCG@20 sonuçları için tek yönlü varyans analizleri	77
Şekil 7.28. SVM ^{rank} ile üretilen çalışmaların ERR@10 sonuçları için tek yönlü varyans analizleri	77
Şekil 7.29. SVM ^{rank} ile üretilen çalışmaların ERR@20 sonuçları için tek yönlü varyans analizleri	78
Şekil 7.30. SVM ^{rank} ile üretilen çalışmaların MAP sonuçları için tek yönlü varyans analizleri	78



SİMGELER VE KISALTMALAR DİZİNİ

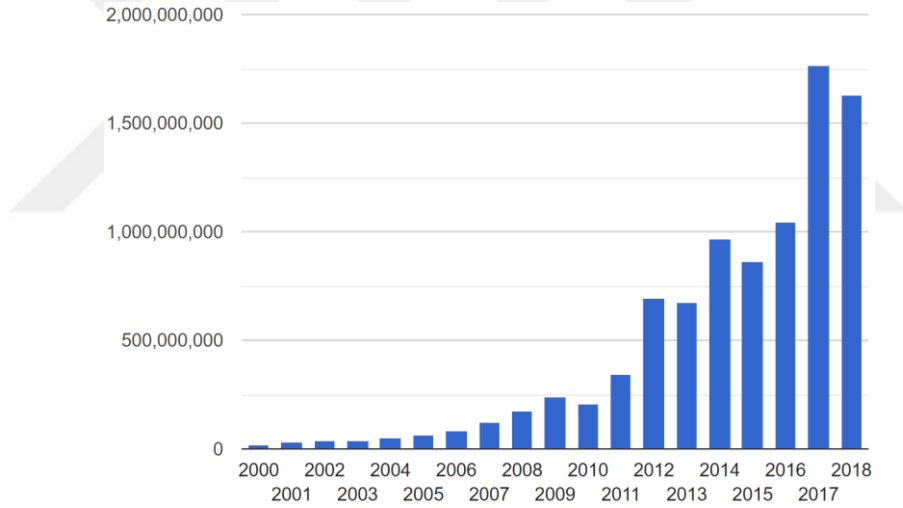
λ	: Alpha
χ	: Chi
ϕ	: Phi
π	: Pi
θ	: Theta
ANOVA	: Analysis of Variance
BM	: Best Match
CDD	: Collection-Document Distance
CLEF	: Cross-Language Evaluation Forum
CTI	: Contribution to Total Inertia
CTR	: Click-Through Rate
DFI	: Divergence from Independence
DFR	: Divergence from Randomness
ERR	: Expected Reciprocal Rank
HTML	: Hypertext Markup Language
IDF	: Inverse Document Frequency
IR	: Information Retrieval
ISR	: Information Storage and Retrieval
KL	: Kullback-Leibler
LGD	: Log-logistic Distribution
LTR	: Learning To Rank
MAP	: Mean Average Precision
MART	: Multiple Additive Regression Trees
nDCG	: Normalized Discounted Cumulative Gain
NTCIR	: NII Testbeds and Community for Information access Research
PL2	: Poisson model with Laplace after-effect and normalization 2
PMI	: Pointwise Mutual Information
PRP	: Probability Ranking Principle
SCQ	: Collection Query Similarity
SCS	: Simplified Clarity Score

SEO	: Search Engine Optimization
SERP	: Search Engine Results Page
SMART	: System for the Mechanical Analysis and Retrieval of Text
SSL	: Secure Socket Layer
SVM	: Support Vector Machines
TF	: Term Frequency
TREC	: Text Retrieval Conference
URL	: Uniform Resource Locator
WWW	: World Wide Web
W3C	: World Wide Web Consortium
WHATWG	: Web Hypertext Application Technology Working Group

1. GİRİŞ

Yazılı bilginin çoğalmasıyla, insanların ihtiyaç duydukları bilgiye erişmesi için bir sisteme ihtiyaç duyulmuştur. Bilgi erişim (information retrieval veya IR) sistemleri kullanıcıların bilgi ihtiyacını gidermek için bilgi içeren belge koleksiyonları içerisinde arama yaparak ilgili belgeleri kullanıcıya getirir. İnternetin yaygınlaşmasından önce bu belge koleksiyonları genellikle kütüphaneler iken günümüzde kayıtlı bilginin büyük bir kısmı internet üzerindedir.

İnternetin yaygınlaşmasıyla birlikte kayıtlı bilgi boyutunda sürekli ve hızlı bir artış gözlemlenmiştir. 1993 yılında 130 olan web sayfası sayısı, 1994 yılında 10,222 sayısına ve 1996 yılında ise 100,000 sayısına ulaşmıştır (Harman, 2019). Günümüzde benzersiz sunucu adına (unique hostname) sahip yaklaşık 2 milyar web sayfası bulunmaktadır. Şekil 1.1 bu web sayfalarının sayısının yıllara göre değişimini göstermektedir (<http-1>).



Şekil 1.1. Web sayfası sayısının yıllara göre değişimi

Web doğası gereği özgürlüktür. Herhangi bir kullanıcı, bir içeriği editör kontrolünden geçmeden bir web sayfası oluşturarak bir bilgiyi yayınlayabilir ve bilgi gereksinimi olan başka bir kullanıcı bu web sayfasındaki bilgiden faydalanabilir. Dolayısıyla kayıtlı bilgi boyutu artarken web belge koleksiyonlarındaki belgeler arası kalite farkı problemi oluşmaktadır.

1.1. Motivasyon

Bilgi erişim sistemleri genellikle belge koleksiyonlarının homojen olduğunu ve belgelerin belirli bir kalite standardını sağladığını varsaymaktadır. Bu varsayım düzgün yapılandırılmış belgelerden oluşan kontrollü koleksiyonlar (örneğin Associated Press ve Wall Street Journal koleksiyonları gibi gazete makalelerinden oluşan koleksiyonlar) için doğru olsa da web belgeleri içeren kontrolsüz belge koleksiyonları için doğru değildir.

Bir web belgesinin kalitesi herhangi bir sorguyla olan alakalılığı ile doğrudan ilişkilidir. Ancak bu ilişki dolayısıyla belge kalitesi ve belge alakalılığı kavramları karıştırılmamalıdır. Belge alakalılığı bir belgenin bir sorguyla olan alaka ilişkisini tanımlarken, belge kalitesi belgenin herhangi bir sorgudan bağımsız olarak kalitesini tanımlamaktadır. Bu sayede belge kalite puanları, bir sorgu olmadan sadece Web belge koleksiyonları üzerinden sorgu zamanı öncesinde çevrimdışı olarak hesaplanabilmektedir. Bununla ilgili olarak Cormack vd. (2011), çalışmalarında sorgu olmaksızın Web belge koleksiyonundaki belgelerin kalitesini ölçmüşler ve belge kalitesinin web belgelerini hedef alan bilgi erişim sistemlerinin başarımını etkileyen önemli bir faktör olduğunu göstermişlerdir. Ayrıca geçmiş çalışmalar da web belgelerinin sonuç listelerinde sıralanması aşamasında belge kalitesini göz önünde bulundurmanın bilgi erişimi sistemi başarımını artırdığını göstermektedir (Bendersky, Croft, & Diao, 2011; Craswell, Robertson, Zaragoza, & Taylor, 2005; Zhou & Croft, 2005).

Bilgi erişimi literatüründeki çalışmaların birçoğu web belgelerinin kalitesini tahmin etmede birden fazla etkenden faydalanmaktadır. Bunlardan en bilineni olan PageRank (Page, Brin, Motwani, & Winograd, 1999), bir web belgesinin kalitesini, o belgenin diğer web belgelerindeki bağlantılar aracılığıyla erişilebilirliği üzerinden hesaplayan bağlantı tabanlı bir ölçüttür. Bu ölçütlere ön belge özellikleri (document prior) de denilmektedir. Benzer şekilde web belgelerinin kalitesini ölçmek için URL (Uniform Resource Locator veya Tekdüzen Kaynak Bulucu) derinliği ve belge uzunluğu vb. birçok ön belge özelliği de zaman içerisinde tanıtılmıştır. Bir web belgesinin kalitesi birçok etkenden etkilenen belirsiz bir kavramdır. Web'in zamanla gelişen yapısı da bu etkenlere gün geçtikçe yenilerinin eklenmesine sebep olmaktadır. Bunun en büyük göstergesi, Web standartlarını geliştiren World Wide Web Consortium ve Web Hypertext Application Technology Working Group organizasyonlarının bu standartları zamanla güncellemeleridir. Bunun sebepleri arasında, kullanıcı alışkanlıkları değişimi (sosyal medya ve akıllı mobil cihazların yaygınlaşması vb.) veya web sayfası kodlarını oluşturan

HTML (Hypertext Markup Language) dilinin versiyonlarının gncellenmesi rnek gsterilebilir. rneęin, web belgeleri iin bir kalite gstergesi olabilecek birok anlamsal etiket (semantic tag), 2008 yılı ncesi web belgelerinde bulunmazken, HTML 5 versiyonuyla tanıtılmıř ve zamanla kullanımı yaygınlařmıřtır (ıplak, Aydın, & Arslan, 2019). Tm bu anlatılanlar gz nnde bulundurulduęunda bir web belgesinin kalitesini lmek, zaman iinde deęiřkenlik gsteren ve tamamlanmayacak zorlu bir iřtir ve her daim iyileřtirmeye aıktır.

1.2. Amalar ve Katkılar

Bu alıřma, bir bilgi eriřim sisteminin bařarımını artırmak amacıyla bir web belgesinin kalitesini len yeni n belge zellikleri sunmaktadır. Bu n belge zelliklerinin oluřturulmasında web belge kalitesi hakkındaki alıřmaların oęunlukla gz ardı ettięi Arama Motoru Optimizasyonu (SEO) alıřmalarındaki uygulamalardan esinlenilmiřtir. n belge zellikleri bir sıralama fonksiyonuna doęrudan entegre edilebileceęi gibi (Bendersky, Croft, & Diao, 2011; Craswell, Robertson, Zaragoza, & Taylor, 2005; Zhou & Croft, 2005) bir ęrenilmiř model tarafından sonu listesinin yeniden sıralanmasında sorgu-baęımsız znitelikler olarak da kullanılabilir (Liu, 2011; Macdonald, Santos, & Ounis, 2012; 2013; Macdonald, Santos, Ounis, & He, 2013; Qin & Liu, 2013).

Bu alıřmada, tanıtılan n belge zellikleri, sorgu-baęımsız znitelikler olarak bir sıralamayı ęrenme daęıtımında (deployment) kullanılmıřtır. Tanıtılan zniteliklerin bilgi eriřim sistemi bařarımına etkisini gzlemlemek iin bu znitelikler, sıralamayı ęrenme alıřmalarında sıklıkla kullanılan sorgu-baęımlı ve sorgu-baęımsız zniteliklerini kullanan bir sıralamayı ęrenme daęıtımına entegre edilerek test edilmiřtir. Ayrıca tanıtılan znitelikler fonksiyonel zelliklerine gre gruplara ayrılarak her bir grubun genel sistem bařarımına etkisi analiz edilmiřtir. alıřmanın katkıları řu řekilde zetlenebilir:

- Web belgelerinin kalitesini lmek zere SEO uygulamalarına dayanan bir znitelik kmesi tanıtılmıřtır.
- Tanıtılan znitelik kmesinin bilgi eriřim sistemi bařarımına katkısının sıralamayı ęrenme daęıtımlarında analiz edilmiřtir.

Tanıtılan öznitelikler fonksiyonel özelliklerine göre gruplara ayrılarak, grupların genel sistem başarımına etkisi analiz edilmiştir.

1.3. Tez Organizasyonu

Tezin kalan kısmı şu şekilde organize edilmiştir:

Bölüm 2 - Bilgi Erişim Sistemleri: Bu bölüm, bilgi erişim sistemleri hakkında genel arka plan bilgisi içermektedir. Bilgi erişim sistemlerinin tarihçesi, sistem süreçleri ve sistemlerin değerlendirme ölçütleri ile ilgili detaylar sunulmuştur.

Bölüm 3 - İlgili Çalışmalar: Bu bölüm, tezin konusu olan ön belge özellikleri ve etkinliklerinin test edilmesi ile ilgili literatürde bulunan çalışmaları tanıtmaktadır.

Bölüm 4 - Sıralamayı Öğrenme: Bu bölüm, çalışmada tanıtılan özniteliklerin etkinliğinin test edilmesinde kullanılan sıralamayı öğrenme kavramını tanıtmaktadır.

Bölüm 5 - SEO-tabanlı Öznitelikler: Bu bölüm, çalışmada tanıtılan öznitelikler, öznitelik grupları ve özniteliklerin hesaplanması hakkında bilgiler içermektedir.

Bölüm 6 - Deneysel Yöntemler: Bu bölüm, çalışmada yapılan deneyler hakkında bilgi içermektedir. Deneylerde kullanılan belge koleksiyonları, sorgu setleri, referans öznitelikler ve deney tasarımı bu bölümde tanıtılmaktadır.

Bölüm 7 - Deney Sonuçları: Bu bölüm, çalışmada yapılan deneylerin sonuçlarını ve sonuçların analizlerini içermektedir.

Bölüm 8 - Sonuç: Bu bölüm, çalışmanın genel değerlendirmesini içermektedir.

2. BİLGİ ERİŞİMİ

“Bilgi Erişimi” bilgi ihtiyacını karşılamak amacıyla, yapılandırılmamış veya yarı-yapılandırılmış bilgi içeren belgeleri bir belge koleksiyonu içerisinde bularak getirme işlemidir. Bir bilgi erişimi sistemi, girdi olarak kullanıcıdan sorgu alır ve çıktı olarak bir belge listesi döndürür. Sorgu, bilgi ihtiyacını özetleyen anahtar kelime veya kelimeler topluluğudur. Bilgi erişim sistemi, kendisine verilen sorguyla alakalı belgeleri koleksiyondan seçerek bir belge listesi oluşturur ve kullanıcıya bu listeyi döndürür (Baeza-Yates & Ribeiro-Neto, 1999; Manning, Raghavan, & Schütze, 2008; Croft, Metzler, & Strohman, 2010).

2.1. Bilgi Erişimi Tarihi

İnsanlık, bilginin yazıya aktarılmasıyla birlikte etkili bir şekilde kullanmak için bilgi kaynaklarını düzenleyerek saklamanın önemli olduğunu anlamıştır. Bu önem, bilginin yazıya aktarılmasının kolaylaşmasıyla (kâğıdın icadı ve yaygınlaşması, matbaanın icadı, vs.) ve yazılı bilgi kaynaklarının çoğalmasıyla daha da artmıştır. Bir kataloğu tarayarak belirli bir dokümanı bulabilen ilk mekanik alet, 1930’lu yıllarda Goldberg tarafından icat edilen “Statistical Machine” isimli bir alettir (Buckland, 2006). Dijital kütüphanelerin geliştirilmesi ve bilgi erişimi alanlarında ilham verici bir başlangıç noktası olarak kabul edilen ise 1945 yılında Bush tarafından önerilen, bilginin nasıl saklanacağını ve bilgiye nasıl erişilebileceğini tarif eden “Memex” adlı kurgu bir alettir (Bush, 1945).

Bilgisayarların gelişmesi ile bilgi erişimi alanındaki çalışmalar hız kazandı. 1951 yılında Calvin Mooers ilk kez “Bilgi Erişimi” terimini kullandı (Mooers, 1950). 1957 yılında Luhn, belgeler için indeksleme birimleri olarak kelimeleri kullanmayı ve erişim için bir kriter olarak kelime eşleşmesini ölçmeyi önerdi (Luhn, 1957). Luhn aynı zamanda bir terimin belgedeki frekansının terimin o belgedeki öneminin ölçülmesinde kullanılabileceğini gösterdi (Luhn, 1958). Bu çalışmalarla birlikte, belgelerin bir sorguyla alakalılıklarına göre sıralanması da başlamıştır. 1960’lı yıllarda Gerard Salton ve ekibi SMART sistemini kurmuş, bilgi erişimi alanında birçok çalışma yapmış ve ISR (Information Storage and Retrieval) raporları adında teknik raporlar yayınlamışlardır (Salton, 1971). Bu çalışmaların en önemlilerinden biri, sorgu ve belgelerin vektörlerle ifade edilerek alakalılıklarının kosinüs benzerliği ile hesaplanmasıdır. Bir diğer önemli

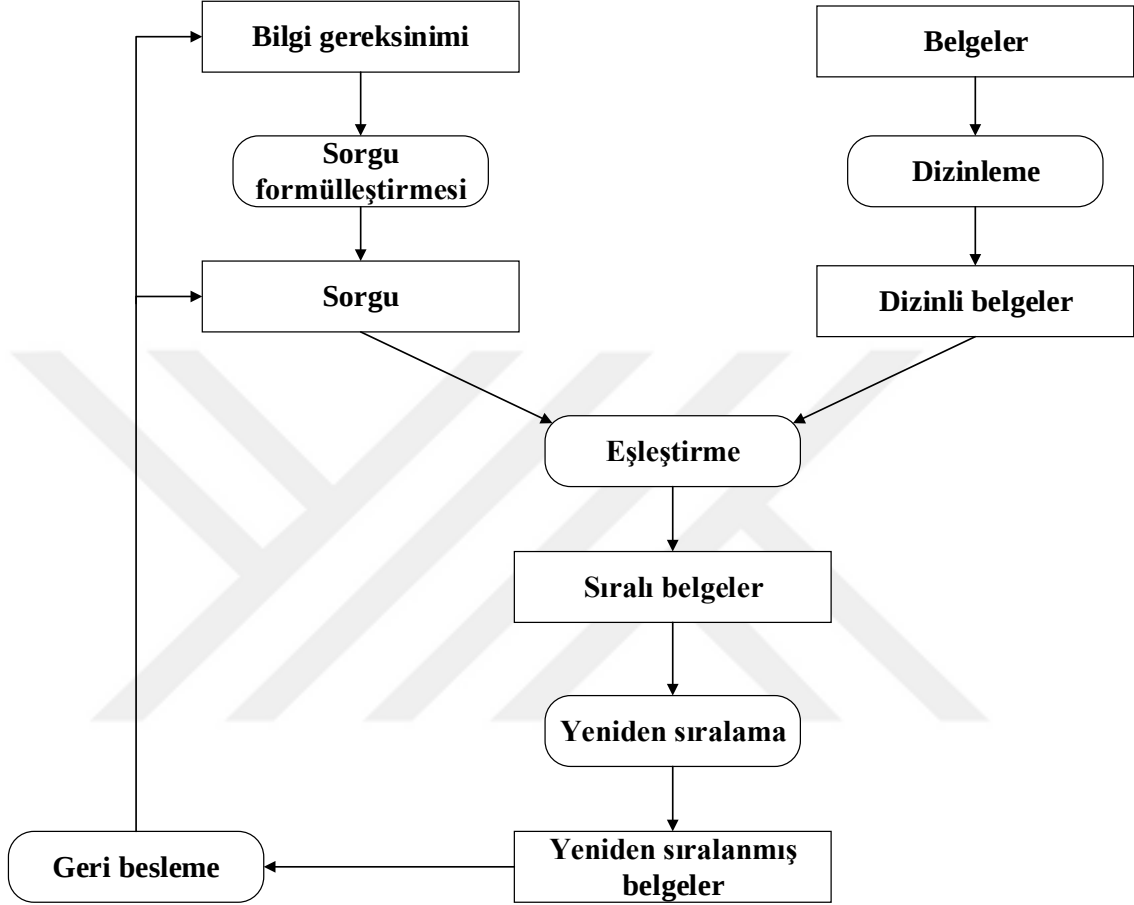
çalışma ise Rocchio algoritması olarak da bilinen alaka düzeyi geri bildirimidir (Rocchio, 1971). Aynı yıllarda Cyril Cleverdon ve ekibi tarafından gerçekleştirilen Cranfield deneyleri sonucunda günümüzde de kullanılan bilgi erişimi sistemlerinin etkililiğini ölçmek için statik bir test koleksiyonu, sorgu-belge alaka yargıları ve standart ölçütleri barındıran bir değerlendirme metodolojisi geliştirilmiştir (Cleverdon C. , 1991).

İlerleyen yıllarda bilgi erişimi alanında birçok çalışma yapılmış ve başta terim ağırlıklandırma modelleri olmak üzere birçok problem için alternatif çözüm yolları türetilmiştir. Ancak akademik çalışmaların test edildiği belge koleksiyonları belge sayısı açısından yeterli değildi. Bunun için Voorhees ve Harman, 1992 yılında çok sayıda uluslararası araştırma grubunun büyük test koleksiyonları oluşturmak için iş birliği yaptığı TREC konferansını (Text REtrieval Conference) başlattı (Voorhees & Harman, 2005). Günümüzde akademik çalışmalarda kullanılan test koleksiyonlarının büyük bir kısmı TREC tarafından oluşturulmuştur. 1990'lı yılların başında Tim Berners-Lee tarafından CERN'de geliştirilen World Wide Web (WWW) sayesinde her geçen yıl kayıtlı belge sayısı hızlı bir biçimde artmaktadır (Berners-Lee, Cailliau, Luotonen, Nielsen, & Secret, 1994). Bu artışla birlikte son yıllarda etkili sıralama modelleri oluşturmak için makine öğrenmesi teknolojilerinden yararlanmak mümkün hale geldi. Makine öğrenmesi yöntemleriyle önceden tanımlanmış özniteliklerin sıralama için nasıl birleştirileceğini öğrenen yöntemlere “Sıralamayı Öğrenme” (Learning-to-Rank) yöntemleri denilmektedir (Liu, 2009).

2.2. Bilgi Erişim Sistemi Süreçleri

Bilgi erişim sistemleri, genellikle metinsel belgeler üzerinde bilgi erişimi işlemlerini gerçekleştiren yazılımlardır. Bilgi erişim sistemi, kullanıcıların ihtiyaç duydukları bilgileri içerebilecek belgelerin varlığı ve yeri hakkında bilgi verir. Önerilen belgelerin bir kısmının kullanıcının bilgi ihtiyacını karşılaması amaçlanmaktadır. Kullanıcının bilgi ihtiyacını karşılayabilecek belgelere alakalı belgeler denir. Mükemmel bir bilgi erişim sistemi, kullanıcıya yalnızca ve tüm alakalı belgeleri getirir. Bir bilgi erişim sisteminde üç temel süreç vardır: belgelerin içeriğinin temsili, kullanıcının bilgi ihtiyacının temsili ve bu iki temsilin karşılaştırılması (Hiemstra, 2009). Bu işlem sonucunda eşleşen belgeler eşleştirme metodunun ürettiği puanlara göre sıralanarak kullanıcıya listelenir. Son yıllarda bilgi erişim sistemlerine dördüncü bir süreç daha eklenmiştir: yeniden sıralama. Bir yeniden sıralama metodu olan sıralamayı öğrenme,

eşleştirme metodu tarafından sıralanan belgeleri makine öğrenmesi aracılığıyla tekrar sıralar ve kullanıcıya sıralı belge listesi döndürür. Bilgi erişim sistemi süreçleri Şekil 2.1’de gösterilmiştir.

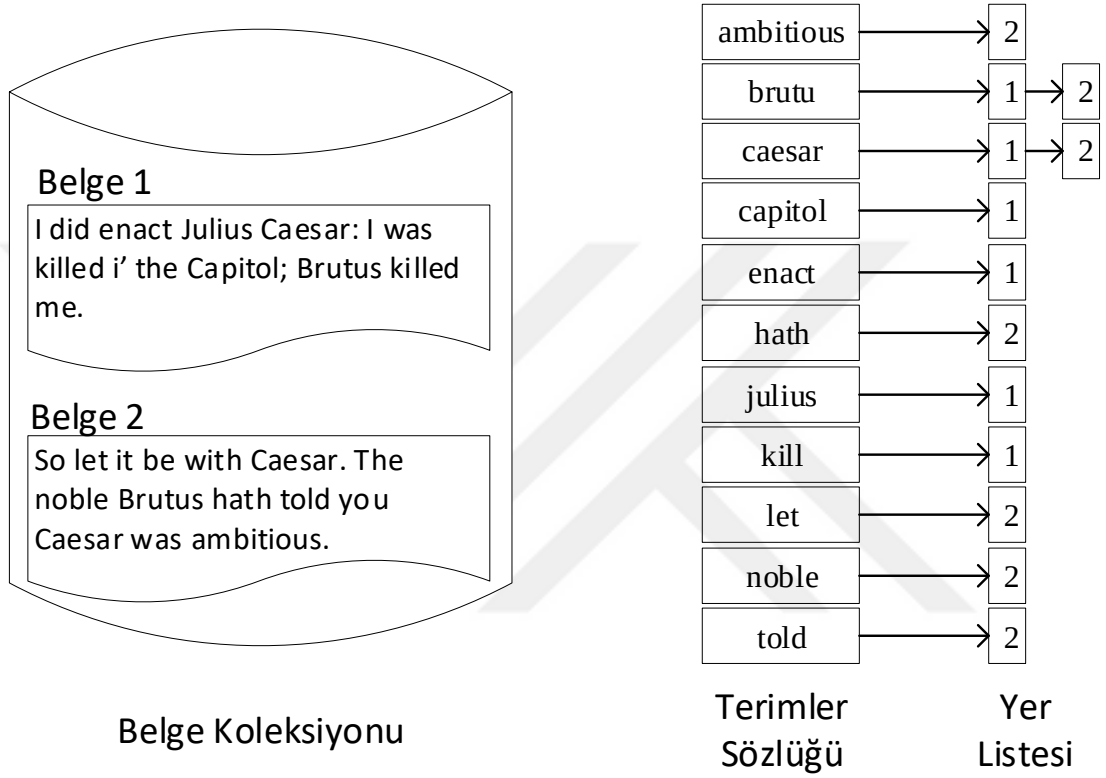


Şekil 2.1. Bilgi erişimi sisteminin süreçleri

2.2.1. Belge içeriği temsili (Dizinleme)

Bilgi erişiminde bir sorgu için alakalı belgelerin getirilmesi aşamasında sorgu terimlerinin belgelerde bulunup bulunmamasına bakılır. Bu işlemde her sorgu terimi için tüm belge koleksiyonunun taranması gerekmektedir. Bu işlem uzun süreceği için belge içerikleri dizinlerle (index) temsil edilmektedir. Belge içeriklerinin dizinlenmesi için kullanılan basit terim-belge matrislerinde satır ve sütunlarda belge koleksiyonundaki terimler ve belgeler bulunur. Matris hücrelerinde ise ilgili terimin belgede bulunması veya bulunmaması durumuna göre 1 veya 0 değerleri yer alır. Bu gösterim çok fazla yer kaplayacağı ve hücrelerin çok büyük bir kısmı 0 değeri alacağı için ters dizin (inverted index) yapıları kullanılmaktadır.

Ters dizin yapılarında öncelikle belge koleksiyonundaki tüm terimleri içeren bir terim sözlüğü oluşturulur. Bu sözlükte her bir terimin işaret ettiği, ilgili terimi içeren belgelerin belirtildiği bir yer listesi (posting list) bulunmaktadır. Böylelikle sorgu terimleri dizinde bulunarak, o terimi içeren belgelere hızlıca ulaşılabilir. Örnek bir ters dizin yapısı Şekil 2.2’de gösterilmiştir.



Şekil 2.2. Ters dizin yapısı

Belge koleksiyonundaki belgeler belirli ön işlemlerden geçerek terim sözlüğü oluşturulur. Bu işlemler belge içeriğinin tokenizasyonu, durak kelimelerin elenmesi, normalizasyon ve gövdelemedir (stemming).

2.2.1.1. Tokenizasyon

Tokenizasyon, bir belge içeriğinin token adı verilen küçük parçalara ayrılma işlemidir. Token, işlem yapılabilen, anlamsal bir birim olarak gruplandırılmış bir karakter dizisidir. Genellikle tokenizasyon işleminde noktalama işaretleri atılır. Tokenizasyon işlemi metnin diline bağlı olarak belirli kurallarla yapılır. Örneğin, çoğu dil için uygulanan kurallardan biri boşluk karakterine göre ayırmakken, Almanca dilinde isim

taamlamalarının birleşik yazılması (“Lebensversicherungsgesellschaftsangestellter” = “Hayat sigortası şirketi çalışanı” vb.) ve çoğu Doğu Asya dillerinde kelimeler arası boşluk bırakılmaması gibi örnekler dolayısıyla farklı dillerde farklı kurallar eklenmesi gerekmektedir (Manning, Raghavan, & Schütze, 2008).

2.2.1.2. Durak kelimelerin elenmesi

Durak kelimeler (stop word), belge koleksiyonunda çok fazla sayıda belgede geçen ve ayırt ediciliği çok az olan kelimelerdir. Türkçede “ve”, “veya”, “ama”, “o”, “bu”, vb. kelimeler durak kelimelere örnek gösterilebilir. Bu kelimeler hem hafızadan tasarruf hem de sistemin hızlı çalışması gibi sebeplerle terim sözlüğüne eklenmezler. Normal bir belge koleksiyonunda durak kelimeler çıkarılarak terim sözlüğünde %40’lık bir tasarruf elde edilebilir. Bu işlemin sistem için büyük faydasının olmasının yanı sıra zararı da vardır. Bunlardan en önemlisi durak kelimelerin kullanıcının bilgi ihtiyacında önemli olması durumudur. Bu durum için en bilinen örnek kullanıcının İngilizce “to be or not to be” ifadesini içeren belgeleri aramak istemesidir (Baeza-Yates & Ribeiro-Neto, 1999).

2.2.1.3. Normalizasyon

Tokenizasyon işleminden sonra elde edilen karakter dizilerinin sorgu ve belgelerde eşleşmesi beklenmektedir. Ancak aynı terimlerin farklı yazımları mevcut olduğundan bunların normalize edilmesi gerekmektedir. Örneğin, sorgusunda “USA” bulunan bir kullanıcı, muhtemelen içerisinde “U.S.A.” geçen belgeleri de incelemek isteyecektir. Normalizasyon işleminde birden fazla gösterimi olan karakter dizileri standart bir hale getirilerek eşleşmeleri sağlanmaktadır. Bu aşamada birden fazla işlem yapılmaktadır. Bunlar arasında örnekte verildiği gibi kısaltmalardaki noktaların veya “-” işaretlerinin atılması gibi işlemler vardır. Bunun haricinde tüm harfleri küçük harflere dönüştürme, aksanların kaldırılması (naïve sözcüğünün naive haline dönüştürülmesi) ve dile özgü başka standartlaştırma işlemleri de normalizasyon aşamalarındandır.

2.2.1.4. Gövdeleme

Kullanıcının girdiği sorgu terimlerinin, kendileri bulunmamasına rağmen farklı varyantları (çoğul eki, yapım eki, zaman eki vb. almış halleri) kullanıcının bilgi ihtiyacını karşılayabilecek alakalı belgelerde bulunabilir. Bu sorun terimlerin gövde (stem)

hallerine getirilmesiyle kısmen çözülmektedir. Gövde, bir kelimenin (kullanılan dile bağlı olarak) ön veya son (bazen her ikisi de) eklerinden arındırılmış halidir. Örneğin, Türkçedeki “okur”, “okuyor”, “okudum” gibi sözcüklerin hepsi gövdeleme algoritması sonucunda “oku” sözcüğüne indirgenmelidir. Gövdeleme, birden fazla varyantı olan bir sözcüğü tek bir gövde olarak ele aldığı için dizin boyutunda tasarruf sağlamaktadır. Bunun yanı sıra, sorgu içerisindeki bir terimin belgelerdeki farklı varyantları ile eşleşmesine olanak sağladığından bilgi erişimi performansını artırdığı düşünülmektedir. Terim sözlüğü oluşturma aşamaları Şekil 2.3’te bir örnek belge üzerinden gösterilmiştir.

Belge 1	«I did enact Julius Caesar: I was killed i' the Capitol; Brutus killed me.»													
Tokenizasyon	I	did	enact	Julius	Caesar	I	was	killed	i	the	Capitol	Brutus	killed	me
Durak Kelimelerin Elenmesi	enactJuliusCaesarkilledCapitolBrutuskilled													
Normalizasyon	enactjuliuscaesarkilledcapitolbrutuskilled													
Gövdeleme	enactjuliuscaesarkillcapitolbrutukill													

Şekil 2.3. Terim sözlüğü oluşturma aşamaları

2.2.2. Bilgi ihtiyacı temsili

Bilgi erişim sistemleri kullanıcının bilgi ihtiyacını giderebilecek belgeleri bulabilmek için bir sorguya ihtiyaç duyar. Buna sorgu formülleştirilmesi denir. Sorgu, bilgi ihtiyacını tarif etmek için oluşturulan anahtar kelimeler topluluğudur. Ancak, belge koleksiyonu ve bilgi erişimi hakkında detaylı bilgi sahibi olmayan bir kullanıcı bilgi ihtiyacını sorgu haline dönüştürmekte zorlanır. Bu durum "Hakkında bilgi bulmak için bilmediğin bir şeyi tanımlama gereği" olarak tarif edilebilir (Tonta, 1995). İlk oluşturulan sorgu için bilgi erişimi sisteminden dönen belgelerin bilgi ihtiyacını tam karşılamadığı durumlarda bilgi ihtiyacında değişiklik yapılabilir veya sorgu tekrar formülleştirilerek sistem tekrar çalıştırılır. Bu tür sorgu yeniden formülleştirilmesi, ilk sorguyu yeni terimlerle genişletme ve genişletilmiş sorgudaki terimleri yeniden ağırlıklandırma yollarıyla yapılabilir.

2.2.3. Sorgu-belge eşleşmesi

Bu aşamada bilgi erişim sistemi, kullanıcıdan alınan sorguyla koleksiyondaki belgeler eşleşme modelleri ile alakalılık üzerinden eşleştirir. Çoğu model bu işlem

sırasında her belgeye bir eşleşme puanı atar ve belgeleri bu puanlara göre sıralayarak kullanıcıya döndürür. Mevcut klasik eşleşme modelleri Boolean model, Vektör model ve Olasılıksal modellerdir (Baeza-Yates & Ribeiro-Neto, 1999).

2.2.3.1. Boolean model

Boolean modeli kümeler teorisine ve Boole cebrine dayanan basit bir modeldir. Bu modelde sorgu terimleri Boole cebir işlemleri (“VE”, “VEYA”, “DEĞİL”) ile birleştirilerek Boolean ifade oluşturulur ve bu işlemlerin sonucunda dönen belgeler kullanıcıya getirilir. Örneğin, kullanıcı Shakespeare oyunlarının olduğu bir koleksiyonda Brutus ve Caesar terimlerinin geçtiği oyunlara “Brutus VE Caesar” sorgusu ile erişebilir. Model, Brutus terimini içeren belgeleri ve Caesar terimini içeren belgeleri ayrı birer küme olarak ele alır ve bu kümelerin kesişiminde olan belgeleri kullanıcıya getirir.

Boolean modelde terimlerin ağırlıkları, terimin belgede bulunma durumuna göre 0 veya 1 olabilir ($w_{i,j} \in \{0,1\}$) ve buna bağlı olarak ikili karar mekanizması vardır: bir belge bir sorguyla tam eşleşir (exact match) veya eşleşmez. Bu yüzden kullanıcıya sadece eşleşen belgeler sırasız olarak getirir.

Boolean modelin basit ve hızlı çalışma gibi avantajlarının yanı sıra bazı dezavantajları da vardır. Bunlar arasında bilgi ihtiyacının Boolean ifadelere çevrilmesinin zor olması (özellikle belge koleksiyonuna hakkında kapsamlı bilgi sahibi olmayan kullanıcılar için), sorgu için dönen belge sayısının çok az veya çok fazla olması ve sorgu için dönen belgelerin sıralanmaması bulunmaktadır.

2.2.3.2. Vektör model

Vektör model, terimleri ağırlıklandırır ve bir sorgu için getirilen belgeleri sıralar. Boolean modelin aksine Vektör modelde kısmi eşleşme (partial match) vardır. Kısmi eşleşme, sorgu ve belge terimlerine ikili olmayan ağırlıkların atanmasıyla gerçekleşir. Terim ağırlıkları, koleksiyondaki belgeler ile sorgu arasındaki benzerlik derecesini hesaplamakta kullanılır. Sorgu terimleriyle kısmen eşleşen belgeler benzerlik derecesine göre sıralanarak kullanıcıya getirilir.

Vektör modelde, t sayıda terim içeren bir terimler sözlüğüne sahip bir koleksiyonda her bir d belgesi ve q sorgusu, t boyutlu bir uzayda birer vektör olarak ifade edilir. Vektörlerin her boyutu sözlükteki bir terimi temsil eder ve atanan ağırlık değerini alır.

Yani sorgu vektörü $\vec{q} = (\omega_{1,q}, \omega_{2,q}, \dots, \omega_{t,q})$ ve belge vektörü $\vec{d} = (\omega_{1,d}, \omega_{2,d}, \dots, \omega_{t,d})$ şeklinde gösterilir. Vektör modeli, sorgu ve belge arasındaki benzerlik (similarity) derecesini $\vec{q}$ ve $\vec{d}$ vektörleri arasındaki korelasyonla belirler. Bu korelasyon iki vektör arasındaki açının (θ) kosinüsü ile Denklem 2.1’de belirtildiği gibi ölçülebilir.

$$sim(d, q) = \cos \theta = \frac{\vec{d} \cdot \vec{q}}{|\vec{d}| \times |\vec{q}|} = \frac{\sum_{i=1}^t \omega_{i,d} \times \omega_{i,q}}{\sqrt{\sum_{i=1}^t \omega_{i,d}^2} \times \sqrt{\sum_{i=1}^t \omega_{i,q}^2}} \quad (2.1)$$

Terim ağırlıklandırması için terimin belgede geçme sayısına (frekans) bağlı yöntemler kullanılır. En basit yaklaşım, ağırlığın d belgesindeki t teriminin frekansı ile orantılı olacak şekilde atanmasıdır. Bu ağırlıklandırma şeması terim frekansı veya tf (term frequency) olarak adlandırılır (Luhn, 1957). Terim frekansı bir terimin belgeyle ilişkisini ifade etse de sorunun belgeyle alaka düzeyinin belirlenmesinde tüm terimlerin eşit öneme sahip olması bir sorundur. Çünkü bazı terimlerin ayırt edicilik gücü çok azdır veya hiç yoktur. Bu sorunun çözümü için ters belge frekansı veya idf (inverse document frequency) kullanılır (Jones, 1972). Belge frekansı bir terimin koleksiyonda geçtiği belge sayısını ifade eder. Çok sayıda belgede geçen (belge frekansı yüksek) bir terimin ayırt edicilik gücü azdır. Bu durumun tersi şeklinde de ters belge frekansı yüksek olan terimlerin ayırt edicilik gücü fazladır. Bu iki ölçütün birleştirilmesiyle $tf - idf$ ölçütü elde edilir. Belge sayısı N olan bir koleksiyondaki bir i teriminin ters belge frekansı ve $tf - idf$ ($\omega_{i,d}$) terim ağırlığı Denklem 2.2’deki gibi hesaplanır.

$$idf_i = \log \frac{N}{df_i} \quad (2.2)$$

$$\omega_{i,d} = tf_{i,d} \times idf_i$$

$tf - idf$, belgeleri sıralamada makul ölçüde etkili bir ölçüt olmasına rağmen, uzun belgeleri önyargıyla üst sıralarda getirdiği için bilgi erişimi performansına zarar verebilmektedir (Singhal, Salton, Mitra, & Buckley, 1996). Uzun belgeler genellikle aynı terimleri tekrar tekrar kullanır ve bu yüzden terim frekans faktörleri büyük olabilir. Uzun belgelerde aynı zamanda çok sayıda farklı terim vardır. Bu durum, bir sorgu ile uzun bir belge arasındaki terim eşleşmelerinin sayısını artırarak, uzun belgelerin kısa belgelere göre üst sıralarda gelme şansını artırmaktadır. Bu önyargıyı telafi etmek için genellikle belge uzunluğu normalizasyonu kullanılır.

2.2.3.3. Olasılıksal modeller

Terim ağırlıklandırma probleminin çözümü için türetilen olasılıksal yaklaşımlar, verilerin daha doğru bir uyumunu elde etmek için, belgelerin terim dağılımları ve modele bağlı olarak alakalılık dağılımları gibi özelliklerini modellemeye çalışmaktadır (Wilkie & Azzopardi, 2014). Bu modeller belgelerin bir sorguyla alakalılık olasılıklarına göre azalan sırayla sıralanması gerektiği ilkesini benimsemektedir. Buna Olasılıksal Sıralama İlkesi veya Probability Ranking Principle (PRP) denir (Robertson & Sparck Jones, 1976; Robertson, 1977). Bu ilkeye göre bir D belgesinin bir Q sorgusu ile alakalılık olasılığı puanı Denklem 2.3'teki gibi hesaplanır.

$$S(Q, D) = \log P(R|D) = \sum_{t \in Q \cap D} \log \frac{P(D_t = 1|R) \cdot P(D_t = 0|\bar{R})}{P(D_t = 0|R) \cdot P(D_t = 1|\bar{R})} \quad (2.3)$$

Denklemde R ve $\bar{R}$ değişkenleri sırasıyla belgenin sorgu ile alakalı olma ve alakalı olmama durumlarını ifade etmektedir. D_t ise sorgu terimi t 'nin D belgesinde görülme veya görülmemesi durumuna göre sırasıyla 1 veya 0 değerini alabilen bir değişkendir. Son olarak $P(D_t = 1|R)$ ifadesi t terimini içeren D belgesinin alakalı olma olasılığını ifade etmektedir. Bu ilkeyi temel alan her biri farklı bir olasılık tahmin tekniğine dayanan birçok olasılıksal model önerilmiştir.

2-Poisson modeli özel ve özel olmayan terimlerin temsili için önerilmiştir (Harter, 1975). Bu modelde özel terimler seçkin belgelerde yoğun olarak bulunan ve bir belgenin bilgilendirme seviyesine katkıda bulunan terimlerdir. 2-Poisson modeli belirli metinler için iyi çalışmasına rağmen pratikte çok iyi performans sergileyememiş ancak En İyi Eşleşme veya Best Match (BM) modelleri olarak bilinen olasılıksal modellerin gelişmesine öncülük etmiştir. Bu modeller arasında en çok kullanılanı, günümüzde de sıkça kullanılan BM25 modelidir. BM25 modeli bir D belgesine bir Q sorgusu için atayacağı puanı Denklem 2.4'te olduğu gibi hesaplar:

$$S(Q, D) = \sum_{t \in Q \cap D} IDF(t) \cdot \frac{tf_{t,D} \cdot (k_1 + 1)}{tf_{t,D} + k_1 \cdot (1 - b + b \cdot \frac{|D|}{avgdl})} \quad (2.4)$$

Denklemde k_1 ve b serbest parametreleri sırasıyla terim frekans doygunluğunu ve belge uzunluğunu normalize etmektedir. $|D|$ ve $avgdl$ ifadeleri ise sırasıyla D belgesinin uzunluğunu ve koleksiyondaki ortalama belge uzunluğunu belirtmektedir.

Dil modeli ifadesi, konuşma tanıma (speech recognition) topluluğu tarafından, dil üretiminin istatistiksel düzenliliklerini yakalayan bir olasılık dağılımına atıfta bulunmak için kullanılır. Konuşma için dil modelleri, sırayla bir sonraki kelimenin olasılığını tahmin etmeye çalışır. Bilgi erişiminde de kullanılan dil modeli yaklaşımında amaç, her belge için bir dil modeli çıkarmak ve bu modellerin her birine göre sorguyu üretme olasılığını tahmin etmektir (Ponte & Croft, 1998). Belgeler tahmin edilen bu olasılıklara göre sıralanırlar. Olasılık hesaplaması Denklem 2.5'te verilmiştir.

$$S(Q, D) = \prod_{t \in Q} P(t|D), \quad P(t|D) = \frac{tf_{t,D}}{|D|} \quad (2.5)$$

Denklemden, sorgu terimlerinden birinin belgede görülmediği bir durumda belgenin olasılık puanı 0 olmaktadır. Bunu önlemek için bazı teknikler kullanılmaktadır. Bunlardan biri belgede görülmeyen terim için belgedeki olasılığı yerine koleksiyondaki olasılığın kullanılmasıdır (Ponte & Croft, 1998). Bunun haricinde sıklıkla kullanılan pürüzsüzleştirme (smoothing) teknikleri arasında Jelinek-Mercer (Jelinek, 1980), Dirichlet Prior (MacKay & Peto, 1995) ve Absolute Discount (Ney, Essen, & Kneser, 1994) bulunmaktadır (Zhai & Lafferty, 2004).

Rastgelelikten Sapma veya Divergence from Randomness (DFR) yapısı, 2-Poisson modelinden ilham alır ve seçkinlik kavramını, anlamsal bilgi teorisi ve Popper'ın bilgi içeriği kavramı (Popper, 1995) ile geliştirir (Amati & van Rijsbergen, 2002). Bir DFR modeli belgelere puan atamak için iki sapma fonksiyonu ve bir normalizasyon fonksiyonu kullanır. Sapma fonksiyonlarından ilki ($Prob_1$) bir terimin rastgeleliğini karakterize etmek için ve ikinci fonksiyon ($Prob_2$) terimin belge tanımlayıcı olarak kullanılma riskini modellemek için kullanılır ve belge puanı Denklem 2.6'da olduğu gibi hesaplanır.

$$S(Q, D) = \sum_{t \in Q \cap D} (1 - Prob_2) \cdot (-\log_2 Prob_1) \quad (2.6)$$

DFR yapısının çalışma prensibinde ilk olarak terimin bilgilendirici içerik olasılığı ölçülür ($Prob_1$). Bunun için binom dağılımı (Poisson veya Sapma yaklaşımları ile), Bose-Einstein istatistiği, ters terim frekansı vb. modeller kullanılır. Bir sonraki adımda, terimin iyi bir belge tanımlayıcısı olarak kabul edildiğindeki bilgi kazanımını hesaplayan ilk normalizasyon uygulanır ($1 - Prob_2$). Bu aşamada kullanılan normalizasyon yöntemleri Laplace'ın Ardıllık Kuralı (Law of Succession) veya Bernoulli işlemleridir. Son olarak

terim frekansı belge uzunluğu ile normalize edilerek ikinci normalizasyon uygulanmış olur.

DFR çatısı altında farklı rastgelelik modelleri ve normalizasyonlar kullanılarak farklı modeller türetilmiştir. Bunların arasında iyi performans sergileyen modellerden biri PL2 modelidir (Amati & van Rijsbergen, 2002). PL2 modeli rastgelelik için Poisson dağılımını ve birinci normalizasyon için Laplace kuralını kullanır:

$$S(Q, D) = \sum_{t \in Q \cap D} \frac{1}{tfn + 1} \left(tfn \cdot \log_2 \frac{tfn}{\lambda} + (\lambda - tfn) \cdot \log_2 e + 0.5 \log_2 (2\pi \cdot tfn) \right) \quad (2.7)$$

$$tfn = tf \cdot \ln \left(1 + \frac{c \cdot avg_l}{l(d)} \right) \quad (2.8)$$

Denklem 2.7’de λ , terimin koleksiyondaki ortalama frekansını göstermektedir. İkinci normalizasyon için kullanılan ve bir c sabiti içeren terim frekansı normalizasyonunu gösteren Denklem 2.8’de ise $l(d)$ gözlemlenen belge uzunluğunu, avg_l ise koleksiyondaki ortalama belge uzunluğunu ifade etmektedir. Parametre kullanmayan, Hipergeometrik dağılım ve Laplace normalizasyonu kullanan DLH, Popper normalizasyonu kullanan DPH (Amati G. , 2006) ve DFRee (Amati G. , 2009) modelleri DFR çatısı altında oluşturulan modellerdir.

Bilgi tabanlı modeller ailesi, Harter (1975) tarafından bahsedilen ve DFR yapısında da kullanılan “bir belgedeki bir kelimenin dağılımı, koleksiyondaki ortalama dağılımından ne kadar saparsa, bu kelimenin incelenen belge için o kadar önemli olması daha olasıdır” ilkesinden ilham almaktadır. Bilgi tabanlı bir model olan LGD (Log-logistic Distribution), terimlerin seçkinliğini temsil etmenin alternatif bir yolu olarak log-logistik dağılımı kullanarak basitleştirilmiş bir DFR modeli oluşturur (Clinchant & Gaussier, 2011). Formülü Denklem 2.9’da verilen LGD modelinde λ_t , Denklem 2.7’deki gibi terimin koleksiyondaki ortalama frekansını ve tfn ise Denklem 2.8’de verilen terim frekansı normalizasyonunu belirtmektedir. Bu model aynı zamanda Jelinek-Mercer metodu ile pürüzsüzleştirilmiş bir dil modeli de içerdiğinden DFR yapısı ile dil modelleri arasında bir köprü olarak da görülmektedir.

$$S(Q, D) = \sum_{t \in Q \cap D} -\log \left(\frac{\lambda_t}{\lambda_t + tfn} \right) \quad (2.9)$$

Bağımsızlıktan sapma veya Divergence from Independence (DFI) modeli DFR yapısının parametrik olmayan karşılığıdır. DFI modeli de DFR gibi terimlerin özel veya özel olmayan şeklinde ayrıştırılmasını frekans dağılımlarındaki farklılıklar üzerinden yapar ancak bunu belirlemek için rastgelelik yerine doymuş bağımsızlık modelini (saturated model of independence) kullanır (Kocabaş, Dinçer, & Karaoğlu, 2014). Bağımsızlık kavramına göre, eğer iki terimin frekansları oranı tüm belgelerde sabitse bu terimlerin belgelerde gözükmesi belgelerden bağımsızdır. Özel olmayan terimler her belgenin içeriğine eşit katkı sağladığından, belgelerde her bir terimin gözlenen sıklığının bağımsızlık altında beklenen sıklıktan farklılığı ölçülerek özel terimler ayırt edilebilir. Bir t teriminin bir D belgesindeki beklenen frekansının hesaplanması Denklem 2.10'da verilmiştir.

$$e_{t,D} = TF_t \cdot \frac{|D|}{N} \quad (2.10)$$

Denklem 2.10'da TF_t terimin koleksiyondaki frekansını ve N koleksiyon uzunluğunu belirtir. DFI modelinden 3 farklı ölçüt oluşturularak puan ataması yapılır. Bunlar doymuş bağımsızlık modelini temel alan DFIB (Denklem 2.11), bağımsızlıktan normalize edilmiş Ki-Kare uzaklığını temel alan DFIC (Denklem 2.12) ve standardizasyonu temel alan DFIZ (Denklem 2.13) ölçütleridir.

$$S(Q, D) = \sum_{t \in Q \cap D} \log_2 \left(\frac{tf_{t,D} - e_{t,D}}{e_{t,D}} + 1 \right) \quad (2.11)$$

$$S(Q, D) = \sum_{t \in Q \cap D} \log_2 \left(\frac{(tf_{t,D} - e_{t,D})^2}{e_{t,D}} + 1 \right) \quad (2.12)$$

$$S(Q, D) = \sum_{t \in Q \cap D} \log_2 \left(\frac{tf_{t,D} - e_{t,D}}{\sqrt{e_{t,D}}} + 1 \right) \quad (2.13)$$

2.2.4. Yeniden sıralama

Yeniden sıralama işlemi sorgu-belge eşleşme aşamasından sonra uygulanmaktadır. Eşleşme sırasında bir sorgu için belgelere eşleşme puanları atanarak belgeler bu puana göre sıralanır. Bu sıralı belgenin çeşitli tekniklerle sistem etkinliğini artırmak amacıyla tekrar sıralanması işlemine yeniden sıralama denir.

Makine öğrenmesi tekniklerinin bilgi erişimine adapte edilmesiyle geliştirilen yeniden sıralama işlemine sıralamayı öğrenme (Learning to Rank veya LTR)

denilmektedir (Liu, 2009). Son yıllarda birçok sıralamayı öğrenme teknikleri önerilmiştir ve bu teknikler bilgi erişimi alanında sıklıkla kullanılmaktadır.

2.3. Bilgi Erişim Sistemi Değerlendirilmesi

Bilgi erişim sistemlerinin değerlendirilmesi, bir sistemin, kullanıcılarının bilgi ihtiyaçlarını ne kadar iyi karşıladığını değerlendirme sürecidir. Sistem değerlendirmesi ve kullanıcı tabanlı değerlendirme olmak üzere iki sınıf değerlendirme vardır. Kullanıcı tabanlı değerlendirme, kullanıcının sistemden memnuniyetini ölçerken sistem değerlendirmesi, sistemin belgeleri ne kadar iyi sıralayabileceğine odaklanır. Kullanıcı tabanlı değerlendirmenin maliyetinin çok olması ve doğru şekilde yapılmasının zor olması sebebiyle bilgi erişim çalışmalarında sistem değerlendirmesi uygulanır (Voorhees E. M., 2002).

Sistem değerlendirmesi testleri ilk olarak Cranfield 2 deneyinde uygulanmıştır (Cleverdon C. , 1997). Deney uygulandığı zamandan itibaren deneysel bilgi erişimi çalışmaları için baskın bir sistem değerlendirme paradigması oluşturmuştur (The Cranfield Paradigm). Araştırmacılar bu paradigmayı temel alarak zamanla standart bir deneysel tasarım geliştirmişler ve bu tasarım Cranfield paradigmasının içsel bir parçası haline gelmiştir. Zamanla geliştirilen ve standardize edilen bu deneysel tasarım baskın bir şekilde birçok bilgi erişimi konferanslarında ve forumlarında kullanılmaktadır. Bunlardan en önemlileri arasında TREC, Cross-Language Evaluation Forum (CLEF) ve NII Testbeds and Community for Information access Research (NTCIR) bulunmaktadır.

Cranfield paradigmasına göre bir test koleksiyonu belgeler, bilgi gereksinimleri ve bir bilgi gereksiniminin bir belge ile alakalı olup olmadığını belirten alakalılık yargıları (relevance judgments veya qrels) olmak üzere üç farklı bileşenden oluşur (Voorhees E. , 2007). Şekil 2.4, örnek bir bilgi gereksinimini göstermektedir. Genellikle bu yapılarda sorgu (query) ve sorgu açıklaması (description) bulunur. Şekil 2.5 ise alakalılık yargıları örneğini göstermektedir. Alakalılık yargılarının farklı test koleksiyonları için farklı gösterimleri olmakla beraber temelde sorgu-belge ikilisinin birbiriyle alakalı olup olmadığını belirtirler. Örnekte her satırda bir alakalılık yargısı bulunmaktadır ve her satır 4 sütundan oluşmaktadır. İlk sütun bilgi gereksinimi numarasını, ikinci sütun iterasyon numarasını, üçüncü sütun belge numarasını ve son olarak dördüncü sütun sorgu-belge ikilisinin alakalılık yargısını belirtir. Bazı test koleksiyonlarında alakalılık değerlendirme düzeyi 0 veya 1 (alakasız veya alakalı) olmak üzere iki kademeli olsa da genellikle çok

kademeli deęerlendirmeler tercih edilmektedir. Bu deęerlendirme t r ne g re 0 yargısı sorgu-belge ikilisinin alakasız olduęunu belirtirken 0'dan y ksek yargılar azdan  oęa doęru alakalılık derecelerini belirtirler. Bazı alakalılık yargılarında bulunan -2 yargısı ise belgenin yararlı olmayan veya zararlı i erięe sahip olduęunu belirtir. Alakalılık yargıları deęerlendiriciler tarafından yapılmaktadır. Genellikle bir sorgu-belge ikilisinin alaka derecesi birden fazla deęerlendirici tarafından deęerlendirilir ve farklı birleřtirme metotlarıyla birleřtirilir.

```
<topic number="280" type="single">  
  <query>view my internet history</query>  
  <description> How can I find my internet browsing history, i.e., the web sites I  
have previously visited?  
</description>  
</topic>
```

řekil 2.4. * rnek bir bilgi gereksinimi*

```
280 0 clueweb12-0000wb-33-21520 2  
280 0 clueweb12-0000wb-33-34663 2  
280 0 clueweb12-0000wb-47-32511 0  
280 0 clueweb12-0000wb-74-01857 0  
280 0 clueweb12-0001wb-81-29716 1  
280 0 clueweb12-0003wb-22-09848 2  
280 0 clueweb12-0003wb-55-09684 1  
280 0 clueweb12-0003wb-92-14460 0  
280 0 clueweb12-0003wb-93-01794 0  
280 0 clueweb12-0004wb-50-30934 0
```

řekil 2.5. * rnek alakalılık yargıları*

Bir test koleksiyonunu hedef alan karřılařtırılacak her bilgi eriřim sistemi, koleksiyondaki her sorgu i in sıralanmış belgelerden sıralı bir belge listesi  retir. Sistem, her bir sorgu i in belgelere bir puan atar ve belgeleri bu puana g re sıralar. Bir sorgu i in puan sırasına g re genellikle  nceden belirlenmiş ilk k adet belge listelenir (top- k retrieval) ve bu iřlem koleksiyondaki her sorgu i in yapılır. Bir sistemin bir sorgu setindeki t m sorgular i in  retmiş olduęu listeye  alıřma (run) denir. Farklı sistemler aynı sorgu seti ve aynı test koleksiyonu i in  alıřmalar  retir. řekil 2.6, TREC formatında hazırlanmış  rnek bir  alıřmadan bir kesiti g stermektedir. Bu formata g re her satırda 6 s t n bulunmaktadır. Bunlardan ilki sorgu numarasını belirtir. İkinci s t n kullanılmayan ve sabit olarak atanmış belirte  barındırır.     nc  s t n belge numarasını, d r  nc 

sütun ise belgenin o sorgu için olan sırasını (rank) belirtir. Beşinci sütunda sıralamanın yapıldığı sorgu-belge ikilisi puanı ve altıncı sütunda ise çalışma etiketi yer alır.

280	Q0	clueweb12-0806wb-32-08913	1	7.726082	BM25k1.2b0.75_KStem_OR_all
280	Q0	clueweb12-1100wb-20-24467	2	7.601938	BM25k1.2b0.75_KStem_OR_all
280	Q0	clueweb12-1808wb-23-20870	3	7.5424395	BM25k1.2b0.75_KStem_OR_all
280	Q0	clueweb12-0806wb-32-08907	4	7.476966	BM25k1.2b0.75_KStem_OR_all
280	Q0	clueweb12-0806wb-34-13812	5	7.447985	BM25k1.2b0.75_KStem_OR_all

Şekil 2.6. Örnek bir çalışma kesiti

Tek bir sorgu için bir sistemin etkinliği, alakalılık yargılarını ve ilgili belgelerin sıralarını göz önünde bulunduran bir değerlendirme ölçütü ile hesaplanır. Sistemin genel etkinliği, daha sonra sorgu setindeki sorgular genelinde ortalama puan olarak hesaplanır ve ortalama puanlar karşılaştırılır. Daha iyi ortalama puanlara sahip çalışmalar üreten sistemler, daha etkili olarak kabul edilir (Voorhees E. , 2019).

2.3.1. Bilgi erişim sistemi değerlendirme ölçütleri

Bir bilgi erişim sisteminin genel kalitesi, sistemin hızı, kullanıcı arayüzü, sorgu dili gibi sistemin birçok farklı yönüne ve kullanıcılarının bilgi arama davranışına bağlıdır. En önemli yönlerden biri, bilgi erişim sisteminin etkinliği, yani kullanıcının bilgi ihtiyacıyla ilgili belgeleri getirme yeteneğidir. Etkinlik ölçütleri, bir bilgi ihtiyacına göre döndürülen belge listesinin alaka düzeyini özetler (Carterette, Kanoulas, & Yilmaz, 2012).

Kesinlik (Precision) ve Duyarlılık (Recall), bilgi erişim sistemlerinde ilk kullanılan ölçütler arasındadır. Her iki ölçüt de ikili alaka düzeyini varsayar, yani bir belge ya kullanıcının bilgi ihtiyacıyla alakalıdır ya da değildir. Ayrıca, sıralanmış listeler yerine belge kümeleri üzerinde çalışırlar; yani, bir erişim sisteminin belgeleri alakalı/alakasız olarak sınıflandırdığını ve yalnızca alakalı olarak sınıflandırılan alt kümeyi döndürdüğünü varsaymaktadırlar. Kesinlik getirilen belgelerin ne kadarının alakalı olduğunu ölçerken, duyarlılık alakalı belgelerin ne kadarının getirildiğini ölçmektedir. Bu ölçütler sınıflandırma alanında da kullanılan durumsallık tablosu (contingency table) ile de Tablo 2.1’de olduğu gibi ifade edilebilmektedir:

Tablo 2.1. Durumsallık tablosu

	Alakalı	Alakasız
Getirilen	gerçek pozitif (tp)	yanlış pozitif (fp)
Getirilmeyen	yanlış negatif (fn)	gerçek negatif (tn)

Tabloya göre kesinlik ve duyarlılık ölçütlerini formüle edilecek olursa:

$$kesinlik = \frac{tp}{tp + fp} \quad duyarlılık = \frac{tp}{tp + fn} \quad (2.14)$$

Kesinlik ve duyarlılık arasında bir ödünleşim (trade-off) vardır. İki varsayımsal uç durum düşünülecek olursa, ilkinde tüm belgeler getirilerek maksimum duyarlılık sağlanırken kesinlik azalmakta, ikincisinde ise tek bir alakalı belge getirilerek maksimum kesinlik sağlanırken duyarlılık çok düşük olacaktır. Duyarlılığı artırmanın yolu genellikle daha fazla belge getirmektir ancak bunun maliyeti kesinliğin azalması olacaktır.

F-measure, hem kesinliği hem de duyarlılığı kullanarak bir sisteme tek bir puan verir ve iki ölçütün ağırlıklı harmonik ortalaması olarak tanımlanır. F-measure ölçütünün en yaygın versiyonu, kesinlik ve duyarlılığa eşit ağırlık veren F_1 ölçütüdür:

$$F_1 = \frac{2 \cdot kesinlik \cdot duyarlılık}{kesinlik + duyarlılık} \quad (2.15)$$

Kesinlik ve duyarlılık ölçütleri kullanıcıya döndürülen belge kümesindeki belgelerin sırasını göz önünde bulundurmaz ancak çoğu bilgi erişim sistemi, bir belge kümesi yerine sıralanmış bir belge listesi döndürür. Kullanıcıların sonuçları bu şekilde deneyimleyeceği ve amacın kullanıcı faydasını ölçmek olduğu düşünüldüğünde, sistem değerlendirmesinde belgelerin sıralamasının göz önünde bulundurulması gerekmektedir. Kullanıcıların belgelerin sıralamasıyla etkileşime girme biçimlerinin altında yatan varsayımlara dayalı olarak zaman içerisinde birçok farklı ölçüt önerilmiştir. Günümüzde bilgi erişimi çalışmalarında sıklıkla kullanılan ölçütlerin başında MAP, nDCG ve ERR ölçütleri gelmektedir.

2.3.1.1. Mean Average Precision (MAP)

Bir k değeri için kesinlik ($P@k$), belge listesindeki ilk k tane belge için hesaplanan kesinlik değeridir. Ortalama kesinlik (AP) ise listedeki bir sorgu için dönen tüm alakalı

belgeler için sırasıyla hesaplanan $P@k$ değerlerinin ortalamasıdır (genellikle ilk 1,000 belge kullanılarak hesaplanır). AP hesaplama formülü Denklem 2.16'da verilmiştir.

$$AP = \frac{1}{|R|} \sum_{k \in R} P@k \quad (2.16)$$

Denklemden R , alakalı (relevant) belgeleri temsil etmektedir ve hesaplanan AP, bir sorgu içindir. Test koleksiyonundaki her bir sorgu için hesaplanan AP değerlerinin ortalamasıyla MAP puanı elde edilir. MAP hesaplanırken belgelerin alakalılık dereceleri göz önünde bulundurulmaz ve iki kademeli değerlendirme üzerinden hesaplama yapılır (alakalı/alakasız).

2.3.1.2. Normalized Discounted Cumulative Gain (nDCG)

nDCG, çok kademeli değerlendirmelerde sıklıkla kullanılan bir ölçüttür. nDCG hesaplamaları bir kesme (cut-off) değeri ile yapılmaktadır. Kesme değeri k olduğunda $nDCG@k$ değeri belge listesindeki ilk k belge için hesaplanır. Hesaplama kazanım (gain) olarak alakalılık dereceleri kullanılır. Sırasıyla ilk k adet belgenin kazanımları toplamı kümülatif kazanımı ($CG@k$) vermektedir. DCG, belge sıralamasını dikkate alarak, alt sıralarda gelen belgelerin değerini düşürmek için alaka puanlarına bir indirim (discount) faktörü uygular. Buna göre, sıralama ne kadar yüksek olursa, kümülatif kazanıma eklenen belge puanının payı o kadar küçük olur (Järvelin & Kekäläinen, 2002). Denklem 2.17, bir k kesme değeri için $DCG@k$ hesaplamasını vermektedir.

$$DCG@k = \sum_i^k \frac{2^{g_i} - 1}{\log_2(1 + i)} \quad (2.17)$$

Denklemden g_i , sıralaması i olan belgeden elde edilen kazanımı (alaka derecesini) göstermektedir. nDCG ise DCG ölçütünün idealDCG puanına bölünerek elde edilen normalize edilmiş bir ölçüttür. Bir sıralı belge listesinin idealDCG puanı, listedeki belgeler alaka düzeylerine göre sıralandıklarında elde edilen DCG puanıdır.

2.3.1.3. Expected Reciprocal Rank (ERR)

Genel olarak kullanıcıların bilgi erişim sistemleriyle olan etkileşimini yansıtan, konum modeli ve kademeli model olarak adlandırılan iki ana model vardır. Konum modeli, farklı konumlardaki belgelerin birbirinden bağımsız olduklarını varsaymakta ve

incelenme olasılığını bir konum fonksiyonu olarak modellemektedir. Kademeli model ise eş zamanlı olarak tüm sonuç listesindeki belgelerin alaka düzeyini ve inceleme olasılığını modellemektedir. Bir konum modeli ölçütü olan DCG, kullanıcının daha önce gördüğü hiçbir belgeyi dikkate almadan yalnızca belgenin sırasına (konumuna) göre indirim uygular. Ancak bir kademeli model ölçütü olan ERR ise daha önce görülen belgelerin alaka düzeyine dayalı olarak indirim uygular.

$$ERR@k = \sum_{r=1}^k \frac{R_r}{r} \prod_{i=1}^{r-1} (1 - R_i) \quad (2.18)$$

Denklem 2.18, ERR ölçütünün formülünü vermektedir. Denklemde verilen R_r ve R_i sırasıyla r ve i sıralarında olan belgelerden edinilen kazanımları göstermektedir. g_i , sıralaması i olan belgeden elde edilen kazanımı (alaka derecesini) ve g_{max} alaka derecelendirmesindeki maksimum dereceyi gösterdiğinde R_i şu şekilde hesaplanır:

$$R_i = \frac{2^{g_i} - 1}{2^{g_{max}}} \quad (2.19)$$

3. İLGİLİ ÇALIŞMALAR

Bu çalışma iki ana kısımdan oluşmaktadır. İlk olarak SEO-tabanlı öznitelikler literatüre tanıtılmaktadır. Bu öznitelikler sorgu-bağımsız olarak belge kalitesini ölçmekte kullanıldığından aynı zamanda ön belge özellikleri olarak da tanımlanmaktadır. İkinci olarak tanıtılan özniteliklerin Web bilgi erişim sistemlerine olan katkısı LTR deneyleri aracılığıyla incelenmektedir. Bu sebeple ilgili çalışmalar bu iki alt başlıkta toplanmıştır.

3.1. Ön Belge Özellikleri

Ön belge özellikleri bir belgenin sahip olduğu ve belge kalitesi hakkında ipuçları barındıran özelliklerdir. Bilgi erişim sistemlerinde ise ön belge özelliği aynı zamanda bir belgenin herhangi bir sorguyla alakalı olma olasılığı olarak tanımlanmaktadır (Bendersky, Croft, & Diao, 2011). Olasılıksal model ve dil modeli gibi bilgi erişim sistemi modelleri ön belge özelliklerini belgenin ön alaka olasılığı olarak sıralama fonksiyonlarına entegre edebilmektedir (Hiemstra, 2009). Denklem 3.1’de Bayes kuralı ile bir D belgesinin T terimini içeren bir sorguyla alakalı olma olasılığı hesaplanmaktadır ve bir sorgu terimiyle alakalı belgeler bu hesaba göre sıralanmaktadır.

$$P(D|T) = \frac{P(T|D)P(D)}{P(T)} \quad (3.1)$$

Burada $P(T|D)$, kullanıcının D belgesindeki bir bilgiye ulaşmak istediğinde T terimini kullanarak bir sorgu oluşturma olasılığıdır. $P(T)$ değeri verilen sorgu terimi için tüm belgelerde aynı olacağından sıralamaya etkisi yoktur. Bu nedenle sıralamaya etkisi olan ikinci olasılık $P(D)$ yani bir belgenin herhangi bir sorgu terimi ile alakalı olma olasılığıdır.

Ön belge özelliklerinin bilgi erişim sistemlerinde (örneğin web sitelerinin ana sayfalarını bulmak için oluşturulan gezinme sorguları¹ (Kraaij, Westerveld, & Hiemstra, 2002) ve e-ticaret ürün arama sistemlerinde (Trotman & Kitchen, 2022) önemi büyüktür. Bunun sebebi ön belge özelliklerinin çoğunlukla belge kalitesiyle ilişkilendirilmesi ve web belgelerinden oluşan belge koleksiyonlarındaki belgelerin kalitelerinin değişkenlik göstermesidir (Bendersky, Croft, & Diao, 2011). Ön belge özellikleri belge sıralama görevlerinin haricinde arama motorlarının verimli çalışmasında da etkin rol

¹ Belirli bir web sayfasını bulma amacıyla oluşturulan sorgulara gezinme sorgusu (navigational query) denir.

oynamaktadır. Web'in büyüklüğü göz önüne alındığında sorgu kelimelerini içeren tüm web sayfalarını sıralamak çok zaman alacağından, tipik bir Web arama motoru dizininde belgeler genellikle ön belge özellikleri ile ölçülen kalitelerine göre sıralanmaktadır. Böylece arama motoru bir sorguyla alakalı k adet belgeyi toplama işlemini kalite sıralı dizinde çok daha erken sonlandırabilir ve yanıt süresini dramatik seviyede düşürebilir. Bu işleme kesin olmayan belge erişimi (inexact top- k document retrieval) denilmektedir (Manning, Raghavan, & Schütze, 2008). Örneğin Google arama motorunda yapılan bir aramanın sonuçlarının "yaklaşık olarak 14,000,000 sonuç bulundu" ibaresi ile gösterilmesi, kesin olmayan belge erişimi yapıldığının göstergesi olarak değerlendirilebilir. Bu bağlamda web arama motorları için web sayfalarının kalitesini ölçülebilir hale getirmek (ölçek geliştirmek) kaçınılmazdır. Buna ilave olarak bir arama motoru düşük kaliteli belgeleri daha az sıklıkla tarayarak (crawl) tarama zamanından tasarruf edebilir (Richardson, Prakash, & Brill, 2006). Literatürde tanıtılan ön belge özellikleri bağlantı analizi (link analysis) tabanlı, içerik tabanlı ve diğer olmak üzere 3 başlıkta incelenebilir.

3.1.1. Bağlantı analizi tabanlı ön belge özellikleri

Çoğu web sayfasında bağlantılar bulunur. Bu bağlantılar sayesinde gerekli durumlarda kullanıcı diğer web sayfalarına yönlendirilir. Bu özellik sayesinde bir koleksiyondaki web belgelerinden bağlantı çizgesi (link graph) oluşturulabilir. Bu çizgede web sayfaları düğümleri (node) oluştururken bağlantılar sayfalar arası kenarları (edge) oluşturur. Bağlantı analizi tabanlı ön belge özellikleri web belge kalitesini tahmin etmede belgelerin bağlantı çizgesi üzerindeki komşuluk ilişkilerini dikkate alırlar.

PageRank (Page, Brin, Motwani, & Winograd, 1999), ön belge özellikleri arasında en bilinen ve en sık kullanılanlardandır. Bağlantı analizi tabanlı bir ön belge özelliği olan PageRank, web sayfalarının kalitesini ölçmede sayfa popülerliğinden faydalanır. Web sayfaları sahip oldukları geri bağlantı (backlink) sayısı açısından büyük farklılıklar gösterir. Geri bağlantı mekanizmasını akademik makalelerdeki alıntılama sistemine benzeten PageRank algoritmasına göre yüksek geri bağlantı sayısı olan sayfalar, az sayıda olan sayfalardan daha önemlidir. Bir web sayfasının PageRank puanı hesaplanırken sayfanın geri bağlantı sayısının yanı sıra o sayfaya bağlantı veren sayfaların kalitesi de göz önünde bulundurulur. Yani bir web sayfasının kalitesi o sayfaya bağlantı veren

sayfaların niceliği ve niteliğiyle ilgilidir. Denklem 3.2 basitleştirilmiş şekliyle bir u web sayfasının PageRank puanının ($R(u)$) hesaplanmasını göstermektedir.

$$R(u) = c \sum_{v \in B_u} \frac{R(v)}{N_v} \quad (3.2)$$

Formülde B_u , N_v ve c sırasıyla u sayfasına bağlantı veren sayfalar kümesini, v sayfasında bulunan bağlantı sayısını ve normalizasyon için kullanılan bir faktör değerini ifade etmektedir.

HITS (Kleinberg, 1999) algoritmasında web sayfalarının kalitesi otorite (authority) ve merkez (hub) kavramları üzerinden açıklanır. Otorite web sayfaları bir konu üzerinde yetkin bilgi içermesi beklenen ve diğer web sayfalarından çok sayıda bağlantı alan sayfaları tanımlarken, merkez web sayfaları bir konu hakkında otorite web sayfalarına çok sayıda bağlantı içeren web sayfalarıdır. Merkezler ve otoriteler karşılıklı pekiştirici (mutually reinforcement) bir ilişki içindedirler. İyi bir merkez birçok iyi otoriteye bağlantı verirken, iyi bir otorite de birçok iyi merkez tarafından bağlantı almaktadır.

3.1.2. İçerik tabanlı ön belge özellikleri

Bir belgenin kalitesini ölçmede belgenin bağlantı özellikleri kadar içerik özellikleri de önem taşır. Bu özellikler belgenin içerik kalitesi üzerinden belge kalitesini ölçerek yüksek kaliteli belgeleri ödüllendirip düşük kaliteli belgeleri cezalandırarak bilgi erişim sisteminin başarımını artırmaktadırlar.

Zhu ve Gauch (2000), çalışmalarında bir web sayfası içeriğinin ne sıklıkla güncellendiği, web sayfasındaki çalışmayan bağlantı sayısı ve içeriğin tek bir konuya ne kadar yoğunlaştığı gibi içerik tabanlı ön belge özellikleri tanıtmışlardır. Tanıtılan özellikler ayrı ayrı ve tümü olacak şekilde statik bir bilgi erişim sistemine entegre edilerek merkezi (centralized) ve dağıtılmış (distributed) arama görevlerinde test edilmişlerdir.

Zhou ve Croft (2005), çalışmalarında belge kalitesini ölçen “information-to-noise” ve “CDD” (Collection-Document Distance veya Koleksiyon-Belge Uzaklığı) olmak üzere iki içerik tabanlı ön belge özelliği tanıtmışlardır. Çalışmada, bu tanıtılan özellikler basit bir sorgu olasılık (query likelihood) modeline entegre edilerek test edilmiştir.

Bendersky vd. (2011), çalışmalarında içerik kalitesinin içeriğin okunabilirliği, düzeni ve gezinme kolaylığı gibi çok sayıda faktöre bağlı olduğunu belirtmişlerdir. Çalışmada bu kavramlar üzerinden başlıktaki terim sayısı, entropi, görünür terim sayısı

gibi çok sayıda içerik tabanlı ön belge özelliği tanıtılmıştır. Tanıtılan bu özellikler belge kalitesini dikkate almayan ve belge kalitesi için PageRank gibi bağlantı analizi tabanlı özellikler kullanan statik bilgi erişim sistemlerine entegre edilerek test edilmiştir.

Cormack vd. (2011), çalışmalarında basit içerik tabanlı bir sınıflandırıcıyla web sayfalarının spam olma olasılığını (“spamminess”) hesaplamışlar ve her bir sayfaya bir spam puanı atamışlardır. Çalışmada bu spam puanları iki farklı şekilde bilgi erişim sisteminin başarımını artırmada kullanılmıştır. Bunlardan ilkinde bir web belge koleksiyonundaki belgeler spam puanlarına göre sıralanarak belirli bir yüzdelik dilimdeki sayfalar filtrelenmiş ve mevcut bilgi erişim sistemleri filtrelenmiş koleksiyon üzerinde çalıştırılarak başarımlarındaki farklar gözlenmiştir. İkincisinde ise bir bilgi erişim sisteminin sorgu için getirdiği sıralı belge listesine spam olma olasılığı yüksek olan belgeler elenmemiş bunun yerine daha alt sıralara alınarak yeniden sıralama yapılmıştır.

3.1.3. Diğer ön belge özellikleri

Bir web sayfasının Web’in bağlantı çizgesinden veya sayfanın içeriğinden elde edilen ön belge özellikleri haricinde de kalitesini belirtebilecek unsurları vardır. Bu unsurlardan bazıları (belge uzunluğu, URL uzunluğu ve derinliği vb.) basit gibi gözükmeyle beraber türetilen ön belge özellikleri literatürde sıklıkla kullanılmış ve etkinlikleri deneysel olarak gösterilmiştir.

Amento vd. (2000), çalışmalarında web sayfa kalitesini PageRank, sayfaya gelen bağlantı sayısı, sayfadan verilen bağlantı sayısı vb. ön belge özellikleri ile ölçerek uzman görüşleri ile karşılaştırmışlardır. Kraaij vd. (2002), çalışmalarında sayfa uzunluğu, sayfaya gelen bağlantı sayısı ve URL formu ön belge özelliklerini bir dil modeline entegre ederek ana sayfa arama (entry page search) görevinde test etmişlerdir. Craswell vd. (2005), çalışmalarında ön belge özelliklerinin statik bir bilgi erişim sistemine entegre edilmesi aşamasındaki uygun olan dönüşüm yöntemlerini (transformation) ve ağırlıklandırılmalarını (weighting) incelemişlerdir. Kamps vd. (2004) ise çalışmalarında belge uzunluğu, URL ile türetilen ve bağlantı topolojisi ile türetilen ön belge özelliklerini bir dil modeline entegre etmişlerdir. Elde edilen bilgi erişim sistemi konu damıtma (topic distillation), ana sayfa bulma (home page finding), adlandırılmış sayfa bulma (named page finding) ve karışık sorgu (mixed query) alt görevlerinde test edilmiştir.

3.2. LTR Deneylerinde Öznitelik Etkinliği

LTR yaklaşımları Web Bilgi Erişim çalışmalarında yaygın olarak kullanılmaktadır. Bir LTR yaklaşımı, bir sorgu için statik bir bilgi erişim sistemi tarafından getirilen belgeleri makine öğrenme teknikleri uygulayarak yeniden sıralar (Liu, 2011). LTR yaklaşımları, bir model öğrenmek için sorgu-bağımlı ve sorgu-bağımsız özniteliklerden yararlanır. Bu bağlamda, sorgu-belge çiftleri için mevcut alaka göstergeleri (terim frekansı vb.) ve mevcut ön belge özellikleri sırasıyla sorgu-bağımlı ve sorgu-bağımsız öznitelikler olarak LETOR ve MSLR (Qin & Liu, 2013) gibi kıyaslama veri setlerine (benchmark dataset) ve LTR çalışmalarına dahil edilmişlerdir.

Macdonald vd. (2013), çalışmalarında bir web sayfasının farklı alanları için elde edilen terim ağırlıklandırma modeli puanlarını LTR deneylerinde sorgu-bağımlı öznitelik değerleri olarak kullanarak bilgi erişim sistemi başarımına katkılarını gözlemlemişlerdir. Çalışmanın sonuçları aynı zamanda sorgu-bağımsız özniteliklerin LTR çalışmalarındaki etkisini de göstermektedir.

Literatürde, LTR deneylerinde kullanılmak üzere yeni öznitelik tanıtan çalışmalarda, deney tasarımı benzer şekilde olmaktadır. Genellikle bir referans öznitelik kümesini ile elde edilen çalışmayla referans öznitelik kümesine çalışmada tanıtılan özniteliklerin eklenmesiyle elde edilen çalışma karşılaştırılmaktadır. Bu bağlamda, Fan vd. (2017) ve van den Akker vd. (2019), çalışmalarında web sayfalarının görsel özelliklerinden sorgu-bağımsız öznitelikler türetmiş ve bu öznitelikleri LTR deneylerinde test etmişlerdir. Web sayfası görüntülerinin (snapshot) sinir ağlarında (neural networks) işlenmesiyle elde edilen görsel öznitelikler referans öznitelik kümesine eklenerek test edilmiştir. Benzer şekilde Ensan vd. (2017), çalışmalarında sinirsel gömme (neural embedding) öznitelikleri tanıtmışlardır. Sorgu ve belge arasındaki ilişkiyi kelime gömme (word embedding) ve belge gömme (document embedding) yöntemleriyle farklı gruplar halinde oluşturulan öznitelikler, referans öznitelik kümesine eklenerek LTR deneylerinde etkinlikleri gözlemlenmiştir. Aynı deney tasarımını kullanan Macdonald vd. (2012), çalışmalarında sorgu öznitelikleri adını verdikleri öznitelik grubunu tanıtarak LTR deneylerinde etkinliklerini test etmişlerdir. Bu çalışmada da tanıtılan öznitelikler sınıflara ayrılmış ve özniteliklerin sınıf bazında veya tekil olarak başarıma katkısı analiz edilmiştir.

Bu çalışmada Bölüm 3.1’de belirtildiği gibi ön belge özellikleri tanıtılmakta ve bu özellikler etkinliklerinin testi için ise Bölüm 3.2’de belirtildiği gibi LTR deneylerinde öznitelik olarak kullanılmaktadır. Literatürdeki çalışmalarda Bölüm 3.2’de belirtilen

referans öznitelik kümesi oluşturmak için iki farklı yol izlenmektedir: Kıyaslama veri setlerindeki öznitelik değerlerini kullanma veya kapsamlı bir öznitelik kümesi oluşturma. Bu çalışmada kullanılan web belge koleksiyonları için kıyaslama veri setlerinde öznitelik değerleri türetilmediği için literatürde kullanılan ikinci yol tercih edilmiştir.



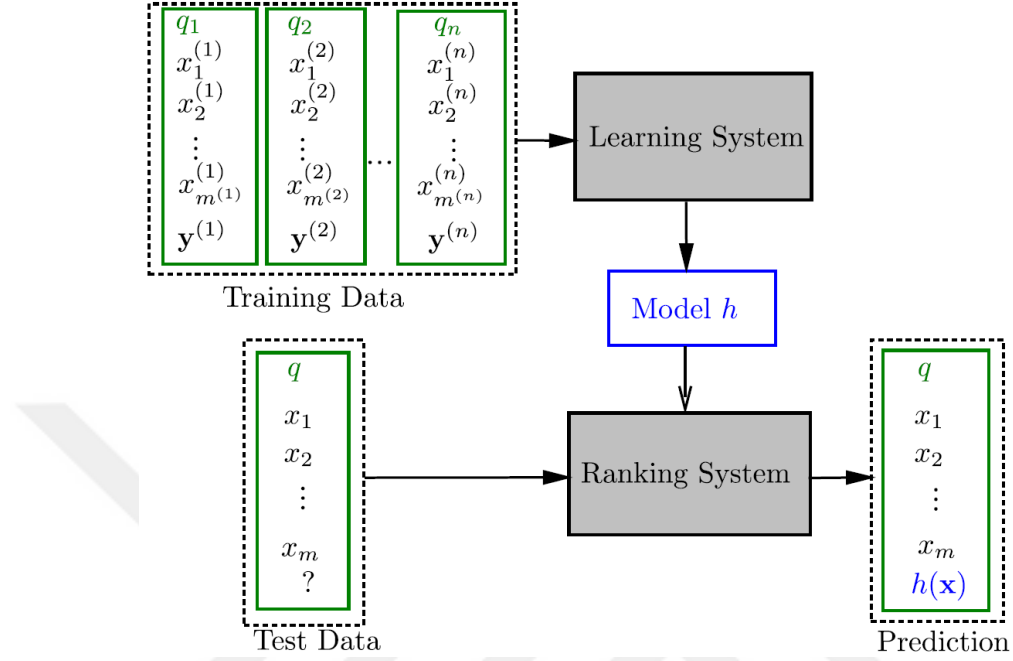
4. SIRALAMAYI ÖĞRENME

Sıralamayı öğrenme, bilgi erişimi alanında belgeleri alaka düzeylerine göre sıralamak için otomatik olarak bir sıralama modeli oluşturma görevidir. Bir bilgi erişim sistemi eşleşme aşamasında bir sorguyla alakalı olduğunu düşündüğü belgeleri alaka sırasına göre sıralar. Sıralamayı öğrenme yöntemleri bu sıralı belge listesini sistem etkinliğini artırmak amacıyla yeniden sıralamaktadır. Son zamanlarda web teknolojilerinin yaygınlaşmasıyla birlikte potansiyel eğitim verisi miktarı göz önüne alındığında, etkili sıralama modelleri oluşturmak için Makine Öğrenmesi (Machine Learning) teknolojilerinden yararlanmak mümkün hale gelmiştir. Belgeleri sıralamak amacıyla ayırt edici öğrenme (discriminative learning) yoluyla önceden tanımlanmış özniteliklerin nasıl birleştirileceğini öğrenen yöntemlere sıralamayı öğrenme yöntemleri denir (Liu, 2009).

Sıralamayı öğrenme yöntemlerinde bir bilgi erişim sistemi tarafından bir sorgu için getirilen sıralı belge listesindeki incelenmekte olan tüm belgeler, belgelerin sorguyla alaka düzeyini yansıtan öznitelik vektörleri ile temsil edilir. Bir q sorgusu ve bu sorgu için sistem tarafından getirilen bir d belgesi ikilisi $\chi = \varphi(d, q)$ şeklinde temsil edilir. Burada φ , öznitelik çıkarma fonksiyonudur. Sıralamayı öğrenme yöntemlerinde sıklıkla kullanılan öznitelikler arasında sorgudaki terim sayısı, sorgu-belge ikilisinin terim ağırlıklandırma modellerinden (BM25, LGD, PL2, vb.) üretilen puanları ve belge uzunluğu gibi öznitelikler yer almaktadır. Sıralamayı öğrenme yöntemleri, çok sayıda özniteliği birleştirebilme yeteneği sayesinde özellikle kullanıcı bilgi ihtiyaçlarının ve belge özelliklerinin çok değişkenlik gösterdiği web tabanlı bilgi erişim sistemlerinde oldukça faydalı olmaktadır.

Şekil 4.1, sıralamayı öğrenme çatısını göstermektedir. Burada eğitim verisi (training data) n adet eğitim sorgusu, bu sorgular için bilgi erişim sistemi tarafından getirilmiş belgeler ve sorgu-belge ikilisi için temel doğruluk etiketlerinden (ground truth label) oluşmaktadır. Her bir $q_i (i = 1, \dots, n)$ sorgusu için öznitelik vektörü olarak temsil edilen belgeler $x^{(i)} = \{x_j^{(i)}\} (j = 1, \dots, m^{(i)})$ olarak ve sorgu-belge ikilisi temel doğruluk etiketleri $y^{(i)} = \{y_j^{(i)}\} (j = 1, \dots, m^{(i)})$ olarak ifade edilir. Daha sonra bu eğitim verisi üzerinde bir sıralama modeli (h) öğrenilir. Bu model eğitim verilerini kayıp fonksiyonunu (loss function) en iyi şekilde optimize etmeye çalışarak öğrenilmektedir. Test aşamasında,

yeni bir sorgu geldiğinde, eğitim aşamasında öğrenilen model uygulanarak belgeler sorgu ile alakalarına göre sıralanır ve sistem ilgili sıralı listeyi geri döndürür.



Şekil 4.1. Sıralamayı öğrenme çatısı

Ayırt edici öğrenme dört bileşenden oluşur: girdi uzayı, çıktı uzayı, hipotez uzayı ve kayıp fonksiyonu. Girdi uzayı genellikle öznitelik vektörleri olarak temsil edilen incelenmekte olan nesneleri, çıktı uzayı, girdi uzayına bağlı olarak elde edilen öğrenme hedeflerini ifade eder. Hipotez uzayı, girdi nesnelerinin öznitelik vektörleri üzerinde çalışan çıktı uzayının formatına göre tahminler yapan fonksiyonları ifade eder. Kayıp fonksiyonu (loss function), hipotez tarafından üretilen tahminin temel doğruluk etiketine ne derece uygun olduğunu ölçer. Kayıp fonksiyonu ile, eğitim setinde deneysel bir risk tanımlanabilir ve optimal hipotez genellikle deneysel risk minimizasyonu yoluyla öğrenilir. Bu bileşenler baz alınarak sıralamayı öğrenme algoritmaları üç farklı yaklaşıma ayrılmıştır.

4.1. Noktasal Yaklaşım (Pointwise Approach)

Noktasal yaklaşımda girdi uzayı her belge için oluşturulan öznitelik vektörlerinden, çıktı uzayı ise belgelerin alaka yargılarından oluşmaktadır. Hipotez uzayı, bir belgenin öznitelik vektörünü girdi olarak alan ve alaka derecesini tahmin eden fonksiyonları içerir. Bu fonksiyonlar genellikle puan atama fonksiyonları olarak adlandırılır ve bu

fonksiyonlara bağılı olarak belgeler tekrar sıralanır. Kayıp fonksiyonu, her belge için alaka derecelerinin doğru tahminini inceler.

Noktasal yaklaşımda kayıp fonksiyonu bir belgenin nihai sıralama listesindeki konumunu dikkate almaz ve böylelikle bir sorgu için getirilen belgeler arası ilişkiyi göz önünde bulundurulmaz. Farklı noktasal sıralama algoritmalarında sıralama; regresyon, sınıflandırma ve sıralı regresyon olarak modellenir. Regresyon tabanlı algoritmalarda, sıralama problemi, çıktı uzayı (alaka dereceleri) gerçek sayılar olarak kabul edilerek bir regresyon problemine indirgenir. Sıralama tabanlı algoritmalarda da regresyon tabanlı algoritmalar benzer şekilde sıralama problemi bir sınıflandırma problemine indirgenmektedir. Ancak sınıflandırma tabanlı algoritmalar, regresyon tabanlı algoritmalarından farklı olarak alaka yargılarını nicel bir değer olarak görmemektedir. Sıralı regresyon ise model sıralamayı öğrenme modeli oluştururken belgelerin alaka yargıları arasındaki sıralı ilişkiyi göz önünde bulundurmaktadır.

Noktasal yaklaşımın bazı problemleri mevcuttur. Noktasal yaklaşımda girdi nesnesi tek bir belge olduğu için belgelerin kendi aralarındaki sıra ilişkisi veya listedeki konumu öğrenme sürecinde dikkate alınmaz. Ayrıca top- k getirim (top- k retrieval) yapılmayan sistemlerde her sorgu için farklı sayıda belge getirilmekte ve bu yaklaşımda girdi uzayında sorgular dikkate alınmadığı için genel kayıp fonksiyonuna çok sayıda belge içeren sorgular hâkim olabilmektedir. Bu sebeplerle noktasal yaklaşım algoritmaları sorgu düzeyinde ve konum tabanlı olan bilgi erişimi değerlendirme ölçütleri tarafından iyi değerlendirilemez.

4.2. İkili Yaklaşım (Pairwise Approach)

İkili yaklaşımda girdi uzayı öznitelik vektörleriyle temsil edilen belge çiftleridir. Çıktı uzayı ise her belge çifti için $\{1, -1\}$ değerlerini alabilen çift tercihidir. Bu değer alaka yargıları (l_u, l_v) olan (x_u, x_v) belge çifti için $y_{u,v} = 2 \cdot I_{\{l_u > l_v\}} - 1$ olarak hesaplanır. Burada $I_{\{l_u > l_v\}}$, $l_u > l_v$ ifadesinin doğru olduğu durumda 1, tersi durumda 0 değerini alır. Hipotez uzayı, girdi olarak bir çift belge alan ve çıktı olarak belgelerin birbirlerine göre sırasını veren iki değer alabilen h fonksiyonlarını içerir. Çoğu ikili yaklaşım algoritmasında hipotez fonksiyonu, bir f puan fonksiyonu için $h(x_u, x_v) = 2 \cdot I_{\{f(x_u) > f(x_v)\}} - 1$ olarak hesaplanır. Kayıp fonksiyonu $h(x_u, x_v)$ ile çıktı uzayındaki $y_{u,v}$ arasındaki tutarsızlığı ölçer.

İkili yaklaşımın amacı, noktasal yaklaşım gibi bir belgenin alaka derecesinden ziyade iki belgenin birbirleriyle olan sırasını tahmin etmeye çalışır. İkili yaklaşımda, sıralama problemi belge çiftleri üzerinde bir sınıflandırma problemine indirgenir. Sınıflandırmanın amacı, ilk belgesi ikinci belgeden daha alakalı olan çiftler hakkında olumlu ve diğer çiftler hakkında olumsuz tahminlerde bulunarak yanlış sınıflandırılmış belge çiftlerinin sayısını en aza indirmektir. İkili yaklaşım da noktasal yaklaşım gibi her sorgu için farklı sayıda belge getirildiği durumlarda genel kayıp fonksiyonuna çok sayıda belge içeren sorguların hâkim olması problemini barındırır. Bu probleme bir çözüm olarak ikili kayıp fonksiyonuna sorgu düzeyinde normalizasyon uygulanabilir. Yani, bir sorgu için ikili kayıp, o sorguyla ilişkili toplam belge çifti sayısı tarafından normalize edilmektedir (Cao, ve diğerleri, 2006; Qin, ve diğerleri, 2008).

4.3. Listesel Yaklaşım (Listwise Approach)

Listesel yaklaşım iki alt kategoriye ayrılmaktadır. Bunlardan ilkinde çıktı uzayı bir sorguyla ilişkilendirilen tüm belgelerin alaka dereceleridir (y) ve kayıp fonksiyonu ise sıklıkla kullanılan bilgi erişimi değerlendirme ölçütlerinin yakınlığına veya sınırlarına göre tanımlanır. İkinci alt kategoride, çıktı uzayı aynı sorguyla ilişkili belgelerin permütasyonunu (π_y) içerir ve kayıp fonksiyonu, hipotez tarafından verilen permütasyon ile alaka yargıları permütasyonu arasındaki farkı ölçer.

Listesel yaklaşım, girdi olarak bir sorguyla ilişkili tüm belgeleri ve çıktı olarak bu belgelerin sıralı listelerini alır. Bu sayede belgelerin hangi sorguyla ilişkilendirildiğini ayırt edebilme ve öğrenme sürecinde belgelerin sıralı listedeki konum bilgilerini de kullanma potansiyeline sahiptir. Bu özellikleriyle sorgu düzeyinde ve konum tabanlı olan bilgi erişimi değerlendirme ölçütleri tarafından değerlendirilebilmektedir.

5. SEO-TABANLI ÖZNİTELİKLER

Sıralamayı öğrenme algoritmaları bir sorguyla ilişkili belgeleri sıralarken özniteliklerden faydalanırlar. Bu öznitelikler arasında sadece belgeyi niteleyen öznitelikler de mevcuttur. Bunlar genelde belgenin kalitesi hakkında fikir veren faktörlerden türetilmiştir. Ortaya çıkış amacı ve genel kullanım alanı düşünüldüğünde SEO teknikleri de benzer faktörlerden faydalanmaktadır. Bu amaçla bu çalışma, belirli SEO tekniklerinden türetilen öznitelikler ile belge kalitesini ölçme ve bu sayede bilgi erişim sistemi etkinliğini artırmayı hedeflemektedir.

5.1. Genel Web Sayfası Yapısı

Bir web sayfası HTML kodlarından oluşmaktadır. Basit bir HTML belgesi yapısı Şekil 5.1’de verilmiştir. 1 numaralı kısım belgeyi HTML belgesi olarak tanımlayan bir belge türü bildirimidir ve web tarayıcılarının belgeyi nasıl yorumlaması gerektiğini belirtir. 2 numaralı kısımda html etiketi bulunmaktadır. Belgedeki bütün elemanlar html etiketleri içerisinde yer almaktadır. HTML elemanı içerisinde belge iki kısma ayrılmaktadır: baş (head) ve gövde (body). 3 numaralı kısımda gösterilen baş elemanı içerisinde sayfa başlığı (title), sayfanın kullandığı stil sayfaları (stylesheet), komutlar (script) ve meta bilgiler gibi belgenin kendisi hakkında açıklayıcı bilgiler bulunmaktadır. 4 numara bir meta bilgi örneğini, 5 numara ise sayfanın başlığı bilgisinin bulunduğu title elemanını göstermektedir. 6 numaralı kısımda sayfanın içeriğini barındıran gövde elemanı bulunmaktadır (Robbins, 2012). Ayrıca her web sayfasının eşsiz bir URL adresi vardır. Web tarayıcıları bu adres sayesinde belirtilen web sayfası içeriğini alarak kullanıcıya gösterebilmektedir.



Şekil 5.1. Basit bir HTML belgesi yapısı

5.2. Arama Motorları ve SEO

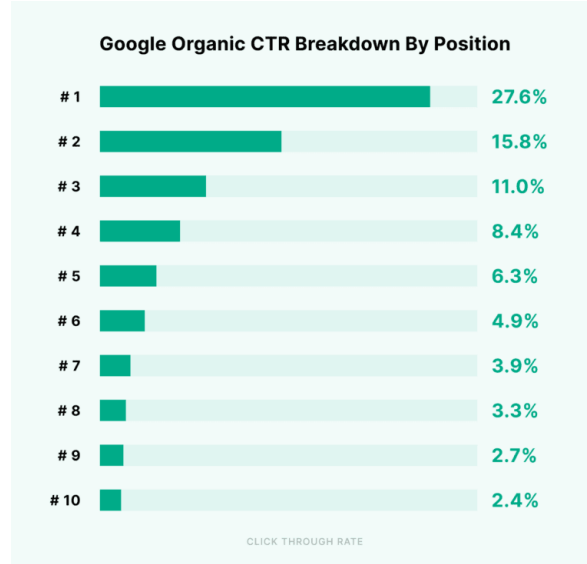
Arama motorları kullanıcıların bilgi ihtiyaçlarını giderme adına kullanıcıdan bir sorgu alarak, dizinlemiş oldukları web belgeleri arasından sıralı bir belge listesini kullanıcıya döndüren yazılımlardır. Arka planda, bir arama motoru, web sayfaları hakkında bilgi bulmak ve toplamak için algoritmalar kullanır. Toplanan bilgiler genellikle, web sayfası içeriği hakkında bilgi veren anahtar kelimeler veya içerik parçaları, sayfanın URL adresi, sayfayı oluşturan kod, sayfaya gelmekte olan ve sayfadan gitmekte olan bağlantılardır. Bu bilgiler dizine eklenerek bir veri tabanında saklanır. Ön planda ise, kullanıcıların belirli bilgileri bulmak için sorgu girdiği ve girilen sorguyla eşleşen web sayfalarına olan bağlantıların sıralanarak listelendiği bir kullanıcı ara yüzü vardır.

İnternet kullanımının geniş kullanıcı kitlelerine yayılmasıyla internetteki web sayfası sayısı yıllar boyunca katlanarak artmaktadır. Web sayfası sayılarındaki artışla birlikte web sayfalarının kaliteleri arasındaki değişkenlik de oldukça artmıştır. Bu durum arama motorlarının kullanıcıdan alınan sorguyla ilişkili web sayfalarını sıralarken sorgu-web sayfası ilişkisine ek olarak web sayfası kalitesini de dikkate alma gerekliliğini getirmiştir. Bu konuyla ilgili en çok tanınan ve ilk çalışmalardan olan PageRank algoritması, akademik çalışmalardaki alıntılama (citation) metodundan ilham alarak ve Web'in geniş bağlantı yapısını kullanarak web sayfalarının kalitesini ölçmektedir. Bir web sayfasının PageRank puanı o sayfaya gelen bağlantıların sayısı ile doğru orantılıdır ve bağlantı kaynağı web sayfalarının PageRank puanları ile ağırlıklandırılır (Page, Brin, Motwani, & Winograd, 1999). En çok kullanılan ticari arama motoru olan Google, PageRank algoritmasını temel almaktadır. Google da dahil olmak üzere ticari arama motorları genellikle birçok farklı faktörleri de dikkate alarak web sayfalarının kalitesini ölçmektedir. Bu ölçümlere göre web sayfaları Arama Motoru Sonuç Sayfası'nda (Search Engine Results Page veya SERP) sıralanmaktadır.

SEO, bir web sayfasının arama motorunun SERP listesinde sıralamada üst sıralara çıkmak amacıyla web sayfasını optimize etme işlemlerine denmektedir. Arama motoru kullanıcıları genellikle bir bilgi ihtiyacını gidermek için SERP listesindeki birinci veya ikinci sayfadaki web sayfalarını (ilk 10-20 web sayfası) ziyaret ederler ve çok az sayıda kullanıcı daha alt sıralardaki web sayfalarını ziyaret eder. Şekil 5.2, SEO çalışmaları yapan bir web sitesinin 4 milyon arama sonucu ile yapmış oldukları bir çalışmanın bir parçasını göstermektedir (http-2). Şekilde Google arama motorunun SERP listesinin ilk

sayfasındaki web sayfalarının tıklanma oranları (Click-Through Rate veya CTR) bilgileri gösterilmektedir. Buna göre SERP listeleri üzerinden ziyaret edilmelerin yarısından fazlası listelerdeki sıralaması ilk 3 olan, çok büyük bir kısmı ise sıralaması ilk 10 olan web sayfalarına olmaktadır. Bu sebeple bir web sayfası yöneticisi (webmaster) için web sayfasını arama sonuçlarında üst sıralara getirmek önemlidir. SEO süreci, hem web sayfalarını üst sıralara taşıyarak keşfedilme olasılığını artırmak hem de kullanıcı deneyimini iyileştirmek için belirli ilkelerin kullanılmasını gerektirir (Khan & Mahmood, 2018).

Ticari arama motorları şirketlerinin sıralama algoritmaları, onların en değerli fikri mülkiyet parçalarıdır ve bu nedenle kamu incelemesinden uzak tutulur. Bununla birlikte, bazı arama motoru şirketleri, web sayfalarını okuma çabalarını kolaylaştıracak “beyaz şapka (white hat)” SEO uygulamalarını teşvik etmek ve sıralama algoritmalarını manipüle etmeye çalışan “kara şapka (black hat)” SEO taktiklerine başvurmadan caydırmak için kısmen kendi çıkarlarına yönelik bir girişimde, web sayfası yöneticilerine SEO tavsiyeleri yayınlamaktadır (Killoran, 2013). Yayımlanan bu tavsiyelerin yanı sıra SEO uygulayıcıları tersine mühendislik (reverse engineering) uygulamaları ile arama motorlarının sıralama algoritmalarına etki eden faktörleri bulmaya çalışmaktadır.



Şekil 5.2. Google arama motoru SERP listesi ilk sayfadaki web sayfalarının tıklanma oranları

SEO teknikleri Sayfa-içi (On-page) ve Sayfa-dışı (Off-page) SEO olmak üzere iki süreçten oluşur. Sayfa-içi SEO, web sayfası yöneticilerinin ilk elden etkisi olan SEO faktörlerinin birleşimidir. Bunlar web sayfasından ulaşılabilen SEO özellikleridir. Sayfa-

içi SEO faktörleri, oluşturulan içeriği, HTML yapısını ve web sitesinin mimarisini kapsar. Sayfa-dışı SEO, Web'in çizge yapısı (graph structure) göz önünde bulundurularak yapılan çalışmalardır. Bağlantı kurma (link building) çalışmaları, Sayfa-dışı SEO çalışmalarına bir örnektir. Bir kullanıcı ziyaret ettiği bir web sayfasındaki (blog, forum, sosyal medya vb. sayfalar da dahil olmak üzere) bağlantıları kullanarak başka web sayfalarını ziyaret edebilir. PageRank algoritmasında olduğu gibi birçok arama motoru da web sayfalarını sıralarken başka web sayfalarından gelen bağlantıları dikkate almaktadır. Bu amaçla bir web sayfası yöneticisinin web sayfası görünürlüğünü başka web sayfaları üzerinden artırma çalışmalarına bağlantı kurma veya geri bağlantı oluşturma (backlinks generation) denir. Sayfa-dışı SEO teknikleri genellikle belirsiz ve ölçümü zor olduğundan bu çalışmada Sayfa-içi SEO teknikleri kullanılmıştır.

5.3. SEO Teknikleri ile Türetilen Öznitelikler

SEO çalışmaları, ticari arama motorlarının bir sorguya karşılık getirilen web sayfalarının sıralamasına etki eden faktörler üzerine yapılmaktadır. Arama motorları web sayfalarını sıralarken sorgu ile olan alaka düzeyinin yanında web sayfası kalitesini de dikkate aldıklarından bu faktörler arasında kalite ölçümüne yardımcı faktörler de bulunmaktadır. Dolayısıyla bu çalışmada web sayfası kalitesine etki edebilecek faktörlerin ölçümü ile elde edilen sorgu-bağımsız özniteliklerin sıralamayı öğrenme algoritmalarında kullanılarak bir bilgi erişim sisteminin etkinliğini artırmak hedeflenmektedir. Bu amaçla temel Sayfa-içi SEO teknikleri kullanılarak Tablo 5.1'deki öznitelikler türetilmiştir.

Tablo 5.1. SEO teknikleriyle türetilen öznitelikler

Öznitelik Tipi	Öznitelik Adı	#
Meta etiket	Number of Meta Tags, Copyright, Viewport, Robots, Description, Keywords, Keyword in Domain, Keyword in First 100 words, Keyword in Image alt text, Keyword in Title, Keyword index in Title	11
İçerik	Number of words in content, Content Length over 1800 words, Headings to Content	3
Bağlantı ve resim	Contact, Favicon, Social media share, Number of images, Alt texted to all images, (Internal-External) to all links, No-follow to all links	7
URL	Https	1
Benzerlik	SimDescHeading, SimDescContent, SimDescKeywords, SimDescTitle, SimTitleContent, SimTitleHeading, SimContentHeading, SimContentKeywords, SimKeywordsHeading	9

5.3.1. Meta etiket öznitelikleri

Meta etiketler, arama motorlarına web sayfası hakkında bilgi sağlayan ve büyük bir çoğunluğu kullanıcıya görünmez olan etiketlerdir. Ayrıca arama motorları SERP listelerinde web sayfaları hakkında kısa bilgiler oluştururken meta etiketlerde bulunan bilgileri de kullanmaktadırlar. Bu nedenle meta etiketler kullanılarak içeriğin en önemli unsurlarının vurgulanarak web sayfasının optimize edilmesi, web sayfasını arama sonuçlarında öne çıkarmanın yanı sıra web sayfasını ziyaret etmeden önce kullanıcı deneyimini de etkilemektedir. Bazı meta etiket türleri sayfa yapısıyla ilgilidir ve web sayfasında gezinmenin kolay olmasını sağlamaktadır. Diğerleri ise arama motorlarına web sayfasının önemli olan bölümlerini veya sayfanın arama motoru tarafından dizine eklenip eklenmeyeceğini belirtir. Örneğin “Description” ve “Keywords” meta etiketleri sayfa içeriği hakkında bilgiler barındırmakta ve belgenin bir sorguyla eşleşmesi aşamasında da yardımcı olabilmektedir (Arslan, 2020) ancak bu çalışmada belge kalitesini ölçmek adına meta etiketlerin özellikleri herhangi bir sorgudan bağımsız olarak incelenmektedir. Meta etiketlerden türetilen öznitelikler aşağıda tanıtılmıştır. Bunlardan “Number of Meta Tags” ve “Keyword index in Title” öznitelikleri haricinde olanlar web sayfasının HTML kodunda ilgili meta etiketin olup olmamasına göre 1 veya 0 değerini alan özniteliklerdir.

- *Copyright*: Bu meta etiketin kullandığı alan telif hakkı yasaları, fikri mülkiyet, lisanslama, patentler vb. ile ilgili herhangi bir konuyu belirtmek için belirlenmiş alandır. Bu etiketin kullanılması, kadar sayfanın güvenilirliği hakkında ipucu vermektedir.
- *Viewport*: Görünüm alanı (viewport), bir web sayfasının kullanıcı cihazı tarafından görülebildiği alandır. Bu meta etiket kullanılarak farklı çözünürlükteki cihazlarda bileşenler dinamik olarak düzenlenerek düzgün web sayfası görünümü elde edilmiş olur. Bu etiket, duyarlı (responsive) ve mobil uyumlu web sayfa tasarımlarında önemli bir rol üstlenerek kullanıcı deneyimini olumlu şekilde etkilemektedir.
- *Robots*: Bu meta etiket, içerisindeki “index” ve “noindex” veya “follow” ve “nofollow” içerikleriyle arama motorlarına web sayfasının dizine eklenip eklenmeyeceğini veya web sayfasındaki bağlantıların takip edilip edilmeyeceğini belirtmektedir.

- *Description*: Bu meta etiketin içeriği web sayfasının konusunu içermektedir. Arama motorlarının SERP listelerinde sayfa içeriği tanıtılırken bu etiket içerisindeki bilgilerden de faydalanılmaktadır. Bu açıdan bakıldığında bu etiket, arama motorları aracılığıyla web sayfasını ziyaret edecek kullanıcıların sayfayı seçmelerinde önemli bir etkiye sahiptir.
- *Keywords*: Anahtar kelimeler (keywords) meta etiketi, içeriğinde web sayfası içeriğini tarif eden anahtar kelimeleri barındırır. Arama motorları sorgularla web sayfalarını eşleştirirken bu anahtar kelimelerden faydalanır. Son yıllarda “anahtar kelime doldurma (keyword stuffing)” adı verilen kara şapka uygulamalardan dolayı bazı arama motorlarının eşleştirme aşamasında anahtar kelimelere eskisi kadar ağırlık vermediği düşünülse de hala sıralamayı etkileyen faktörler arasında yer almaktadır.
- *Keyword in Domain, Keyword in First 100 words, Keyword in Image alt text, Keyword in Title, Keyword index in Title*: Arama motoruna girilen bir sorguyla bir web sayfasında anahtar kelime meta etiketinde bulunan bir anahtar kelimenin eşleşmesi, sorgu ve web sayfası arasındaki alaka göstergelerinden biridir. Bu sebeple anahtar kelimeler meta etiketine, başta arama motorlarında çokça kullanılan sorgu terimleri olmak üzere web sayfasıyla alakalı olmayan anahtar kelimeler eklenerek arama motorlarını manipüle etmeye çalışan kara şapka uygulamalar mevcuttur. Bu durumu denetlemek için meta etikette bulunan anahtar kelimelerden birinin web sayfasının önemli alanlarından bir kısmında bulunup bulunmadığını gösteren öznitelikler türetilmiştir. Bu alanlar arasında web sayfasının alan adı (domain), içeriğin ilk 100 kelimesi (100 kelimedenden az olan içerikler için içeriğin tamamı incelenmektedir), resimleri tarif eden “alt text” alanındaki metin ve sayfa başlığı (title) bulunmaktadır. Ayrıca sayfa başlığında anahtar kelimenin bulunduğu konum da önemli olduğundan, başlıkta geçen ilk anahtar kelimenin konumu da bir öznitelik olarak tanımlanmıştır.
- *Number of meta tags*: Meta etiket tipinde türetilen özniteliklerin dikkate aldığı etiketler haricinde kullanılan birçok meta etiket bulunmakta ve bu etiketler de arama motorlarına web sayfası hakkında bilgiler sunmaktadır. Bu sebeple web sayfasındaki meta etiket sayısı da bir öznitelik olarak tanımlanmıştır.

5.3.2. İçerik öznitelikleri

Web sayfası içeriği, arama motorları tarafından sorgu-web sayfası alaka ilişkilendirmelerinde en çok dikkate alınan kısımdır. Arama motorları tarafından yayınlanan SEO uygulamaları tavsiyelerinde sayfa içeriği çokça vurgulanmaktadır. Literatürde de içerik kalitesini ölçmekle ilgili çalışmalar mevcuttur (Zhou & Croft, 2005; Bendersky, Croft, & Diao, 2011). Bu çalışmalara ek olarak bu çalışmada SEO uygulamaları da göz önünde bulundurularak türetilen öznitelikler aşağıda tanıtılmıştır. Bu çalışmada web sayfası içeriği sayfa başlığı ve sayfa gövdesi metnlerinin birleştirilmesiyle elde edilmiştir.

- *Number of words in content*: Belge uzunluğu bilgi erişimi çalışmalarında kullanılan bir özelliktir. Genellikle bir belgenin hafızada kapladığı alan olarak ölçümlenir. SEO uygulamalarında ise içerikteki kelime sayısı dikkate alınmaktadır. Bu sebeple bu öznitelik de içerik kelime sayısı üzerinden hesaplanmaktadır.
- *Content Length over 1800 words*: SEO çalışmalarında SERP listelerinde üst sıralarda yer alan web sayfalarının (özellikle blog sayfaları) içerik kelime sayılarına bakıldığında ortalama olarak yaklaşık 1800 kelimeden oluştukları gözlemlenmiştir. Bu bilgiden yola çıkarak türetilen bu öznitelik, içerik kelime sayısının 1800 kelimeyi geçip geçmemesi durumuna göre 1 veya 0 değerlerini almaktadır.
- *Headings to Content*: Başlıklar (headings) bir bölümün başında bulunan ve bölüm hakkında kısa ve açıklayıcı bilgi veren ifadelerdir. Bir web sayfasında başlıkların kullanımı sayfaya genel bakışı kolaylaştırır ve sayfa içeriğinde hiyerarşi sağlar. Ayrıca başlıklar, ziyaretçilerin web sayfasını hızlı bir şekilde taramasına ve arama motorlarının sayfanın yapısını ve içeriğini hızlı bir şekilde anlamalarına yardımcı olurlar. Başlık kullanımının içerik uzunluğuyla orantılı olması gerekmektedir. Bu sebeple türetilen bu öznitelikte web sayfasındaki toplam başlık kelime sayısının içerik kelime sayısına oranı hesaplanmaktadır.

5.3.3. Bağlantı ve resim öznitelikleri

Web sayfaları, içeriğe veya web sayfasının kendisine ek bilgi sağlamak için web sayfasındaki resimlerden ve bağlantılardan yararlanır. Resimler içeriğin anlatımına görsel olarak katkıda bulunurlar. Bir web sayfasındaki bağlantılar dahili (internal) ve harici (external) olmak üzere ikiye ayrılır. Dahili bağlantılar, aynı web sitesindeki iki sayfa arasında bulunan bağlantılardır. Harici bağlantılar ise farklı bir web sitesine yönlendirme yapmaktadır. Bir web sayfasındaki resim ve bağlantı özelliklerinden türetilen öznitelikler aşağıda tanıtılmıştır.

- *Contact*: Bir web sayfasındaki iletişim bilgileri, kullanıcılar için bir iletişim kanalının varlığını ifade eder. Bu kanalın varlığı web sayfasının güvenilirliğini olumlu etkiler. Bu öznitelik iletişim bilgisinin varlığını sayfa içerisinde bir e-posta bağlantısı olup olmamasına göre 1 veya 0 değeri olarak belirtmektedir.
- *Favicon*: URL veya başlık ile görünen küçük boyutlu bir resim olarak “favicon”, bir web sayfasının logosu olarak düşünülebilir. Bir web sayfasının tanınabilirlik ve akılda kalıcılık sağlayarak kullanım kolaylığını ve güvenilirliğini gösterir. Web sayfasının favicon içerip içermemesine göre bu öznitelik 1 veya 0 değeri almaktadır.
- *Social Media Share*: Sosyal medya, insanların sosyal ağ sitelerini geliştirirken ve sürdürürken kullandıkları teknolojileri veya uygulamaları temsil eder (Facebook, Instagram, Twitter, vb.). Kullanıcılar bu uygulamalar üzerinden multimedya metin, resim, ses, video gibi türlerde paylaşım yapabilmektedir (Fuchs, 2014). Sosyal medya paylaşım özelliği adından da anlaşılacağı üzere bir web sayfası içeriğinin sayfa üzerinden sosyal medya platformlarında paylaşılabilme özelliğidir. Bu sayede web sayfasının tanınırlığı artabilmektedir. Web sayfasının bu özelliği içerip içermemesine göre öznitelik 1 veya 0 değerlerini almaktadır.
- *Number of images*: İçeriğin görsellerle desteklenmesi, anlatımın etkinliğini ve web sayfasının kullanım kolaylığını artırmaktadır. Adından da anlaşılacağı üzere bu özniteliğin değeri web sayfasındaki görsel sayısıdır.
- *Alt texted to all images*: Web sayfasında bulunan bir görsel arama motorları tarafından anlamlandırılmaz. Bunun için görsellerde görselin tarif

edilebilmesi için bir alt text alanı bulunur. Bu alandaki içerikler aynı zamanda görme engelli bireylerin kullandıkları özel okuyucu programları tarafından okunabilmekte ve sayfanın erişilebilirliğine katkı sunmaktadır. Bu öznitelik bir web sayfasında alt text içeren görsel sayısının sayfadaki tüm görsel sayısına oranı olarak değer alır.

- *(Internal-External) to all links*: Dahili bağlantılar, güncel otorite oluşturmaya, PageRank puanını dağıtmaya ve görünürlüğü iyileştirmeye yardımcı olur. Harici bağlantılar ise genellikle içeriğe daha fazla bağlam eklemek ve bir okuyucuyu mevcut konuya ek ve önemli bilgiler ekleyecek başka bir kaynağa yönlendirmek için içerik içinde kullanılırlar. SEO uygulamalarında bağlantıların bu iki bağlantı türüne olabildiğince eşit olarak dağıtılması tavsiye edilmektedir. Bu öznitelik dahili ve harici bağlantı sayısı farkının tüm bağlantı sayısına oranı olarak hesaplanır.
- *No-follow to all links*: Takip edilmeyen bağlantılar (no-follow links), kullanıcıları bağlantı veren web sayfasının kalitesinden ödün vermeden başka bir web sayfasına yönlendirerek bir web sayfasının içeriğine ek bilgi sağlar. Kullanıcı tarafından takip edilmeyen bağlantılar normal bağlantılardan ayırt edilemez. Bağlantı analizi algoritmaları bu özelliği olan bağlantıları web sayfaları için kalite puanı atama süreçlerinde kullanmazlar. Ancak, bu özelliği olan bağlantıların sayısının sayfadaki tüm bağlantı sayısına oranı web sayfasının kalitesi hakkında ipucu verebilmektedir.

5.3.4. URL öznitelikleri

URL bilgisine dayalı URL derinliği, URL uzunluğu gibi öznitelikler, LTR deneylerinde yaygın olarak kullanılmaktadır. Bu özniteliklere ek olarak bu çalışmada “Https” özniteliği kullanılmaktadır. SSL (Secure Socket Layer veya Güvenli Yuva Katmanı) sertifikasına sahip bir web sayfası, “https” ile başlayan bir URL adresine sahiptir. SSL sertifikası, şifreleme yoluyla istemci (client) ve sunucu (server) arasında güvenli veri alışverişi sağlar. Bu nedenle, SSL sertifikası sahibi olmak, bir web sayfasının güvenilirliğinin göstergelerindendir.

5.3.5. Benzerlik öznitelikleri

Bir web sayfasının alan (başlık, açıklama, anahtar kelimeler, sayfa içi başlıklar ve içerik) içeriklerinin birbiriyle tutarlı olması beklenir. Bu tutarlılık web sayfasının bir konu hakkındaki uzmanlığını belirlemekle birlikte kullanıcı deneyimini de etkilemektedir. Bu çalışmada alan içerikleri arası tutarlılık benzerlik ölçme yöntemleriyle ölçülmektedir. Bir web sayfasının alan içerikleri arasındaki benzerlik ölçümü aynı zamanda spam tanıma (spam detection) tekniklerinde de kullanılmaktadır (Pera & Ng, 2008). Benzerlik ölçümü için metin benzerliği ölçümlerinde sıklıkla kullanılan kosinüs benzerliği kullanılmıştır (Denklem 2.1).

Belirtilen alan içeriklerinin ikili kombinasyonlarından elde edilen çiftlerin her birinin benzerliği bir benzerlik özniteliği olarak bu çalışmada kullanılmıştır. Yapılan deney sonuçlarında başlık ve anahtar kelimeler alan içeriklerinden elde edilen benzerlik özniteliği olmadan oluşturulan öznitelik kümesi tüm düzeneklerde daha iyi sonuç verdiğinden bu öznitelik, kümeye dahil edilmemiştir. Tablo 5.2, belirtilen alanları ve bu alan içeriklerinden elde edilen benzerlik özniteliklerini göstermektedir.

Tablo 5.2. Web sayfası alan içeriklerinden SEO teknikleriyle türetilen benzerlik öznitelikleri

	Açıklama	Anahtar Kelime	Sayfa İçi Başlık	İçerik
Başlık	SimDescTitle	-	SimTitleHead.	SimTitleCon.
Açıklama		SimDescKey.	SimDescHead.	SimDescCon.
Anahtar Kelime			SimKey.Head.	SimCon.Key.
Sayfa İçi Başlık				SimCon.Head.

6. DENEYSEL YÖNTEMLER

Bu çalışmanın amacı temel SEO tekniklerine dayanan sorgu-bağımsız öznitelikler üretmek ve bu özniteliklerin LTR deneylerinde test edilerek belge kalitesini ölçme yoluyla bir bilgi erişim sisteminin etkinliği üzerindeki etkilerinin değerlendirilmesidir. Bu amaç doğrultusunda deneyler, en kapsamlı ve güncel web belge koleksiyonu (web document collection) olan ClueWeb09 ve ClueWeb12 koleksiyonları ile gerçekleştirilmiştir. Deneylerde ayrıca, bu koleksiyonları kullanan çeşitli konferanslar ve atölye çalışmaları (workshop) tarafından yayınlanan sorgu setleri ve alaka yargıları (qrel) kullanılmıştır. Deneylerde, sorgular ve sorgularla eşleşen belgeler için referans (base) ve SEO öznitelikleri hesaplanmış ve SEO özniteliklerinin LTR sisteminin başarımına olan etkileri incelenmiştir. Bu bölümün alt başlıklarında deneylerde kullanılan belge koleksiyonları, sorgu setleri, referans öznitelikler ve deney tasarımı anlatılmaktadır.

6.1. Kullanılan Belge Koleksiyonları

Deneylerde, çalışmanın hedefine uygun olarak web belge koleksiyonları kullanılmıştır. ClueWeb09 ve ClueWeb12 koleksiyonları bilgi erişimi alanındaki çalışmaları desteklemek için oluşturulan güncel, literatürde sıklıkla kullanılan ve genel web yapısına oldukça yakın web belge koleksiyonlarıdır.

6.1.1. ClueWeb09

ClueWeb09 koleksiyonu 2009 yılının ocak ve şubat aylarında toplanan, 10 dilden yaklaşık 1 milyar web belgesi barındıran ve yaklaşık 25 TB boyutunda bir koleksiyondur. Çalışmamızda bu koleksiyonun Category A (ClueWeb09A) ve Category B (ClueWeb09B) olmak üzere iki alt kümesi kullanılmaktadır. ClueWeb09A, ClueWeb09 koleksiyonundaki İngilizce web belgelerini içeren (~500 milyon web belgesi) koleksiyondur. ClueWeb09B koleksiyonu ise ClueWeb09A koleksiyonunun bir alt kümesi olup 50 milyon İngilizce web belgesi içerir. ClueWeb09B koleksiyonu en yüksek tarama önceliğini (highest crawl priority) yansıttığından, Wikipedia sayfaları da dahil olmak üzere görece yüksek kaliteli belgeleri barındırdığı söylenebilir (Santos, Macdonald, & Ounis, 2011).

6.1.2. ClueWeb12

ClueWeb12 koleksiyonu, 10 Şubat 2012 ile 10 Mayıs 2012 tarihleri arasında toplanan yaklaşık 733 milyon İngilizce web belgesi barındıran bir koleksiyondur. Bu koleksiyon ayrıca yaklaşık 50 milyon belge barındıran bir alt küme koleksiyonu içerir. ClueWeb12-B13 isimli bu koleksiyon, ClueWeb12 koleksiyonundaki belgelerden temsili veya tek tip (uniform) %7'lik bir örnek alınarak oluşturulmuştur. Çalışmamızda ClueWeb12 koleksiyonunun iki versiyonu (tüm koleksiyon ve ClueWeb12-B13) da kullanılmaktadır.

ClueWeb12 koleksiyonu, selefi ClueWeb09 koleksiyonuna göre bir miktar daha filtrelenmiştir. Koleksiyondaki belgeleri toplamak için yapılan tarama işleminde (crawl) kullanılan tohum bağlantı adresleri (seed URLs), ClueWeb09 koleksiyonundaki yüksek PageRank puanı alan ve spam olasılığı düşük web sayfalarından alınmıştır. Buna ilave olarak zararlı yazılım veya pornografik öğeler içeren web sayfaları da tarama işlemi sırasında filtrelenmiştir.

6.2. Kullanılan Sorgu Setleri

Bilgi erişim sistemlerinin değerlendirilebilmesi için çeşitli konferanslar ve atölye çalışmaları sorgular yayınlarlar. Katılımcı grupların sistemleri bu sorgularla hedef belge koleksiyonundaki belgeleri eşleştirerek birer çalışma (run) üretirler. İlgili organizasyon çalışmalardaki belgeleri kullanarak farklı algoritmalarla bir belge havuzu oluşturur (pooling). İnsan değerlendiriciler tarafından bu belge havuzundaki belgelerle sorgular arasındaki alaka belirli puanlara göre işaretlenerek alaka yargı belgesi (qrel) oluşturulur. Organizasyon sonrası tüm araştırmacıların sistemlerini test edebilmesi için sorgu setleriyle birlikte bu alaka yargı belgesi de yayınlanır.

6.2.1. Web Track

Web Track sorgu setleri TREC organizasyonu tarafından Web Track görevleri için 2009 ve 2014 yılları arasında yayınlanmıştır. Her Web Track sorgu setinde 50 adet sorgu bulunur. 2009-2012 yılları arası yayınlanan 4 sorgu setindeki (WT09-WT12) sorgulara 1-200 arası numaralar verilmiştir ve sorguların hedef koleksiyonu ClueWeb09 koleksiyonudur. Bu çalışmada belirtilen 200 sorgudan, belge havuzunda bir sorgu ile eşleşen hiç belge olmaması veya alaka yargı belgesinde bir sorgu için hiç belge

işaretlenmemesi gibi sebeplerle 197 sorgu kullanılmıştır ve CW09A sorgu seti olarak adlandırılmıştır. 2013 ve 2014 yıllarında yayınlanan 2 sorgu setindeki (WT13 ve WT14) sorgulara 251-300 arası numaralar verilmiştir ve sorguların hedef koleksiyonu ClueWeb12 koleksiyonudur (Clarke, Craswell, & Soboroff, 2009; Collins-Thompson, Bennett, Diaz, Clarke, & Voorhees, 2014).

6.2.2. Million Query Track

Million Query Track görevleri TREC organizasyonu tarafından 2007-2009 yılları arasında 3 kez gerçekleştirilmiştir. Bu görevlerin en önemli özelliği çok sayıda sorgu ve bir sorgu için görece az sayıda sorgu-belge alaka yargısı içermesidir. Bu görevler iki önemli amaca hizmet etmektedir. Bunlardan birincisi, çok sayıda sorgu ve geniş bir web belge koleksiyonu ile ad hoc bilgi erişiminin araştırılmasıdır. İkincisi ise sistemlerin değerlendirilmesinde yüzeysel olarak değerlendirilen çok sayıda sorgunun veya kapsamlı olarak değerlendirilen az sayıda sorgunun daha iyi olduğunun araştırılmasıdır.

Million Query 2007 ve Million Query 2008 görevlerinde hedef belge koleksiyonu GOV2 web belge koleksiyonudur. Bu koleksiyon sadece .gov uzantılı web sayfalarından oluştuğundan bu görevlerde kullanılan sorgu setleri bu çalışmada kullanılmamıştır. Million Query Track 2009 görevinde ise hedef koleksiyon ClueWeb09B web belge koleksiyonudur. Bu görevde kullanılan sorgu setinde (MQ09) 562 sorgu bulunmaktadır ve bu sorguların ilk 50 tanesi WT09 sorgu setindeki sorgulardan oluşmaktadır (Carterette, Pavlu, Fang, & Kanoulas, 2009).

6.2.3. Tasks Track

Tasks Track görevleri TREC organizasyonu tarafından 2015 ve 2016 yıllarında 2 kez gerçekleştirilmiştir ve hedef koleksiyonu ClueWeb12 koleksiyonudur. Bu görevlerin gerçekleştirilmesindeki motivasyon, standart değerlendirme mekanizmalarının bir erişim sisteminin kalitesini ölçerken yalnızca sorgu için getirilen belgelerin alakasına odaklanması ve kullanıcının sorguyu oluşturma amacını ve kullanıcı memnuniyetini (user satisfaction) göz ardı etmesidir. Buna bağlı olarak bu görevlerin 3 değerlendirme amacı vardır: Görev anlama (task understanding), görev tamamlama (task completion) ve ad hoc bilgi getirimidir. Tasks Track görevlerinde 2 adet 50 sorgu barındıran sorgu seti (TT15 ve TT16) olmak üzere toplamda 100 sorgu ve bahsedilen değerlendirme amaçları

için alaka yargı belgeleri yayınlanmıştır. Bu çalışmada bu sorguların 85 tanesi (Bölüm 6.2.1’de belirtilen sebeplerden) ve değerlendirme için ad hoc alaka yargı belgeleri kullanılmıştır (Yılmaz, ve diğerleri, 2015). Bu çalışmada, TT15 ve TT16 ile WT13 ve WT14 sorgu setlerinin birleşimiyle elde edilen CW12A sorgu seti kullanılmaktadır.

6.2.4. We Want Web

Bilgi erişimi alanındaki konferans ve atölye çalışmalarının ad hoc bilgi erişimi görevlerinden uzaklaşarak farklı görevlere yönelmeleri üzerine NTCIR organizasyonu, içeriğinde geleneksel ad hoc bilgi erişimi olan We Want Web (WWW) görevlerini gerçekleştirmiştir. WWW görevlerinde hedef koleksiyon ClueWeb12-B13 web belge koleksiyonudur. İlk WWW görevi olan WWW-1 2017 yılında gerçekleşmiştir (Luo, ve diğerleri, 2017). Bu çalışmada WWW-1, WWW-2 ve WWW-3 görevlerinde kullanılan, sorgu sayıları sırasıyla 100, 80 ve 80 olmak üzere toplamda 260 sorgu içeren 3 sorgu seti kullanılmaktadır ve NTCIR sorgu seti olarak adlandırılmaktadır. Bu sorgulardan bir tanesi için (144. sorgu) sistem hiçbir belge döndürmediği için sonuçlar 259 sorgu üzerinden değerlendirilmektedir.

6.3. Referans Öznitelikler

Bu çalışmada türetilen SEO-tabanlı özniteliklerin bir bilgi erişim sisteminin etkinliğine etkisini incelemek amacıyla referans öznitelikler kullanılmaktadır. Bu amaçla önceki çalışmalarda olduğu gibi (Macdonald, Dinçer, & Ounis, 2015; Macdonald, Santos, & Ounis, 2012; 2013; Macdonald, Santos, Ounis, & He, 2013) literatürdeki LTR çalışmaları kapsamlı bir şekilde taranmış ve olabildiğince geniş bir referans öznitelik kümesi oluşturulmuştur. Oluşturulan referans öznitelik kümesi 168 öznitelikten oluşmakta ve ayrıca LTR çalışmalarında sıklıkla kullanılan kıyaslama veri setlerinde (LETOR ve MSLR (Qin & Liu, 2013)) bulunan öznitelikleri de içermektedir.

Bu çalışmada kullanılan referans öznitelikler Tablo 6.1’de listelenmiştir. Literatürde bulunan öznitelikler bazı kaynaklarda sorgu-bağımlı ve sorgu-bağımsız olmak üzere 2 sınıfa, bazı kaynaklarda ise sorgu öznitelikleri (query features veya Type-Q), sorgu-belge öznitelikleri (query-document features veya Type-QD) ve belge öznitelikleri (document features veya Type-D) olmak üzere 3 sınıfa ayrılmaktadır. Bu

çalışmada ikinci sınıflandırma (Type-Q, Type-QD ve Type-D sınıflandırılması) kullanılmaktadır.

Tablo 6.1. Çalışmada kullanılan referans öznitelikler

Öznitelik Sınıfı	Öznitelik Adı
Type-Q	WordCount, Gamma, Omega, AvgPMI, SCS (Hauff, Azzopardi, & Hiemstra, 2009; He & Ounis, 2006)
Type-Q	Mean and Variance of IDF, CTI, Skewness, Kurtosis, SCQ (Hauff, Azzopardi, & Hiemstra, 2009; He & Ounis, 2006; Zhao, Scholer, & Tsegay, 2008)
Type-QD	TF Sum, Min, Max, Mean, Variance for 5 fields (Qin & Liu, 2013)
Type-QD	TF-IDF Sum, Min, Max, Mean, Variance for 5 fields (Qin & Liu, 2013)
Type-QD	Sum, Min, Max, Mean, Variance of term counts divided by text length for 5 fields (Qin & Liu, 2013)
Type-QD	BM25, DFIC, DFRee, DLH13, DPH, LGD, PL2, LM.JM, LM.ABS, LMIR.DIR for 5 fields (Macdonald, Santos, & Ounis, 2013; Macdonald, Santos, Ounis, & He, 2013; Qin & Liu, 2013)
Type-QD	Covered query term number for 5 fields (Qin & Liu, 2013)
Type-QD	Covered query term ratio for 5 fields (Qin & Liu, 2013)
Type-QD	Minimum coverage, in the title of the document, of all query terms (Dadachev, 2015)
Type-QD	Minimum coverage, in the body of the document, of all query terms (Dadachev, 2015)
Type-D	Number of child pages, Inlink count, Outlink count, PageRank (Qin & Liu, 2013)
Type-D	SpamScore (Cormack, Smucker, & Clarke, 2011)
Type-D	Entropy, URL Depth, Average term length, Fraction of anchor text, Fraction of table text, Number of title terms, Stopword coverage, Stopword ratio, Text/Document length (Bendersky, Croft, & Diao, 2011)
Type-D	URLWiki (Macdonald, Santos, & Ounis, 2013)
Type-D	CDD (Zhou & Croft, 2005)

6.3.1. Sorgu öznitelikleri

Sorgu terimlerinin, sorgu veya tüm koleksiyon terimleri içerisindeki istatistiksel veya frekans değerleri üzerinden hesaplanan öznitelikler sorgu öznitelikleri olarak adlandırılır. Doğası gereği sorgu öznitelikleri belge bağımsızdır, yani bir sorgu özniteliğinin değeri bir sorgu için getirilen tüm belgeler için aynıdır. Böyle bir durumda teorik olarak sorgu özniteliklerinin LTR algoritmalarının öğrenme aşamasında bir katkısı olmayacağı düşünülebilir. Ancak deneysel çalışmalar pratikte sorgu özniteliklerinin ağaç tabanlı LTR algoritmalarında kullanılmasının bilgi erişim sisteminin etkinliğini artırdığını göstermiştir (Macdonald, Santos, & Ounis, 2012). Bu çalışmada literatürde kullanılan 15 adet sorgu özniteliği kullanılmıştır (He & Ounis, 2006; Hauff, Azzopardi, & Hiemstra, 2009):

WordCount: Bir sorgudaki terim sayısını ifade eder.

Gamma: Bir sorguyu oluşturan terimlerin bilgilendirme miktarının dağılımını ifade eder ve sorgu terimleri arasındaki minimum *idf* değerinin maksimum *idf* değerine oranı olarak hesaplanır.

Omega: Bir sorgunun genellik/özgüllük durumunu belirten sorgu kapsamı (query scope) değeridir. Örneğin “Olympic games” sorgusu “Olympics Sydney” sorgusundan daha genel bir sorgudur. Bir sorgunun omega değeri sorgunun herhangi bir terimini içeren belge sayısının tüm koleksiyondaki belge sayısına oranı üzerinden hesaplanır.

AvgPMI: Noktasal karşılıklı bilgi katsayısı (pointwise mutual information veya PMI), iki terimin belge koleksiyonunda birlikte gözükme istatistiklerini gösteren bir ölçüttür. *AvgPMI* ise bir sorgudaki terim çiftlerinin PMI değerlerinin ortalamasıdır ve yüksek *AvgPMI* değeri birbiriyle ilgili terimler içeren sorguyu belirtir (Carmel & Yom-Tov, 2010).

SCS: Sorgu netliği (query clarity), sorgunun belirsizliği (ambiguity) ile ters orantılıdır. Örneğin “Jordan” sorgusu ile basketbol oyuncusu Michael Jordan veya ülke olan Jordan (Ürdün) hakkında arama yapıldığı belirsizdir. Netlik puanı (Clarity Score) ile ölçülen sorgu netliğinin bilgi erişim sistemi etkinliği üzerinde önemli bir etkisi vardır (Cronen-Townsend, Zhou, & Croft, 2002). Ancak bu puanı hesaplamak fazla maliyetli olduğundan sonraki çalışmalarda basitleştirilmiş versiyonu olan SCS (Simplified query Clarity Score) kullanılmıştır (He & Ounis, 2006). SCS, sorgu özgüllüğünün bir göstergesi olarak basitleştirilmiş sorgu dili modelinin koleksiyon dili modelinden Kullback-Leibler sapmasını ölçer.

IDF: Ters belge frekansı (Inverse Document Frequency veya IDF), belirli bir belge koleksiyonunda bir terimin yaygın mı yoksa nadir mi görüldüğünü ölçen bir terim özgüllüğü ölçütüdür. Koleksiyondaki toplam belge sayısının, terimi içeren belge sayısına bölünmesiyle elde edilir (Jones, 1972).

CTI: Toplam eylemsizliğe katkı (Contribution to Total Inertia veya CTI), IDF gibi terim özgüllüğü ölçütüdür. Bağımsız dağılımda, bir terimin koleksiyon frekansının koleksiyondaki belgelere belge boyutuyla orantılı dağılması ve bu durumun toplam eylemsizliğe katkısı olmaması beklenir. Bilgi verme amacı olan terimlerin ise bu durumu bozması ve toplam eylemsizliğe katkısı olması beklenir. CTI değeri bir terimin bir belgedeki gerçekleşen ve beklenen terim frekansları arasındaki fark üzerinden hesaplanır (Kocabaş, Dinçer, & Karaoğlu, 2014).

Skewness ve Kurtosis: Çarpıklık (Skewness) ve Basıklık (Kurtosis), terim frekans dağılımını ifade eden tabirlerdir. Çarpıklık, dağılımın simetrik dağılımdan farklı olması durumunda gerçekleşir. Basıklık ise dağılımın grafik görünümünün şeklinin basıklığını ifade eder. Çarpıklık ve basıklık sırasıyla 3. ve 4. standardize edilmiş moment üzerinden hesaplanmaktadır.

SCQ: Sorgunun koleksiyon ile benzerliğini ölçen bir ölçüt olan SCQ (Collection Query Similarity), terimin koleksiyon terim frekansı ve ters belge frekansı üzerinden hesaplanmaktadır (Zhao, Scholer, & Tsegay, 2008).

Bu çalışmada sorgu terimlerinin IDF, CTI, Skewness, Kurtosis ve SCQ değerlerinin ortalaması ve varyansları olmak üzere ikişer öznitelik kullanılmıştır.

6.3.2. Sorgu-belge öznitelikleri

Sorgu terimlerinin koleksiyondaki belgelerle olan benzerlik derecesini tarif eden özniteliklere sorgu-belge öznitelikleri denmektedir. Literatürde kullanılan sorgu-belge özniteliklerinin çoğu terim ağırlıklandırma metotlarına dayanır. Bu çalışmada kullanılan sorgu-belge özniteliklerinin birçoğu kıyaslama veri setlerinde kullanılan özniteliklerdir. Bu çalışmada kullanılan sorgu-belge özniteliklerinin çoğunluğu sorgu terimlerinin bir belge üzerindeki aynı öznitelik değerlerinin farklı birleştirme (aggregation) metotları sonucu elde edilen farklı varyasyonlarıdır. Bu çalışmada kullanılan birleştirme metotları toplam, minimum, maksimum, ortalama ve varyanstır. Benzer şekilde bazı sorgu-belge öznitelikleri bir özniteliğin, web sayfasının farklı alanları (field) için elde edilen sonuçlarından da oluşmaktadır. Bir web sayfası 5 alandan oluşmaktadır. Bunlar başlık (title), gövde (body), çapa metin veya bağlantı metni (anchor), URL ve web sayfasının bütünü, yani diğer 4 alanın birleşimidir (whole). Bu çalışmada literatürde kullanılan 137 sorgu-belge özniteliği kullanılmıştır (Qin & Liu, 2013; Macdonald, Santos, & Ounis, 2013):

TF Sum, Min, Max, Mean, Variance for 5 fields: Sorgudaki her bir terimin bir belgedeki frekans değerlerinin belgedeki 5 alan için farklı birleştirme metotları ile elde edilen sonuç değerlerini ifade eder.

TF-IDF Sum, Min, Max, Mean, Variance for 5 fields: Sorgudaki her bir terimin bir belgedeki frekans değerleri ile aynı terimlerin ters belge frekansından elde edilen tf-idf değerlerinin belgedeki 5 alan için farklı birleştirme metotları ile elde edilen sonuç değerlerini ifade eder.

Sum, Min, Max, Mean, Variance of term counts divided by text length for 5 fields: Sorgudaki her bir terimin belgedeki frekansının belge uzunluğuna oranının belgedeki 5 alan için farklı birleştirme metotları ile elde edilen sonuç değerlerini ifade eder.

BM25, DFIC, DFRee, DLH13, DPH, LGD, PL2, LM.JM, LM.ABS, LMIR.DIR for 5 fields: Sorgunun belgenin 5 alanı için hesaplanan terim ağırlıklandırma metodu puanlarını ifade eder.

Covered query term number/ratio for 5 fields: Sorgudaki terimler içerisinde belge içinde bulunan terim sayısı ve bu sayının toplam terim sayısına oranı değerlerini ifade eder.

Minimum coverage, in the title/body of the document, of all query terms: Belgenin başlık ve gövdesinde sorgunun tüm terimlerini kapsayan en kısa kesim uzunluklarını ifade eder (Dadachev, 2015).

6.3.3. Belge öznitelikleri

Belge öznitelikleri sadece belgelerden türetilir ve sorgu bağımsızdır. Bu öznitelikler çoğunlukla belge kalitesi ile ilişkilendirilir (bu çalışmada türetilen SEO-tabanlı öznitelikler de belge öznitelikleridir). Bu çalışmada referans öznitelik kümesi için literatürde kullanılan 16 adet belge özniteliği kullanılmıştır (Zhou & Croft, 2005; Cormack, Smucker, & Clarke, 2011; Bendersky, Croft, & Diao, 2011; Qin & Liu, 2013; Macdonald, Santos, & Ounis, 2013):

Number of child pages: Bir web sayfası ve bu sayfanın altındaki sayfalar ile aralarında “parent-child” ilişkisi vardır. Bir child web sayfasının sadece bir parent sayfası olurken, bir parent web sayfasının birden çok child sayfası olabilmektedir. Bu öznitelik bir web sayfasının altındaki child sayfa sayısını belirtmektedir.

Inlink count: Bir web sayfasına diğer web sayfalarından verilen bağlantı sayılarının toplamını ifade eder.

Outlink count: Bir web sayfasından diğer web sayfalarına verilen bağlantı sayılarının toplamını ifade eder.

PageRank: Bir web sayfasının PageRank (Page, Brin, Motwani, & Winograd, 1999) puanını ifade eder. Bu çalışmada ClueWeb09 ve ClueWeb12 koleksiyonlarını yayınlayan Lemur Project tarafından yayınlanan resmi PageRank puanları kullanılmıştır.

SpamScore: Bir web sayfasının spam olmama olasılığını hesaplayan spam puanlarını (Cormack, Smucker, & Clarke, 2011) ifade eder. Bu çalışmada Lemur Project tarafından yayınlanan resmi SpamScore puanları kullanılmıştır.

Entropy: Belge içeriğinin entropisini ifade eder ve Denklem 6.1'deki formülle hesaplanır.

$$Entropy = - \sum_{\omega \in D} p_D(\omega) \cdot \log p_D(\omega) \quad (6.1)$$
$$p_D(\omega_i) = \frac{tf_{\omega_i, D}}{\sum_{\omega_j \in D} tf_{\omega_j, D}}$$

Denklemdaki $p_D(\omega)$, D belgesindeki ω teriminin frekansının belgedeki tüm terimlerin frekansı toplamına oranıdır ve terimin o belgede bulunma olasılığını ifade eder.

URL Depth: Bir web sayfasının URL yolunda bulunan ters eğik çizgi (backslash) karakteri sayısını ifade eder ve sayfanın alan adı (domain) hiyerarşisindeki derecesini tahmin etmede kullanılır.

Average term length: Belgedeki terimlerin ortalama karakter uzunluğunu ifade eder ve sayfanın okunabilirliği hakkında fikir verir.

Fraction of anchor text: Bir web sayfasındaki bağlantı metinlerinin (anchor text) uzunluğunun sayfanın toplam metin uzunluğuna oranını ifade eder.

Fraction of table text: Bir web sayfasında tablo içerisinde yer alan metinlerin uzunluğunun sayfanın toplam metin uzunluğuna oranını ifade eder.

Number of title terms: Bir web sayfasının başlığındaki terim sayısını ifade eder.

Stopword ratio: Belgedeki durak kelimelerin (stopword), aynı belgedeki durak harici kelimelere oranını ifade eder.

Stopword coverage: Belgedeki durak kelimelerin, durak kelime listesindeki kelime sayısına oranını ifade eder.

Text/Document length: Bir web sayfasındaki kullanıcıya gözüken kelime sayısının sayfanın tamamındaki kelime sayısına oranını ifade eder.

URLWiki: Bir web sayfasının Wikipedia sayfası olup olmadığını ifade eder.

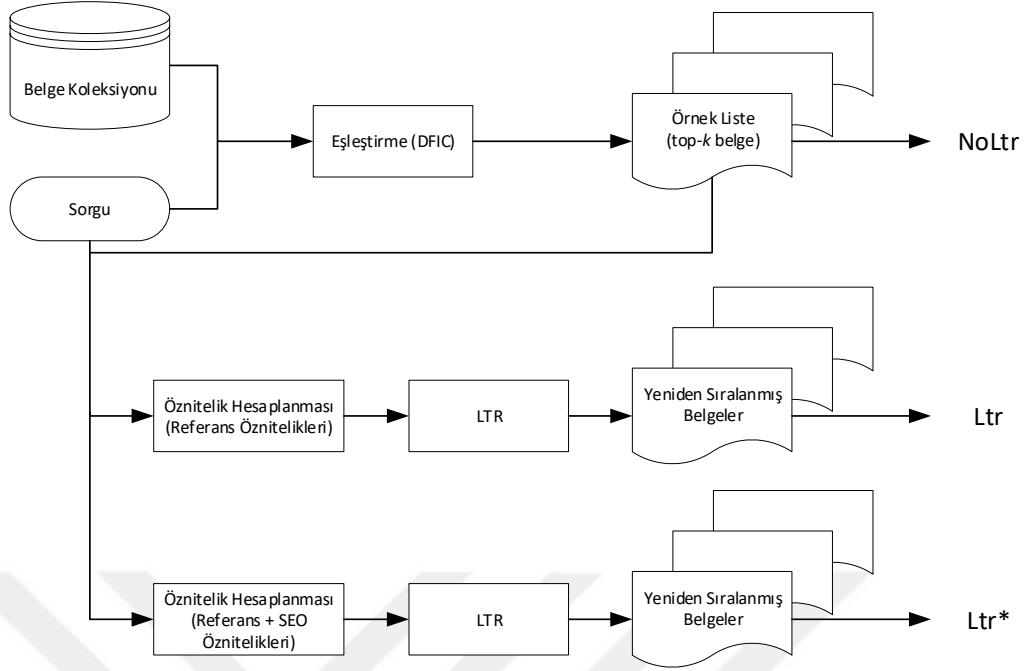
CDD: Koleksiyon-Belge Uzaklığı (Collection-Document Distance), koleksiyonla belge unigram dil modeli arasındaki göreceli entropiyi veya Kullback-Leibler sapmasını (KL divergence) ifade eder ve Denklem 6.2'deki formülle hesaplanır.

$$\begin{aligned}
CDD &= \sum_{\omega} P_{coll}(\omega|C) \cdot \log \frac{P_{coll}(\omega|C)}{P(\omega|D)} \\
P(\omega|D) &= \lambda P_{doc}(\omega|D) + (1 - \lambda) P_{coll}(\omega|C) \\
P_{doc}(\omega|D) &= \frac{tf_{\omega,D}}{\|D\|}, P_{coll}(\omega|C) = \frac{tf_{\omega,C}}{\|C\|}
\end{aligned} \tag{6.2}$$

Denklemdaki $P_{doc}(\omega|D)$ ve $P_{coll}(\omega|C)$ ifadeleri sırasıyla ω teriminin D belgesinde ve C koleksiyonunda bulunma olasılığını ifade eder ve ω teriminin terim frekansının D belgesi ve C koleksiyonu uzunluğuna oranı ile hesaplanır.

6.4. Deney Tasarımı

Bu çalışmada gerçekleştirilen deney aşamaları genel hatlarıyla şu şekildedir: İlk aşamada ilgili sorgular ile alakalı belgeler, bir terim ağırlıklandırma metodu puanlarına göre sıralı listelenerek her bir sorgu kümesi için bir örnek liste (sample) oluşturulmuştur. Bu işleme “top- k retrieval” da denir ve terim ağırlıklandırma puanına göre ilk k adet belge seçilerek örnek liste oluşturulur. Her bir sorgu seti için oluşturulan listelerdeki belgeler LTR algoritmaları ile tekrar sıralanmışlardır. Bu çalışmada türetilen özniteliklerin değerlendirilmesi bu aşamada yapılmaktadır. Oluşturulan örnek listeler için önceki çalışmalarda önerilen ve LTR deneylerinde kullanılan öznitelikler hesaplanarak referans tasarım matrisleri (baseline design matrices) oluşturulmuştur. Mevcut özniteliklerle bu çalışmada türetilen SEO-tabanlı öznitelikler birleştirilerek ve öznitelik değerleri hesaplanarak yeni tasarım matrisleri oluşturulmuştur. Son olarak referans tasarım matrisleri ve oluşturulan yeni tasarım matrisleri ile LTR deneyleri yapılmış ve sırasıyla Ltr ve Ltr* çalışmaları türetilmiştir. Türetilen bu çalışmalar bilgi erişim sistemi değerlendirme ölçütleri ile değerlendirilerek türetilen özniteliklerin sistem etkinliğinde meydana getirdiği değişimler analiz edilmiştir. Şekil 6.1, bu çalışmanın deney tasarımını göstermektedir.



Şekil 6.1. Çalışmanın deney tasarımı

Özellikle bilgi erişim sistemleri çalışmalarında arzu edilen ve teşvik edilen özellikler arasında tekrarlanabilirlik (repeatability) bulunmaktadır. Bu amaçla bu alt başlığın devamında çalışmanın deney tasarımı detaylandırılmaktadır. Çalışma Apache Lucene (Bialecki, Muir, & Ingersoll, 2012) çatısı (framework) altında açık kaynaklı bir Java projesi ile yürütülmüştür. Proje, GitHub deposunda genel erişime açık olarak bulunmaktadır (<http-3>).

Bu çalışmada Bölüm 2.2’de yer alan bilgi erişimi sisteminin süreçleri takip edilmiştir. Sürecin ilk aşamasında sorgu formülleştirilmesi ve belge dizinleme işlemleri yer almaktadır. Bu işlemlerde metinler önışlem (preprocess) aşamalarından geçmektedir. Ancak bu çalışmada kullanılan belge koleksiyonları web sayfalarından oluştuğundan ilk olarak web sayfalarından metinlerin çıkarılması gerekmektedir. Bunun için web sayfaları jsoup kütüphanesi ile ayrıştırılmış (parse), sonrasında sayfa başlığı ve sayfa içeriği ile metin belgeleri haline getirilerek metin işlemleri yapılmasına uygun hale getirilmiştir. Bu çalışmada önışlemler olabildiğince minimum seviyede ve hafif olacak şekilde uygulanmıştır. Lucene çatısı önışlemler için metin analizi işlemlerini gerçekleştiren ve özelleştirilebilen bir çözümleyici (analyzer) yapısı barındırır. Bu çözümleyici içerisinde Tokenizer ve TokenFilter özellikleri sırasıyla metni küçük parçalara (token) ayırmakta ve bu parçalar üzerinde işlem yapılmasına olanak sağlamaktadır. Bu çalışmada metni parçalara ayırma işlemi için gramer tabanlı StandardTokenizer kullanılmıştır. Bu işlem

için StandardTokenizer Unicode Consortium tarafından yayınlanan Unicode Metin Bölümlemesi kurallarını takip eder. Bu çalışmada, çözümleyicide TokenFilter olarak “lowercase”, “englishpossessive” ve “kstem” kullanılmıştır. Bu filtrelerle birlikte metin parçaları küçük harf notasyonuna çevrilmiş, iyelik eklerinden arındırılmış ve KStem (Krovetz, 1993) gövdeleme algoritması ile eklerinden ayrılmıştır. Parçalara ayrılan belgeler Lucene çatısındaki dizinleyiciler (indexer) ile dizinlenmişlerdir (McCandless, Hatcher, & Gospodnetić, 2010).

Eşleştirme aşamasında önışlemden geçmiş olan sorgular dizinlerde bulunan belgeler ile parametrik olmayan bir terim ağırlıklandırma metodu olan DFIC kullanılarak eşleştirilmişlerdir. Bu aşamada bir sorgu için koleksiyondaki belgelere bir eşleştirme puanı atanır ve belgeler bu puanlara göre çoktan aza doğru sıralanır. Sıralı belgelerin ilk k tanesi alınarak (top- k retrieval) örnek liste oluşturulur. Bu çalışmada k değeri literatürdeki çalışmalara benzer şekilde 1,000 olarak alınmıştır (Qin, Liu, Xu, & Li, 2010). Oluşturulan örnek liste aynı zamanda deneylerde NoLtr çalışması olarak da kullanılmaktadır.

Örnek listelerdeki sorgu-belge çiftleri için gerekli öznitelikler hesaplanarak tasarım matrisleri oluşturulmuş ve LTR algoritmaları bu girdileri kullanarak belgeleri yeniden sıralamışlardır. Öznitelik değer hesaplamaları için gerekli bilgiler önceki bölümlerde bulunmakla birlikte implementasyon bilgileri de belirtilen açık kaynak kodlu projede bulunabilmektedir. Bu çalışmanın LTR deneylerinde literatürde sıklıkla kullanılan 3 adet LTR algoritması kullanılmıştır: LambdaMART (λ MART), RankBoost ve SVM^{rank}.

LambdaMART (Burges, 2010), LambdaRank ve MART algoritmalarından faydalanan bir LTR algoritmasıdır. LambdaMART, MART algoritmasının tahminleme (prediction) görevi için kullandığı Gradyan Arttırmalı Karar Ağaçları’nı (Gradient Boosting Decision Trees) LambdaRank algoritmasının kullandığı maliyet fonksiyonu (cost function) ile kullanarak sıralama yapmaktadır. LambdaMART sıralayıcılarının da olduğu bir sistemin “Yahoo! Learning To Rank Challenge” yarışmasını kazanmasıyla (Chapelle & Chang, 2011), LambdaMART algoritmasının gerçek dünyadaki sıralama problemlerini çözmek için başarılı bir algoritma olduğunu kanıtlanmıştır ve günümüzde de “state-of-the-art” LTR algoritması olarak kullanılmaktadır. Bu çalışmada LambdaMART implementasyonu için JForests kütüphanesi kullanılmıştır.

RankBoost (Freund, Iyer, Schapire, & Singer, 2003), bir makine öğrenmesi metodu olan arttırma (boosting) yaklaşımını temel alan bir LTR algoritmasıdır. Arttırma yaklaşımında çok sayıda zayıf öğrenici (weak learner) birleştirilerek yüksek derece doğru tahmin kuralları oluşturulur. Bu yaklaşımı sıralama probleminde uygulayan RankBoost, her iterasyonda belge çiftlerinin dağılımlarını güncelleyerek eğitilen zayıf sıralayıcıları sonunda birleştirerek sıralı belge listesi oluşturur. Bu çalışmada RankBoost algoritması implementasyonu için RankLib kütüphanesi kullanılmıştır.

SVM^{rank} veya RankSVM (Joachims, 2006), lineer Destek Karar Vektörlerini (Support Vector Machines veya SVM) eğiterek belge çiftleri arasında ikili sınıflandırma (binary classification) yoluyla sıralama problemine çözüm sunmaktadır. SVM^{rank} , geleneksel SVM çözümlerinin artan belge sayısı ile süper-lineer olarak ölçeklenmesi problemine çözüm olarak düzlem kesme (cutting-plane) algoritması kullanmaktadır. Bu çalışmada LTR deneylerinde 5-katlamalı çapraz doğrulama (5-fold cross validation) yöntemi, 3 parça eğitim, 1 parça test ve 1 parça da doğrulama aşamasında kullanılacak şekilde uygulanmıştır.

7. DENEY SONUÇLARI

Deneyler, Bölüm 6.4’de detaylandırılan deney tasarımına uygun olarak gerçekleştirilmiştir. CW09A, CW12A, MQ09 ve NTCIR sorgu setleri için gerçekleştirilen deneylerde her bir sorgu seti için NoLtr, Ltr ve Ltr* çalışmaları oluşturulmuştur. Ayrıca Ltr ve Ltr* çalışmaları deney tasarımında tanıtılan her bir LTR algoritması için farklı oluşturulmuştur. NoLtr çalışması terim-ağırlıklandırma metodu sonucunda oluşturulan sıralı belge listesini ifade ederken, Ltr çalışması bu listenin LTR algoritmasının sadece referans öznitelikleri ile çalışması sonucu yeniden sıralanan belge listesini, Ltr* çalışması ise NoLtr çalışmasındaki sıralı belge listesinin LTR algoritmasının referans ve SEO öznitelikleri ile çalıştırılması sonucu elde edilen yeniden sıralı belge listesini ifade etmektedir. Ltr ve Ltr* çalışmaları arasındaki tek fark Ltr* çalışmalarında SEO özniteliklerinin dahil edilmesi olduğundan bu çalışmaların bilgi erişim sistemi etkinlikleri arasındaki fark SEO özniteliklerinin etkinliğini göstermekte ve bu nedenle deney sonuçlarının yorumlanması bu fark üzerinden yapılmaktadır. Belirtilen çalışmaların değerlendirmeleri LTR deneylerinde sıklıkla kullanılan nDCG, ERR ve MAP ölçütleri ile yapılmıştır. Sonuçlarda, nDCG ve ERR ölçütleri için belirlenen kesme değeri (cut-off value) 10 ve 20 olmak üzere ikişer puan ve MAP puanı sunulmuştur.

7.1. Ortalama Sistem Başarımı Analizi

Tablo 7.1-Tablo 7.5 her bir sorgu seti ve her bir LTR algoritması ile elde edilen NoLtr, Ltr ve Ltr* çalışmaları sonuçlarının farklı değerlendirme ölçütlerinin ortalama puanlarını sunmaktadır. Ortalama puanlar, bir sorgu setindeki her bir sorgu için elde edilen puanların ortalamasıdır. Tablolarda NoLtr çalışma sonuçlarının da sunulmasının nedeni LTR algoritmalarının ve kullandıkları özniteliklerin bir bilgi erişim sistemi başarımındaki etkisinin gözlemlenmesi içindir. Bu çalışmanın amaçları arasında olan SEO-tabanlı özniteliklerin etkinliğinin incelenmesi için tablolardaki Ltr ve Ltr* çalışmaları arasındaki karşılaştırmalar değerlendirilmektedir. Tablolardaki koyu renkli puanlar, ilgili düzenekte Ltr* çalışmasının Ltr çalışmasından başarılı olması (ilgili değerlendirme ölçütüne göre daha yüksek puan elde etmesi) durumunu belirtmektedir. Tablolardaki † sembolü ise ilgili düzenekte Ltr* çalışmasının Ltr çalışmasından başarılı olduğu durumlarda bu iki çalışma arasındaki puan farkının iki yönlü eşleştirilmiş t-teste (paired t-test) göre istatistiksel olarak anlamlı ($p < 0.05$) olduğunu belirtmektedir.

Tablo 7.1. Her bir sorgu seti için NoLtr, Ltr ve Ltr* çalışmalarının ortalama nDCG@10 puanları

		CW09A	CW12A	MQ09	NTCIR
	NoLtr	0.03398	0.14175	0.18362	0.32252
LambdaMART	Ltr	0.21405	0.24213	0.32296	0.40084
	Ltr*	0.25297[†]	0.25348	0.33011[†]	0.40516
RankBoost	Ltr	0.25112	0.25756	0.31082	0.39859
	Ltr*	0.26552[†]	0.26435	0.31210	0.40019
SVM ^{rank}	Ltr	0.25873	0.26100	0.32925	0.40761
	Ltr*	0.26661	0.26341	0.32997	0.40935

Tablo 7.1 her bir sorgu seti için NoLtr çalışmasının ve LTR algoritmaları ile elde edilen Ltr ve Ltr* çalışmalarının ortalama nDCG@10 puanlarını sunmaktadır. Tabloya göre LambdaMART aracılığıyla tüm sorgu setleri için elde edilen Ltr* çalışmaları aynı şekilde elde edilen Ltr çalışmalarından daha başarılıdır. Ayrıca CW09A ve MQ09 sorgu setlerinden elde edilen çalışmalar arasındaki fark istatistiksel olarak anlamlıdır. Benzer şekilde RankBoost aracılığıyla tüm sorgu setleri için elde edilen Ltr* çalışmaları aynı şekilde elde edilen Ltr çalışmalarından daha başarılıdır ve CW09A sorgu setinden elde edilen çalışmalar arasındaki fark istatistiksel olarak anlamlıdır. Yine benzer şekilde SVM^{rank} aracılığıyla tüm sorgu setleri için elde edilen Ltr* çalışmaları aynı şekilde elde edilen Ltr çalışmalarından daha başarılıdır.

Tablo 7.2. Her bir sorgu seti için NoLtr, Ltr ve Ltr* çalışmalarının ortalama nDCG@20 puanları

		CW09A	CW12A	MQ09	NTCIR
	NoLtr	0.04218	0.15335	0.21946	0.33122
LambdaMART	Ltr	0.20632	0.24079	0.35354	0.40205
	Ltr*	0.24133[†]	0.25409[†]	0.36020[†]	0.40487
RankBoost	Ltr	0.24342	0.26223	0.34480	0.40410
	Ltr*	0.25989[†]	0.26709	0.34468	0.40586
SVM ^{rank}	Ltr	0.25040	0.27285	0.36002	0.41589
	Ltr*	0.26153[†]	0.27321	0.35780	0.41725

Tablo 7.2 her bir sorgu seti için NoLtr çalışmasının ve LTR algoritmaları ile elde edilen Ltr ve Ltr* çalışmalarının ortalama nDCG@20 puanlarını sunmaktadır. Tabloya göre LambdaMART aracılığıyla tüm sorgu setleri için elde edilen Ltr* çalışmaları aynı şekilde elde edilen Ltr çalışmalarından daha başarılıdır. Ayrıca CW09A, CW12A ve MQ09 sorgu setlerinden elde edilen çalışmalar arasındaki fark istatistiksel olarak anlamlıdır. RankBoost ve SVM^{rank} aracılığıyla CW09A, CW12A ve NTCIR sorgu setleri için elde edilen Ltr* çalışmaları aynı şekilde elde edilen Ltr çalışmalarından daha başarılıdır ve CW09A sorgu setinden elde edilen çalışmalar arasındaki fark istatistiksel olarak anlamlıdır.

Tablo 7.3. Her bir sorgu seti için NoLtr, Ltr ve Ltr* çalışmalarının ortalama ERR@10 puanları

		CW09A	CW12A	MQ09	NTCIR
	NoLtr	0.02543	0.07131	0.05395	0.37344
LambdaMART	Ltr	0.14565	0.13437	0.10354	0.43888
	Ltr*	0.16410[†]	0.13965	0.10635[†]	0.4489
RankBoost	Ltr	0.15979	0.12735	0.10113	0.45042
	Ltr*	0.17124[†]	0.12944	0.10215	0.44881
SVM ^{rank}	Ltr	0.16615	0.13347	0.10497	0.44314
	Ltr*	0.16923	0.13036	0.10493	0.44728

Tablo 7.3 her bir sorgu seti için NoLtr çalışmasının ve LTR algoritmaları ile elde edilen Ltr ve Ltr* çalışmalarının ortalama ERR@10 puanlarını sunmaktadır. Tabloya göre LambdaMART aracılığıyla tüm sorgu setleri için elde edilen Ltr* çalışmaları aynı şekilde elde edilen Ltr çalışmalarından daha başarılıdır. Ayrıca CW09A ve MQ09 sorgu setlerinden elde edilen çalışmalar arasındaki fark istatistiksel olarak anlamlıdır. RankBoost aracılığıyla CW09A, CW12A ve MQ09 sorgu setleri için elde edilen Ltr* çalışmaları aynı şekilde elde edilen Ltr çalışmalarından daha başarılıdır ve CW09A sorgu setinden elde edilen çalışmalar arasındaki fark istatistiksel olarak anlamlıdır. SVM^{rank} aracılığıyla CW09A ve NTCIR sorgu setleri için elde edilen Ltr* çalışmaları aynı şekilde elde edilen Ltr çalışmalarından daha başarılıdır.

Tablo 7.4. Her bir sorgu seti için NoLtr, Ltr ve Ltr* çalışmalarının ortalama ERR@20 puanları

		CW09A	CW12A	MQ09	NTCIR
	NoLtr	0.03020	0.07856	0.05971	0.38032
LambdaMART	Ltr	0.15329	0.14127	0.10897	0.44551
	Ltr*	0.17275[†]	0.14719[†]	0.11180[†]	0.45458
RankBoost	Ltr	0.16847	0.13589	0.10684	0.45673
	Ltr*	0.17942[†]	0.13805	0.10763	0.45507
SVM ^{rank}	Ltr	0.17479	0.14279	0.11038	0.44887
	Ltr*	0.17773[†]	0.13991	0.11017	0.45300

Tablo 7.4 her bir sorgu seti için NoLtr çalışmasının ve LTR algoritmaları ile elde edilen Ltr ve Ltr* çalışmalarının ortalama ERR@20 puanlarını sunmaktadır. Tabloya göre LambdaMART aracılığıyla tüm sorgu setleri için elde edilen Ltr* çalışmaları aynı şekilde elde edilen Ltr çalışmalarından daha başarılıdır. Ayrıca CW09A, CW12A ve MQ09 sorgu setlerinden elde edilen çalışmalar arasındaki fark istatistiksel olarak anlamlıdır. RankBoost aracılığıyla CW09A, CW12A ve MQ09 sorgu setleri için elde edilen Ltr* çalışmaları aynı şekilde elde edilen Ltr çalışmalarından daha başarılıdır ve CW09A sorgu setinden elde edilen çalışmalar arasındaki fark istatistiksel olarak anlamlıdır. SVM^{rank} aracılığıyla CW09A ve NTCIR sorgu setleri için elde edilen Ltr*

çalışmaları aynı şekilde elde edilen Ltr çalışmalarından daha başarılıdır ve CW09A sorgu setinden elde edilen çalışmalar arasındaki fark istatistiksel olarak anlamlıdır.

Tablo 7.5. Her bir sorgu seti için NoLtr, Ltr ve Ltr* çalışmalarının ortalama MAP puanları

		CW09A	CW12A	MQ09	NTCIR
	NoLtr	0.0553	0.1152	0.1332	0.3180
LambdaMART	Ltr	0.1214	0.1666	0.2091	0.3615
	Ltr*	0.1484[†]	0.1704	0.2127[†]	0.3610
RankBoost	Ltr	0.1587	0.1838	0.2053	0.3695
	Ltr*	0.1665[†]	0.1886[†]	0.2050	0.3698
SVM ^{rank}	Ltr	0.1650	0.1931	0.2146	0.3865
	Ltr*	0.1691[†]	0.1928	0.2147	0.3841

Tablo 7.5 her bir sorgu seti için NoLtr çalışmasının ve LTR algoritmaları ile elde edilen Ltr ve Ltr* çalışmalarının MAP puanlarını sunmaktadır. Tabloya göre LambdaMART aracılığıyla CW09A, CW12A ve MQ09 sorgu setleri için elde edilen Ltr* çalışmaları aynı şekilde elde edilen Ltr çalışmalarından daha başarılıdır. Ayrıca CW09A ve MQ09 sorgu setlerinden elde edilen çalışmalar arasındaki fark istatistiksel olarak anlamlıdır. RankBoost aracılığıyla CW09A, CW12A ve NTCIR sorgu setleri için elde edilen Ltr* çalışmaları aynı şekilde elde edilen Ltr çalışmalarından daha başarılıdır. Ayrıca CW09A ve CW12A sorgu setinden elde edilen çalışmalar arasındaki fark istatistiksel olarak anlamlıdır. SVM^{rank} aracılığıyla CW09A ve MQ09 sorgu setleri için elde edilen Ltr* çalışmaları aynı şekilde elde edilen Ltr çalışmalarından daha başarılıdır ve CW09A sorgu setinden elde edilen çalışmalar arasındaki fark istatistiksel olarak anlamlıdır.

Tablolar genel olarak incelendiğinde Ltr ve Ltr* çalışmaları arasında yapılan 60 farklı başarı karşılaştırması içerisinde Ltr* çalışmaları 50 durumda Ltr çalışmalarından daha başarılı sonuçlar elde etmiştir. Ayrıca Ltr* çalışmalarının Ltr çalışmalarından daha başarılı olduğu bu 50 durum içerisinde 21 durumda çalışmalar arasındaki fark istatistiksel olarak anlamlıdır. Genel sonuçlara bakıldığında bu çalışmada türetilen SEO-tabanlı özniteliklerin LTR teknikleri kullanılan bir bilgi erişim sisteminde kullanıldıklarında sistemin başarısını artırdığı söylenebilir. Burada dikkat edilebilecek diğer bir husus da SEO-tabanlı özniteliklerin sistem başarımını artırdığı ve çalışmalar arası istatistiksel olarak anlamlı sonuçların elde edildiği durumların büyük bir çoğunluğu LTR çalışmalarında sıklıkla kullanılan “state-of-the-art” LTR tekniği olan LambdaMART algoritması aracılığıyla elde edilen çalışmalarında gözlemlenmiştir.

Tablo 7.1-Tablo 7.5 tablolarındaki sonuçlara göre tüm LTR teknikleri ile elde edilen Ltr ve Ltr* çalışmalarının tüm değerlendirme ölçütü puanlarındaki arasındaki başarı farkı göz önünde bulundurulduğunda CW09A sorgu seti için elde edilen fark diğer sorgu setleri için elde edilen farklardan daha fazladır. Bunun sebeplerinden birisi sorgu setlerinin hedef aldığı belge koleksiyonlarıdır. Bölüm 6.2’de belirtildiği gibi CW09A ve MQ09 sorgu setleri ClueWeb09 belge koleksiyonunu, CW12A ve NTCIR sorgu setleri de ClueWeb12 belge koleksiyonunu hedef almaktadır. Bölüm 6.1’de belirtildiği gibi ClueWeb09, Web’in filtrelenmemiş bir versiyonu iken ClueWeb12 koleksiyonunun elde edilme aşamasında zararlı veya kalitesiz olduğu düşünülen web sayfalarını elemek için bir filtre uygulanmıştır. Buna göre ClueWeb09 belge koleksiyonunun ClueWeb12 koleksiyonuna oranla daha fazla sayıda düşük kaliteli web sayfası içerdiği söylenebilir. Bununla birlikte MQ09 sorgu seti de ClueWeb09 belge koleksiyonunun sadece Category B alt kümesini kullanmaktadır ve bu alt küme Bölüm 6.1.1’de belirtildiği gibi Wikipedia sayfaları gibi görece yüksek kaliteli web sayfalarını içermektedir. Dolayısıyla, CW09A sorgu setinin hedef aldığı belge koleksiyonu diğer sorgu setlerinin hedef aldıkları koleksiyonlara göre daha fazla düşük kaliteli web sayfası içermektedir. Bu bilgiler göz önünde bulundurulduğunda CW09A sorgu seti için Ltr ve Ltr* çalışmaları arasındaki farkın diğer sorgu setlerindeki çalışmalar arası farklardan fazla olması ile, bu çalışmada türetilen SEO-tabanlı özneliliklerin web sayfalarının belge kalitesini ölçerek sistem başarımını artırdığı çıkarımı yapılabilmektedir.

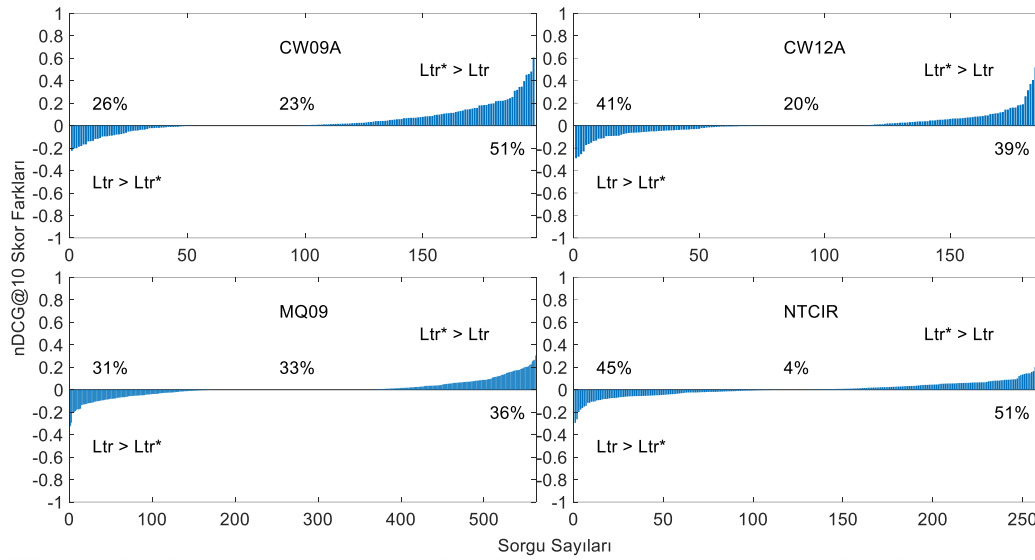
Tablo 7.1-Tablo 7.5 tablolarında Ltr ve Ltr* çalışmalarına ilave olarak NoLtr çalışmalarının sonuçları da sunulmuştur. Bunun nedenlerinden biri LTR algoritmalarının yeniden sıralama yoluyla sistem başarımına etkisini göstermektir. Tablolardaki sonuçlar tüm LTR algoritmalarının tüm sorgu setlerinde sistem başarımını artırdığını göstermektedir. Sonuçlar arasında CW09A sorgu seti ile elde edilen NoLtr ile Ltr çalışmaları arasındaki farkın diğer sorgu setleri ile elde edilen çalışmalar arasındaki farka göre daha yüksek olması dikkat çekmektedir. Bu farkın yüksek olmasının nedenlerinden biri, filtrelenmemiş bir web belge koleksiyonu olan ClueWeb09 koleksiyonunu hedef alan CW09A sorgu setindeki sorgular için terim ağırlıklandırma modeli puanlarına göre sıralanan örneklemdeki belgelerin düşük kaliteli veya spam web sayfası olma olasılığı diğer sorgu setleri için üretilen örneklemdeki belgelere oranla daha yüksektir. Herhangi bir sorguyla alakalı olması muhtemel olmayan düşük kaliteli web sayfalarının (Cormack, Smucker, & Clarke, 2011) örneklemde üst sıralarda bulunması çalışmanın etkinliğini

azaltacağından CW09A sorgu seti için elde edilen NoLtr çalışmalarının sonuçlarının düşük çıkması olağandır. Buna bağlı olarak belge kalitesini ölçmekle ilişkilendirilen sorgu-bağımsız öznelilikleri (PageRank, Waterloo spam puanları vb.) kullanan LTR algoritmalarının düşük kaliteli web belgelerini örneklem listesinde aşağı sıralara iterek sistem etkinliğini artırması beklenen bir durumdur.

7.2. Sorgu Başına (Per-Query) Sistem Başarımı Analizi

Bir bilgi erişim sisteminde istenilen durum sistemin olabildiğince az sayıda sorguda başarısız olmasıdır. Ortalama başarısı iyi olan bir bilgi erişim sistemi bazı sorgularda kötü sonuçlar verebilmektedir (Carmel & Yom-Tov, 2010). Bölüm 7.1’de literatürde de sistem başarımı karşılaştırmalarında genellikle kullanılan yöntem olan ortalama puanlar üzerinden Ltr ve Ltr* çalışmaları karşılaştırması sunulmuştur. Bunun yanı sıra bu bölümde çalışmaların deneylerde kullanılan sorgu setlerinde birbirlerine karşı sistem başarımı olarak üstünlük sağladıkları sorgu sayıları sunulmaktadır.

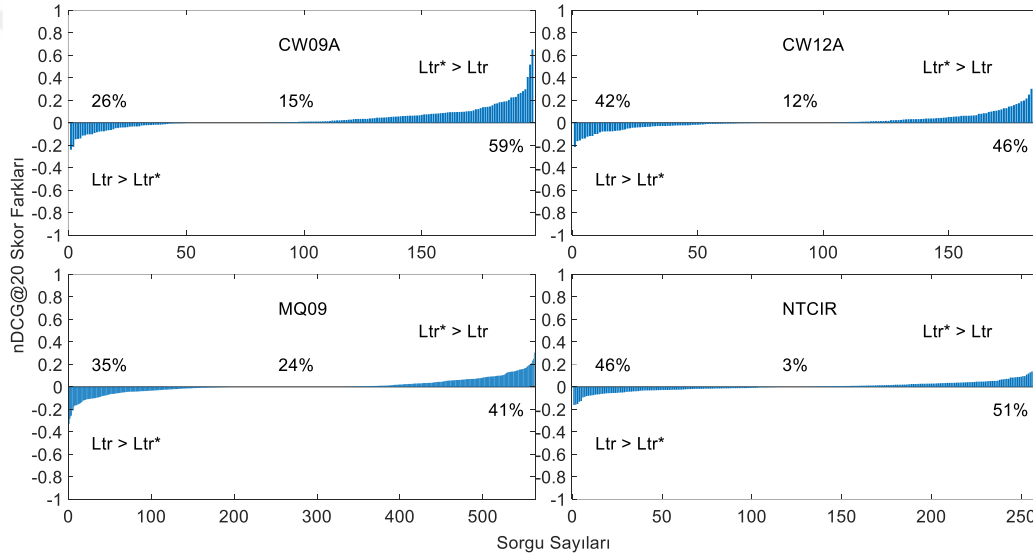
Şekil 7.1 deneylerde kullanılan tüm sorgu setleri için LambdaMART ile üretilen Ltr ve Ltr* çalışmalarının her bir sorgu için $nDCG@10$ puan farklarını ($Ltr^* - Ltr$) olacak şekilde göstermektedir. Şekilde dikey eksen $nDCG@10$ puan farkları ve yatay eksen sorgu sayıları gösterilmektedir. Her bir alt şekil, ilgili sorgu setinde elde edilen çalışmalar arasındaki farkları ve sorgu sayılarını göstermektedir. Alt şekillerde sol tarafta bulunan yüzdelik ifade Ltr çalışmasının Ltr* çalışmasından daha yüksek puanlar elde ettiği sorguların ilgili sorgu setindeki toplam sorgu sayısına oranının yüzdelik ifadesidir. Benzer şekilde ortadaki yüzdelik ifade Ltr ve Ltr* çalışmalarının aynı puanı elde ettiği ve sağ tarafta bulunan yüzdelik ifade ise Ltr* çalışmasının Ltr çalışmasından daha yüksek puan elde ettiği sorgu oranlarını göstermektedir.



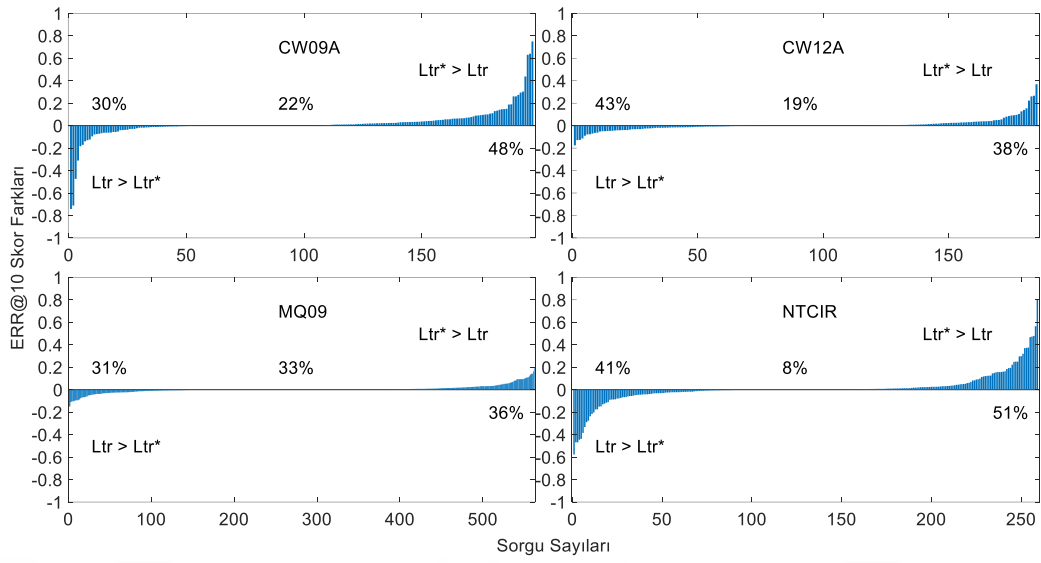
Şekil 7.1. LambdaMART ile üretilen çalışmaların her bir sorgu için $nDCG@10$ puan farkları

Şekilde CW09A sorgu seti ile ilgili sonuçlar incelendiğinde 197 sorgunun %51'inde Ltr* çalışmasının Ltr çalışmasına göre daha yüksek, %23'ünde aynı, %26'sında ise daha düşük $nDCG@10$ puanları elde ettiği gözükmemektedir. CW12A ile ilgili sonuçlarda ise 185 sorgunun %39'unda Ltr* çalışması Ltr çalışmasına göre daha yüksek, %20'sinde aynı, %41'inde ise daha düşük puanlar elde etmiştir. MQ09 ile ilgili sonuçlarda ise 562 sorgunun %36'sında Ltr* çalışması Ltr çalışmasına göre daha yüksek, %33'ünde aynı, %31'inde ise daha düşük puanlar elde etmiştir. Son olarak NTCIR ile ilgili sonuçlarda ise 259 sorgunun %51'inde Ltr* çalışması Ltr çalışmasına göre daha yüksek, %4'ünde aynı, %45'inde ise daha düşük puanlar elde etmiştir. Tablo 7.1'de LambdaMART ile elde edilen çalışmaların sonuçları incelendiğinde tüm sorgu setlerinde elde edilen Ltr* çalışmalarının ortalama $nDCG@10$ puanları Ltr çalışmalarından yüksektir ve Şekil 7.1'deki sonuçlarda da CW12A sorgu seti haricinde tüm sorgu setlerinde Ltr* çalışmalarının Ltr çalışmalarından yüksek $nDCG@10$ puanı elde ettiği sorgu sayısı fazladır. Dolayısıyla CW12A sorgu seti için elde edilen Ltr* çalışması daha az sorguda Ltr çalışmasından yüksek puan elde etmesine rağmen başarılı olduğu sorgulardaki puan farkı büyüklük (magnitude) olarak fazla olduğundan ortalamada daha yüksek sonuç elde etmiştir. CW12A sorgu seti için bu durum diğer LTR ve değerlendirme ölçütleriyle elde edilen sonuçlar incelendiğinde çoğunlukta değildir (yapılan 15 karşılaştırmanın 9'unda Ltr* çalışmaları daha fazla sorguda Ltr çalışmalarından başarılı yüzdelere elde etmektedir) ancak ortalama ve sorgu başına sonuç analizi farkı açısından iyi bir örnektir.

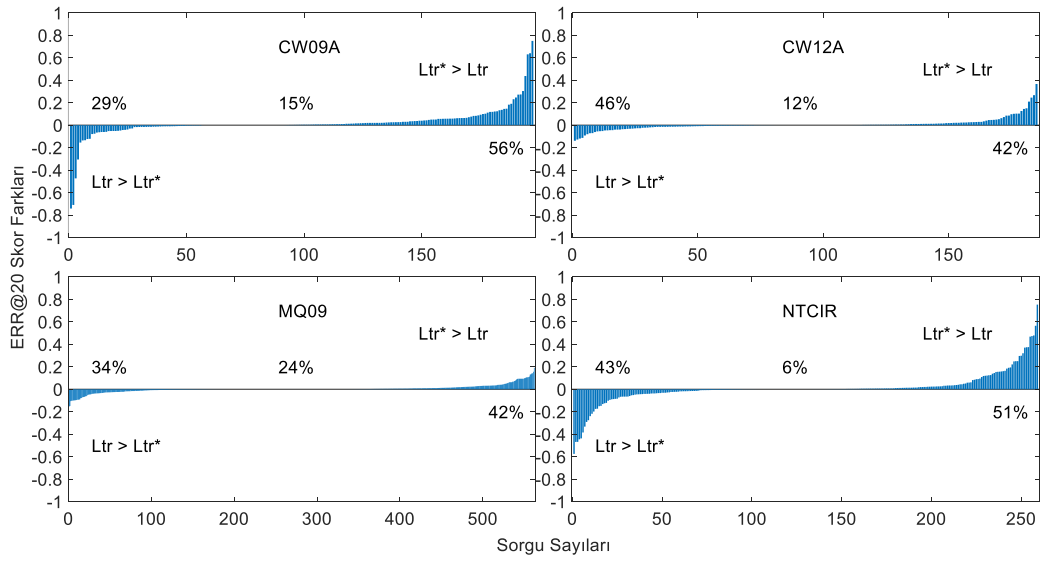
LambdaMART ile elde edilen çalışmaların diğer değerlendirme ölçütlerine (nDCG@20, ERR@10, ERR@20 ve MAP) göre sorgu başına puan farkları ve diğer LTR algoritmalarıyla elde edilen çalışmaların tüm değerlendirme ölçütlerine göre sorgu başına puan farkları bölümün kalan kısmında sunulmuştur (Şekil 7.2-Şekil 7.15). Sonuçlar genel olarak incelendiğinde CW09A sorgu setinde elde edilen Ltr* çalışmaları 15 karşılaştırmaların tamamında Ltr çalışmalarından daha yüksek puan elde ettiği sorgu sayısı daha fazladır. CW12A sorgu seti için bu oran 15 karşılaştırmada 9, MQ09 ve NTCIR sorgu setleri için 12'dir (ayrıca bu iki sorgu setindeki kalan 3'er karşılaştırmada birer beraberlik gözlemlenmiştir).



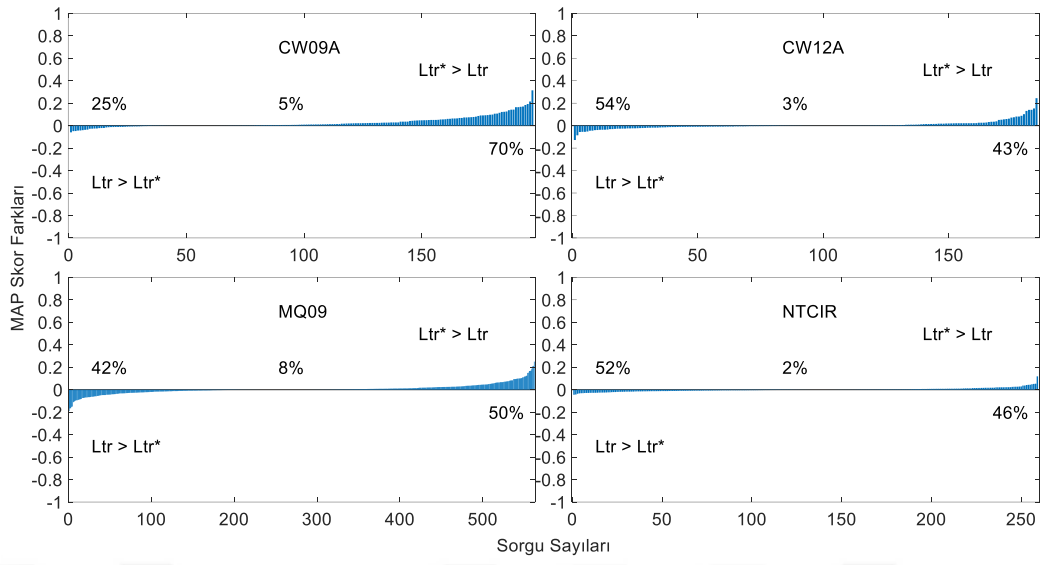
Şekil 7.2. LambdaMART ile üretilen çalışmaların her bir sorgu için nDCG@20 puan farkları



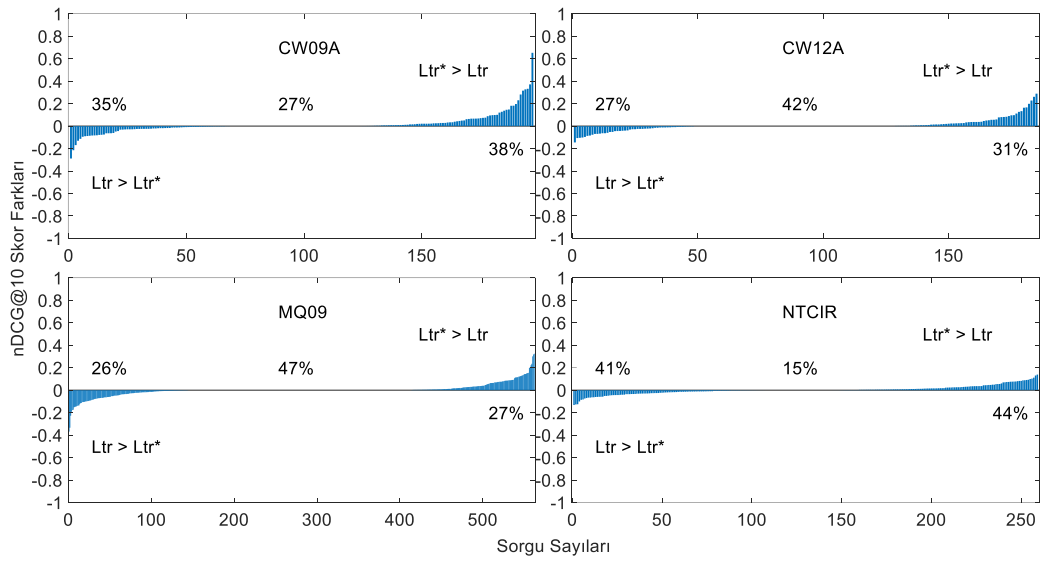
Şekil 7.3. LambdaMART ile üretilen çalışmaların her bir sorgu için ERR@10 puan farkları



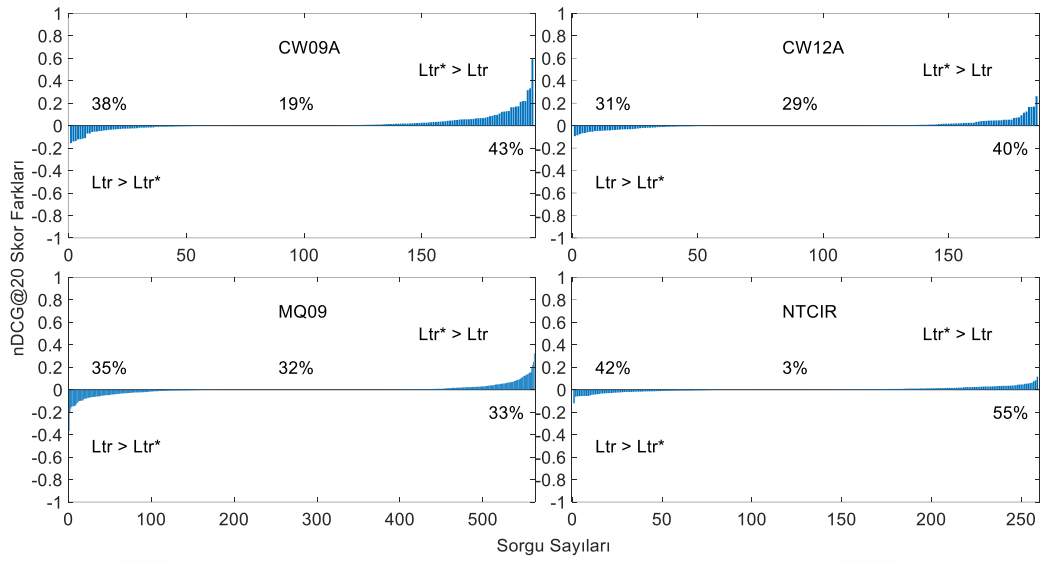
Şekil 7.4. LambdaMART ile üretilen çalışmaların her bir sorgu için ERR@20 puan farkları



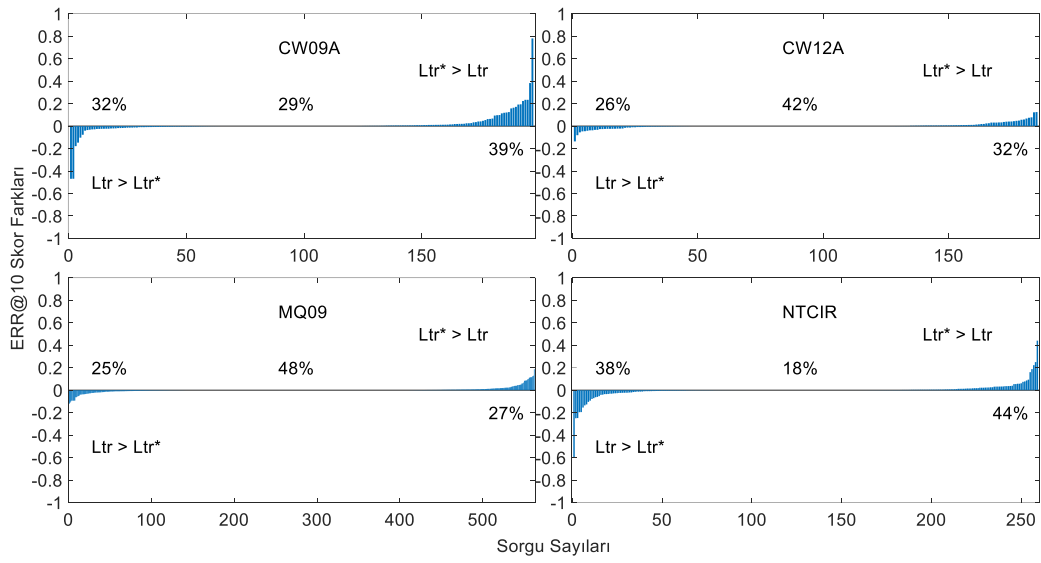
Şekil 7.5. LambdaMART ile üretilen çalışmaların her bir sorgu için MAP puan farkları



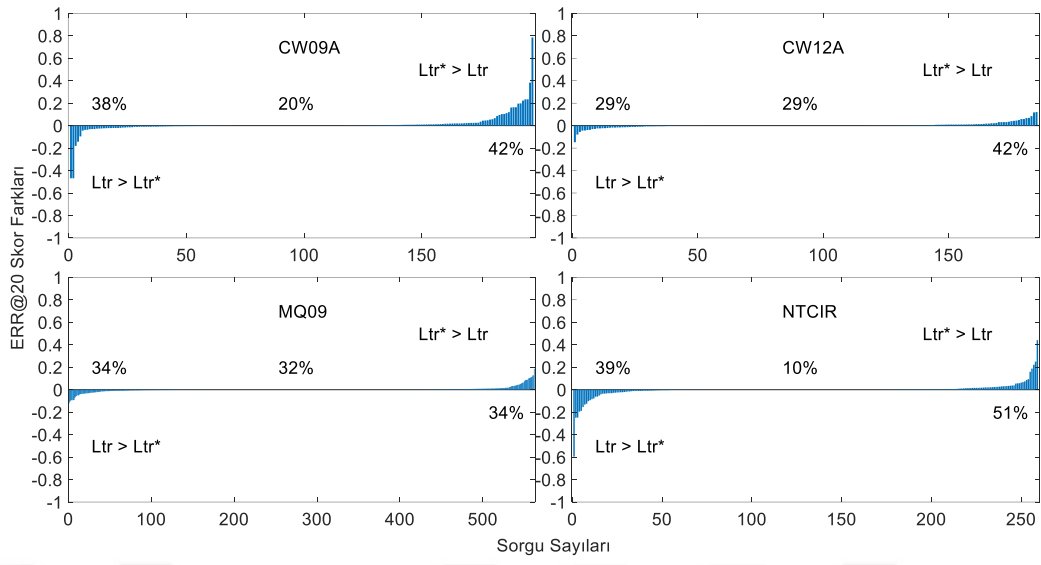
Şekil 7.6. RankBoost ile üretilen çalışmaların her bir sorgu için nDCG@10 puan farkları



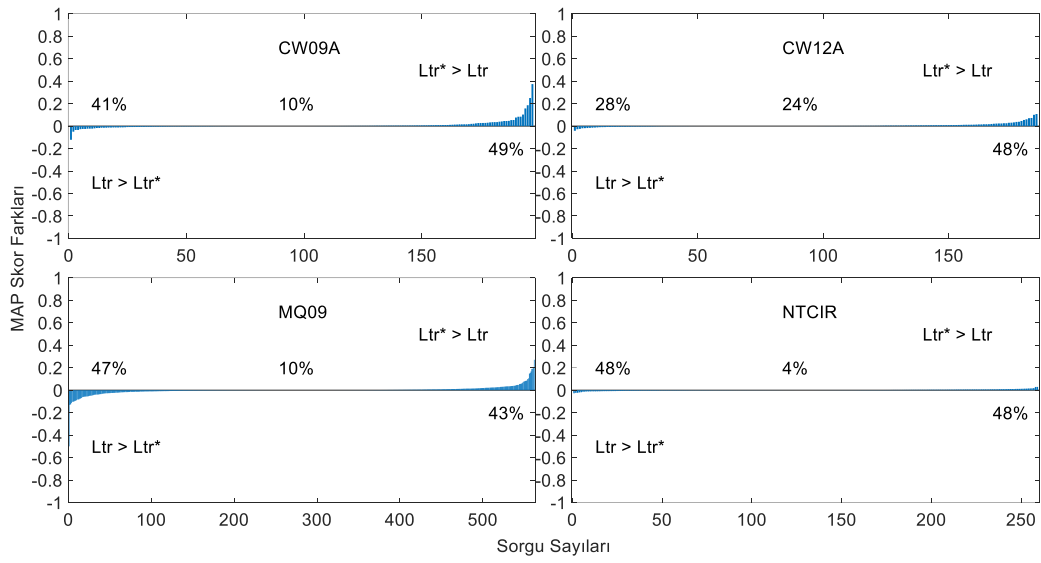
Şekil 7.7. RankBoost ile üretilen çalışmaların her bir sorgu için nDCG@20 puan farkları



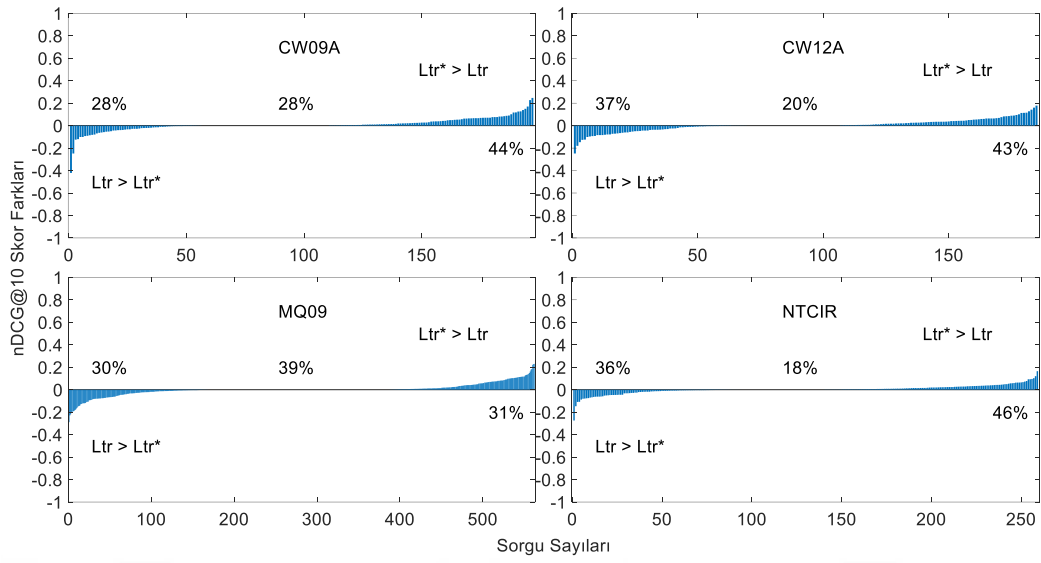
Şekil 7.8. RankBoost ile üretilen çalışmaların her bir sorgu için ERR@10 puan farkları



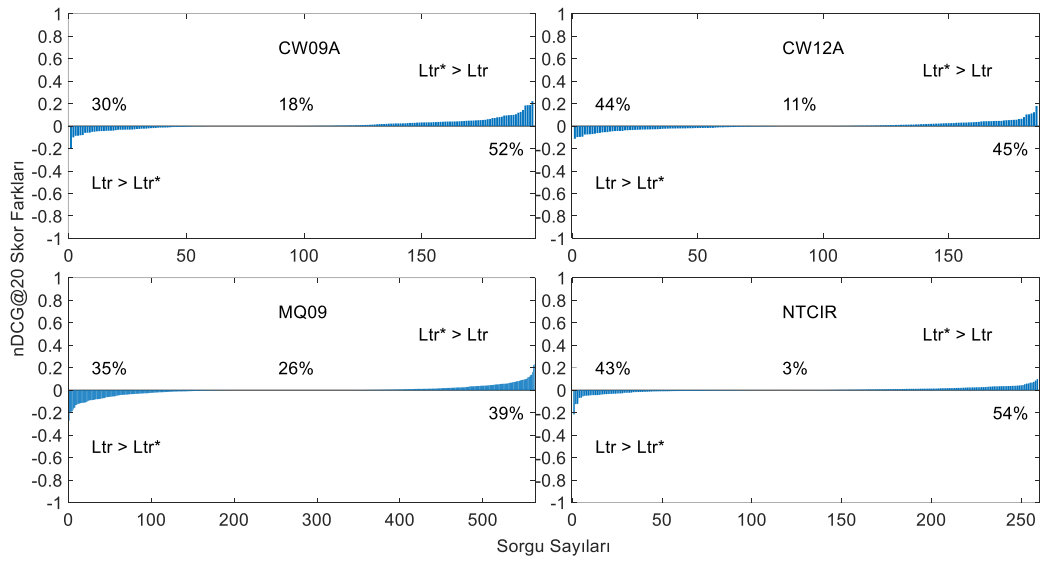
Şekil 7.9. RankBoost ile üretilen çalışmaların her bir sorgu için ERR@20 puan farkları



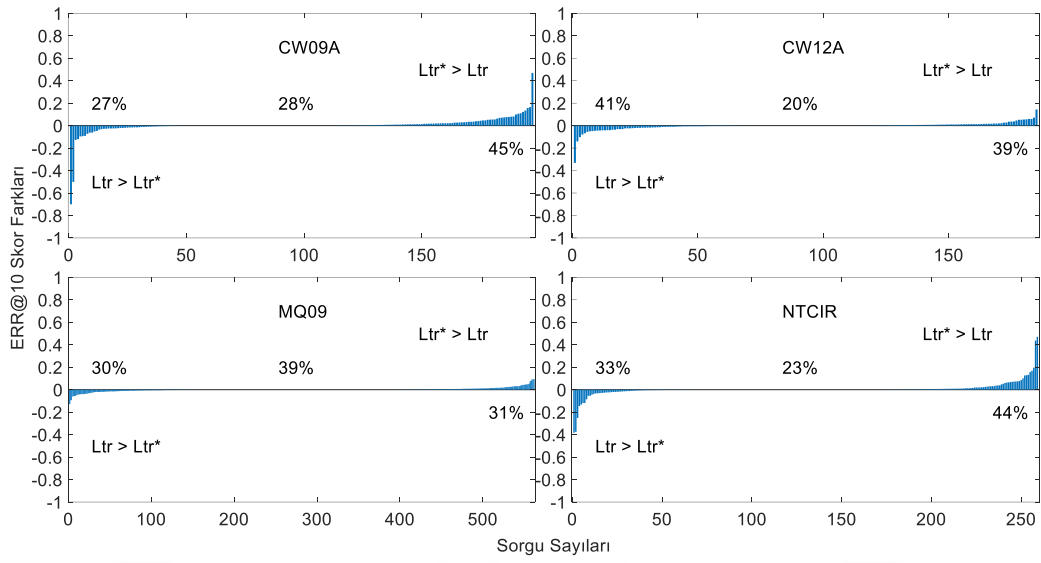
Şekil 7.10. RankBoost ile üretilen çalışmaların her bir sorgu için MAP puan farkları



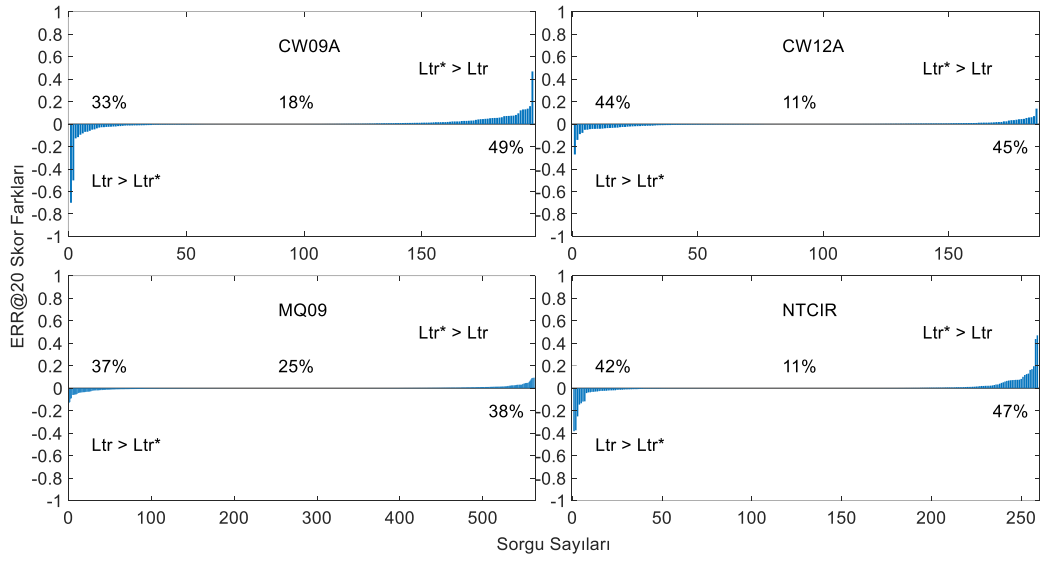
Şekil 7.11. SVM^{rank} ile üretilen çalışmaların her bir sorgu için $nDCG@10$ puan farkları



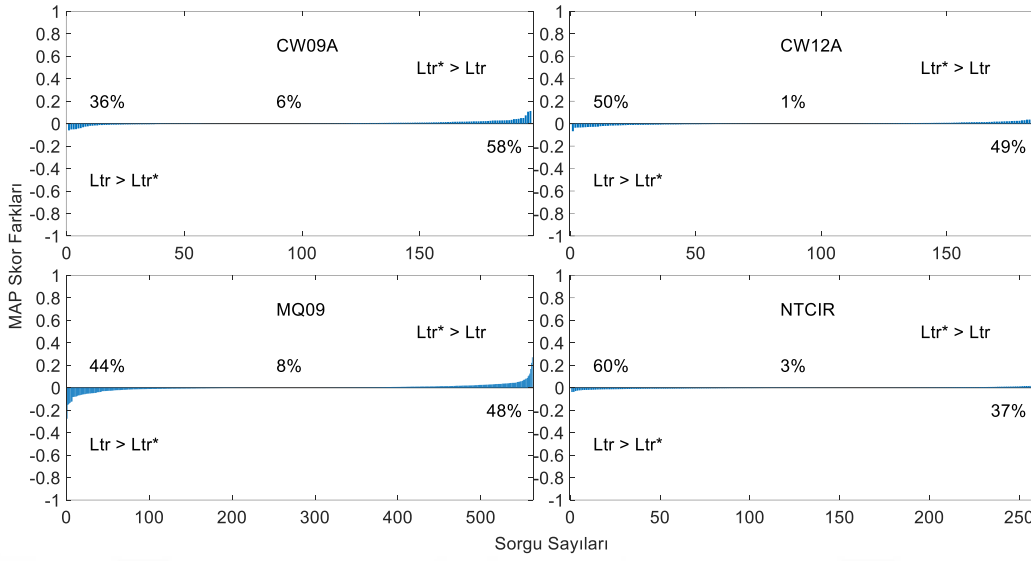
Şekil 7.12. SVM^{rank} ile üretilen çalışmaların her bir sorgu için $nDCG@20$ puan farkları



Şekil 7.13. SVM^{rank} ile üretilen çalışmaların her bir sorgu için ERR@10 puan farkları



Şekil 7.14. SVM^{rank} ile üretilen çalışmaların her bir sorgu için ERR@20 puan farkları

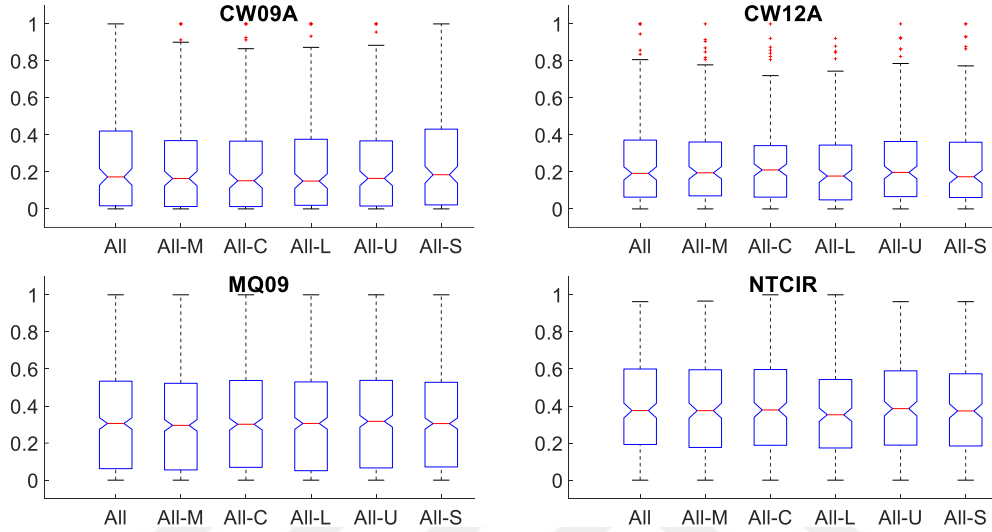


Şekil 7.15. SVM^{rank} ile üretilen çalışmaların her bir sorgu için MAP puan farkları

7.3. SEO-tabanlı Özniteliklerin Ortalama Sistem Başarımına Katkısı

Bu çalışmada tanıtılan SEO-tabanlı öznitelikler, Bölüm 5.3'te belirtildiği gibi beş gruba ayrılmıştır. Bu özniteliklerin grup bazında etkisini incelemek için çıkarma (ablation) metodu kullanılmıştır. Bu metot ile bir öznitelik grubunun sisteme katkısını incelemek için tüm öznitelik kümesinden istenilen öznitelik grubu çıkarılarak iki durumdaki sonuçlar karşılaştırılır. Bu çalışmada tanıtılan beş grup özneliğin sisteme katkısını incelemek için tek yönlü varyans analizi (one-way ANOVA) kullanılmıştır. Şekil 7.16 farklı sorgu setleri için üretilen çalışmaların farklı değerlendirme ölçütü sonuçları için tek yönlü varyans analizlerinin kutu grafiği (boxplot) temsillerini göstermektedir. Her alt şekil, ilgili sorgu seti için üretilen çalışma sonuçlarını göstermektedir. Şekillerdeki “All”, Tablo 5.1 ve Tablo 6.1 tablolarındaki tüm öznitelikleri (bu çalışmada tanıtılan SEO-tabanlı öznitelikler ve deneylerde kullanılan referans özniteliklerinin birleşimi) kullanan LTR algoritması ile üretilen çalışmanın sonuçlarını göstermektedir. “All-M”, “All-C”, “All-L”, “All-U” ve “All-S” ise sırasıyla tüm özniteliklerden Bölüm 5.3'te tanıtılan meta etiket öznitelikleri, içerik öznitelikleri, bağlantı ve resim öznitelikleri, URL öznitelikleri ve benzerlik özniteliklerinin çıkarılması ile elde edilen öznitelik kümesini kullanan LTR algoritması ile üretilen çalışmaların sonuçlarını göstermektedir. Şekillerdeki her kutu için kutunun ortasındaki çizgi medyanı, alt ve üst çizgiler ise sırasıyla %25 ve %75'lik kısımdaki sonuçları temsil etmektedir. Bıyıklar (whiskers) olarak ifade edilen kutu dışındaki çizgi sınırları bu yüzdelik dilimler

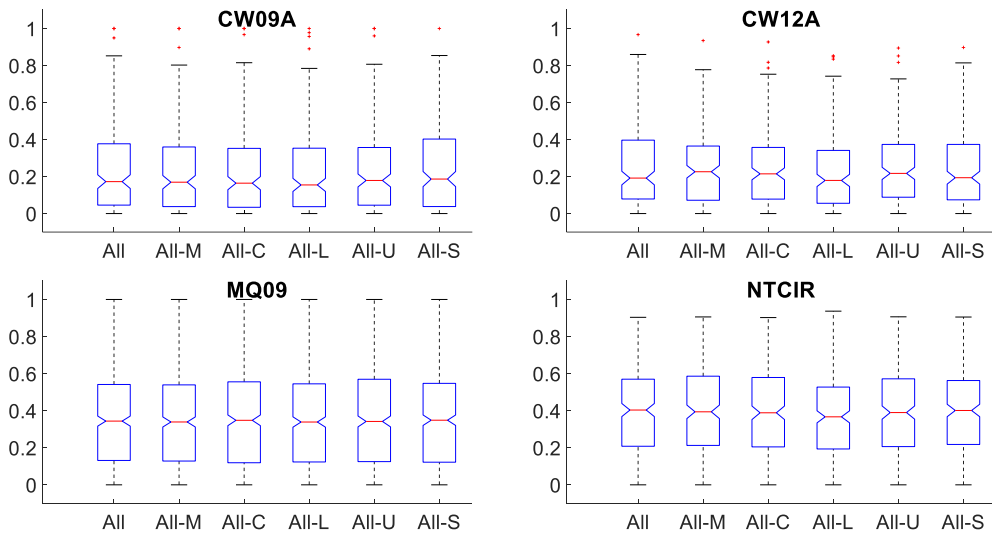
ve aykırı değerler (outlier) haricindeki sonuçları temsil etmektedir. Şekillerdeki kırmızı “+” işaretleri aykırı değerleri belirtmektedir. Kutuların üzerinde medyan çizgisi hizasında kutunun sağ ve sol taraflarında çentikleri (notch) bulunmaktadır. İki kutunun çentiklerinin örtüşmemesi medyanların %95 güven aralığında farklılaştığını göstermektedir.



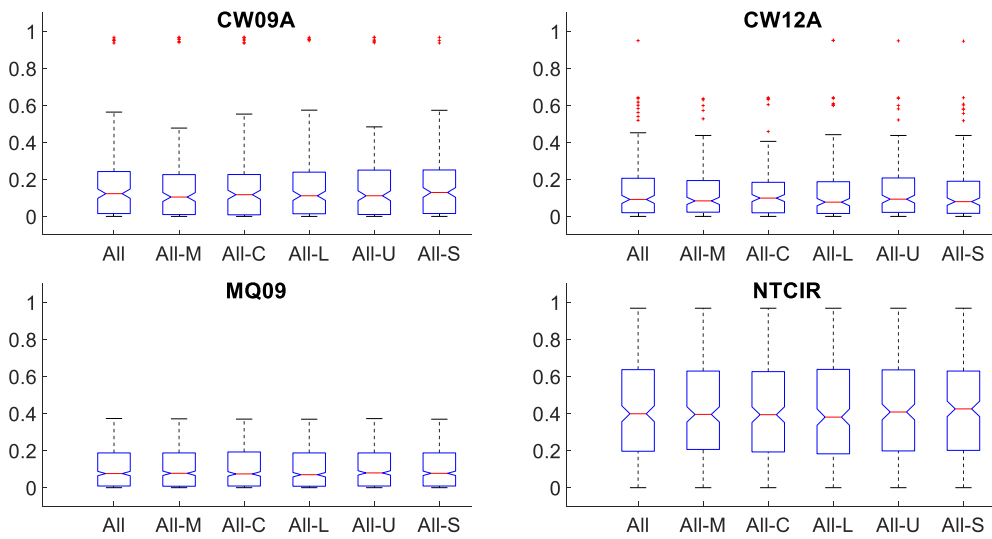
Şekil 7.16. LambdaMART ile üretilen çalışmaların $nDCG@10$ sonuçları için tek yönlü varyans analizleri

Şekil 7.16 sistemde LTR algoritması olarak LambdaMART ve değerlendirme ölçütü olarak $nDCG@10$ kullanıldığı durumda SEO-tabanlı öznitelik gruplarının ortalama sistem başarımına katkısını tek yönlü varyans analizlerinin kutu grafiği gösterimi aracılığıyla göstermektedir. Şekle göre CW09A sorgu seti için tüm öznitelik kümesini kullanan “All” çalışmasıyla karşılaştırıldığında “All-M”, “All-C”, “All-L” ve “All-U” çalışmalarının medyan puanları daha düşük, “All-S” çalışmasının medyan puanı ise daha yüksektir. Bu durum meta etiket öznitelikleri, içerik öznitelikleri, bağlantı ve resim öznitelikleri ve URL özniteliklerinin ortalama sistem başarımına katkısı olduğunu göstermektedir. CW12A sorgu seti için bağlantı ve resim öznitelikleri ve benzerlik öznitelikleri ortalama sistem başarımına katkıda bulunmaktadır. MQ09 sorgu seti için katkıda bulunan öznitelik grupları meta etiket öznitelikleri, içerik öznitelikleri ve benzerlik öznitelikleri iken NTCIR sorgu seti için ise meta etiket öznitelikleri, bağlantı ve resim öznitelikleri ve benzerlik öznitelikleridir. Ancak şekilde çalışmaları gösteren kutulardaki çentikler incelendiğinde çentiklerin birbirleriyle örtüştükleri ve öznitelik gruplarının etkilerinin farklılaşmadığını gözlemlenmektedir.

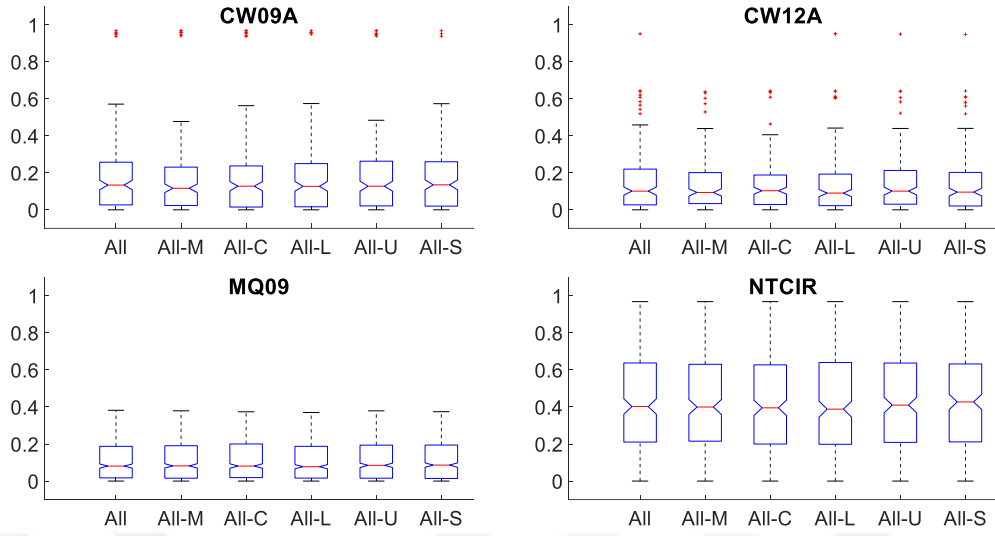
LambdaMART ile üretilen çalışmaların diğer değerlendirme ölçütleriyle (nDCG@20, ERR@10, ERR@20 ve MAP) elde edilen ve RankBoost ve SVM^{rank} ile üretilen çalışmaların tüm değerlendirme ölçütleriyle elde edilen puanlarının tek yönlü varyans analizlerinin kutu grafiği gösterimleri bölümün kalan kısmında sunulmuştur (Şekil 7.17-Şekil 7.30). Şekillerin yorumları Şekil 7.16 ile benzerlik göstermektedir ve SEO-tabanlı öznitelik gruplarının ortalama sistem başarımına katkıları arasında bir fark gözlemlenmemiştir.



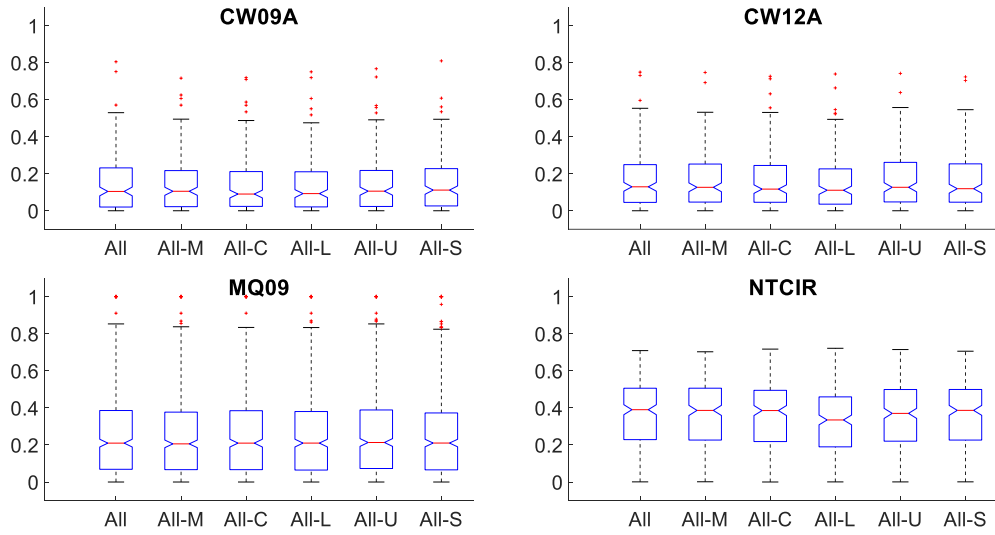
Şekil 7.17. LambdaMART ile üretilen çalışmaların nDCG@20 sonuçları için tek yönlü varyans analizleri



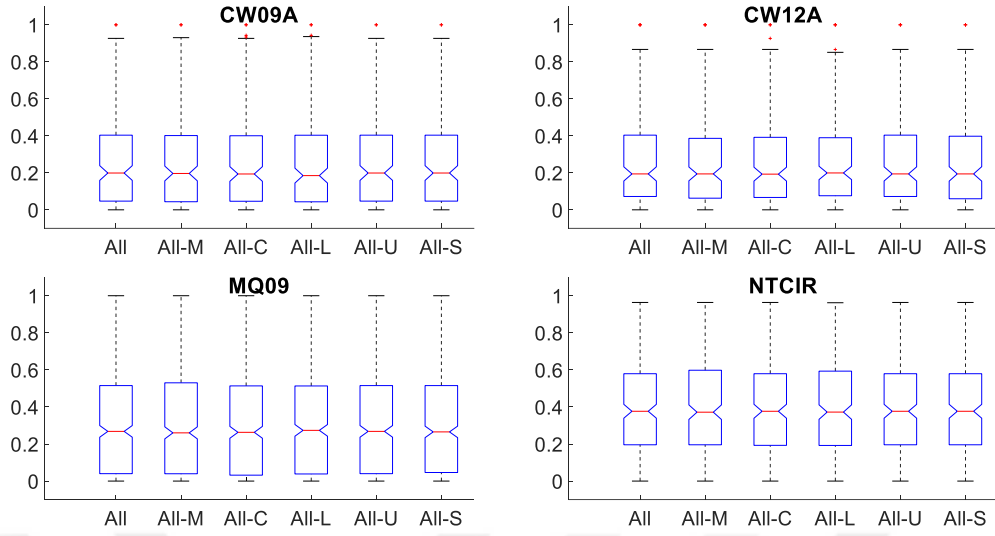
Şekil 7.18. LambdaMART ile üretilen çalışmaların ERR@10 sonuçları için tek yönlü varyans analizleri



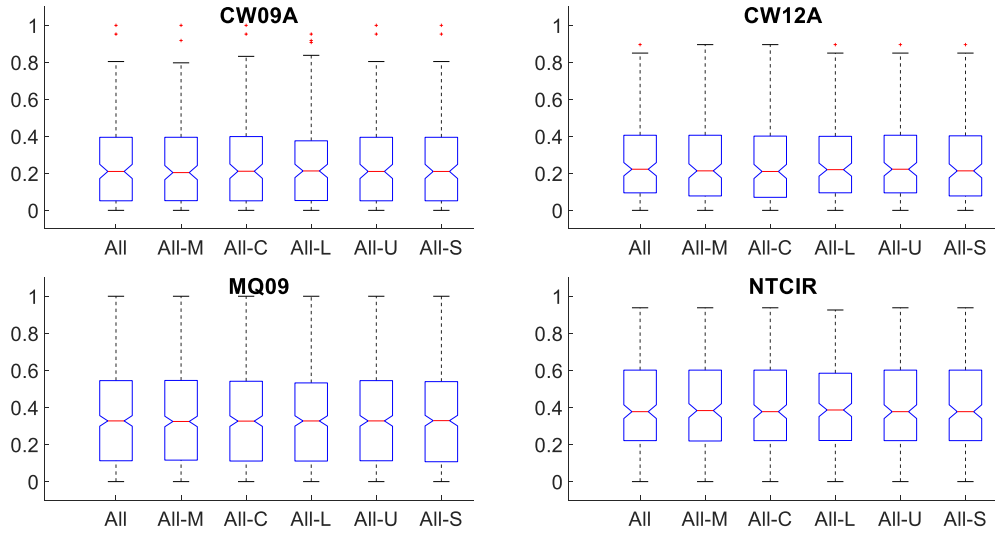
Şekil 7.19. LambdaMART ile üretilen çalışmaların ERR@20 sonuçları için tek yönlü varyans analizleri



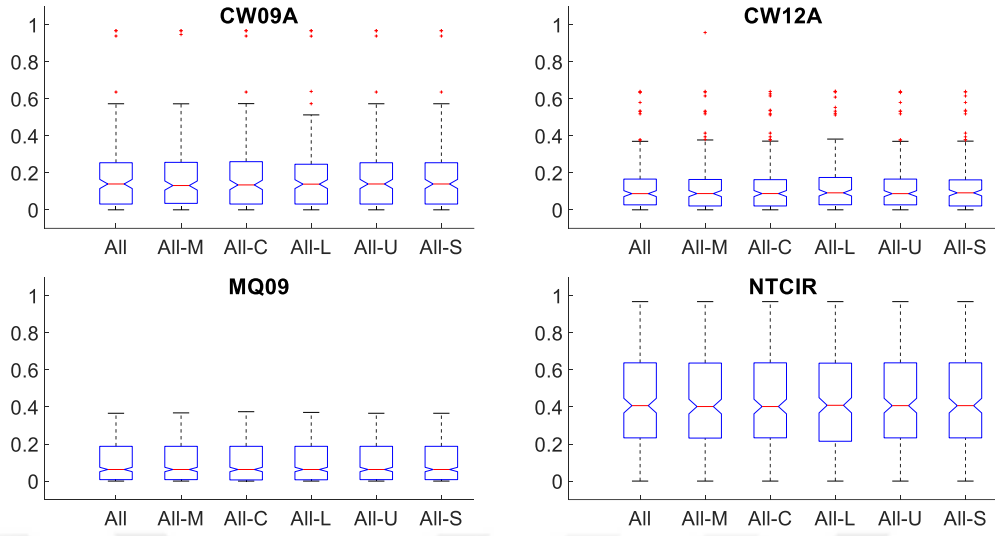
Şekil 7.20. LambdaMART ile üretilen çalışmaların MAP sonuçları için tek yönlü varyans analizleri



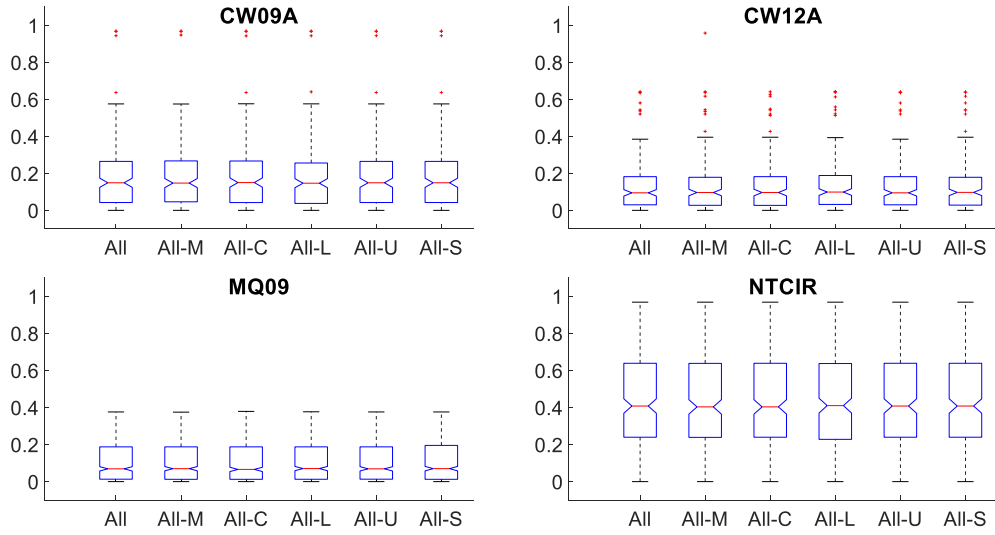
Şekil 7.21. RankBoost ile üretilen çalışmaların $nDCG@10$ sonuçları için tek yönlü varyans analizleri



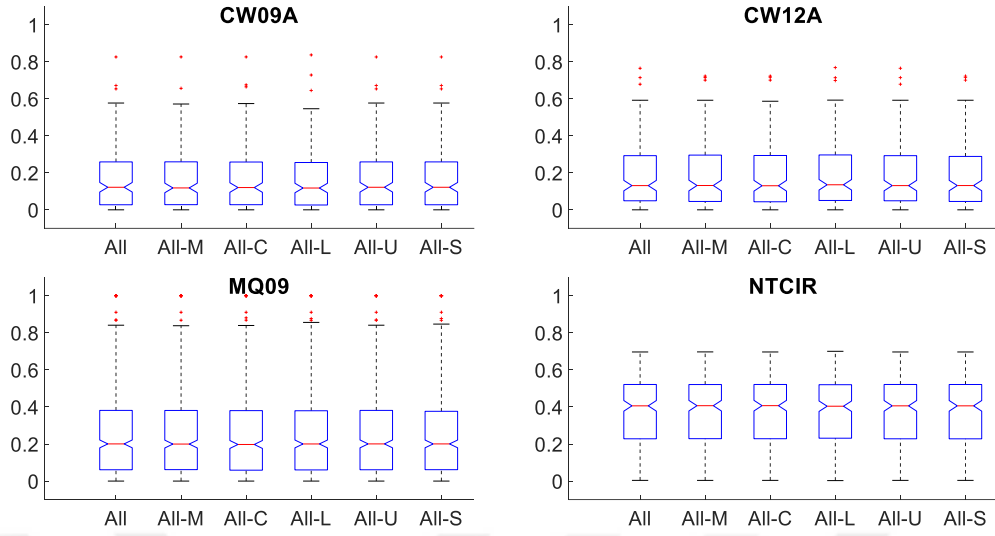
Şekil 7.22. RankBoost ile üretilen çalışmaların $nDCG@20$ sonuçları için tek yönlü varyans analizleri



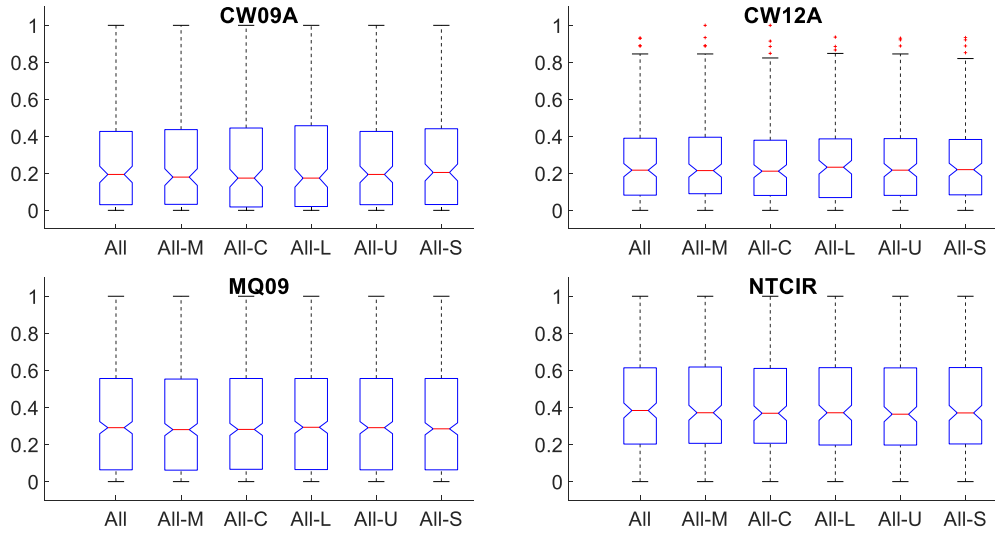
Şekil 7.23. RankBoost ile üretilen çalışmaların ERR@10 sonuçları için tek yönlü varyans analizleri



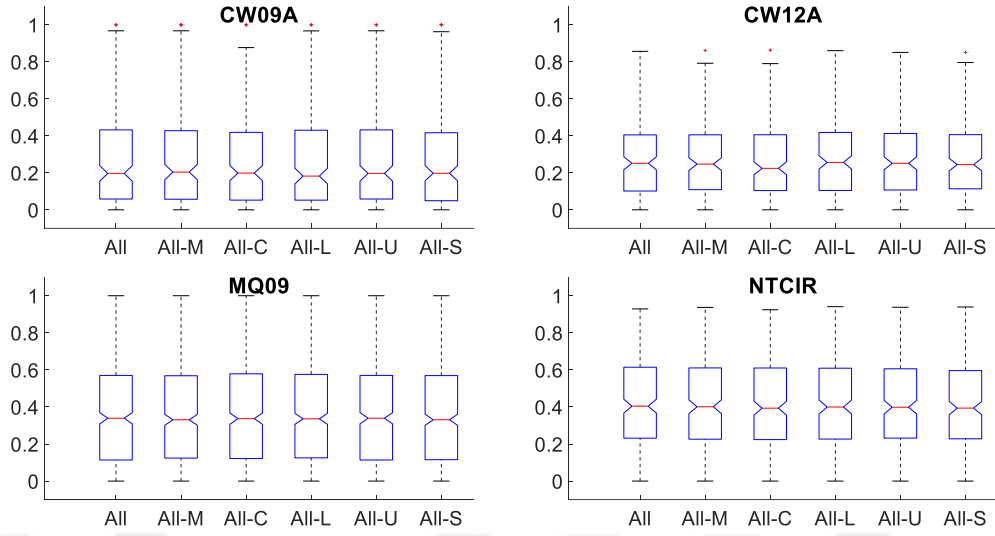
Şekil 7.24. RankBoost ile üretilen çalışmaların ERR@20 sonuçları için tek yönlü varyans analizleri



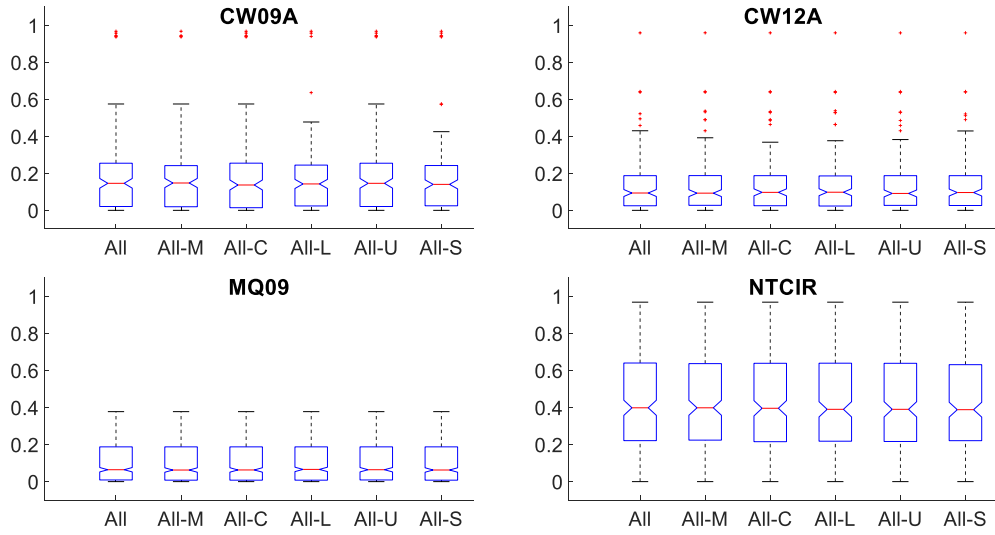
Şekil 7.25. RankBoost ile üretilen çalışmaların MAP sonuçları için tek yönlü varyans analizleri



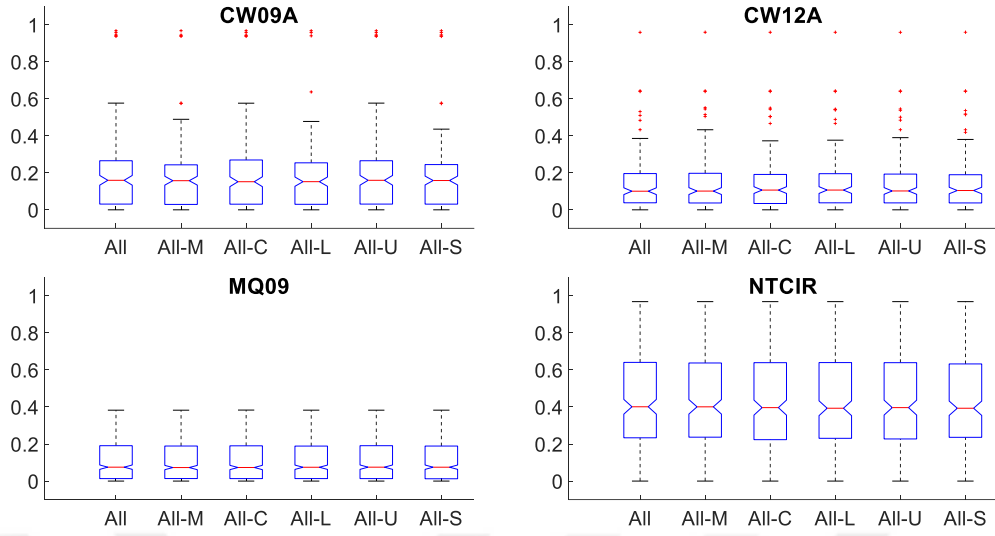
Şekil 7.26. SVM^{rank} ile üretilen çalışmaların nDCG@10 sonuçları için tek yönlü varyans analizleri



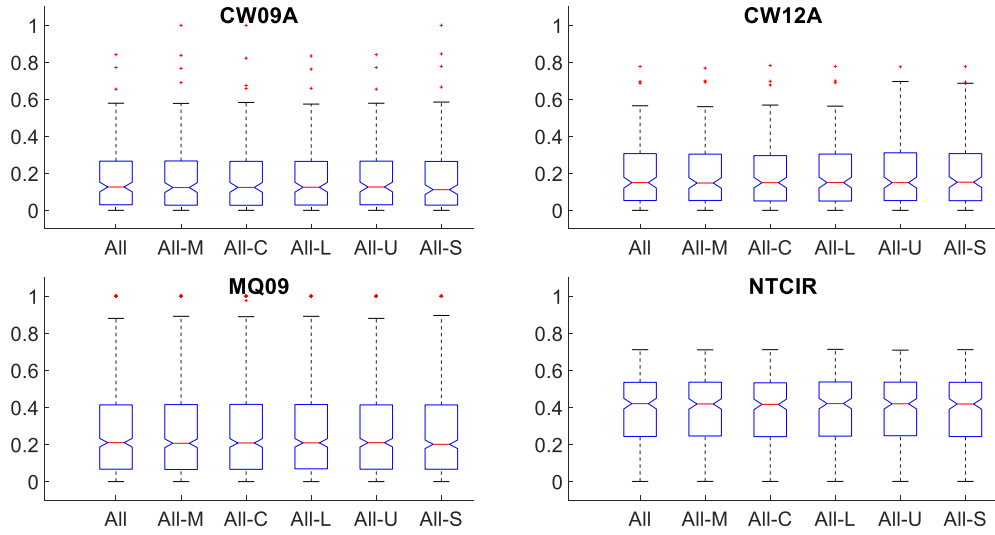
Şekil 7.27. SVM^{rank} ile üretilen çalışmaların $nDCG@20$ sonuçları için tek yönlü varyans analizleri



Şekil 7.28. SVM^{rank} ile üretilen çalışmaların $ERR@10$ sonuçları için tek yönlü varyans analizleri



Şekil 7.29. SVM^{rank} ile üretilen çalışmaların ERR@20 sonuçları için tek yönlü varyans analizleri



Şekil 7.30. SVM^{rank} ile üretilen çalışmaların MAP sonuçları için tek yönlü varyans analizleri

8. SONUÇLAR VE GELECEK ÇALIŞMALAR

Web bilgi erişimi çalışmaları çoğunlukla sorgu ve web belgeleri arasındaki ilişkiye odaklanmakta ve belge kalitesinin bilgi erişim sistemi başarımı üzerindeki etkisini göz ardı etmektedir. Önceki çalışmalarda web belgesi kalite öznitelikleri önerilmiş olmasına rağmen web belgeleri için yeni belge kalitesi öznitelikleri oluşturma konusu ticari arama motorları için önem arz ederken bu konuya akademik ilgi nispeten daha azdır.

Bu çalışmada, ticari arama motorlarının web sayfalarının kalitesi hakkında bilgi edinmek için uyguladıkları veya uyguladıkları düşünülen uygulamalardan ilham alınarak Web bilgi erişim sistemleri için SEO-tabanlı öznitelikler türetilmiştir. Belge kalitesini ölçmeyi amaçlayan bu özniteliklerin LTR içeren bir bilgi erişim sisteminin başarımına olan etkisi gözlemlenmiştir. Bunun için deneylerde “state-of-the-art” LTR tekniği olan LambdaMART ve deneylerin genellenebilirliğini göstermek için RankBoost ve SVM^{rank} olmak üzere iki farklı LTR tekniği daha kullanılmıştır. Deneyler mevcut en büyük ve güncel web belge koleksiyonları olan ClueWeb09 ve ClueWeb12 koleksiyonları üzerinde ve bu koleksiyonları hedef alan ilgili TREC ve NTCIR sorgu setleri ile gerçekleştirilmiştir. Deneylerde referans çalışmalar (baseline runs) olarak, ilgili LTR metodu tarafından LTR deneylerinde standart olarak kullanılan sorgu-bağımlı ve sorgu-bağımsız özniteliklerinden oluşturulan referans öznitelik kümesini kullanarak üretilen çalışmalar kullanılmıştır. Türetilen SEO-tabanlı özniteliklerin etkinliğini göstermek amacıyla referans çalışmaların üretilme süreci ile aynı şekilde ve sadece referans öznitelik kümesine SEO-tabanlı öznitelikler eklenerek yeni çalışmalar üretilmiş ve bu çalışmalar referans çalışmalarla bilgi erişim sistemi etkinliği (retrieval effectiveness) üzerinden farklı değerlendirme ölçütleriyle karşılaştırılmıştır.

Deney sonuçları bu çalışmada türetilen SEO özniteliklerinin LTR tekniklerine başarılı bir şekilde entegre edilebileceğini ve standart öznitelikleri kullanan bilgi erişim sistemlerinin başarımında artışa yol açtığını göstermektedir. Deneylerin güncel ve farklı veri setleri ile ve farklı LTR teknikleriyle benzer sonuçlar vermesi türetilen özniteliklerin etkilerinin genellenebilirliğini göstermektedir. Bununla birlikte deney sonuçlarında ClueWeb09-CategoryA koleksiyonunu hedef alan çalışmalarda SEO-tabanlı özniteliklerin sistem başarımını diğer çalışmalara oranla daha fazla artırdığı gözlemlenmiştir. Bölüm 6.1’de belirtilen web belgesi koleksiyonları özellikleri göz önünde bulundurulduğunda ClueWeb09-CategoryA koleksiyonunun diğer

koleksiyonlara oranla daha fazla düşük kaliteli web sayfası içermesi, türetilen SEO özniteliklerinin web belge kalitesini ölçmedeki başarısını göstermektedir.

Deney sonuçlarında, bu çalışmada tanıtılan SEO-tabanlı özniteliklerden faydalanılarak elde edilen çalışmalarla (Ltr*) bu öznitelikleri kullanmadan elde edilen çalışmalar (Ltr) arasındaki bilgi erişim sistemi başarımları farkları sorgu bazında da incelenmiştir. Bu inceleme sonucunda SEO-tabanlı özniteliklerin deneylerde kullanılan sorguların büyük bir çoğunluğunda sistem başarımına katkıda bulunduğu gözlemlenmiştir. Ayrıca türetilen SEO-tabanlı öznitelikler 5 gruba ayrılmış ve her bir grubun sistem başarımına etkisi incelenmiştir. Bunun için tüm öznitelikleri (standart LTR öznitelikleri ve SEO-tabanlı öznitelikler) içeren öznitelik kümesinden sırayla SEO öznitelik grupları çıkarılarak elde edilen çalışmalar karşılaştırılmıştır. Deney sonuçlarında çalışmalar arasında istatistiksel olarak anlamlı bir farka rastlanmamıştır. Dolayısıyla bu çalışmada türetilen SEO-tabanlı öznitelik gruplarının sistem başarımına eşit derecede katkı sağladığı söylenebilir.

Bu çalışmada tanıtılan öznitelikleri kullanan bir bilgi erişim sisteminin başarımı NTCIR-15 organizasyonunun WWW-3 English alt görevinde test edilmiştir (Aydın, Göksel, Arslan, & Dinçer, 2020). Alt görevde katılımcı gruplar, organizasyon tarafından belirlenen bir sorgu seti (WWW3) için belirli bir web belge koleksiyonundaki (ClueWeb12-B13) alakalı belgeleri sıralayarak çalışmalar üretmekte ve bu çalışmaların çeşitli değerlendirme ölçütü sonuçları karşılaştırılmaktadır. Alt görevin resmi açıklanan sonuçlarında SEO-tabanlı öznitelikleri kullanan bir bilgi erişim sistemi tarafından üretilen çalışmaların elde ettiği sıralamalar gösterilmektedir (Sakai, ve diğerleri, 2020; Sakai, ve diğerleri, 2022).

8.1. Çalışma Kısıtları

Cranfield paradigmasının kısıtları bu çalışmanın kısıtları arasında da yer almaktadır. Bunlardan en bilineni sorgu-belge alaka yargılarının oluşturulma yöntemidir. Bu yargılar insanlar tarafından oluşturulmaktadır. Bir sorgu-belge çifti için birden fazla insan tarafından alaka yargıları oluşturulmakta ve bunlar çeşitli metotlarla birleştirilmektedir. Ancak bu durum hem insan faktörünün doğruluk kısıtlarını hem maliyet kısıtlarını beraberinde getirmektedir. Maliyeti azaltmak gibi sebeplerle “pooling” denilen teknikle bir sorgu için tüm belgeler yerine değişik bilgi erişim sistemleri tarafından üst sıralarda getirilen belgelerin birleşimindeki belgeler değerlendirilmektedir

(Sakai, Tao, & Zeng, 2022). Dolayısıyla tüm sorgular ve belge koleksiyonu düşünüldüğünde alaka yargısı bulunan sorgu-belge çifti sayısı oldukça azdır (Buckley, Dimmick, Soboroff, & Voorhees, 2006).

Çalışma kısıtlarından bir diğeri, kullanılan web belge koleksiyonlarından çevrimdışı olmasından ve Web'e dair tüm bilgileri içermemesinden kaynaklanmaktadır. Ticari arama motorları (Google, Yahoo!, Yandex, Bing, DuckDuckGo, YaCy vb.), web arama görevlerinde, web belgelerinin kalitesini ölçmede birçok zorlukla karşılaşmakta ve farklı çözüm yolları geliştirmektedirler. Ancak bu yolların birçoğu test koleksiyon tabanlı değerlendirmede (Cranfield paradigması) taklit edilememektedir. Örneğin, web sayfasında bulunun bozuk bağlantılar (broken links) veya web sayfasının yüklenme hızı gibi özellikler SEO uygulamalarında sıklıkla kullanılırken, web belge koleksiyonlarında bu bilgilere ulaşılamamaktadır. Bunun yanı sıra bu çalışmada kullanılan web belge koleksiyonları mevcut güncel koleksiyonlar olmasına rağmen, 2009 ve 2012 yıllarında oluşturulmuşlardır. Günümüze kadar olan süreçte web teknolojilerinde gelişmeler olmuş, web sayfalarında yeni özellikler kullanılmaya başlanmıştır. SEO uygulamalarında da kullanılan bu özellikler mevcut koleksiyonlarda yok veya düşük oranlarda bulunmaktadır. Örneğin, ClueWeb12-B13 koleksiyonunda anlamsal etiketler içeren web sayfası oranı yaklaşık %4'tür (Çıplak, Aydın, & Arslan, 2019).

8.2. Gelecek Çalışmalar

Bu çalışma, gelecek birçok çalışmaya öncülük edebilecek kapasitededir. Gelecek çalışma olasılıklarından ilki, web belgelerine bu çalışmada tanıtılan ön belge özelliklerinden türetilcek tekil bir kalite puanı atanmasıdır. Literatürdeki benzer çalışmaların (Cormack, Smucker, & Clarke, 2011; Page, Brin, Motwani, & Winograd, 1999) uygulama yolları takip edilerek belge kalite puanlarına göre web belge koleksiyonu filtrelenebilir veya sonuç listesi tekrar sıralanabilir. Bu çalışmada kullanılan ClueWeb09 ve ClueWeb12 koleksiyonlarına ilave olarak yakın zamanda ClueWeb22 web belge koleksiyonu kullanıma sunulmuştur. Selefî koleksiyonlara göre çok daha fazla belge içeren (10 milyar web sayfası) ClueWeb22 koleksiyonu, 2022 yılının ilk yarısında dizinlenen web belgelerinden oluşturulduğundan diğer koleksiyonlara göre daha güncel web sayfalarını içerdiği düşünülmektedir. Koleksiyonların elde edilme süreleri arasında web sayfası oluşturma aşamasında kullanmak üzere tanıtılan yeni HTML özellikleri de incelenerek yeni ve daha kapsamlı ön belge özellikleri kümesi oluşturulabilir ve tekil

kalite puanı hesaplamalarına entegre edilebilir. Bununla birlikte elde edilen kalite puanı mevcut kalite puanları (Cormack, Smucker, & Clarke, 2011; Page, Brin, Motwani, & Winograd, 1999) ile karşılaştırılabilir etkinliği literatürde olduğu gibi farklı web belge koleksiyonlarında analiz edilebilir (Yılmazel & Arslan, 2019).

Bir diğer gelecek çalışma olasılığı da bu çalışmada tanıtılan ön belge özelliklerinin farklı görevler için de kullanılabilmesidir. Bu görevlerden ilki spam sınıflandırmasıdır. Bir web sayfasının düşük kaliteli olması ile spam olma olasılığı arasında bir ilişki vardır (Cormack, Smucker, & Clarke, 2011). Dolayısıyla kalite ölçmede etkin sonuçlar veren web sayfası özelliklerinin spam sınıflandırma görevinde de etkili olması olasıdır. Bir gelecek çalışma olasılığı da elde edilen kalite ölçütün, sonuç listelerinde yakın kopya (near duplicate) web belgelerinin elenmesinde veya sıralamasında kullanılmasıdır (Khoshdel Nikkhoo, 2011; Fröbe, ve diğerleri, 2021). Bunun yanı sıra elde edilen ölçüt, arama sistemlerinin verimini (efficiency) artırma adına uygulanan kesin olmayan belge erişimi uygulamasında şampiyon listelerin oluşturulmasında statik kalite puanı olarak kullanılabilir (Manning, Raghavan, & Schütze, 2008).

KAYNAKÇA

- Amati, G. (2006). Frequentist and bayesian approach to information retrieval. *European Conference on Information Retrieval*, (s. 13–24).
- Amati, G. (2009). *Divergence from Randomness Models*. Springer.
- Amati, G., & van Rijsbergen, C. J. (2002). Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Transactions on Information Systems (TOIS)*, 20, 357–389.
- Amento, B., Terveen, L., & Hill, W. (2000). Does “authority” mean quality? Predicting expert quality ratings of Web documents. *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, (s. 296–303).
- Arslan, A. (2020). On the usefulness of HTML meta elements for web retrieval. *Eskişehir Technical University Journal of Science and Technology A-Applied Sciences and Engineering*, 21(1), 182-198.
- Aydın, A., Göksel, G., Arslan, A., & Dinçer, B. T. (2020). ESTUCeng at the NTCIR-15 WWW-3 Task: Experimenting with new document quality features. *Proceedings of NTCIR-15*.
- Baeza-Yates, R., & Ribeiro-Neto, B. (1999). *Modern Information Retrieval*.
- Bendersky, M., Croft, W. B., & Diao, Y. (2011). Quality-biased ranking of web documents. *Proceedings of the fourth ACM international conference on Web search and data mining*, (s. 95–104).
- Berners-Lee, T., Cailliau, R., Luotonen, A., Nielsen, H., & Secret, A. (1994). The world-wide web : Internet technology. *Communications of The ACM*, 37(8), 76-82.
- Białecki, A., Muir, R., & Ingersoll, G. (2012). Apache Lucene 4. *Proceedings of the SIGIR 2012 Workshop on Open Source Information Retrieval*, (s. 17–24). Portland.
- Buckland, M. (2006). *Emanuel Goldberg and his knowledge machine : information, invention, and political forces*.
- Buckley, C., Dimmick, D., Soboroff, I., & Voorhees, E. (2006). Bias and the limits of pooling. *SIGIR '06: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (s. 619–620). New York, NY, USA: ACM Press.
- Burges, C. J. (2010). From RankNet to LambdaRank to LambdaMART: An Overview. *Learning*, 11, 81.
- Bush, V. (1945). *As We May Think*.
- Cao, Y., Xu, J., Liu, T.-Y., Li, H., Huang, Y., & Hon, H.-W. (2006). Adapting ranking SVM to document retrieval. *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, (s. 186-193).

- Carmel, D., & Yom-Tov, E. (2010). *Estimating the query difficulty for information retrieval*. Morgan & Claypool Publishers.
- Carterette, B., Kanoulas, E., & Yilmaz, E. (2012). Evaluating Web Retrieval Effectiveness. *Web Search Engine Research* (Cilt 4, s. 105-137).
- Carterette, B., Pavlu, V., Fang, H., & Kanoulas, E. (2009). Million Query Track 2009 Overview. *Proceedings of the 18th text retrieval conference*. Gaithersburg.
- Chapelle, O., & Chang, Y. (2011). Yahoo! learning to rank challenge overview. *Proceedings of the learning to rank challenge*, (s. 1–24).
- Clarke, C. L., Craswell, N., & Soboroff, I. (2009). Overview of the TREC 2009 Web Track. *Proceedings of the 18th text retrieval conference*. Gaithersburg.
- Cleverdon, C. (1991). The significance of the Cranfield tests on index languages. *Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval*, (s. 3-12).
- Cleverdon, C. (1997). *The Cranfield tests on index language devices* (Cilt 19).
- Clinchant, S., & Gaussier, É. (2011). Retrieval constraints and word frequency distributions a log-logistic model for IR. *Information Retrieval*, 14(1), 5-25.
- Collins-Thompson, K., Bennett, P., Diaz, F., Clarke, C., & Voorhees, E. (2014). TREC 2013 Web Track Overview. *Proceedings of the 22th text retrieval conference*. Gaithersburg.
- Cormack, G. V., Smucker, M. D., & Clarke, C. L. (2011). Efficient and effective spam filtering and re-ranking for large web datasets. *Information Retrieval*, 14, 441-465. doi:10.1007/s10791-011-9162-z
- Craswell, N., Robertson, S., Zaragoza, H., & Taylor, M. (2005). Relevance weighting for query independent evidence. *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, (s. 416–423).
- Croft, W. B., Metzler, D., & Strohman, T. (2010). *Search engines: Information retrieval in practice*. Addison-Wesley Reading.
- Cronen-Townsend, S., Zhou, Y., & Croft, W. (2002). Predicting query performance. *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, (s. 299-306). Tampere.
- Çıplak, G., Aydın, A., & Arslan, A. (2019). The usage statistics of new HTML5 semantic elements in the ClueWeb12 dataset. *International Conference on Advanced Technologies, Computer Engineering and Science (ICATCES 2019)*, (s. 173–176).
- Dadachev, B. (2015). *On the Helmholtz principle for text mining*. Ph.D. dissertation, Cardiff University.
- Ensan, F., Bagheri, E., Zouaq, A., & Kouznetsov, A. (2017). An empirical study of embedding features in learning to rank. *CIKM '17: ACM Conference on Information and Knowledge Management*, (s. 2059-2062).

- Fan, Y., Guo, J., Lan, Y., Xu, J., Pang, L., & Cheng, X. (2017). Learning visual features from snapshots for web search. *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, (s. 247–256).
- Freund, Y., Iyer, R., Schapire, R., & Singer, Y. (2003). An efficient boosting algorithm for combining preferences. *Journal of machine learning research*, 4, 933–969.
- Fröbe, M., Bevendorff, J., Gienapp, L., Völske, M., Stein, B., Potthast, M., & Hagen, M. (2021). CopyCat: Near-Duplicates within and between the ClueWeb and the Common Crawl. *The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, (s. 2398–2404).
- Fuchs, C. (2014). *Social media: a critical introduction*. Sage.
- Harman, D. (2019). Information Retrieval: The Early Years. *Foundations and Trends in Information Retrieval*, 13(5), 425–577.
- Harter, S. P. (1975). A probabilistic approach to automatic keyword indexing. Part I: On the distribution of specialty words in a technical literature. *Journal of the American Society for Information Science (JASIS)*, 26, 197–216.
- Hauff, C., Azzopardi, L., & Hiemstra, D. (2009). The combination and evaluation of query performance prediction methods. *European Conference on Information Retrieval*, (s. 301–312).
- He, B., & Ounis, I. (2006). Query performance prediction. *Information Systems*, 31, 585–594. doi:10.1016/j.is.2005.11.003
- Hiemstra, D. (2009). Information Retrieval Models. *Information Retrieval: Searching in the 21st Century* (s. 1-19). Wiley.
- Järvelin, K., & Kekäläinen, J. (2002). Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems (TOIS)*, 20, 422–446.
- Jelinek, F. (1980). Interpolated estimation of Markov source parameters from sparse data. *Proc. Workshop on Pattern Recognition in Practice, 1980*, 381–397.
- Joachims, T. (2006, 8). Training linear SVMs in linear time. *Proceedings of the SIGIR 2012 Workshop on Open Source Information Retrieval*, (s. 217–226). Philadelphia.
- Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 60(1), 493–502.
- Kamps, J., Mishne, G., & de Rijke, M. (2004). Language Models for Searching in Web Corpora. *TREC*.
- Khan, M., & Mahmood, A. (2018). A distinctive approach to obtain higher page rank through search engine optimization. *Sadhana*, 43, 1–12.
- Khoshdel Nikkhoo, H. (2011). *The impact of near-duplicate documents on information retrieval evaluation*. Master Thesis, University of Waterloo.

- Killoran, J. B. (2013). How to use search engine optimization techniques to increase website visibility. *IEEE Transactions on professional communication*, 56, 50-66.
- Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46, 604–632.
- Kocabaş, İ., Dinçer, B. T., & Karaoğlu, B. (2014). A nonparametric term weighting method for information retrieval based on measuring the divergence from independence. *Information retrieval*, 17, 153–176.
- Kraaij, W., Westerveld, T., & Hiemstra, D. (2002). The importance of prior probabilities for entry page search. *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, (s. 27–34).
- Krovetz, R. (1993). Viewing Morphology As an Inference Process. *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (s. 191–202). Pittsburgh: ACM.
- Liu, T.-Y. (2009). Learning to Rank for Information Retrieval. *Foundations and Trends in Information Retrieval*, 3, 225–331. doi:10.1561/15000000016
- Liu, T.-Y. (2011). *Learning to rank for information retrieval*. Springer Science & Business Media.
- Luhn, H. (1957). A statistical approach to mechanized encoding and searching of literary information. *Ibm Journal of Research and Development*, 1(4), 309-317.
- Luhn, H. (1958). The automatic creation of literature abstracts. *Ibm Journal of Research and Development*, 2(2), 159-165.
- Luo, C., Sakai, T., Liu, Y., Dou, Z., Xiong, C., & Xu, J. (2017). Overview of the NTCIR-13 We Want Web Task. *Proceedings of NTCIR-13*, (s. 394-401).
- Macdonald, C., Dinçer, B. T., & Ounis, I. (2015). Transferring learning to rank models for web search. *Proceedings of the 2015 international conference on the theory of information retrieval*, (s. 41–50).
- Macdonald, C., Santos, R. L., & Ounis, I. (2012). On the usefulness of query features for learning to rank. *Proceedings of the 21st ACM international conference on Information and knowledge management*, (s. 2559–2562).
- Macdonald, C., Santos, R. L., & Ounis, I. (2013). The whens and hows of learning to rank for web search. *Information Retrieval*, 16, 584–628.
- Macdonald, C., Santos, R. L., Ounis, I., & He, B. (2013). About learning models with multiple query-dependent features. *ACM Transactions on Information Systems (TOIS)*, 31, 1–39.
- MacKay, D., & Peto, L. (1995). A hierarchical Dirichlet language model. *Natural Language Engineering*, 1(3), 289-308.
- Manning, C., Raghavan, P., & Schütze, H. (2008). *Introduction to Information Retrieval*.
- McCandless, M., Hatcher, E., & Gospodnetić, O. (2010). *Lucene in action* (Cilt 2). Manning Greenwich.

- Mooers, C. (1950). Information retrieval viewed as temporal signaling. *Proceedings of the international congress of mathematicians*, 1, s. 572-573.
- Ney, H., Essen, U., & Kneser, R. (1994). On structuring probabilistic dependences in stochastic language modelling. *Computer Speech & Language*, 8(1), 1-38.
- Page, L., Brin, S., Motwani, R., & Winograd, T. (1999). *The PageRank Citation Ranking: Bringing Order to the Web*. Stanford InfoLab. Stanford InfoLab.
- Pera, M. S., & Ng, Y.-K. (2008). Identifying spam web pages based on content similarity. *International Conference on Computational Science and Its Applications*, (s. 204–219).
- Ponte, J. M., & Croft, W. B. (1998). A Language Modeling Approach to Information Retrieval. *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (s. 275–281). New York, NY, USA: Association for Computing Machinery. doi:10.1145/290941.291008
- Popper, K. (1995). *The Logic of Scientific Discovery*. Routledge, London.
- Qin, T., & Liu, T.-Y. (2013). Introducing LETOR 4.0 Datasets. *CoRR*, abs/1306.2597.
- Qin, T., Liu, T.-Y., Xu, J., & Li, H. (2010). LETOR: A benchmark collection for research on learning to rank for information retrieval. *Information Retrieval*, 13, 346–374.
- Qin, T., Zhang, X.-D., Tsai, M.-F., Wang, D.-S., Liu, T.-Y., & Li, H. (2008). Query-level loss functions for information retrieval. *Information Processing and Management*, 44(2), 838-855.
- Richardson, M., Prakash, A., & Brill, E. (2006, May). Beyond PageRank: machine learning for static ranking. *Proceedings of the 15th international conference on World Wide Web*, (s. 707-715). Edinburgh.
- Robbins, J. N. (2012). *Learning Web Design: A beginner's guide to HTML, CSS, JavaScript, and web graphics*. O'Reilly Media, Inc.
- Robertson, S. E. (1977). The Probability Ranking Principle in IR. *Journal of Documentation*, 33, 294–304. doi:10.1108/eb026647
- Robertson, S. E., & Sparck Jones, K. (1976). Relevance weighting of search terms. *Journal of the American Society for Information Science (JASIS)*, 129–146.
- Rocchio, J. (1971). Relevance feedback in information retrieval. *The Smart Retrieval System-Experiments in Automatic Document Processing*, 313-323.
- Sakai, T., Tao, S., & Zeng, Z. (2022). Relevance Assessments for Web Search Evaluation: Should We Randomise or Prioritise the Pooled Documents? *ACM Transactions on Information Systems (TOIS)*, 40, 1-35.
- Sakai, T., Tao, S., Maistro, M., Chu, Z., Li, Y., Chen, N., . . . Liu, Y. (2022). Corrected Evaluation Results of the NTCIR WWW-2, WWW-3, and WWW-4 English Subtasks. *arXiv preprint arXiv:2210.10266*.

- Sakai, T., Tao, S., Zeng, Z., Zheng, Y., Mao, J., Chu, Z., . . . Soboroff, I. (2020). Overview of the NTCIR-15 We Want Web with CENTRE (WWW-3) Task. *Proceedings of NTCIR-15*.
- Salton, G. (1971). *The SMART Retrieval System—Experiments in Automatic Document Processing*.
- Santos, R. L., Macdonald, C., & Ounis, I. (2011). Effectiveness beyond the first crawl tier. *Proceedings of the 20th ACM international conference on Information and knowledge management*, (s. 1937–1940).
- Singhal, A., Salton, G., Mitra, M., & Buckley, C. (1996). Document length normalization. *Information Processing & Management*, 32(5), 619-633.
- Tonta, Y. (1995). Bilgi Erişim Sistemleri. *Türk Kütüphaneciliği*, 9(3), 302-314.
- Trotman, A., & Kitchen, V. (2022). Quality metrics for search engine deterministic sort orders. *Information Processing and Management*, 59(6).
- van den Akker, B., Markov, I., & de Rijke, M. (2019). ViTOR: learning to rank webpages based on visual features. *The World Wide Web Conference*, (s. 3279–3285).
- Voorhees, E. (2007). TREC: Continuing information retrieval's tradition of experimentation. *Communications of The ACM*, 50(11), 51-54.
- Voorhees, E. (2019). The Evolution of Cranfield. *Information Retrieval Evaluation in a Changing World*, 45-69.
- Voorhees, E. M. (2002). The philosophy of information retrieval evaluation. *Evaluation of Cross-Language Information Retrieval Systems. Proceedings of CLEF 2001, LNCS 2406*, (s. 355–370).
- Voorhees, E., & Harman, D. (2005). *TREC: Experiment and Evaluation in Information Retrieval (Digital Libraries and Electronic Publishing)*.
- Wilkie, C., & Azzopardi, L. (2014). Best and Fairest: An Empirical Analysis of Retrieval System Bias. *ECIR 2014 Proceedings of the 36th European Conference on IR Research on Advances in Information Retrieval - Volume 8416*, (s. 13-25).
- Yilmaz, E., Verma, M., Mehrotra, R., Kanoulas, E., Carterette, B., & Craswell, N. (2015). Overview of the TREC 2015 Tasks Track. In: Voorhees, EM and Ellis, A, (eds.) *National Institute of Standards and Technology (NIST) (2015)*.
- Yilmazel, İ. B., & Arslan, A. (2019). An intrinsic evaluation of the Waterloo spam rankings of the ClueWeb09 and ClueWeb12 datasets. *Journal of Information Science*, 41-57.
- Zhai, C., & Lafferty, J. (2004). A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems (TOIS)*, 22, 179–214.
- Zhao, Y., Scholer, F., & Tsegay, Y. (2008). Effective pre-retrieval query performance prediction using similarity and variability evidence. *European conference on information retrieval*, (s. 52–64).

Zhou, Y., & Croft, W. B. (2005). Document quality models for web ad hoc retrieval. *Proceedings of the 14th ACM international conference on Information and knowledge management*, (s. 331–332).

Zhu, X., & Gauch, S. (2000). Incorporating quality metrics in centralized/distributed information retrieval on the World Wide Web. *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, (s. 288–295).

http-1: <https://www.internetlivestats.com/total-number-of-websites> (Erişim Tarihi: 15.10.2022)

http-2: <https://backlinko.com/google-ctr-stats> (Erişim Tarihi: 15.10.2022)

http-3: <https://github.com/iorixxx/lucene-clueweb-retrieval> (Erişim Tarihi: 15.10.2022)



ÖZGEÇMİŞ

ORCID NO: 0000-0001-8035-1932

Ad Soyad : Ahmet Aydın

Yabancı Dil : İngilizce

Eğitim ve Mesleki Geçmişi:

- 2015-, Eskişehir Teknik Üniversitesi, Lisansüstü Eğitim Enstitüsü, Bilgisayar Mühendisliği, Bilgisayar Bilimleri (DR)
- 2012-2015, Anadolu Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği, Bilgisayar Bilimleri (YL) (TEZLİ)
- 2005-2010, Anadolu Üniversitesi, Mühendislik-Mimarlık Fakültesi, Bilgisayar Mühendisliği, Bilgisayar Mühendisliği Pr. (İngilizce)
- 2008-2012, Anadolu Üniversitesi, İşletme Fakültesi, İşletme Bölümü, İşletme Pr. (Açıköğretim)
- 2020-2021, Anadolu Üniversitesi, Açıköğretim Fakültesi, Hukuk Bölümü, Adalet Pr. (Açıköğretim)
- 2018-, Araştırma Görevlisi, Eskişehir Teknik Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Bilgisayar Bilimleri Anabilim Dalı
- 2012-2018, Araştırma Görevlisi, Anadolu Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Bilgisayar Bilimleri Anabilim Dalı
- 2012-2015, Yazılım Mühendisi, Anadolu Üniversitesi, Bilgisayar Araştırma ve Uygulama Merkezi (BAUM)

Yayınları ve/veya Bilimsel/Sanatsal Faaliyetleri:

- AYDIN AHMET, GÖKSEL GÖKHAN, ARSLAN AHMET, DİNÇER BEKİR TANER (2020). ESTUCeng at the NTCIR-15 WWW-3 Task: Experimenting with new document quality features. NTCIR 15 Conference: Proceedings of the 15th NTCIR Conference on Evaluation of Information Access Technologies, December 8-11, 2020 Tokyo Japan, 253-257. (Tam Metin Bildiri/Sözlü Sunum) (Yayın No:6649298)
- ÇIPLAK GÖKHAN, AYDIN AHMET, ARSLAN AHMET (2019). The usage statistics of new HTML5 semantic elements in the ClueWeb12 dataset. International Conference on Advanced Technologies, Computer Engineering and Science (ICATCES 2019), 173-176. (Tam Metin Bildiri/Sözlü Sunum) (Yayın No:5000589)
- AYDIN AHMET, TELÇEKEN SEDAT (2015). Comparison of Three Search Algorithms for Mobile Trip Planner for Eskişehir City. 2015 International Symposium on INnovations in Intelligent SysTems and Applications (Tam Metin Bildiri/) (Yayın No:2231342)
- AYDIN AHMET, TELÇEKEN SEDAT (2015). Artificial Intelligence Aided Recommendation Based Mobile Trip Planner For Eskişehir City. THE 10th IEEE Conference on Industrial Electronics and Applications (Tam Metin Bildiri/) (Yayın No:2231182)
- AYDIN AHMET, TELÇEKEN SEDAT (2015). Mobil Gezin Uygulaması için A* Algoritması ve Karınca Kolonisi Optimizasyonu Karşılaştırılması. IEEE 23. Sinyal İşleme ve İletişim Uygulamaları Kurultayı (Tam Metin Bildiri/) (Yayın No:2230970)