
GAZİ ÜNİVERSİTESİ
Gazi
1926

WEB ÖLÇEĞİNDE BÜYÜK VERİ KAYNAKLARINDAN BİLGİ ÇIKARIMI VE DOĞRULANMASI

Alisettar HÜSEYİNLİ

DOKTORA TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANA BİLİM DALI

GAZİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

AĞUSTOS 2023

ETİK BEYAN

Gazi Üniversitesi Fen Bilimleri Enstitüsü Tez Yazım Kurallarına uygun olarak hazırladığım bu tez çalışmasında;

- Tez içinde sunduğum verileri, bilgileri ve dokümanları akademik ve etik kurallar çerçevesinde elde ettiğimi,
- Tüm bilgi, belge, değerlendirme ve sonuçları bilimsel etik ve ahlak kurallarına uygun olarak sunduğumu,
- Tez çalışmasında yararlandığım eserlerin tümüne uygun atıfta bulunarak kaynak gösterdiğimi,
- Kullanılan verilerde herhangi bir değişiklik yapmadığımı,
- Bu tezde sunduğum çalışmanın özgün olduğunu,

bildirir, aksi bir durumda aleyhime doğabilecek tüm hak kayıplarını kabullendiğimi beyan ederim.

Alisettar HÜSEYİNLİ

04/08/2023

WEB ÖLÇEĞİNDE BÜYÜK VERİ KAYNAKLARINDAN BİLGİ ÇIKARIMI VE DOĞRULANMASI

(Doktora Tezi)

Alisettar HÜSEYİNLİ

GAZİ ÜNİVERSİTESİ

FEN BİLİMLERİ ENSTİTÜSÜ

Ağustos 2023

ÖZET

Bilgi tabanları, belirli bir alanda derinleşmiş veya genel bilgiyi kapsayan olguların oluşturduğu bilgi kümelerini ifade etmektedir. Yapay zekâ çalışmalarından karar destek sistemlerine, soru cevap uygulamalarından arama motorlarına birçok alanda altyapının oluşturulması için bilgi tabanları önemli rol oynamaktadır. Bilgi tabanlarının önemi kadar barındırdığı bilgilerin doğruluğu da önemlidir. Bilgi tabanları yaygın kullanıma sahip olmakla birlikte yanlış bilgi içerme konusunda eksikler barındırmaktadır. Bu eksikliği ortadan kaldırmak için literatürde düzeltme ve tamamlamaya yönelik farklı çalışmalar yapılmıştır. Bu çalışmalar üçlüleri, ilişkileri, ilişki türlerini, sayısal değerleri düzeltmeyi veya yeni üçlü ve ilişkileri ortaya çıkararak bilgi tabanını zenginleştirmeyi hedeflemektedir. Bu tez çalışmasında doğrulama işlemi için üçlülerin güven değerlerini temel alan yayılma yaklaşımı önerilmektedir. Bu yöntem güven değerinin etkisinin tek bir üçlü ile sınırlı kalmadan bilgi tabanı üzerinde yayılmasını sağlamaktadır. Bu sayede güçlü bağlantıları daha da güçlendirerek ve zayıf bağlantıları da ortadan kaldırarak bilgi tabanını devamlı olarak daha istikrarlı duruma getirmektedir. Mevcut çalışmaların bir diğer eksikliği arındırma işlemini tek seferlik işlem olarak ele almaları ve işlem performansını ikinci planda tutmalarıdır. Ancak gerçek dünya BT'leri canlı, dinamik ve sürekli gelişen sistemlerdir. Bu nedenle önerilen yaklaşım sürekli arındırmayı desteklemelidir. Bunu ölçebilmek için farklı veri boyutu ve farklı hatalı üçlü oranlarında deneyler hazırlanmıştır. Doğrulama çalışmalarında aktif kullanılan FB15K, NELL, WN18 ve YAGO3-10 veri kümeleri ile yapılan deneylerde veri kümesinden bağımsız olarak veri boyutu ve yanlış bilgi oranının artmasına rağmen ortalama %87 doğruluk ve %98 hassaslık sonuçları elde edilmiştir.

Bilim Kodu	:	92404
Anahtar Kelimeler	:	Bilgi tabanı, bilgi çizgesi, bilgi doğrulama, yayılma, üçlü güven değeri
Sayfa Adedi	:	116
Danışman	:	Prof. Dr. M. Ali AKCAYOL

INFORMATION EXTRACTION AND VERIFICATION IN WEB-SCALE BIG DATA SOURCES

(Ph. D. Thesis)

Alisettar HÜSEYİNLİ

GAZİ UNIVERSITY

GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

August 2023

ABSTRACT

Knowledge bases refer to sets of knowledge formed by facts that cover a particular field or commonsense knowledge. Knowledge bases play an essential role in creating infrastructure in various areas, ranging from artificial intelligence studies to decision support systems and from question-and-answer applications to search engines. The accuracy of the information they contain is as crucial as the importance of knowledge bases. Although knowledge bases are widely used, they suffer from incompleteness. In order to address this deficiency, different studies have been carried out in the literature for correction and completion. These studies aim to improve triples, relationships, types of relationships, and literals, or to enrich the knowledge base by revealing new triples and relationships. This thesis proposes a propagation approach based on the confidence values of triples for the verification process. This method ensures that the effect of the confidence value is spread throughout the knowledge base without being limited to a single triple. In this way, it constantly stabilizes the knowledge base by further strengthening the strong links and eliminating the weak ones. Another shortcoming of the current studies is that they treat the verification process as a one-time procedure and neglect the ongoing process performance. However, real-world knowledge bases are live, dynamic, and constantly evolving systems. Therefore, the proposed approach should support continuous verification. To measure this, experiments were conducted with different data sizes and false triple rates. In experiments with FB15K, NELL, WN18, and YAGO3-10 datasets, which are actively used in validation studies, an average 87% accuracy and 98% recall results were obtained. These results were achieved regardless of the increase in data size and false information rate across the datasets.

Science Code : 92404

Key Words : Knowledge base, knowledge graph, knowledge verification, propagation, triple confidence

Page Number : 116

Supervisor : Prof. Dr. M. Ali AKCAYOL

TEŞEKKÜR

Her konuda desteğini ve bilgisini benden esirgemeyen değerli tez danışmanım Prof. Dr. M. Ali AKCAYOL'a, değerli fikirleri ile çalışmama katkı sağlayan Tez İzleme Kurulu üyeleri Prof. Dr. Kemal LEBLEBİCİOĞLU ve Prof. Dr. Hacer KARACAN hocalarıma göstermiş oldukları tüm yardımlarından dolayı teşekkürü kendime borç bilirim.

Çalışmamıza, BİDEB 2244 Sanayi Doktora Programı, 118C127 numaralı "İnternette Heterojen Veri Kaynaklarından Veri Toplanması, Doğrulanması ve Sorgulanması" projesi kapsamında verdikleri destekten dolayı Türkiye Bilimsel ve Teknolojik Araştırma Kurumu'na (TÜBİTAK) ve Huawei Telekomünikasyon Ltd. Şti.'ye teşekkür ederiz.

Beni çocukluğumdan beri hayat boyu öğrenmeye teşvik eden, bu yolda her zaman doğrudan ve dolaylı olarak bana destek olan babam Arif HÜSEYİNLİ ve annem Zenfira HÜSEYİNLİ'yi rahmetle anıyorum.

Tez önerimin matematiksel modellenmesi aşamasında yardımcı olan kıymetli arkadaşım Dr. Tekin KARADAĞ'a da katkılarından dolayı teşekkür ederim. Deneysel çalışmalarım sırasında veri setlerinin dağılımları ile ilgili değerli görüşlerini paylaşarak çalışmama katkıda bulunan arkadaşım Dr. Savaş GAYAKER'e de teşekkürü kendime borç bilirim.

Doktora çalışmamın tüm aşamalarında manevi desteğini benden hiçbir zaman esirgemeyen, en zor anlarımda bile her daim yanımda olan sevgili eşim Dilara HÜSEYİNLİ'ye de sonsuz teşekkürlerimi sunarım.

İÇİNDEKİLER

Sayfa

Sayfa

Sayfa

ÇİZELGELERİN LİSTESİ

Çizelge **Sayfa**

ŞEKİLLERİN LİSTESİ

Şekil **Sayfa**

Şekil **Sayfa**

RESİMLERİN LİSTESİ

Resim **Sayfa**

SİMGELER VE KISALTMALAR

Bu çalışmada kullanılmış simgeler ve kısaltmalar, açıklamaları ile birlikte aşağıda sunulmuştur.

Simgeler	**Açıklamalar**
E	Varlıklar kümesi
R	İlişkiler kümesi
(h, r, t)	Özne, ilişki ve nesne üçlüsü
$(\mathbf{h}, \mathbf{r}, \mathbf{t})$	Özne, ilişki ve nesne vektörleri
Conf(a)	*a* üçlüsünün güven değeri
$f_r(h, t)$	Puan fonksiyonu
$h(x)$	*x* elemanı için hash fonksiyonu
$\mathbf{M}_r$	Projeksiyon matrisi
$\mathcal{N}(\mu, \Sigma)$	Gauss dağılımı
$\mathbb{R}^d$	*d* boyutlu reel vektör uzay
$\mathbb{C}^d$	*d* boyutlu karmaşık vektör uzay
$\mathbb{H}^d$	*d* boyutlu hiper karmaşık vektör uzay

Kısaltmalar	**Açıklamalar**
BG	Basic Conceptual Graphs
BGP	Basic Graph Pattern
BT	Bilgi Tabanı
CPN	Coloured Petri Nets
FCG	Full Conceptual Graphs
FCPN	Fuzzy Coloured Petri Nets
FPN	Fuzzy Petri Nets
IE	Information Extraction
GNN	Graph Neural Networks
KGE	Knowledge Graph Embedding
KRL	Knowledge Representation Learning

Kısaltmalar	Açıklamalar
MLN	Markov Logic Network
OWL	Web Ontology Language
PN	Petri Nets
PRA	Path Ranking Algorithm
PSL	Probabilistic Soft Logic
RDF	Resource Description Framework
SBF	Scalable Bloom Filter
SG	Simple Conceptual Graphs
URL	Uniform Resource Locator

1. GİRİŞ

Son yıllarda bilgi teknolojilerindeki gelişmelerin büyük bir hızla artması, üretilen verilerin çok büyük boyutlara çıkmasına neden olmuştur. Bu gelişmelerle birlikte, oluşan veriden anlamlı bilgilerin çıkarılması ve işlenmesi de aynı şekilde çok önemli bir çalışma konusu haline gelmiştir. Yapılan araştırmalara göre son birkaç yılda insanlık tarihinde şimdiye kadar üretilen tüm bilgiden daha fazla bilgi üretilmiştir ve bu süreç ivmeli olarak artmaya devam etmektedir [1, 2]. Bu kadar büyük ölçekte üretilen ve sayısallaştırılan bilginin boyutu dikkate alındığında, bu bilginin işlenmesinin önemi ve aynı zamanda zorluğu da önem kazanmaktadır.

Üretilen bu verinin çoğu zaman yapılandırılmamış veya yarı yapılandırılmış şekilde olması, bilgisayar tarafından bu veriyi işleme ve anlamlandırmayı zorlaştırmaktadır. Genel bilgi (*commonsense knowledge*) veya belli bir alana özgü özel bilgi, karar destek sistemleri, otomatik soru cevaplama, Web arama motorları ve yapay zekaya yönelik çalışmalarda yoğun bir şekilde kullanılmaktadır [3-5]. Günümüzde büyük ölçekteki veri kaynağına en önemli örnek olarak Web gösterilebilir. Web'in dağıtık yapısı kendi içinde yarı yapılandırılmış veya yapılandırılmamış doküman ağını barındırır. Bu dokümanlar Uniform Resource Locator (URL) bağlantıları aracılığı ile birbirlerine bağlanmakta ve büyük bir doküman ağı altyapısını ortaya çıkarmaktadır. 1990'larda Web'in ortaya çıkmasından kısa bir süre sonra anlamsal ağ kavramı da bilgisayar bilimlerinde giderek yaygın kullanılır hale gelmeye başlamıştır. Web'in kurucularından Tim Berners-Lee 2001 yılında anlamsal ağ ile ilgili şunları söylemiştir: "Anlamsal ağ, mevcut Web yapısının uzantısı olmakla beraber, insanların ve makinaların ortak çalışmasını desteklemek amacıyla bilginin daha iyi tanımlanmış semantik anlamını ortaya koymak içindir." [3].

Anlamsal içeriğin sağlandığı bilgi tabanlarının (*knowledge base*) oluşturulması Web'in tarihinden çok daha eskiye, 1980'lere dayanmaktadır [6]. Bilgi tabanları ile ilgili yapılmış erken dönem çalışmaları daha çok karar destek sistemleri için bilgi çıkarımına katkı sağlayacak altyapının oluşturulmasını amaçlamıştır. 2012 yılında Google tarafından geliştirilen "Google Knowledge Graph" [7] Bilgi Tabanı (BT) bu tür bilgi gösterimlerinin literatürde bilgi çizgesi (*knowledge graph*) olarak isimlendirilmesine yol açmıştır. Anlamsal içeriği barındırması açısından bilgi tabanları veri tabanlarından önemli farklılıklara sahiptir. İlişkisel ve ilişkisel olmayan veri tabanları bilgiyi salt veri olarak

barındırmakta ve bu veriler arasındaki anlamsal ilişkiden daha çok mantıksal ilişkilerle ilgilenmektedir. Bilgi tabanlarındaki yaygın yaklaşım ise verinin üçlüler (*triple*) şeklinde ifade edilmesidir. Bir üçlü yapı kendi içinde iki argüman (*argument*) ve bir ilişkiyi (*relation*) barındırır. Literatürde bazı çalışmalar argüman ve ilişki terimleri yerine özne (*subject*), nesne (*object*) ve yüklem (*predicate*) veya baş (*head*), kuyruk (*tail*) ve ilişki (*relation*) terimlerini tercih etmektedir [6].

Bilgi tabanları, anlamsal ağ için oluşturulmuş gösterim standartlarını bilginin üçlüler olarak ifade edilmesi için kullanmaktadır. Bu standartların en yaygınlarına örnek olarak Resource Description Framework (RDF) ve Web Ontology Language (OWL) gösterilebilir.

Veri yönetiminin karmaşık boyutlara ulaşması bilgi tabanlarının da büyük ölçekli bilgi tabanlarına dönüşmesine neden olmuştur [8]. Bu tür bilgi tabanları belirli bir alana özgü olmaktan ziyade daha çok alan bağımsız (*domain-independent*) ve genel bilgiyi içerecek şekilde oluşturulmaktadır. DBpedia [9], YAGO [2], NELL [10], Elementary [11], Freebase [12], ConceptNet [13] gibi araştırma projeleri, Google Knowledge Graph [7] ve Knowledge Vault [14], Microsoft's Satori ve Probase, IBM Watson soru cevaplama sistemi gibi ticari projeler bilgi tabanlarına örnek olarak gösterilebilir.

Yapılan çalışmalar büyük ölçekli bilgi tabanlarının oluşturulmasında önemli gelişmelere imza atmakla beraber bazı eksikliklere de sahiptir. Dian, yaptığı doktora tezinde bu eksikliklerden bazılarına dikkat çekmektedir [15]. İlk olarak, geliştirilen bilgi tabanlarının neredeyse tamamı Vikipedi gibi var olan belirli Web kaynaklarından "*infobox*" bilgilerini işleyerek üçlüleri ortaya çıkarmaktadır. Resim 1.1'de Vikipedi "*infobox*" örneği gösterilmiştir.

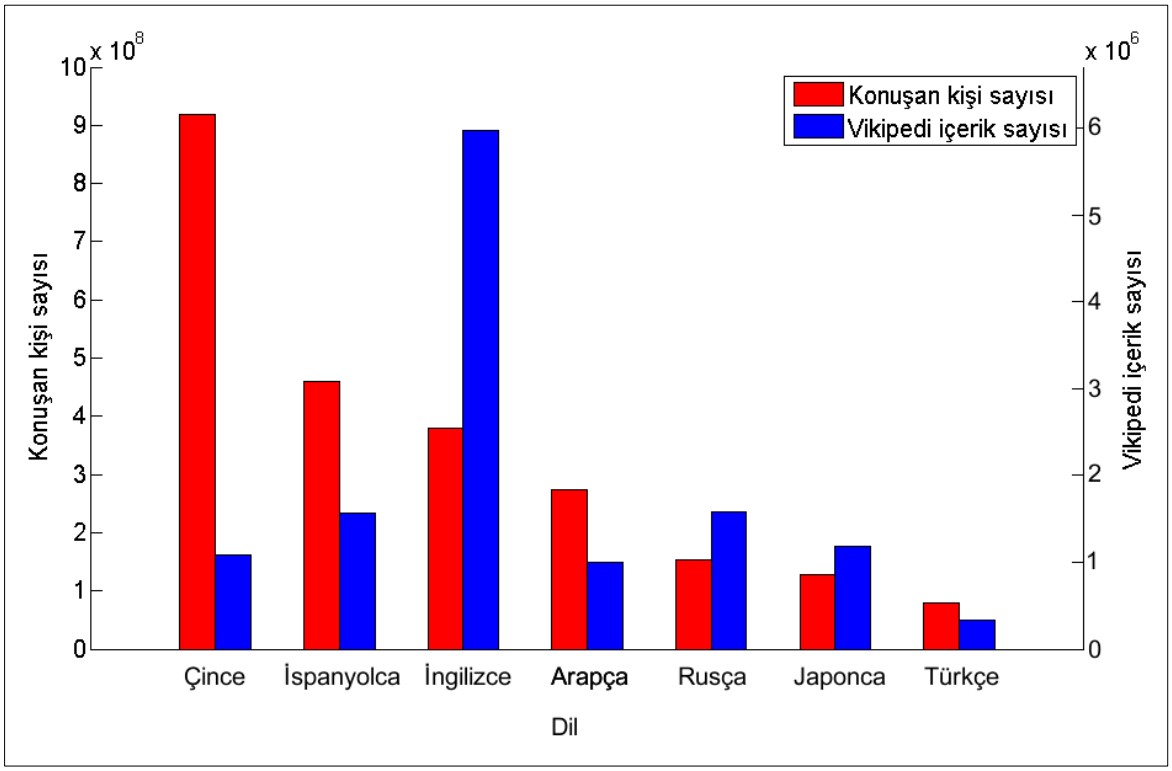
Vikipedi'de yer alan infoboxların içerikleri gönüllü katılımcılar tarafından sıklıkla güncellenmektedir. Bu nedenle bilgi tabanlarındaki üçlülerin de sürekli güncellenmesi gerekmektedir. Bu tür bilgi kaynaklarındaki diğer bir problem ise bilgi tabanlarının eksik ilişkilere sahip olmasıdır. Örneğin, Freebase'in son sürümü kendi içinde 1,9 milyar üçlü barındırmasına rağmen, Freebase'i kullanan uygulamalar halen tamamlanmamış üçlü bağlantılarından dolayı olumsuz etkilenmektedir [15]. Üçüncü ve en önemli eksikliklerden birisi de yapılan çalışmaların büyük bir bölümünde dil olarak İngilizcenin kullanılmasıdır.

Şekil 1.1'de dünya genelinde yaygın kullanılan diller için Vikipedi içerikleri ve bu dili konuşan kişi sayısı karşılaştırmalı olarak verilmiştir [16, 17].

Resim 1.1. Vikipedi infobox örneği

İnternet ortamının gün geçtikçe daha yaygın hale gelmesi bilginin artması ile beraber bilgi kirliliğinin de artmasına neden olmaktadır. Tekrarlı ve zaman aşımına uğramış bilgi ile beraber farklı kaynaklardan kasıtlı veya kasıtsız şekilde yanlış, sahte ve taraflı bilgi üretimi de bu süreçte artış göstermiştir. Facebook, Twitter, Youtube gibi sosyal medya ağları gibi açık platformlar, kişisel bloglar ve diğer medya kaynakları özellikle bu tür istenmeyen bilginin artmasına neden olmakta ve sağlanan bilginin büyük boyutta olması bilgi doğrulama işlemlerinin elle yapılmasını olanaksız hale getirmektedir [18]. Kasıtlı olarak üretilen sahte bilgiler ve taraflı yayınlar geniş kitlelerde olumsuz algıların ve yanlış bilgilenmenin ortaya çıkmasına neden olmaktadır [19]. Bu anlamda, bilginin elde edilmesi kadar doğru bilginin elde edilmesi de önemli konulardan biri haline gelmiştir. Özellikle, Web gibi doğası gereği dağıtık ve denetimsiz bir kaynaktan elde edilen bilginin doğruluğunun sorgulanması günümüzde zorunluluk oluşturmaktadır.

Yukarıda verilen bilgiler ışığında, günümüzde "*bilgi doğrulama*" kendi içinde belirli zorlukları barındıran ayrı bir araştırma konusu olarak giderek önemini artırmaktadır. Her ne kadar literatürde doğrulama (*verification*) ve olgu kontrolü (*fact checking*) birbirinin yerine kullanılsa da yeni çalışmalar bu kavramların ayrıştırılması gerektiğini ortaya koymuştur. Doğrulama, bilimsel yöntemler ile doğru olguların ortaya çıkarılmasını, olgu kontrolü ise iddianın mantıksal ve bağlamsal açıdan doğruluğu ve tutarlılığını ifade etmektedir. Bu anlamda doğrulama işlemi olgu kontrolü için ön aşama olarak görülebilir [17-19].



Şekil 1.1. Yaygın diller için Vikipedi içerikleri ve bu dili konuşan kişi sayısı

Hem bilginin ortaya çıkarılmasında hem de doğrulanması ve kontrol edilmesinde çok sayıda açık araştırma alanı bulunmaktadır. Thorne tarafından yapılan çalışma bu alanlara detaylı bir şekilde ışık tutmaktadır [19]. Bu çalışmada vurgulanan önemli eksikliklerden birisi olguların doğrulanmasında yaygın olarak kapalı dünya varsayımının (*close world assumption*) kullanılmasıdır. Bu varsayım, bilgi tabanındaki bilgilerin kendi içinde mutlak doğruluğunu ve yeterliliğini kabul etmektedir [23]. Bu şekilde bir yaklaşım, örneğin, belirli iki şehir arasında tanımlı tarihte bulunan uçak seferlerinin olması gibi bazı durumlar için yeterli olmasına karşın, gerçek dünya problemlerinin çoğunda eksik kalmakta ve açık dünya varsayımına (*open world assumption*) başvurmayı gerekli kılmaktadır. Thorne,

medya kaynaklarının metin olarak ifade edilmesinin derin öğrenme ve doğal dil işleme yöntemleri ile olanaklı hale gelmesine vurgu yaparak olguların denetimi için sadece metin tabanlı kanıtların değil, resim ve video gibi çoklu ortam kaynaklarının da kullanılabileceğini öne sürmektedir [19].

Bilgi tabanı oluşturmaya yönelik ilk dönem çalışmalar 1980'li yıllara dayanmaktadır. Bu konudaki ilk çalışma olarak Lenat tarafından 1984 yılında yapılan Cyc projesi gösterilebilir [6]. Projenin amacı, bilginin makine formatına dönüştürülerek ifade edilmesi ve terimler arasındaki ilişkilendirmeleri yapmaktır. Günümüze kadar geliştirilmeye devam eden Cyc bilgi tabanı 500000 terim ve bu terimlerle ilişkili 7 milyon kanıyı kendi içinde barındırır.

Bilgilerin Web ortamından toplanmasına yönelik başka bir çalışma Matuszek tarafından yapılmış ve Cyc bilgi tabanının genişletilmesini sağlamıştır [6]. Bu çalışmada İngilizce terimler Google arama motoru üzerinden sorgulanarak dönen sonuçlara göre bilgi çıkarımı ve doğrulaması yapılmaktadır.

Genel bilgiyi kapsayan bir başka çalışma olan ConceptNet [13], MIT tarafından Open Mind Common Sense (OMCS) bilgi tabanı üzerine inşa edilmiş anlamsal bir ağdır. 2002 yılından itibaren ConceptNet projesi geliştirilmeye devam etmekte ve 2017 yılı itibariyle 21 milyon ilişkiden oluşan 8 milyon varlığı barındırmaktadır [13]. ConceptNet ağırlıklı olarak İngilizce içeriğe sahip olmasına rağmen diğer dilleri de desteklemektedir.

YAGO ve YAGO2 projeleri Suchanek ve arkadaşları tarafından Max Planck Enstitüsünde geliştirilen bilgi tabanı oluşturmaya yönelik çalışmalardır [24]. YAGO2 projesi YAGO projesinin geliştirilmesi sonucunda ortaya çıkmıştır. Bu bilgi tabanı mevcut hali ile 9,8 milyon varlık ve 447 milyon kanı içermektedir. Yapılan deneysel çalışmalar YAGO bilgi tabanının %95 seviyesinde doğru bilgi içerdiğini göstermiştir [24].

Max Planck Enstitüsünde yapılan başka bir çalışmada WebChild [25] isimli bir bilgi tabanı geliştirilmiştir. Bu çalışma ile Web içeriklerinden "genel ortak bilgi" tabanının ortaya çıkartılması amaçlanmıştır [26].

Freebase ve DBpedia gibi çalışmalar da genel ortak bilgiyi kapsayacak şekilde geliştirilen bilgi tabanlarına örnek gösterilebilir. RDF standardını temel alan bu bilgi tabanları birbiri

ile bağlantı kurarak daha fazla içeriği barındırma özelliğine sahip olabilmektedir. DBpedia, özellikle açık kaynaklı bir yaklaşım ile günümüzde halen geliştirilmeye devam etmektedir [6].

Akademik çalışmaların yanı sıra bilgi tabanlarının geliştirilmesinde aktif ticari girişimler de mevcuttur. Google tarafından geliştirilen ve şu anda da Google arama motorunun altyapısında kullanılan "Google Knowledge Graph" projesi ticari bilgi tabanlarına örnek olarak gösterilebilir [7]. Ticari bir yaklaşım olmasından dolayı bu bilgi tabanının çalışma şekli ile ilgili çok fazla bilgi yayınlanmamıştır. Freebase sisteminin bünyesine katılması ile beraber Google diğer bir bilgi tabanı olan "Knowledge Vault" projesini başlatmıştır [14]. Bu bilgi tabanı kendi içinde 45 milyon varlığı ve 271 milyon kanıyı barındırmaktadır. Yaygın bilinen bilgi tabanlarının karşılaştırmalı tablosu Çizelge 1.1.'de sunulmuştur.

Çizelge 1.1. Bazı yaygın bilgi tabanlarının içerik açısından karşılaştırılması [27]

Adı	Varlık	Olgu	Varlık Türü	İlişki Türü
Google Knowledge Graph	500 000 000	3 500 000 000	1500	35 000
Freebase	47 560 817	2 903 361 537	26 507	37 781
Knowledge Vault	45 000 000	271 000 000	1100	4469
YAGO	4 595 906	25 946 870	488 469	77
DBpedia	4 580 000	583 000 000	685	2795
Wikidata	15 759 256	43 189 154	23 157	1203
OpenCyc	118 499	2 413 894	45 153	18 526
NELL	1 908 694	441 807	274	296

Bilgi tabanlarında doğruluk ve mevcut olguların kontrol edilmesine ilişkin çalışmalar da son yıllarda giderek artmaktadır. Özellikle, manuel yöntemlerden ziyade otomatikleştirilmiş doğrulama kavramları geniş bilgi tabanlarının ortaya çıkması ile beraber literatürde yer almaya başlamıştır. Doğrulama işlemi için istatistiksel veya bilgi tabanının çizgesine yönelik yöntemler bulunduğu gibi veri madenciliği, Web arama ve doğal dil işleme yöntemlerini ele alan farklı yaklaşımlar da bulunmaktadır [27-29].

Doğrulama işlemlerinde kanıtların kaynağı, doğrulama için gereken girdi/çıktı ve kullanılacak yöntemler en önemli noktalardır. Wang, bu kavramları ele alan çalışmasında, doğrulama işlemi için bilginin kim tarafından ve hangi medya aracılığı ile sağlandığı gibi kaynağa ilişkin *meta* verileri dikkate almaktadır. Bu yöntem bilginin doğrulanması için yeterli olmasa da sınıflandırma başarısının artırılması için ek bilgi sunmaktadır [31].

Doğrulama işlemi literatürde geniş kapsamlı olarak ele alınmaktadır. Bilgi tabanlarının doğruluğunu sağlamak için farklı yöntemleri içeren arındırma çalışmaları yapılmaktadır. Genel itibariyle arındırma çalışmaları tamamlama (*complete*) ve düzeltme (*correction*) yaklaşımlarına sahiptir [32]. Tamamlama çalışmaları bağlantı tahmini (*link prediction*) çalışmaları [33] ile yakından ilgilidir ve kapalı dünya varsayımını temel alan yöntemler ile var olan BT'de yeni ilişkilerin ortaya çıkarılmasını amaçlamaktadır. Hata düzeltme çalışmaları ise ilişkilerin, ontolojik anlamda varlık türlerinin ve üçlünün özellikleri olarak ifade edilebilecek yardımcı bilgilerin (*literal)* düzeltilmesini ele alır.

Son BT oluşturma ve arındırma çalışmalarında öne çıkan yeni konulardan biri de ilişkiler için güven değerlerinin dikkate alınmasıdır. Güven değerleri üçlünün ilişkisine bağlı bir değerdir ve NELL [10] gibi bazı BT'lerde ve olgu doğrulamayı [34] temel alan arındırma çalışmalarında kullanılmıştır. Güven değerleri kapalı dünya yaklaşımlarında (*çevrimdışı olarak da anılacaktır*) dönüştürme tabanlı modellerde maliyet fonksiyonu ile, kural tabanlı modellerde ise t-norm, co-norm, olasılıksal hafif mantık (Probabilistic Soft Logic- PSL) veya Markov mantık ağı (Markov Logic Network - MLN) yöntemleri ile belirlenmektedir. BT dışı kaynaklar üzerinden güven değerlerinin hesaplanmasına yönelik açık dünya yaklaşımlı (*çevrimiçi olarak da anılacaktır*) çalışmalar da mevcuttur. Bu çalışmalarda güven değeri bazen kaynağın güvenilirliğini [14], bazen bilginin doğruluğunu [10] bazen de her iki değerin birleştirilmesi [34] ile elde edilmiş ortak değeri ifade etmektedir.

Bu tez çalışması kapsamında BT oluşturma ve doğrulama kavramları ele alınarak incelemelerde bulunulmuştur. Yapılan literatür taramaları doğrultusunda BT'lerin oluşturulması ve doğrulanmasına yönelik açık alanlar ve eksiklikler tespit edilerek bu yönde çözüm üretmeye yönelik çalışmalar yapılmıştır. Bu tez çalışmasında BT'lerde doğrulama işlemine yönelik arındırama aşaması için güven değerleri ve üçlülerin diğer üçlülerle ilişkisini temel alan yaklaşım önerilmektedir. Bu yaklaşım çevrimdışı yaklaşımlarla BT'nin içinde oluşturulan veya çevrimiçi yaklaşımlarıyla dış kaynaklardan elde edilen güven değerinin sadece mevcut üçlü ile sınırlı kalmadan çizge üzerinde yayılımını ele almaktadır. Bu şekilde yüksek güven değerine sahip güçlü ilişkilerin daha da güçlenmesini, düşük güven değerine sahip zayıf ilişkilerin ise ortadan kaldırılmasını hedeflemektedir.

BT'lerin arındırılması tek seferlik bir işlem değildir. Dinamik ve sürekli büyüyen bir çizge yapısı ile çalışılması gerektiği göz önünde bulundurulmalıdır. Bu nedenle verinin boyutu ve çizgenin topolojisinin dikkate alınması gerekmektedir. Ayrıca, Paulheim'in çalışmasında [27] vurgulandığı gibi arındırma işlemini performans kriterlerini dikkate almadan ele almak önerilen yöntemin başarısını değerlendirirken yanılgıya sebep olacaktır. Bu nedenle çalışmada yanlış bilgiyi temizleme başarısı ile beraber veri boyutundan bağımsız olarak temizleme için harcanan toplam işlem süresi ve önerilen yöntemin ölçeklenebilirliği gibi performans kriterleri de göz önünde bulundurulmuştur. Bu açıdan önerilen yöntem yenilikçi yaklaşımıyla arındırma işlemini tek seferlik işlem olarak ele almadan, BT oluşturulurken ve genişlerken sürekli devam eden arındırma sağlamaktadır.

Tez kapsamında ayrıca, BT'ler dinamik ve sürekli büyüyen yapısı göz önünde bulundurularak veri yapısı ve veri depolama işlemleri açısından performans kısıtları incelenmiştir. Bu anlamda akan veri üzerinden BT'nin oluşturulması, güncellenmesi, silinmesi ve mevcut BT yapısında arama yapılmasına yönelik indeksleme çalışması yapılmıştır. Bloom filtresi ve B+ ağaçlarının birleştirilmesi ile oluşturulan indeksleme yöntemi BT üzerindeki veri işlemlerinde mevcut indeksleme yöntemlerinden işlem süreleri anlamında daha iyi sonuçlar ortaya koymuştur.

Tezin ilerleyen bölümleri aşağıdaki şekilde organize edilmiştir. İkinci bölümde bilgi tabanları, bilgi tabanı oluşturma yöntemleri, bilgi gösterimi yaklaşımları ve bilgi yerleştirme çalışmalarından bahsedilecektir. Daha sonra BT'lerde doğrulama işlemleri kapsamında çevrimiçi ve çevrimdışı yöntemler sunulacaktır. Üçüncü bölümde Web ölçeğinde bilgi çıkarımı, web kaynakları üzerinden doğrulama ve elde edilen bilginin güven değerine ilişkin çalışmalardan bahsedilecektir. Dördüncü bölümde tez kapsamında önerilen BT modeli ve bilginin güven değerine bağlı olarak arındırma işlemine yönelik önerilen yöntem detaylı olarak açıklanacaktır. Beşinci bölümde BT'de performans kısıtları ve ölçeklenebilir Bloom filtresi yardımıyla performans optimizasyonuna yönelik önerilen indeksleme yöntemi ele alınacaktır. Altıncı bölümde güven değerinin yayılması ve performans optimizasyonuna yönelik önerilen yöntemlerin deneysel çalışmaları ve sonuçları paylaşılacaktır. Son bölümde tez kapsamında elde edilen sonuçlar ve bu konuya yönelik gelecek çalışmalar tartışılarak sonuçlandırılacaktır.

2. BİLGİ TABANLARI VE BİLGİ DOĞRULAMA

BT belirli bir konuda dikey bilgiyi veya günlük genel kanıları ifade eden yatay bilgiyi yapılandırılmış şekilde bir arada barındıran bilgi kümelerini ifade etmektedir. Bu bilgi kümeleri yapay zekâ uygulamaları, soru-cevaplama ve karar destek uygulamaları gibi birçok alanda arka plan sistemleri olarak aktif rol almaktadır. Bu anlamda BT'lerin oluşturulması ve doğrulanması önemli araştırma konularıdır. Etkin BT sistemlerinin oluşturulmasında doğru bilgi gösterim modelinin belirlenmesi ilk aşama olarak görülmektedir. Özellikle, oluşturulacak BT'nin dinamik yapısı göz önüne alındığında hızlı sorgulama, güncelleme ve ekleme/çıkarma işlemleri BT'nin oluşturulması için öncelikli gereksinimler arasında yer almaktadır. Tezin bu bölümünde BT'lerin oluşturulmasında kullanılan yöntemler, bilgi gösterim yaklaşımları ve BT'lerde bilgi doğrulama başlıkları ele alınacaktır.

2.1. Bilgi Tabanı Oluşturma Yöntemleri

Literatürde BT'lerin oluşturulması için geliştirilmiş farklı yöntemler sunulmuştur. BT'yi baştan oluşturmaya yönelik yöntemlerinin yanı sıra, BT yapısının oluşturmasından sonra yeni olguların ortaya çıkarılması ve güncel verilerin mevcut bilgi tabanına eklenmesi amacıyla kullanılan güncelleme yöntemleri de BT'nin genişletilmesinde önemli rol oynamaktadır [23, 38, 39]. Bu anlamda hem BT'nin baştan oluşturulması hem de güncellenmesine yönelik yöntemler BT'nin oluşturulması kapsamında değerlendirilebilir.

BT'leri baştan oluşturma yöntemleri tarihsel gelişme süreci de dikkate alındığında dört farklı başlık altında toplanabilir [6]. Bu yöntemler Çizelge 2.1'de gösterilmiştir.

Çizelge 2.1. Bilgi tabanı oluşturma yöntemleri

Yöntem	Kaynak	Teknik
İnsan emeği	İnsan	Bilgi mühendisliği
Etkileşim	İnsan	İnsan-bilgisayar etkileşimi
Madencilik	Metin	Doğal dil işleme
Muhakeme	Bilgi tabanları	Mantıksal çıkarım

İlk yöntem erken dönem bilgi tabanı oluşturma çalışmalarında daha çok kullanılmakla beraber insan emeğini temel alır ve kullanıcı dostu ara yüzlerden gerçek insan bilgisinin girilmesi ile elde edilen olguları bilgi tabanının oluşturulması için kullanır. Bu yöntem

gönüllü kullanıcıların bilinçli bir şekilde veri girişi yaptığı platformlarda uygulanabildiği gibi kişilerin günlük hayatta kullandığı uygulamalar üzerinden arka plan verisi toplanarak da gerçekleştirilebilmektedir.

Etkileşim yöntemlerinde edinilen bilginin kaynağı yine insan zihni olarak görülmektedir. İlk yöntemden farklı olarak burada veri girişi insanlar aracılığıyla doğrudan yapılmamaktadır. Burada insanların bilgisayar ile etkileşimi temel alınarak örneğin davranış analizi veri girişi olarak kabul edilmektedir. Bilgisayar oyunlarından çıkarım yapılması buna örnek olarak gösterilebilir. Oyun tabanlı yöntemlerde bilgi elde edilmesi daha eğlenceli hale getirilmekte, genel bilginin çıkarımı ise sistemin yan özelliği olarak işlev görmektedir [38, 40].

Daha güncel yaklaşımlar madencilik ve muhakemeye dayalı yöntemleri tercih etmektedir. Güncel bilgi tabanlarının ve yapılan çalışmaların ölçeği dikkate alındığında ilk iki yöntemin büyük ölçekli bilgi edinme için çok verimli yaklaşımlar olmadığı görülmektedir. Madencilik ve muhakeme yaklaşımlarının ortaya çıkmasındaki bir diğer etken ise belirli bir alana yönelik bilgi tabanlarının geliştirilmesinden öte genel bilginin tek çizge üzerinde gösterilmesini amaçlayan çalışmalara gerek duyulması olmuştur. Bilgi tabanının oluşturulması için otomatikleştirilmiş bu yöntemlerin tercih edilmesinin en önemli nedeni son dönemde veri üretiminin devasa boyutlara ulaşması, özellikle de Web gibi büyük veri kaynaklarının ortaya çıkmasıdır [2].

Madencilik yaklaşımları metin gibi yapılandırılmamış veya yarı yapılandırılmış veri kaynaklarından genel bilginin doğal dil işleme yöntemleri ile ortaya çıkarılmasını amaçlamaktadır.

Muhakeme başlığı altında toplanmış yöntemler var olan bilgi tabanlarını kaynak olarak kullanmakta ve mantıksal çıkarımlar aracılığıyla yeni olgular elde etmektedir. Bu yöntemlerde kavramlar arasında analoji kurulması, tümevarım yaklaşımı ve istatistiksel çıkarım yöntemleri tercih edilmektedir [41, 42].

Var olan bilgi tabanlarının genişletilmesine yönelik yöntemler de kendi içinde kapalı dünya varsayımını ve açık dünya varsayımını temel alan yöntemler olarak iki gruba ayrılmaktadır [27]. Çevrimdışı yöntemler bilgi tabanının kendisini veri kaynağı olarak

kullanmaktadır. Sınıflandırma, istatistiğe dayalı yöntemler bilgi kaynağının genişletilmesi için tercih edilmektedir. Çevrimdışı yöntemlerde dikkat edilmesi gereken en önemli nokta Thorne'un araştırmasında vurguladığı kapalı dünya yaklaşımının eksik ve yanlış olguların türetilmesine neden olabileceğidir [19].

Bilgi tabanının genişletilmesine yönelik çevrimiçi yaklaşımlar da kendi içinde sınıflandırma, doğal dil işleme, yarı yapılandırılmış veriden bilgi çıkarımı ve bilgi tabanı füzyonu olarak sıralanabilir. Çevrimiçi yöntemler temel bilgi kaynağı haricindeki kaynakları referans olarak kabul ettiğinden daha geniş bilgi kaynağı ile çalışmaktadır. Bu başlık altındaki yöntemlerin karşılaştırmalı değerlendirmesi veri büyüklüğü, veri akış hızı, beklenen doğrulama oranı gibi kriterlere göre yapılabilmektedir [27]. Daha önce de belirtildiği gibi açık dünya yaklaşımı kendine özgü zorlukları ve kısıtlarına rağmen daha güvenilir bilginin ortaya çıkmasını sağlamaktadır. Tez çalışmasında bu yaklaşım izlenerek Thorne'un çalışmasında [19] ele aldığı sorunların ortadan kaldırılması hedeflenmiştir.

2.2. Bilgi Gösterimi

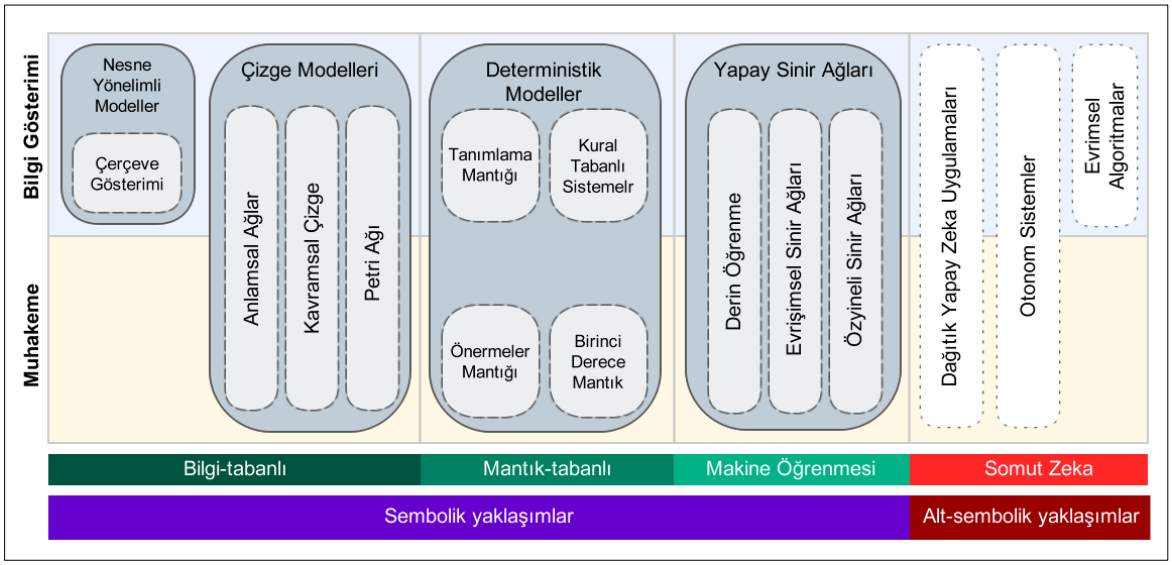
Bilgi gösterimi modelleri üzerine yapılan ilk çalışmalar 19. yüzyılın sonlarına dayanmaktadır [40]. Bir araştırma konusu olarak bilgi gösterimi, yapay zekâ ile ilgili çalışmaların yaygınlaştığı 1950'li yıllardan beri güncelliğini korumaya devam etmektedir. Bilgi, tanımı gereği geneldir ve karmaşıktır. İçinde çıkarım yapılabilen malumatı barındırma özelliğine sahiptir. Zaten bu karmaşıklığa sahip değilse ve içerdiği veri ile ilgili bir genelleme sunmuyorsa sadece işlenmiş veriyi bir başka deyimle malumatı ifade eder. Burada vurgulanması gereken en önemli nokta farklı ayrıntı (*granularity*) seviyelerinde bir bilginin veri veya bir verinin bilgi olarak ele alınabileceğidir [41].

Bilgi gösterimi için literatürde farklı yaklaşımlar bulunmaktadır. Bu yaklaşımların her biri doğrudan veya dolaylı olarak insan zihninin bilgiyi nasıl işlediğine yönelik felsefi görüşleri esas alır. Bilgi gösterimi yöntemleri sembolik ve alt-sembolik yaklaşımlar olarak iki ana başlık altında toplanabilir [42]. Bazı kaynaklarda bu iki yaklaşıma ek olarak bağlantıcı (*connectionist*) yaklaşım da ayrı başlık altında ele alınmaktadır [43]. Bağlantıcı yaklaşım doğrudan insan beyninin yapısını temel alan yöntemleri içine almakla birlikte girdi olarak sembolik veri kullandığı için sembolik yaklaşımların altında değerlendirilebilir. Bilgi

gösterimi ve muhakeme için kullanılan farklı yaklaşımlar Şekil 2.1'de görselleştirilmiştir. Bu görsel [42] ve [44]'ten esinlenerek oluşturulmuştur.

2.2.1. Sembolik yaklaşımlar

Sembolik modeller bilginin semboller ve kurallardan oluşan mantıksal ifadeler yardımıyla gösterimini hedefler. Bu modeli temel alan yaklaşımlar insan zihninin çalışma mantığının da bir nevi sembol manipülasyonu olduğunu kabul etmektedir. Bu kapsamda değerlendirilen tüm modeller varlıkların, kavramların, olguların ve kuralların ifade edilmesi için sembollerden oluşan sistemlerin yeterliliğini öngörmektedir. Örneğin, yazı olarak "kırmızı" kelimesi sembolik olarak kırmızı kavramını tanımlamaktadır. Bu yaklaşım bilgi gösterimine dayalı muhakemenin (*reasoning*) de tanımlanan sembollerin manipülasyonu ile mümkün olduğunu savunur. Sembolik yaklaşımlar kendi içinde mantık-tabanlı, bilgi-tabanlı ve makine öğrenmesi yaklaşımları olarak üç başlığa ayrılır.



Şekil 2.1. Bilgi gösterimi ve muhakeme yaklaşımları

2.2.2. Alt-sembolik yaklaşımlar

Bu yaklaşımlar bilgiyi tek başına ve ayrık olarak ele almamaktadır. Bilgi sembolik bir ifadeden öte etkileşimi, görselleştirmeyi ve hareketi içinde barındıran bir bütün olarak kabul edilmektedir. Somut zekaya (*embodied intelligence*) yönelik çalışmalar bu yaklaşımlara örnek olarak gösterilebilir. Somut zekâ çalışmaları yüksek zihinsel faaliyetin zihin dışında bir bedene ihtiyacı olduğunu savunmaktadır [42]. Tez kapsamında ele alınan

BT'ler alt-sembolik gösterimler kapsamında değerlendirilmediğinden bu yaklaşımlar detaylı olarak irdelenmeyecektir.

2.3. Bilgi Gösterimi için Temel Prensipler

Bilgi gösterimi için önerilen modelin hangi yaklaşımı temsil etmesinden bağımsız olarak aşağıdaki temel prensiplere cevap verebilmesi hedeflenir [45]:

1. Biçimsellik: bilginin anlamını biçimsel simgelerle ortaya koymalı
2. Mantıksallık: mantıksal temellere dayanmalı
3. Yapısallık: bilginin yapısal gösterimine olanak tanımalı
4. Hesaplanabilirlik: bilgi üzerinde hesaplama işlemleri yapabilmeli
5. Anlamsal Yakınlık: kullanıcı açısından bilginin gerçek dünya ve temsili kavramları arasında farkı en aza indirgemeli

Yukarıdaki temel prensipler genel çerçevesi sabit kalmakla birlikte R. Davis ve ark. çalışmasında da farklı ifadelerle yer almaktadır [46]. Bahsedilen tüm prensipleri aynı anda sağlayan bilgi gösterim modeli günümüzde maalesef ortaya koyulamamıştır. Bununla birlikte farklı gösterim yaklaşımlarının temel prensipler açısından artı ve eksi yönleri değerlendirilebilir [45]. Sembolik yaklaşımların karşılaştırma tablosu Çizelge 2.2'de gösterilmiştir.

Çizelge 2.2. Bilgi gösterim yöntemlerinin karşılaştırılması [42]

Yöntem	Biçimsellik	Mantıksallık	Yapısallık	Hesaplanabilirlik	Anlamsal Yakınlık
Önermeler Mantığı	Sağlanıyor	Sağlanıyor	Sağlanamıyor	Sağlanıyor	Sağlanamıyor
Tanımlama Mantığı	Sağlanıyor	Sağlanıyor	Sağlanamıyor	Sağlanıyor	Sağlanamıyor
Birinci Derece Mantık	Sağlanıyor	Sağlanıyor	Sağlanamıyor	Sağlanıyor	Sağlanamıyor
Kural Tabanlı Sistemler	Sağlanıyor	Sağlanıyor	Sağlanamıyor	Sağlanamıyor	Sağlanamıyor
Çerçeve Gösterimi	Sağlanıyor	Sağlanıyor	Sağlanıyor	Sağlanamıyor	Kısmen sağlanıyor
Çizge Modelleri	Sağlanıyor	Sağlanıyor	Sağlanıyor	Kısmen sağlanıyor	Kısmen sağlanıyor

2.4. Bilgi Gösterimi Yöntemleri

Bilgi çıkarımı yöntemleri ile elde edilen bilgilerin düzenli bir yapıda gösterimi bilginin erişimi, güncellenmesi ve doğrulanması açısından önemlidir. Bilgi gösterimi yöntemleri yapay zekâ çalışmalarında arka plan sistemlerinin oluşturulması açısından ayrı bir araştırma konusu olarak öne çıkmaktadır. Bilgi gösterim yöntemleri mantıksal gösterim, çizge gösterimi, çerçeve gösterimi ve yapım kuralları olarak dört ana başlık altında irdelenebilir [46–49].

Mantıksal gösterim

Bu gösterim şeklinde bilgi mantıksal ifadeler kullanılarak temsil edilir. Bilgi belirli sözdizimsel *(syntax)* ve anlamsal *(semantics)* kurallar çerçevesinde mantıksal ifadelere dönüştürülür. Mantıksal gösterim şeklinin avantajları olarak elde edilen ifadelerden mantıksal çıkarımların yapılabilmesi ve programlama dillerine kolay bir şekilde aktarılması gösterilebilir. Bununla birlikte mantık kurallarından kaynaklı kısıtlamalar ve mantıksal gösterim sonucunda ortaya çıkan bilginin doğal yapıda olmaması bu gösterim şeklini kullanmayı zorlaştırmaktadır.

Çizge gösterimi

Bu gösterim şeklinde bilgi çizge olarak ifade edilmektedir. Çizge üzerinde düğümler ve kenarlar kullanılarak nesneler, kavramlar ve ilişkiler gösterilir. Bilginin gösteriminde çizge yaklaşımlarının kullanımı Peirce'in 1896 yılında mantıksal cümleleri çizge olarak gösterdiği çalışmaya dayanmaktadır [40, 45]. Çizge veri yapısı ile bilgi gösterimi için birçok yaklaşım bulunmakla birlikte bunlardan en yaygın olanları anlamsal ağ ve kavramsal çizge *(conceptual graph)* yaklaşımlarıdır. Çizge veri yapısı kullanılarak görselleştirme bilginin doğal bir gösterimini sunduğu için anlaşılması daha kolaydır. Aynı zamanda algoritmik açıdan karmaşıklık düzeyinin yüksek olması ve kenarların tanımlanmasında standartların olmaması bu yöntemin dezavantajı olarak öne çıkmaktadır.

Çerçeve (*frame*) gösterimi

Çerçeve, kendi içinde öznitelikler ve değerleri bulunduran kayıt *(record)* olarak gösterilebilir. Her bir çerçeve gerçek dünyada bir varlığı temsil etmektedir. Çerçeve

içerisinde tanımlı değerler aynı zamanda başka bir varlığa bağlantı verebilecek şekilde tanımlanır. Bu anlamda çerçeve gösterimi nesne yönelimli gösterim olarak da tanımlanabilir. Kolay programlanabilir olması ve esnek yapı sayesinde genişleyebilmesi bu gösterim şeklinin avantajları olarak gösterilebilir. Diğer yönden mantıksal çıkarım yapmanın zorluğu ve çok genel bir yaklaşıma sahip olması bu yöntemin dezavantajlarıdır.

Yapım kuralları

Bu gösterim şekli durum ve eylem çiftleri ile tanımlanır. Durum eylem çifti "Eğer DURUM oluşursa EYLEM gerçekleşir" şeklinde de açıklanabilir. Bu gösterim İngilizcede "IF condition THEN action" olarak tanımlanır. Yapım kuralları ile bilgi gösterimi yöntemi doğal dil şeklinde ifade edildiğinden ve kural tanımlamanın esnekliğinden kaynaklı avantajlara sahiptir. Bu gösterim şeklinin dezavantajları olarak mantıksal çıkarım yapmanın zorluğu ve belirli bir durum için birden fazla kuralın aynı anda gerçekleşebilir olması dezavantaj olarak gösterilebilir.

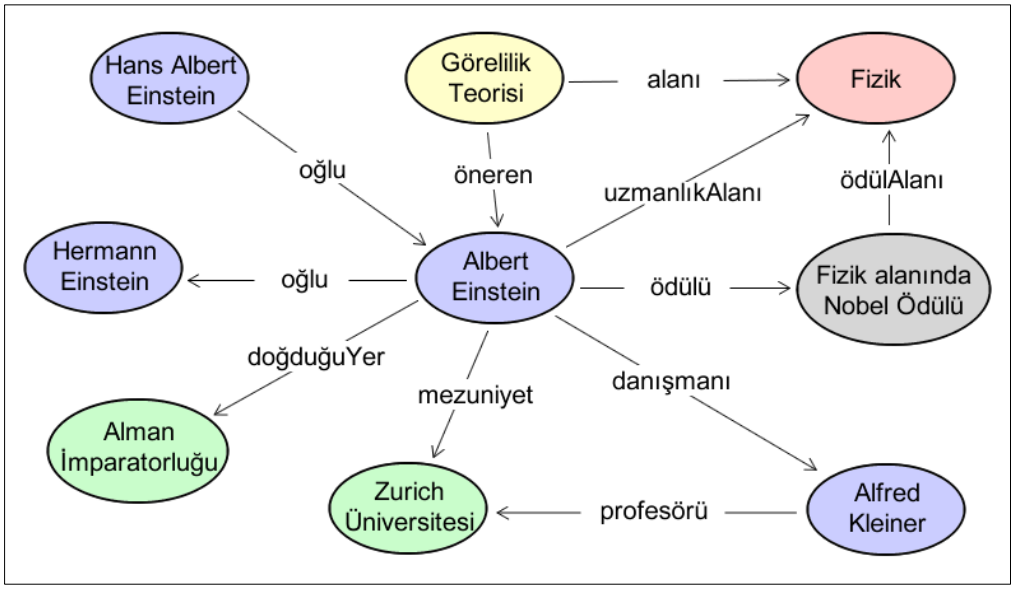
Bilginin gösterim yöntemleri bu başlıklarla sınırlı değildir. Literatürde bunların dışında yapay sinir ağları ile bağlantıcı model yaklaşımı da bulunmaktadır [50].

Bilginin gösteriminde çizge yaklaşımları diğer gösterim yöntemlerine oranla daha yaygın kullanıma sahiptir. Özellikle, anlamsal ağ ve kavramsal çizge yaklaşımları bilgi tabanı oluşturma çalışmalarında sıkça tercih edilmektedir. Anlamsal ağlarda düğümler nesneleri, kenarlar ise ilişkileri temsil etmektedir. Anlamsal ağlar nesneler arası ilişkiyi tanımlamakla birlikte nesneleri farklı türler altında kategorileştirir. Günümüzde anlamsal ağ gösterimi de kendi içinde çizge yapısına göre farklı alt gösterim şekillerine ayrılmıştır. Cognitive Semantic Networks, Multilayered Extended Semantic Networks (MultiNets), Hierarchical Semantic Form, Resource Description Framework bu çizge türlerine örnek olarak gösterilebilir [50].

Cognitive Semantic Networks nesneler arası bağlantıları "*is-a*", "*has-a*", "*a-kind-of*" türü bağlaçlarla ilişkilendirmektedir. Bu gösterim şekli sürekli genişleyen yapısı ile öğrenme benzeri bir davranış sergilemek için kullanılır [51]. Multilayered Extended Semantic Networks özellikle soru cevaplama uygulamalarında doğal dilin anlaşılması için tercih

edilmektedir [52]. Hierarchical Semantic Form yöntemi ise iki farklı türde düğüm kullanarak nesneleri çizge içinde kategorileştirmektedir [53].

Anlamsal ağlardan farklı olarak kavramsal çizge yaklaşımında hem nesneler/kavramlar hem de ilişkiler düğümlerle ifade edilmektedir. Bu yöntemde yönlü çizge kullanılmaktadır ve çizge üzerindeki kenarlar etiketlenmemektedir. Bu şekilde oluşturulan çizgeden birinci derece mantık *(first-order logic)* yardımıyla mantıksal önermeler çıkarılmaktadır. Kavramsal çizge yaklaşımı da zaman içinde geliştirilerek yeni farklı alt yaklaşımlar oluşturulmuştur. Basic Conceptual Graphs (BG), Simple Conceptual Graphs (SG), Full Conceptual Graphs (FCG) bunlara örnek olarak gösterilebilir [43, 45]. Bu yaklaşımlar birbirinden çizgenin yapısına göre ayrışmaktadır. Bağlantı şekline göre farklı çizgeler üzerinden farklı mantıksal önermeler çıkarılmaktadır [45].



Şekil 2.2. Bilgi çizgesinde örnek varlık ve ilişkiler olarak üçlülerin ifade edilmesi [54]

2.5. Bilginin Çizge Gösterimi

Son yıllar BT'ye yönelik yapılan çalışmalar çoğunlukla bilgi gösterimi yaklaşımı olarak çizge gösterimini tercih etmektedir [27, 55]. Çizge gösterimi bu anlamda BT çalışmaları için *de facto* standart haline gelmiştir. Çizelge 2.2'de de görüldüğü gibi temel prensipler açısından çizge gösterim yöntemleri diğer yöntemlere göre avantaj sağlamaktadır. Özellikle yapısallık ve gerçek dünya tanımlamaları ile anlamsal yakınlık farkını büyük ölçüde kapatabilmesi çizge yöntemlerinin güçlü yönleri olarak gösterilmektedir. Aynı

zamanda çizge üzerinde bazı kısıtları barındırmasıyla birlikte algoritmik hesaplamaların gerçekleştirilebiliyor olması BT'nin oluşturulması için gereken temel prensipleri yerine getirdiğini göstermektedir [45].

Literatürde kullanılan çizge gösterim yaklaşımları bazı farklılıklarla birlikte üçlülerin (özne, yüklem ve nesne üçlüsü) içerdiği varlık ve ilişkilerin çizgeye yerleştirilmesi ile oluşturulmaktadır [56]. Bahsedilen farklılıklar üçlüde ifade edilen bilgiye ek olarak yardımcı bilgilerin (*literal*) çizgeye yerleştirilmesi veya ilişkilerin de varlıklar gibi birer düğüm olarak ifade edilmesidir [57]. Şekil 2.2'de yönlü çizge içerisinde örnek üçlü yapıların yerleştirilmesi gösterilmiştir. Bu yapıda ilişkiler çizgenin kenarları olarak ifade edilmiştir.

Bu şekilde oluşturulan ağırlıklandırılmış ve yönlü çizge $E \times R \times E$ şeklinde üç boyutlu seyrek tensör G olarak gösterilebilir. Burada E varlıkların, R ise ilişkilerin sayısını ifade etmektedir. h varlığından t varlığına r ilişkisinin bulunması $G(h, r, t) = 1$ olarak, ilişkinin mevcut olmaması ise $G(h, r, t) = 0$ olarak ifade edilir [14]. Tez kapsamında kullanılan tensör tanımı uygulamalı matematikteki kullanım şekli ile çok boyutlu diziye karşılık gelmektedir [58].

Başka bir çizge yaklaşımında ise varlık ve ilişkiler doğrudan G tensörüne haritalanmadan önce kelime yerleştirme yöntemleri (word2vec [59], GloVe [60], Elmo [61] vb.) ile önce (**h**, **r**, **t**) vektörlerine dönüştürülür. Daha sonra ise doğrudan bu vektörler veya bu vektörlerden oluşan matrisler üzerinde işlem yapılır.

Farklı çizge gösterimi yaklaşımlarını temel alan BT çalışmalarında BT'nin oluşturulması, düzenlenmesi, BT üzerinde çevrimdışı yöntemlerle yeni bilgi keşfi, mevcut bilgilerin doğrulanması ve BT'nin arındırılması gibi işlemler bilgi yerleştirme (*knowledge embedding*) başlıkları altında incelenebilir.

2.6. Bilgi Yerleştirme Yöntemleri

Bilgi yerleştirme yöntemleri literatürde Knowledge Representation Learning (KRL), Knowledge Graph Embedding (KGE), istatistik tabanlı ilişkisel öğrenme veya çoklu ilişkisel öğrenme olarak isimlendirilmektedir [54]. Bu yöntemler çizgeyi oluşturan varlık

ve ilişkilerin, üzerinde hesaplama yapılabilmesi için düşük boyutlu öznitelik uzayına semantik olarak haritalanmasını hedeflemektedir [56]. Bilgi yerleştirme yöntemlerinin temel amacı BT üzerindeki sınıflandırma, olgu denetimi ve yeni olguların çıkarımı işlemleri için kullanılan hesaplama karmaşıklığını en aza indirgemektir. Bilgi yerleştirme çalışmaları BT'nin çevrimdışı yöntemlerle doğrulanması ve arındırılmasında da kullanılmaktadır. Bilgi yerleştirme yöntemleri üçlü olgu tabanlı ve açıklama tabanlı yöntemler olarak iki başlıkta ele alınabilir. Üçlü olgu tabanlı yöntemler tez kapsamında geliştirilen yöntemle ilişkili olduğundan bu modeller detaylı olarak irdelenecektir.

2.6.1. Üçlü olgu tabanlı yerleştirme

Üçlü olgu tabanlı yerleştirme yöntemleri BT'de bulunan üçlüler üzerinde işlem yapmaktadır. *(h, r, t)* üçlüsü dışında bulunan yardımcı bilgiler ve varlıklara ait özellikler (*literals*) dikkate alınmamaktadır. Üçlü olgu tabanlı yerleştirme yöntemleri kendi içinde dönüştürme tabanlı (*translation-based*), tensör ayrıştırma ve sinir ağı tabanlı modeller olarak üç başlıkta toplanabilir [62–64].

Dönüştürme tabanlı modeller

Dönüştürme tabanlı modellerde kelime yerleştirme yöntemleri temel alınarak vektör uzayına haritalanmış varlık ve ilişkilerin yakınlık değerleri üzerinden olgu denetimi ve yeni olguların tespiti yapılmaktadır.

Bu konuda yapılmış ilk çalışma TransE yerleştirme modelidir [65]. TransE modelinde varlıklar ve ilişkiler aynı $\mathbb{R}^d$ uzayına haritalanmaktadır. Dönüştürme tabanlı modellerde *(h, r, t)* üçlüsüne karşılık gelen **h**, **r** ve **t** vektörleri arasında Eş. 2.1'de bulunan uzamsal yakınlığın sağlanması beklenmektedir.

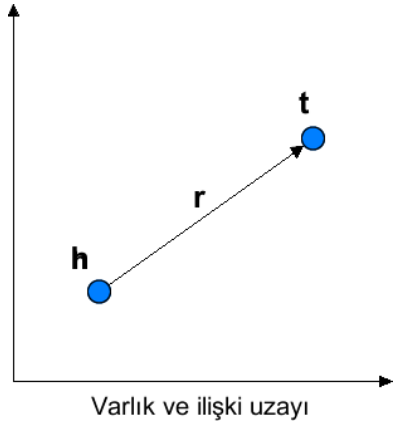
$$\mathbf{h} + \mathbf{r} \approx \mathbf{t} \tag{2.1}$$

Olgunun denetlenmesi için ise puan fonksiyonu (*score function*) kullanılmaktadır. Uzamsal yakınlığın TransE için puan fonksiyonu Eş. 2.2'de gösterilmiştir.

$$f_r(h,t) = \|\mathbf{h} + \mathbf{r} - \mathbf{t}\|_{l_1/l_2} \tag{2.2}$$

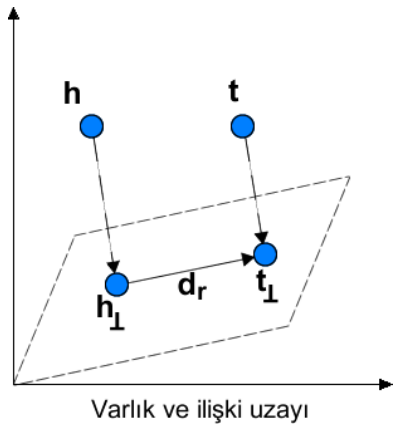
Uzaklık ölçümü l_1-*norm* veya l_2-*norm* olarak hesaplanmaktadır.

TransE modeli birebir ilişkilerde (ülke – başkent ilişkisi gibi) başarılı sonuçlar üretmekle birlikte $1 - N$ ve $N - N$ ilişki türlerinde yüksek başarı sağlayamamaktadır. TransE modeli Şekil 2.3'te gösterilmiştir.

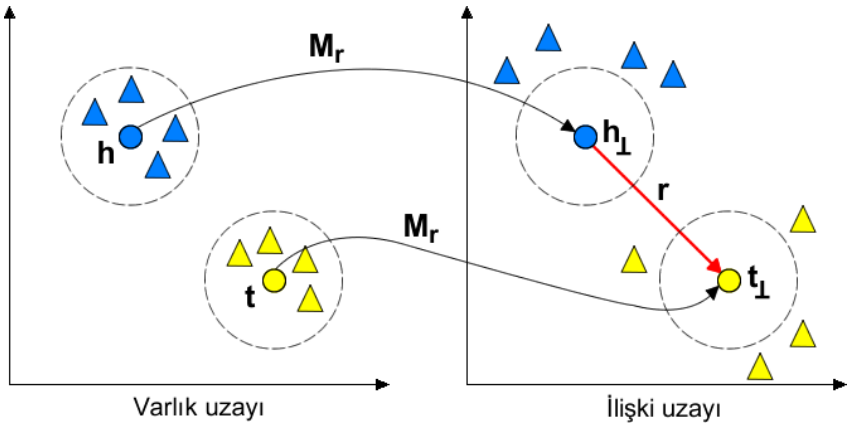


Şekil 2.3. TransE modeli gösterimi

TransE'nin $1 - N$ ve $N - N$ ilişkilerdeki eksik yönlerini ortadan kaldırmak için TransH yöntemi önerilmiştir [66]. Bu yöntem varlıklar arasındaki uzaklığı her bir ilişkinin kendi hiperdüzlemine izdüşümünü alarak hesaplamaktadır. TransH modeli Şekil 2.4'te gösterilmiştir.



Şekil 2.4. TransH modeli gösterimi



Şekil 2.5. TransR modeli gösterimi

Bu model için puan fonksiyonu ise Eş. 2.3'te görülebilir.

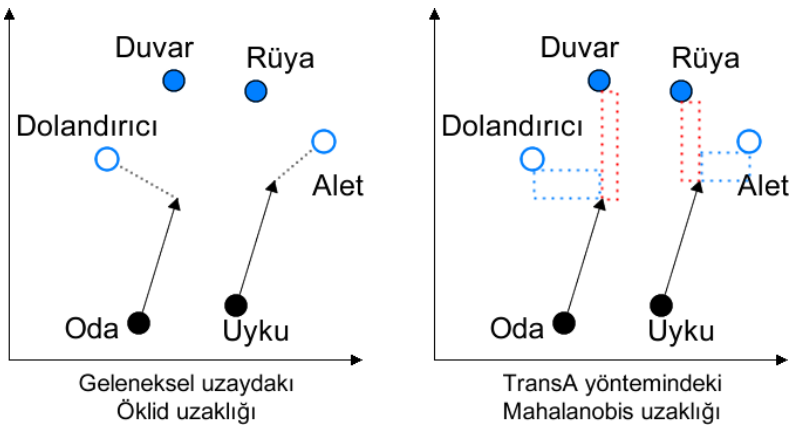
$$f_r(h,t) = \|\mathbf{h}_\perp + \mathbf{d}_r - \mathbf{t}_\perp\|_2^2 \tag{2.3}$$

Bu eşitlikte $\|\cdot\|_2^2$ işlemi Öklid uzaklığının karesini ifade etmektedir.

TransR [67] yöntemi TransH yaklaşımını genişleterek varlık ve ilişkilerin temsil edildiği uzayları ayrıştırmıştır (Şekil 2.5). Puan fonksiyonu TransH ile aynı olmakla birlikte **h** ve **t** vektörlerinin izdüşümünü hesaplamak için $\mathbf{M}_r \in \mathbb{R}^{k\times d}$ projeksiyon matrisi kullanılmaktadır. TransR modeli için $\mathbf{h}_\perp$ ve $\mathbf{t}_\perp$ vektörleri aşağıdaki şekilde hesaplanır.

$$\mathbf{h}_\perp = \mathbf{M}_r\mathbf{h},\ \mathbf{t}_\perp = \mathbf{M}_r\mathbf{t} \tag{2.4}$$

Öklid uzaklık ölçütü yerine uyarlamalı Mahalanobis uzaklığını temel alan TransA çalışması da dönüştürme tabanlı modellere örnek gösterilebilir [68].



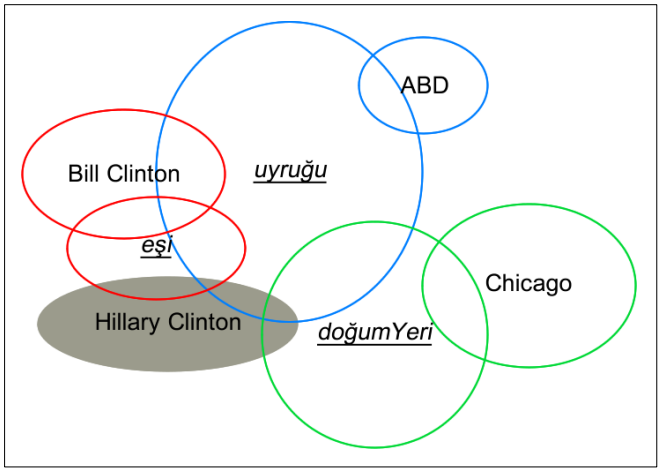
Şekil 2.6. TransA modeli gösterimi

TransA modeli Öklid uzaklığı ile yanlış belirlenen olguları **h** vektörünün **t** vektörlerinden oluşan kümeye uzaklığını hesaplayarak düzeltmektedir (Şekil 2.6). Mahalanobis uzaklığına dayalı puan fonksiyonu Eş. 2.5'te gösterilmiştir.

$$f_r(h,t) = (|\mathbf{h} + \mathbf{r} - \mathbf{t}|)^{\top} \mathbf{W}_r (|\mathbf{h} + \mathbf{r} - \mathbf{t}|) \tag{2.5}$$

Bu eşitlikte $\mathbf{W}_r$ ilişki tabanlı simetrik ağırlık matrisini göstermektedir.

Öklid uzayında vektör uzaklıklarını temel alan yaklaşımlara TransE'den türetilmiş TransSparse [69], STransE [70] ve TransD [71] modelleri de örnek gösterilebilir.



Şekil 2.7. KG2E modeli gösterimi

Dönüştürme tabanlı yaklaşımlar sadece Öklid uzayı ile sınırlı kalmamaktadır. Örneğin, KG2E modeli varlık ve ilişkilerin gösterimi için çoklu Gauss dağılımları kullanmaktadır [72]. KG2E modeli Şekil 2.7'te gösterilmiştir. Bu model varlık ve ilişkileri ortalama vektörleri ve kovaryans matrisleri şeklinde tanımlamaktadır (Eş. 2.6).

$$\mathbf{h} \sim \mathcal{N}(\mu_h, \Sigma_h), \mathbf{r} \sim \mathcal{N}(\mu_r, \Sigma_r), \mathbf{t} \sim \mathcal{N}(\mu_t, \Sigma_t) \tag{2.6}$$

KG2E modeli **h** – **t** ve **r** vektörleri arasındaki uzaklık yaklaşımını da $\mathcal{N}(\mu_h - \mu_t, \Sigma_h - \Sigma_t)$ dağılım fonksiyonuna genişletmiştir. KG2E gibi Gauss dağılımını temel alan modellere TransG [73] modeli de örnek gösterilebilir.

2.6.2. Tensör ayrıştırma modelleri

Tensör ayrıştırma yöntemlerinde BT'de bulunan varlık ve ilişkiler $\mathcal{X} \in \mathbb{R}^{n \times n \times m}$ ikili (*binary*) tensör olarak ifade edilmektedir. Bu konu ile ilgili detaylı bilgi Bölüm 2.3'te verilmiştir. Daha sonra tensör ayrıştırma yöntemleri kullanılarak 3B tensörün öznitelik sayıları indirgenmektedir [58].

Tensör ayrıştırma kullanan RESCAL [74] modeli BT'yi *d*-rank ayrıştırma yaparak *k*. dilim için aşağıdaki matris çarpımını elde etmektedir.

$$\mathcal{X}_k \approx AR_kA^T, k = 1,2, \dots, m \tag{2.7}$$

Burada $A \in \mathbb{R}^{n \times d}$ matrisi varlıkların, $R_k \in \mathbb{R}^{d \times d}$ matrisi ise *k.* ilişkinin sıkıştırılmış anlamsal gösterimini temsil etmektedir. Verilen varlıklar için $\mathbf{h}_i$ ve $\mathbf{t}_j$ vektörleri *A* matrisinde *i.* ve *j.* satırlara karşılık geleceğinden Eş. 2.7 temel alınarak aşağıdaki puan fonksiyonu tanımlanabilir.

$$f_r(h,t) = \mathbf{h}^\top \mathbf{M}_r \mathbf{t} \tag{2.8}$$

DistMult [75] yöntemi RESCAL modelindeki işlem karmaşıklığını azaltarak performans artışı sağlamıştır. DistMult modelinde ilişkilerin temsil edildiği $\mathbf{M}_r$ matrisi diag(**r**) diyagonal matrisi ile kısıtlandırılmıştır.

Complex [76] yöntemi DistMult modelinin simetrik [77] ve asimetrik [78] ilişkileri ayrıştırmadaki eksikliğini gidermek amacıyla varlık ve ilişkileri karmaşık uzaya taşımaktadır. Bu durumda puan fonksiyonu aşağıdaki şekilde değiştirilmiştir.

$$f_r(h,t) = \mathrm{Re}(\mathbf{h}^\top \mathrm{diag}(\mathbf{r}) \bar{\mathbf{t}}) \tag{2.9}$$

Bu eşitlikte Re(·) karmaşık değerin reel kısmını, $\bar{\mathbf{t}}$ ise **t** vektörünün karmaşık eşleniğini göstermektedir.

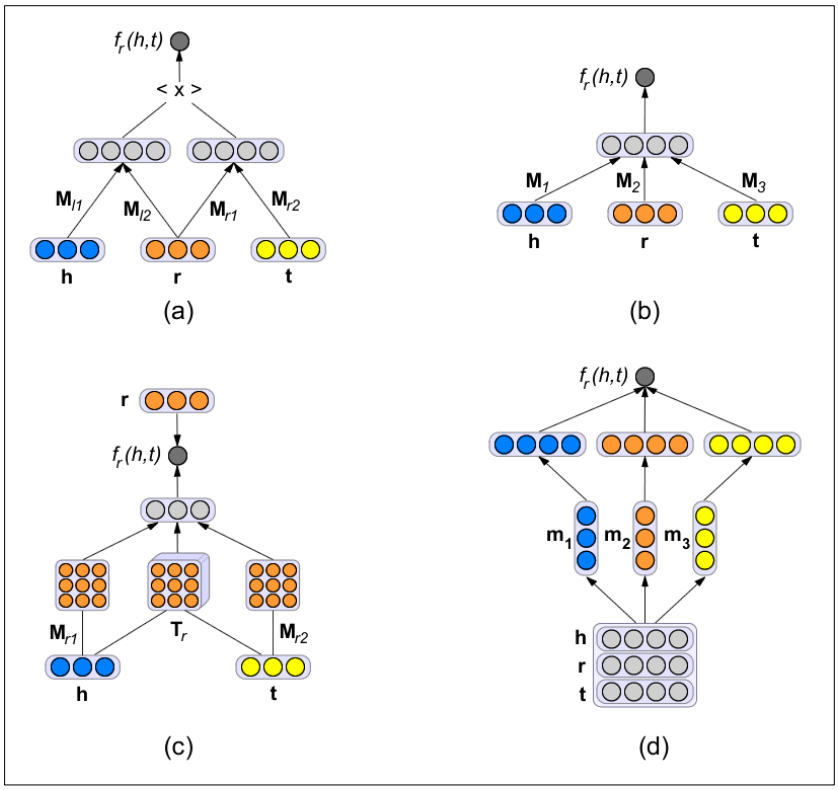
Tensör ayrıştırmada sıkça kullanılan Canonical Polyadic (CP) [58, 79] yönteminin varlıkları birbirinden bağımsız olarak ele almasından kaynaklanan eksikliğini gidermek

için SimplE [80] modeli üçlüleri çift yönlü değerlendirerek ortalama puan fonksiyonu hesaplamaktadır (Eş. 2.10).

$$f_r(h,t) = \frac{1}{2}(\mathbf{h} \circ \mathbf{rt} + \mathbf{t} \circ \mathbf{r}'\mathbf{h}) \tag{2.10}$$

Bu eşitlikte ◦ işlemi Hadamard çarpımını, $\mathbf{r}'$ ise ters ilişki matrisini ifade etmektedir.

Bu modeller dışında Euler eşitliğini temel alan RotatE [81] ve dört boyutlu hiper karmaşık uzayda puan fonksiyonunu hesaplayan QuatE [82] modelleri de tensör ayrıştırma yöntemlerine örnek olarak gösterilebilir.



Şekil 2.8. Yapay sinir ağı tabanlı modellerin gösterimi (a) SME, (b) MLP, (c) NTN, (d) ConvKB

2.6.3. Sinir ağı tabanlı modeller

Bilgi yerleştirme için uygulanan bir diğer yaklaşım sinir ağı modellerini temel almaktadır. Özellikle, yapay sinir ağlarının doğrusal olmayan karmaşık problemlere çözüm üretme yeteneği bu yöntemlerin uygulanmasına olanak tanımaktadır. Sinir ağı tabanlı modellerde

h, **r** ve **t** vektörleri sinir ağına girdi olarak alınmakta ve enerji puanı hesaplanarak çıktı üretilmektedir [62]. Çıktı değeri üçlünün güven değerine karşılık gelmektedir.

Bilgi yerleştirme için sinir ağı modeli öneren çalışmalara örnek olarak SME [83] gösterilebilir (Şekil 2.8 (a)). SME modeli varlık ve ilişki vektörlerini girdi alarak ara katmana göndermektedir. Burada üçlüde bulunan *özne* ve *nesne*nin *ilişki* ile olan bağlantısı değerlendirilerek son katmanda güven değeri hesaplanmaktadır. Ara katmandaki değerler $g_{sol}(\mathbf{h},\mathbf{r})$ ve $g_{sağ}(\mathbf{t},\mathbf{r})$ fonksiyonları ile tanımlanmaktadır (Eş. 2.11).

$$g_{sol}(\mathbf{h},\mathbf{r}) = \mathbf{M}_{l1}\mathbf{h} + \mathbf{M}_{l2}\mathbf{r} + \mathbf{b}_l \qquad (2.11)$$
$$g_{sağ}(\mathbf{t},\mathbf{r}) = \mathbf{M}_{r1}\mathbf{t} + \mathbf{M}_{r2}\mathbf{r} + \mathbf{b}_r$$

Elde edilen $g_{sol}(\mathbf{h},\mathbf{r})$ ve $g_{sağ}(\mathbf{t},\mathbf{r})$ matrisleri puan fonksiyonunda kullanılmaktadır (Eş. 2.12).

$$f_r(h,t) = g_{sol}(\mathbf{h},\mathbf{r})^\top g_{sağ}(\mathbf{t},\mathbf{r}) \qquad (2.12)$$

Güven değerinin hesaplanması için başka bir yaklaşım Knowledge Vault [14] çalışmasında önerilen MLP modelidir. Bu model SME modelinin basitleştirilmiş şekli olarak önerilmiştir. Şekil 2.8 (b)'de MLP yapay sinir ağının diyagramı gösterilmiştir. MLP modeli için puan fonksiyonu Eş. 2.13'deki şekilde tanımlanır.

$$f_r(h,t) = \mathbf{m}^\top f(\mathbf{M}_1\mathbf{h} + \mathbf{M}_2\mathbf{r} + \mathbf{M}_3\mathbf{t}) \qquad (2.13)$$

Bu eşitlikte $\mathbf{M}_1, \mathbf{M}_2, \mathbf{M}_3 \in \mathbb{R}^{d\times d}$ birinci katmanın, **m** ise ikinci katmanın ağırlık değerlerini göstermektedir.

NTN [84] modeli klasik matris yaklaşımı yerine ilişkilerin tanımlanması için tensör önermektedir. Bu modelin bir diğer farkı da tensör katmanında elde edilen çıktının tekrardan **r** ilişki vektörü ile çarpılarak puan fonksiyonunun elde edilmesidir (Eş. 2.14). Bu yöntemin dezavantajı her bir ilişki için ayrı $\mathbf{T}_r \in \mathbb{R}^{d\times d\times k}$ tensör tanımlamasının yapılmasıdır. NTN modeli Şekil 2.8 (c)'de gösterilmiştir.

$$f_r(h,t) = \mathbf{r}^\top f(\mathbf{h}^\top \mathbf{T}_r \mathbf{t} + \mathbf{M}_{r1}\mathbf{h} + \mathbf{M}_{r2}\mathbf{t} + \mathbf{b}_r) \qquad (2.14)$$

NTN modelinin özel hali ($\mathbf{T}_r = 0$) SLM modeli olarak önerilmiştir [84]. Bu modelin geliştirilmesinde amaç ilişki tensörünün oluşturduğu hesaplama karmaşıklığını azaltmaktır. SLM modeli için puan fonksiyonu Eş. 2.15'de gösterilmiştir.

$$f_r(h,t) = \mathbf{r}^\top f(\mathbf{M}_{r1}\mathbf{h} + \mathbf{M}_{r2}\mathbf{t} + \mathbf{b}_r) \tag{2.15}$$

ConvKB [85] modeli evrişimsel sinir ağı kullanarak sinir ağı tabanlı yaklaşımların kapsamını genişletmektedir. ConvKB evrişimsel filtre kullanarak girdi matrisi (**h**, **r** ve **t** vektörlerinin birleşmesinden elde edilmiş matris) üzerinde farklı öznitelikleri ortaya çıkarır (Şekil 2.8 (d)). ConvKB için puan fonksiyonu aşağıda gösterilmiştir.

$$f_r(h,t) = \mathrm{C}(g(\mathbf{A} * \Omega))\mathbf{w} \tag{2.16}$$

Bu eşitlikte Ω filtre kümesini, $\mathbf{A} * \Omega$ işlemi **A** girdi matrisine uygulanmış evrişim işlemini, $\mathbf{w} \in \mathbb{R}^{3d}$ ağırlık vektörünü, C ise birleştirme (*concatenation*) işlemini göstermektedir.

Sinir ağını temel alan yaklaşımlar bu modellerle sınırlı değildir. Son zamanlarda, özellikle derin öğrenme yöntemlerinin popülerlik kazanması çizge sinir ağları [86], çekişmeli üretici ağlar [87] gibi farklı yapay sinir ağı modellerini kullanan bilgi yerleştirme yaklaşımlarının da ortaya çıkmasına sebep olmuştur.

2.6.4. Açıklama tabanlı ve diğer yerleştirme modelleri

Bu modeller bilgi yerleştirme işlemi için BT'de bulunan üçlüleri kullanmakla birlikte ek bilgiler veya BT'nin çizge yapısını da dikkate almaktadır. Üçlü dışında metin tabanlı ek bilgileri kullanan yöntemler üçlü olgu tabanlı modellerin uzantısı olarak görülebilir. Dönüştürme tabanlı, tensör ayrıştırma tabanlı veya sinir ağı tabanlı yöntemlerin temel alınarak genişletilmesi bu yöntemler için esin kaynağı olmuştur [62, 63].

Benzer şekilde çizge yapısı temel alınarak çizge kenarları üzerinden yeni olguların çıkarımı ve mevcut olguların denetimi de bilgi yerleştirme yaklaşımlarına örnek olarak gösterilebilir. Örneğin, TransE modelinin genişletilmiş hali olarak PtransE [88] yöntemi puan fonksiyonunu hesaplamak için ele alınan üçlünün puan fonksiyonu ile çizge üzerinde olguyu ortaya çıkaran üçlülerin puan fonksiyonlarının ortalamasını dikkate almaktadır.

Açıklama tabanlı ve çizge tabanlı yaklaşımlara ek olarak bilginin zamansal boyutunu da modele entegre eden çalışmalar bulunmaktadır. Bu tür çalışmalar, var olan bilginin belirli zaman aralığında geçerliliğinden kaynaklı eksikliği ortadan kaldırmayı hedeflemektedir [89–91].

2.7. Bilgi Tabanı Doğrulama Yöntemleri

Bilgi tabanlarının doğrulanması amacıyla geliştirilmiş yöntemlerinin tarihi BT oluşturma yöntemleri kadar eski olmasa da bu alanda da literatürde önemli çalışmalar bulunmaktadır [18, 32, 43]. Doğrulama yöntemleri ağırlıklı olarak denetimli (*supervised*) yöntemlerden oluşmaktadır. Denetimli doğrulama yöntemlerine sınıflandırma işlemi örnek olarak gösterilebilir.

Sınıflandırma yöntemi elde edilen bilginin daha önceden öğrenme setinde tanımlanmış etiketlerle öğrenilerek test veri setinde sınanmasını temel alır. Bu yöntem bilgi tabanlarının doğrulanması için de benzer şekilde kullanılmaktadır. Yöntemin dezavantajı sınıflandırma işlemi için herhangi bir dış bilgiyi temel almadan mevcut bilgileri kullanmasıdır. Bilgi doğrulama alanında yapılan metin sınıflandırma çalışmalarındaki temel yaklaşım üretilmiş ve yanlış bilgi içeren bilgilerin dilbilimsel açıdan değerlendirilerek tespit edilmesidir [94].

Bilgi tabanlarının doğrulanması amacıyla literatürde çizge üzerinde uygulanan algoritmalardan da faydalanılmıştır [44-47]. Path Ranking Algortihm (PRA) bu amaçla kullanılan algoritmalara örnek olarak gösterilebilir. Bu ve benzeri yöntemlerde amaç, çizge üzerindeki varlıklar arasında yolun (*path*) değerlendirilerek bilginin doğrulanmasıdır [31, 48].

Metin benzerlik ölçütleri dikkate alınarak yapılan doğrulama işlemleri de bilgi tabanlarının doğrulanmasında sıkça kullanılmakta ve metin özdeşliği üzerinden farklı yaklaşımlar geliştirilmektedir. “Recognizing Textual Entailment” çalışması bu yönteme örnek olarak gösterilebilir [101]. Bu yöntemde amaç, iddia edilen olgunun hangi kaynaklar üzerinden edinildiği ve mevcut kaynaklar ile özdeşliğinin değerlendirilmesidir. Yapılan çalışmalar bilgi kaynağını doğruluk puanına göre puanlamakta ve sonraki kaynak doğrulama işlemlerinde bu doğruluk (güvenilirlik) puanı da göz önüne alınmaktadır [31].

Literatürde BT doğrulama yöntemlerinin alt alanı olarak BT arındırma veya temizleme (*refinement*) yöntemleri de önemli yere sahiptir. Temizleme işlemleri mevcut BT'lerde bulunan yanlış olguları ortadan kaldırarak BT'yi bilginin doğruluğu açısından daha istikrarlı duruma getirmeyi hedeflemektedir. BT oluşturma yöntemlerinde olduğu gibi doğrulama yöntemlerinde de yaklaşımlar kapalı dünya varsayımlı ve açık dünya varsayımlı olarak iki başlık altında değerlendirilebilir.

2.7.1. Kapalı dünya varsayımlı çalışmalar

Daha önce de değinildiği gibi kapalı dünya varsayımlı çalışmalar BT'nin bütünlük (*completeness*) varsayımına dayanmaktadır ve bu çalışmalarda doğrulama, hata giderme ve gürültü temizleme işlemleri BT çizgesi ele alarak uygulanmaktadır. Kapalı dünya varsayımlı çalışmalar kendi içinde de dönüştürme tabanlı ve kural tabanlı olmak üzere alt başlıklara ayrıştırılabilir.

Dönüştürme tabanlı çalışmalar

TransE [65] ile başlayan dönüştürme tabanlı yöntemler üçlüdeki ilişkiyi (*relation*) özne ve nesne arasındaki dönüşüm olarak *head* + *relation* ≈ *tail* şeklinde ifade etmektedir. Dönüştürme uzayının türlü varyasyonları ile TransE çalışması farklı şekillerde devam ettirilmiştir [66, 67, 71, 88, 102–104]. TransE kolay uygulanabilirlik sunmakla birlikte karmaşık ilişkiler ele alındığında limitli performans sergilemektedir. BT'deki yol (*path*) bilgisini de kullanarak karmaşık ilişkilerin kodlanması için PTransE [88] TransE çalışmasını genişletmiştir. Bununla birlikte PTransE de farklı ilişki örüntülerini modellemede kısıtlı kalmaktadır. PaTyBRED [105] iki varlık arasındaki ilişkilerden oluşan yol bilgisi ve varlıkların tür bilgisini birleştirerek bu özellikleri de sınıflandırmaya dahil etmiştir. UKGE [106] TransE'de tanımlanan dönüştürme değeri ile birlikte güven değerini de üçlü değerlendirmesine dahil etmektedir. Ancak TransE'nin yaşadığı zorluklar bu çalışma için de geçerlidir. CKRL [107] çalışması güven değerine bağlı çerçeve sunarak TransE çalışmasını farklı yöne doğru genişletmiştir. Bu çalışmada güven değeri sadece varlıklar arasındaki ilişkilerle sınırlı kalmamaktadır. Bu değer hem yerel hem de global yol üzerinden hesaplanmaktadır. DSKRL [108] CKRL çalışmasına ek olarak varlıkların hiyerarşik tür bilgisi ve farklılık bilgisini de ekleyerek gürültü temizleme yapmaktadır. Güncel hata giderme çalışmalarına bir diğer örnek olarak INDIGO gösterilebilir. INDIGO

[109] Graph Neural Networks (GNN) kullanarak BT'yi GNN olarak ifade etmektedir. Daha sonra oluşturulan bu sinir ağı kullanılarak yeni üçlüler doğrulanmaktadır.

Kural tabanlı çalışmalar

Kural tabanlı bilgi çıkarımı yöntemleri ağırlıklı olarak bilgi yerleştirme ve yeni bilgi çıkarımı için tercih edilmektedir. Bu yöntemler mantık kurallarını temel alarak mevcut bilgilerin doğrulanması için de kullanılmaktadır. *(Y, oğlu, X)* ← *(X, ebeveyni, Y) ve (Y, cinsiyet, Erkek)* kural tabanlı bilgi çıkarımına örnek gösterilebilir. Kural tabanlı hata giderme çalışmaları mantık kuralları ile çıkarım yapmakta veya ön tanımlı kurallar yardımıyla hatalı bilgileri tespit etmektedirler. Örnek olarak, "Correction Tower" [110] çalışması kolay implementasyon sunarak sapan değerlerin ve hatalı ilişkilerin tespitinde başarı sağlamaktadır. Buna karşılık bu çalışma büyük BT yapılarında başarısız olmaktadır. KALE [111] çalışması da birleşme, kesişme, tersini alma gibi mantık işlemlerini t-norm bulanık mantık kuralları ile hesaplayarak yeni bilgi çıkarımı yapmaktadır. Jeyaraj ve ark. tarafından yapılan başka bir çalışma [112] PSL kullanarak kuralların değerini olasılık dağılımına göre hesaplamaktadır. Bir diğer çalışma pLogicNet [113] ise kuralların tanımlanması ve yeni değerlerin hesaplanması için Markov mantık ağı (MLN) [114] tercih etmiştir. Benzer şekilde Bayes ağı (*Bayesian network*) tercih eden çalışmalar da mevcuttur [115].

PTrustE [116] çalışması kural tabanlı ve dönüşüm tabanlı çalışmaları tek modelde toplayarak farklı bir yaklaşım sunmuştur. BT'nin özelliklerini ortaya çıkarmak amacıyla önce varlıklara ait yol bilgilerini dönüştürmekte (*embedding*) daha sonra olasılıksal mantık ağı (*probability logic network*) yardımıyla global ve yerel güven değerleri hesaplamaktadır. Son olarak Bi-directional Gated Recurrent Unit yardımıyla yol için doğruluk değeri (*path trustworthiness*) oluşturmakta ve bu değeri üçlünün doğruluğunu belirlemek için kullanmaktadır.

Çevrimdışı yaklaşımlar BT'nin arındırılması için başarılı yöntemler ortaya koymakla birlikte BT'nin tamamı için periyodik çalıştırılması gereken modellerdir. Özellikle, BT'nin çizge yapısında bulunan öznitelikleri ortaya koymak için yapılan gösterimler, varlık ve ilişkiler arasındaki semantik bağlantıyı haritalamaya çalışan dönüşümler veya üçlülerin mantıksal kurallarının belirlenmesini sağlayan kural tabanlı modeller temizleme işlemleri

için her seferinde BT'nin tamamının topyekûn ele almaktadır. Tahmin edilebileceği gibi büyük BT yapılarında bu yöntemler karmaşık ve hesaplama maliyeti yüksek işlemlere yol açmaktadır.

2.7.2. Açık dünya varsayımlı çalışmalar

Açık dünya yaklaşımları BT'nin tamamlanmamış olması varsayımını temel almaktadır [117]. Bu nedenle bu çalışmalarda üçlülerin doğrulanması ve hataların temizlenmesi BT'nin iç kaynakları ile sınırlı kalmamaktadır. Web kaynakları, sosyal medya ve başka BT'ler dış kaynak şeklinde kullanılarak üçlülerin doğruluğu teyit edilmektedir. NELL [10], Web kaynaklarını kullanarak bilgi çıkarımı yapmakla birlikte aday ilişkilere sezgisel yöntemlerle güven değeri atamaktadır. Web kaynağını kullanan Knowledge Vault [14] çalışması da kaynaklardan elde edilen bilgileri veri harmanlama (*data fusion*) yöntemleri ile harmanlayarak güven değeri hesaplamaktadır.

Doğrulama işlemlerinde kanıtların kaynağı, doğrulama için gereken girdi/çıktı ve kullanılacak yöntemler önemli parametrelerdir. Wang, bu kavramları ele alan çalışmasında, doğrulama işlemi için bilginin kim tarafından ve hangi medya (*ortam*) aracılığı ile sağlandığı gibi kaynağa ilişkin meta verileri dikkate almaktadır. Bu yöntem bilginin doğrulanması için yeterli olmasa da sınıflandırma başarısının artırılması için ek bilgi sunmaktadır [31].

Var olan bilgi tabanlarını kullanarak temizleme yapan başka bir çalışma "Tracy", Mohamed tarafından yapılmıştır [100]. Bu çalışmada doğrulanması hedeflenen olgu parçalanarak ilişkili varlıklar için bilgi tabanındaki olgular toplanmakta ve bu olgular için mantıksal çıkarım yapılmaktadır.

Web arama motorlarını temel alan başka bir doğrulama yaklaşımı Gerber ve ark. tarafından yapılan "DeFacto – Deep Fact Validation [34]" isimli çalışmada önerilmiştir. Bu çalışmada elde edilen olgu web arama motoru üzerinde araştırılmaktadır. Elde edilen sayfalar uygunluk ve güven düzeyine göre değerlendirilmekte ve sonuçlar üzerinden karar verilmesi için kullanıcıya sunulmaktadır.

Web ve BT kaynaklarından elde edilen güven değerleri ile birlikte doğal dil işleme yöntemleri ile metinler üzerindeki kanıtlar kullanılarak da bilginin doğrulanması yapılmaktadır. Örnek olarak, Zafar ve arkadaşlarının yaptığı "FactCheck [118]" çalışmasında metin tabanlı işlemler ile RDF üçlüleri üzerinde doğrulama yapılmaktadır. Metin işleme yöntemleri ile yapılan başka bir örnekte, Du ve arkadaşları tarafından mantıksal ve metin tabanlı kanıtların birleştirilmesi gerçekleştirilmiştir [94]. Doğrulama işlemi için bahsedilen yöntemlerle birlikte veri madenciliği, Web arama ve doğal dil işleme yöntemlerini ele alan diğer yaklaşımlar da bulunmaktadır [27, 28].

Son yıllarda BT'lerde arındırma işlemini konu alan detaylı tarama çalışmaları da yapılmıştır. Alan ile ilgili daha detaylı araştırma için Paulheim'in makale taraması [27], Hogan'ın "Knowledge Graphs" çalışması [32] ve Wang'ın çalışması [56] incelenebilir.

Çevrimiçi yöntemler her bir bilgi parçacığından üçlülerin çıkarımı sırasında kaynağın güvenilirliğine dayalı güven değeri hesaplayabilmektedir. Bu yaklaşımda her bir üçlü tekil olarak ele alındığından çevrimdışı yöntemlerde olduğu gibi ölçeklendirme sorunu yaşanmamaktadır. Ancak gürültünün azaltılması ve hataların giderilmesi konusunda bu modeller de eksik kalmaktadır. BT'ye yeni eklenen veya güncellenen üçlüye ait güven değeri sadece bu üçlü ile sınırlı kalmakta ve BT'nin temizlenmesi açısından etki oluşturmamaktadır.

3. WEB ÖLÇEĞİNDE BİLGİ ÇIKARIMI VE BİLGİNİN GÜVEN DEĞERİ

Güncel BT oluşturma çalışmalarında Web ve İnternet kaynakları en önemli bilgi kaynağı olarak yer almaktadır. Web doğası gereği kendi içinde yapılandırılmış, yarı yapılandırılmış ve yapılandırılmamış bilgi türlerini barındırmaktadır. Bu da kendi içinde Web'den bilgilerin elde edilmesi, ön işlenmesi ve bilgi çıkarım yöntemleri yardımıyla BT yapısına uygun yapısal forma dönüştürülmesi süreçlerini zorunlu hale getirmektedir.

Web kaynakları BT'nin oluşturulmasıyla birlikte bilgilerin doğruluğunun teyit edilmesi amacıyla da sıkça başvurulan kaynak olarak tercih edilmektedir. Bilginin arama motorları üzerinden aratılması, Web sayfaları üzerinden teyit edilmesi, kaynağın güvenilirliğini dikkate alarak güvenilirlik katsayısının oluşturulması literatürde kullanılan doğrulama yöntemleri arasında yer almaktadır.

Bu bölümde Web ölçeğinde bilgi çıkarımı yöntemleri ve doğrulama işlemleri için bilgi güvenilirliğine dayalı çalışmalar ele alınacaktır.

3.1. Web Ayrıştırma

Web ayrıştırma işlemi İnternet ortamından elde edilen dokümanlardan hem bilgi çıkarımı hem de doğrulama aşamaları için ön işlem aşamasıdır. Web ayrıştırma yöntemleri arama motorlarından ürün karşılaştırma uygulamalarına kadar birçok alanda kullanıma sahip yöntemleri içermektedir. Bu yöntemler manuel ve otomatik olarak iki farklı grupta ele alınabilir.

1. Metin örüntü eşleştirme: Bu yöntemde düzenli ifadeler (*regular expressions*) yardımı ile manuel olarak metinlerin ayrıştırılması gerçekleştirilir. Bu yöntem en temel yöntemlerden biri olarak görülmektedir. UNIX grep komutu veya programlama dillerindeki düzenli ifade belirleyicilerin yardımı ile yarı yapılandırılmış metinden ön tanımlı içerikler çıkarılmaktadır.
2. Hyper Text Markup Language (HTML) ayrıştırma: HTML etiketlerine sahip metinde ayrıştırma işlemi gerçekleştirilir. Bu yöntem Web ayrıştırma işlemi için sıkça tercih edilmektedir. XML sorgulama dillerindeki mantığa benzer olarak HTML etiketleri üzerinde XQuery veya HTQL benzeri sorgulama dili ile ayrıştırma yapılmaktadır.

3. Document Object Model (DOM) ayrıştırma: Scriptler ile oluşturulan dinamik HTML etiketlerinin ayrıştırılması geçekleştirilmektedir. Güncel Web sitelerinin büyük kısmı sayfa içeriğinden öte sayfa elementlerini de script yardımı ile oluşturmaktadır. DOM ayrıştırma yönteminde sayfa içindeki yapılar ağaç olarak ele alınmakta ve bu ağaç ayrıştırılarak HTML elementleri ortaya çıkarılmaktadır.
4. Semantik açıklama (semantic annotation) tanıma: Web dokümanlarına yerleştirilen semantik açıklamaların ayrıştırılması gerçekleştirilmektedir. Semantik açıklamalar Web sayfalarında direkt kullanıcıya gösterilmeyen, daha çok arama motoru optimizasyonu amacı ile yerleştirilmiş bilgilerdir. Bu kısımlar sayfa içeriği ve bağlamı ile ilgili bilgiler sunmakta, güncelleme durumu ve tarihi, içerik oluşturucusu gibi bilgileri içermektedir. Günümüzde Web sayfalarının büyük çoğunluğu semantik açıklama metinlerini içeriğinde bulundurmaktadır. Bu içeriklerin ayrıştırılarak analiz edilmesi sayfanın bağlamı ile ilgili ek bilgiler sunmaktadır.
5. Bilgisayar görüntülemeli Web sayfa analizi: Diğer yöntemlere göre daha fazla zorluk barındıran ve bilgisayar destekli görsel analiz yapan yöntemlerdir. Çok fazla tercih edilememesine rağmen bu alanda da yapılmış bazı çalışmalar bulunmaktadır. Bu yöntemde amaç makine öğrenmesi ve bilgisayarlı görü kullanılarak Web sayfası içeriğinin görüntüden çıkarılmasıdır [119].

3.2. Bilgi Çıkarımı

Bilgi Çıkarımı (Information Extraction - IE) yarı yapılandırılmış veya yapılandırılmamış veriden istenen yapılandırılmış verinin elde edilmesini temel alan yöntemlerdir. Doğal dildeki metinler yaygın olarak bu yöntemlerde girdi olarak kullanılmaktadır. Bu yöntemlere ek olarak son yıllarda özellikle, derin öğrenme algoritmalarının gelişmesi ile beraber resim ve video gibi çoklu ortam verilerindeki içeriklerin yazıya dönüştürülmesi de bilgi çıkarımı olarak ele alınabilir. Bilgi çıkarımı yöntemleri Web üzerinden elde edilen veriler üzerinden BT oluşturulmasında önemli yere sahiptir. Bu amaçla geliştirilmiş bazı yöntemler aşağıda listelenmiştir [120].

1. Üçlü yapıları oluşturma: Metin içindeki cümlelerden üçlü olguların ortaya çıkarılması işlemidir. Doğal dil işleme yöntemleri ile cümlelerde nesne, özne ve yüklemlerin bulunması gerçekleştirilir. Elde edilen üçlü yapılar da daha sonra bilgi gösterimi yöntemleri ile ifade edilmekte ve varlıklar arasında ilişkilendirme yapılmaktadır.

2. Varlık ismi tanıma: Cümle içinde tanımlanmış, bilinen varlık isimlerinin (isim, konum vb. özel isimler) belirlenmesidir. Örnek; "Fatih okula gidiyor" cümlesinde Fatih'in kim olduğu daha önceden bilinmemesine rağmen varlık ismi olarak belirlenmesi gereklidir. Bu yöntem ile elde edilmiş varlıklar genelde tekil ayrıştırıcılar ile tanımlanmaktadır [121]. Varlıkların ayrıştırılması için metin bağlamı veya diğer cümleler kaynak olarak kullanılmaktadır.
3. Eşgönderge çözümlemesi (*coreference resolution*) [122]: Cümle içinde önceden belirlenmiş varlığın bağlamsal olarak ilişkilendirilmesidir. Örnek; "Kerem okula gidiyor. O, lise öğrencisi." cümlelerindeki "Kerem" ve "O" varlık isimlerinin aynı kişiye işaret ettiğinin belirlenmesi gerekir. Eşgönderge çözümlemesi tekrarlı varlıkların ortaya çıkarılmasını engellemek amacıyla doğru bir şekilde yapılmak zorundadır.
4. Kural belirleme: Elde edilen üçlü ilişkilerden kuralların ortaya çıkarılmasıdır. Örnek; "Ahmet, Gazi Üniversitesinde okuyor" cümlesinden "KİŞİ KURUMDA okuyor" kuralının belirlenmesi. Varlık ismi tanıma ve eşgönderge çözümlemesi işlemi sonrasında ortaya çıkan üçlü yapılardan kurallar belirlenmektedir [121].

3.3. Açık Bilgi Çıkarımı

Açık Bilgi Çıkarımı (Open Information Extraction - OIE), yaygın olarak metinlerden üçlü veya n-li (*n-ary*) şekilde elde edilen yapısal verinin bilgisayarın anlayacağı şekilde ortaya çıkarılmasını amaçlamaktadır. Bu yöntemler bilgi çıkarımı yöntemlerinin alt kümesi olarak görülebilir. OIE ile elde edilen bilgi ikili mantıksal operatör ile hesaplanabilir doğru (*true*) veya yanlış (*false*) bir doğruluk değerine sahip olur. Metinden elde edilen olası olgular, argümanlar ve ilişkiden oluşan bir küme şeklinde ifade edilir. OIE yöntemleri, bilgi tabanlarının oluşturulması, ilişki çıkarımı (*relation extraction*), soru cevaplama vb. gibi alanlarda kullanılmaktadır. Bilgi çıkarma işlemine göre daha geniş bilgilerin ortaya çıkarılması amacıyla yapılmış bazı çalışmalar OIE kapsamında değerlendirilmektedir. "TextRunner" Web ölçeğinde yapılmış bu tür çalışmaların ilki olarak gösterilebilir [123].

ReVerb: Metin içinde anlamsal olarak daha kapsamlı bilginin elde edilmesidir. Örnek: "Ahmet Mehmet'le anlaşma yaptı" cümlesinden "(Ahmet, anlaşma, yaptı)" üçlüsü çıkarılabileceği gibi "(Ahmet, Mehmet'le, anlaşma yaptı)" üçlüsü de çıkarılabilir. Mevcut örnek için çıkarılan ikinci üçlü anlamsal olarak daha kapsayıcıdır [124].

OLLIE: Belirli bir olgu eksikliğinin ortadan kaldırılmasını amaçlamaktadır. IE'deki eşgönderge çözümlemesi yöntemlerini temel alır. Örnek: "Obama, eski ABD başkanı, Hawaii'de doğmuştur" cümlesindeki "eski ABD başkanı" argümanının aslında gizli özne olan Obama'ya ait olduğu bilgisini ortaya çıkarmaktadır.

CSD: OIE'nin minimalize edilmesi ve iç içe (*nested*) yapıda ifade edilmesini öne süren bir yaklaşıma sahiptir. Örnek: "Büyükelçilik, "Pakistanda 6700 ABD vatandaşı vardır", dedi" cümlesinde "(Pakistanda, 6700 ABD vatandaşı, vardır)" bilgisi *arg2* olarak tanımlanırsa, "(Büyükelçilik, arg2, dedi)" olgusu iç içe geçmiş iki olguyu ifade etmiş olur.

3.4. Web Kaynağının Güvenilirliği

İnternet ve Web'in yaygınlaşması milyonlarca Web sayfasının ortaya çıkmasını da beraberinde getirmiştir. Web üzerinde bulunan sayfaların güvenilirliği 1990'lardan beri aktif araştırma konusu olarak güncelliğini korumaktadır. Kaynağın güvenilirliği yanlış bilginin yayılmasının önlenmesi, gereksiz bilgi (*spam*) akışının durdurulması, arama motorlarında anlamlı sıralamanın yapılması ve Web'den otomatik bilgi çıkarımlarında güven değerlerinin tanımlanması için önemlidir. Özellikle, arama motorları için Web sayfalarının ilgili aramalarda gösterim sırası ticari olarak da ciddi önem arz etmektedir. Web sayfalarının güvenilirliğini belirlemek için literatürde hem arama motorlarına sahip ticari kuruluşlar tarafından hem de akademik kurumlar tarafından yürütülen çalışmalar bulunmaktadır. Web üzerinde kaynak güvenilirliğini sağlamak için W3C tarafından da "W3C Provenance" çalışmaları başlatılmıştır [125]. Google tarafından uzun süre aktif olarak kullanılmış PageRank [126], Stanford Üniversitesi ve Yahoo tarafından geliştirilmiş TrustRank [127], J. Kleinberg tarafından önerilen HITS [128] algoritmaları Web kaynağının güvenilirliğini hesaplamak için yapılmış ilk dönem çalışmalara örnek olarak gösterilebilir. Benzer çalışmalar günümüzde de güncelliğini korumakla birlikte sadece bağlantı (*link*) odaklı yaklaşımlar sundukları göz önünde bulundurulmalıdır.

Web sayfaları sadece bağlantı odaklı bilgiler barındırmamaktadır. Web sayfalarında kullanıcı davranışlarından ziyaret istatistiklerine, içerikten görsel tasarıma birçok işlenebilir veri bulunmaktadır. Kaynak güvenilirliğinin belirlenmesi için bu verileri de ele alan farklı çalışmalar mevcuttur. Özellikle, sosyal medyanın aktif kullanımı ve Web

sayfalarında sosyal medya entegrasyonlarının yaygınlaşması bahsedilen verilerin daha da zenginleşmesine katkı sağlamıştır. [129].

3.4.1. Web sıralama algoritmaları

PageRank [126] ve HITS [128] gibi geleneksel Web sıralama algoritmaları sayfa güvenilirliklerini (*credibility*) sayfadaki bağlantıları temel alarak hesaplamaktadır. Bu algoritmalardaki temel yaklaşım akademik yayınlardaki atıf mantığını güven değeri olarak kabul etmektir. "Daha çok atıf verilen çalışma daha güvenilirdir" yaklaşımı daha çok dış bağlantıya sahip sayfalar için de uygulanmaktadır. Zaman içinde bu yöntemlerin kırılganlığı ve manipülasyona açık olduğu ortaya çıkmıştır. Spam sayfalar oluşturularak arama motorlarının yanıltılması bu manipülasyonlara örnek olarak gösterilebilir. Bu sorunları ortadan kaldırmak için bağlantılardan oluşan çizge yapısı kullanılarak spam bağlantıların önlenmesi, istatistiksel anomalilerin tespit edilmesi, güven yayılımına bağlı sıralamanın yapılması ve çizge düzenlemesini temel alan çalışmalar yapılmıştır [129].

3.4.2. İçerik tabanlı yaklaşımlar

Bu yaklaşımlar Web sayfasının güvenilirlik değerini sadece bağlantıları temel alarak değil tüm içeriği analiz ederek belirlemektedir. Bir Web sayfası başlık (*head*) ve içerik (*body*) kısmından oluşmaktadır. İçeriği temel alan çalışmalarda metnin karakter bazında analizi, anlamsal ve dilsel değerlendirmeler, tipografik inceleme, kelime sayısı, içeriğin tekrarlanma oranı dikkate alınmaktadır. Makine öğrenmesi yaklaşımlarının yaygınlaşması ile içeriğin kategorize edilmesi, konu başlıklarının belirlenmesi, kullanıcı yorumları ve sosyal medya entegrasyonlarından elde edilen sosyal medya puanlaması da içerik tabanlı kaynak güvenilirliği belirleme yöntemlerinde önem kazanmıştır. İçerik tabanlı yaklaşımlara örnek olarak [130, 131] gösterilebilir. Bu yaklaşımlara başka bir örnek de Wawer ve ark. tarafından yapılan çalışmadır [132]. Bu çalışmada öznitelik olarak duraklama işaretlerinin sıklığı, duygu analizi, içeriğin kategorisi ve gramer hataları gibi parametreler tercih edilmiştir.

3.4.3. Görsel yaklaşımlar

Günümüzde çok popüler olmamakla birlikte Web sayfalarının tasarımını dikkate alarak güven ölçümü yapan çalışmalar bulunmaktadır. Wouter ve ark. tarafından yapılan bir deneysel çalışmada aynı içeriğe sahip Web sayfaları farklı renklendirmelerle kullanıcılara sunulmuştur [133]. Sağlık, devlet kurumları ve kurumsal sayfalar için yapılan deneylerde farklı renklerin kullanıcı açısından farklı güven algısı oluşturduğu gösterilmiştir. Bu çalışma Almanya ve Hollanda gibi yakın iki ülkede bile sonuçların kültürel farktan kaynaklı değişiklik gösterdiğini ortaya koymuştur. Görsel yaklaşımların incelendiği başka bir çalışma da Lindgaard ve ark. tarafından yapılmıştır [134]. Bu çalışmada da mobil ve masaüstü kullanılabilirliği ve Web sayfası kullanıcı arayüzünün estetik tasarımı gibi etkenlerin kullanıcı güvenini etkilediği vurgulanmaktadır. Görsel tasarım çalışmaları ile elde edilen sonuçlar günümüzde önerilen bütünleşik çözümlerde bir öznitelik olarak değerlendirilmektedir.

3.4.4. Kullanıcı tabanlı yaklaşımlar

Bazı yaklaşımlar kullanıcı davranışları ile birden fazla yöntemin birleştirilmesini önermektedir. Örneğin, Jessen ve ark. tarafından yapılan bir çalışma sosyal medya profilleri, beğeni sayıları ve Web sayfasına verilen bağlantıların hangi domainlerden geldiği bilgisine dayalı güvenilirlik hesaplaması yapmaktadır [135].

Kullanıcı davranışını temel alan yaklaşımlar da kullanıcı tabanlı yaklaşımlara örnek olarak gösterilebilir. Tıklama sayıları ve sıklığını inceleyen yönteme Dou ve ark. çalışması [136], davranış temelli yaklaşıma ise Liu ve ark. yaptığı çalışma [137] örnek gösterilebilir.

3.4.5. Ziyaret sayılarının hesaplanması

Amazon tarafından hizmete sunulan ve 2022 yılında sonlandırıldığı duyurulan “Alexa Web” arama robotları da Web sayfalarına ziyaret sıklığını temel alarak sıralama (*rank*) değeri oluşturmaktadır. Bu yöntem de sayfa bağlantıları üzerinden işlem yapan Web sıralama algoritmaları gibi Web sayfalarının otomatik ziyaret botları ile ziyaret edilerek manipüle edilmesine karşı korunaksızdır [138].

3.5. Kaynak Güvenilirliği Parametreleri

Kaynak güvenilirliğinin hesaplanması için incelenen güncel çalışmalarda Dong ve ark. [139] ve Esteves ve ark. [138] çalışmaları tez çalışmasının kapsamı açısından özellikle öne çıkmaktadır.

Web sayfasının güvenilirliği ile elde edilen bilginin güvenilirliği karşılıklı etki oluşturmaktadır. Kaynağın güvenilir olması elde edilen bilginin güvenilirliği ön kabulünü oluşturduğu gibi bilginin güvenilirliği de kaynağın güvenilirliğini belirleyebilir. Bu nedenle yapılan güncel çalışmalar PageRank, TrustRank gibi mevcut sıralama yöntemlerine de alternatif oluşturmaktadır [139].

Esteves ve ark. yaptığı çalışma [138] Web üzerinde sahte haberlerin belirlenebilmesi için kaynak güvenilirliğine önem vererek haberin güvenilirliğini belirlemektedir. Çalışmada seçilen öznitelikler içerik tabanlı ve sosyal tabanlı olmak üzere iki başlıkta toplanmıştır. İçerik tabanlı öznitelikler kendi içinde metin, görünüm ve meta bilgi alt başlıklarına sahiptir. Sosyal tabanlı öznitelikler ise sosyal medya popülerliği, genel popülerlik ve link mimarisi başlıklarında ele alınmıştır. Çalışmada tercih edilen öznitelikler açıklamaları ile beraber aşağıda sunulmuştur. Benzer öznitelikler birleştirilerek aynı başlık altında gösterilmiştir.

1. Güncellik: sayfanın değiştirilme zamanı açısından güncelliğini ifade etmektedir. Önbellekte tutulan ilk ve son sayfa zamanlarının farkı alınarak değerlendirilir.
2. Domain: sayfanın kök domain bilgisini ifade etmektedir. (Örneğin; org, gov, com vb.)
3. Yetkili tespiti: sayfanın HTML içeriğinde e-posta, adres, telefon bilgisi gibi yetkililerin kimliğini doğrulamaya yönelik bilgilerin bulunma durumunu ifade etmektedir.
4. Dış bağlantılar: Web tabanlı protokolleri kullanan dış bağlantıların sayısını göstermektedir.
5. Metin kategorisi: Naive Bayes sınıflandırıcı ile metnin kategori olasılığını göstermektedir. LexRank [140] ve LSA [141] yöntemleri ile belirlenen metin kategorileri de öznitelik olarak dikkate alınmıştır. Örnek kategoriler iş, eğlence, siyaset, din, spor vb. şekilde ifade edilmiştir.
6. Okunabilirlik: Si ve Callan [142] tarafından belirlenen *R* okunabilirlik ölçümlerinin vektör karşılığını ifade etmektedir.

7. Spam ölçümü: Spam sınıflandırma sonucunda elde edilen değeri göstermektedir.
8. Sosyal etiketler: Sayfada bulunan sosyal medya etiketlerinin sıklığını göstermektedir.
9. Kaynak erişimi: Sayfa erişiminin herkese açık olup olmadığını ifade etmektedir.
10. PageRankCC: CommonCrawl Corpus tarafından hesaplanan PageRank değerini göstermektedir.
11. Dilsel içerik: Dilsel hata puanını ve duygu analizi sonucunu göstermektedir.
12. HTML2Seq: *bag-of-words* [143] yaklaşımından yola çıkarak HTML etiketlerinin dağılımını göstermektedir.

3.6. Web Kaynaklarından Doğrulama

Web kaynaklarının güvenilirliği tez kapsamında değerlendirilebilecek farklı akademik çalışmalarda [144–148] ele alınmıştır. Bu çalışmalar kaynak güvenilirliğini tanımlamadan öte bilginin Web'de doğrulanmasını amaçlamaktadır. Her ne kadar tez kapsamında bilginin öncelikli olarak Web'den doğrulanması değil de güven değeri ile elde edilmesi amaçlansa da Web üzerinden elde edilen bilginin güven değerinin hesaplanmasında kullanılan yöntemler açısından bu çalışmalar da incelenmiştir. Bu çalışmalar bilginin doğrulanması ile beraber BT'nin arındırılması açısından da değerli kaynak niteliğindedir [27, 149].

Bu yayınlara örnek olarak Huaman ve ark. tarafından BT üçlüsünün güven değerini hesaplamak amacıyla yapılmış çalışma gösterilebilir [150]. Çalışma özellikle adres ve iletişim bilgilerinin "Yandex StreetMap" üzerinde doğrulanmasını amaçlamaktadır. Bu başlıkta değerlendirilebilecek bir başka çalışma Trummer tarafından pekiştirmeli öğrenme yardımıyla yapılmıştır [151]. Örnek bir başka çalışma da Souza ve ark. tarafından kaynak doğrulama yapısının oluşturulması için hazırlanmıştır [152].

"Web kaynaklarından doğrulama" yaygın bir araştırma başlığı olmamakla birlikte bu konuda ele alınabilecek en kapsamlı çalışma Gerber ve ark. tarafından hazırlanan "DeFacto" makalesidir [153]. Bu çalışmada Web sayfalarının değerlendirilmesi ayrı bir başlık olarak ele alınmıştır. Yazarlar Web sayfasının uygunluğunu üç kriter altında değerlendirmektedirler.

1. Konu başlığının Web'de yaygınlığı: Bu değer konu başlığına sahip farklı Web kaynaklarının sayısını göstermektedir.

2. Konu başlığının arama sonuçlarında yaygınlığı: Doğrulanması amaçlanan bilgi için arama sonuçlarında konu başlığının yaygınlığını ifade etmektedir.
3. Konu başlığının kapsayıcılığı: Belirli bir Web sayfasında konu başlığının bulunma sıklığının toplam elde edilen Web sayfalarındaki sıklığa oranını ifade etmektedir.

4. GÜVEN DEĞERLERİNE BAĞLI BİLGİ TABANI DOĞRULAMA

Bu bölümde tez kapsamında önerilen çizge yöntemi ve BT modeli ana hatlarıyla ele alınmıştır. Ayrıca oluşturulan BT üzerinde doğrulama ve temizleme işlemleri için önerilen çizge yöntemine dayalı yayılma işlemi ve bulanık Petri ağlarını temel alan yayılma işlemi detaylı olarak açıklanmıştır.

4.1. Geliştirilen Çizge Gösterimi

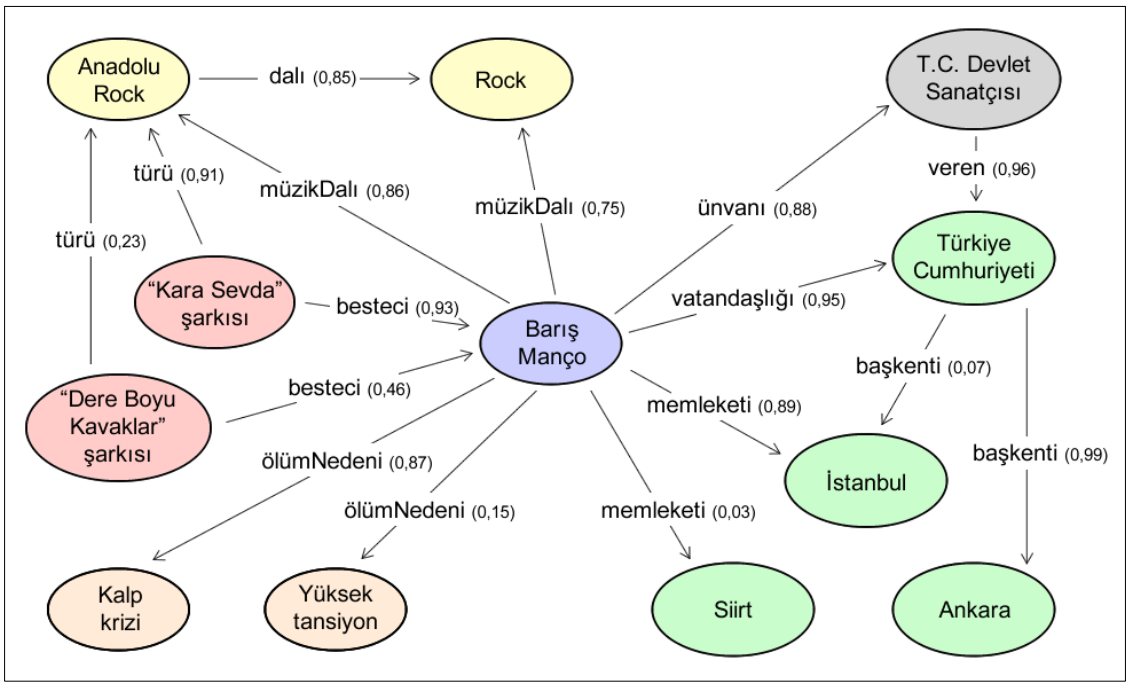
Yapılan literatür taraması ve örnek BT incelemeleri sonucunda tez kapsamında oluşturulan BT için bilgi gösterimi olarak çizge veri yapısının kullanılmasına karar verilmiştir. Önerilen çizge, özne *(s)*, yüklem *(p)* ve nesne *(o)* üçlülerinden oluşmaktadır. Özne ve yüklemler çizge üzerinde birer düğüm, yüklem veya ilişki ise bu düğümler arasındaki bağlantı olarak gösterilmektedir. Bahsedilen çizgenin yönlü (*directed*) bir çizge olduğu göz önünde bulundurulmalıdır. Burada çizgenin yönü ilişkinin yönünü ifade etmektedir. BT'nin yönlü çizgelerden oluşması kabul edilmekle birlikte her bir tekil özne ve nesne düğümleri arasında tanımlı *p* yüklemi için ters yüklem (*p'*) veya ters ilişkinin varlığı göz önünde bulundurulmalıdır. Üçlüler dışında çizge üzerinde barındırılan bilgiler *literal* olarak tanımlanmaktadır. *Literal* hem düğüm hem de ilişki üzerindeki özellikleri tanımlamak için ifade edilir. Örnek bir çizge yapısı Şekil 4.1'de gösterilmiştir.

Önerilen BT'nin mevcut çalışmalardan en önemli farkı ilişkilerin bilginin elde edilme yöntemi ve kaynağına göre 0 ve 1 arasında normalleştirilmiş güven değerine sahip olmasıdır. Güven değeri *Conf(h, r, t)* olarak ifade edilmektedir. Bu durum bir özne için aynı ilişkinin birden fazla nesne ile bağlantı yaptığı anlamına gelmektedir. Dolayısıyla önerilen BT'nin mevcut BT'lerden bir diğer farkı daha fazla ilişki barındırıyor olmasıdır. Tez kapsamında öne sürülen modeldeki *Conf(h, r, t)* değerinin hesaplanması aşağıdaki üç farklı durum için güncellenmiştir. Bunlar, olgunun dış kaynaktan elde edilmesi, mevcut değerin güncellenmesi ve güven değerinin yayılma etkisi ile hesaplanmasıdır.

Olgunun dış kaynaklardan elde edilmesi: Web kaynağından elde edilen üçlüler için *Conf(h, r, t)* değeri bilginin elde edildiği sayfanın güvenilirliği (*trust rank*), aynı bilginin farklı sayfalarda tekrarlanma sayısı, Web sayfası başlığı ile üçlüde bulunan varlıkların yakınlığı gibi parametrelere göre hesaplanır [153].

Olgunun mevcut güven değerinin güncellenmesi: Dış kaynaktan elde edilen üçlü çizgeye yerleştirilirken var olan güven değeri eski güven değeri ile karşılaştırılarak yeni güven değeri elde edilir. Bu işlem güven değerinin güçlenmesi, zayıflaması veya mevcut değeri koruması olarak sonuçlandırılır.

Olgunun güven değerinin BT üzerinde yayılma etkisi ile hesaplanması: Bu durumda yeni gelen olgunun güven değeri çevresinde bulunan üçlüler üzerinde yayılma etkisi oluşturarak mevcut güven değerlerinin değişmesini tetiklemektedir. Bu işlem çizge yapısı üzerinde zincirleme bir işlem olarak devam etmektedir. Yayılma etkisinin modellenmesi Bölüm 4.3'de deneysel sonuçları ise Bölüm 6.6'da detaylı olarak ele alınmıştır.

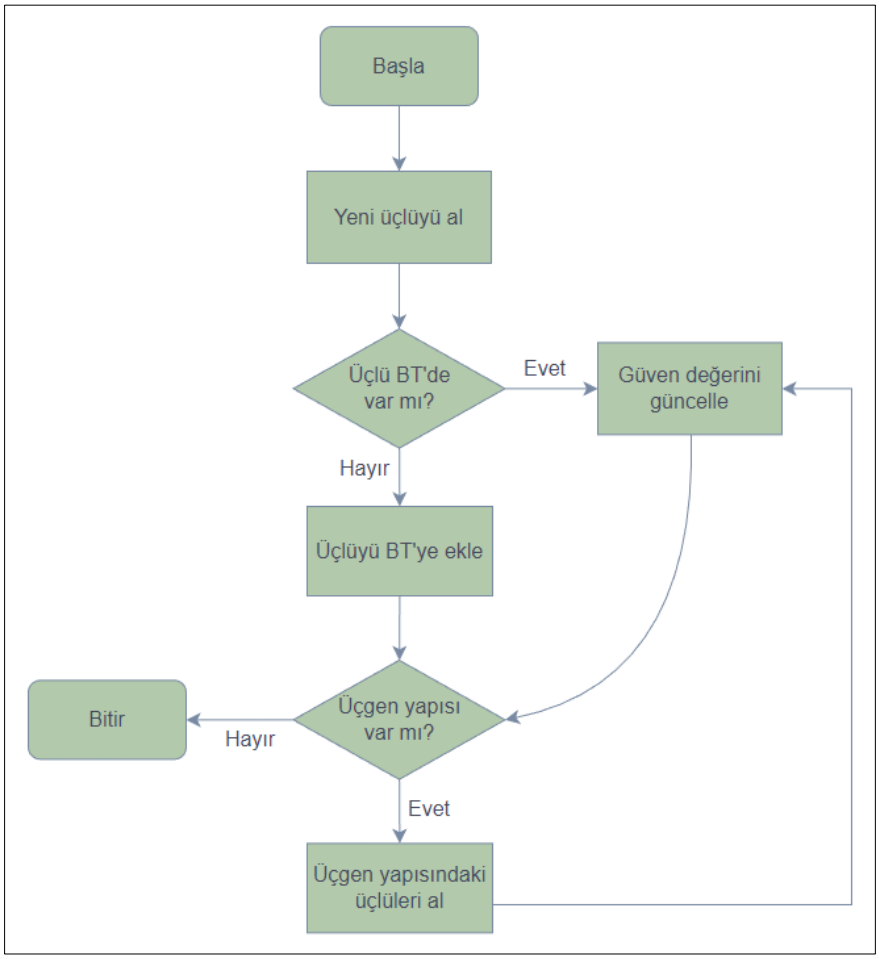


Şekil 4.1. Önerilen BT için örnek çizge yapısı

4.2. Önerilen BT Modeli

Tez kapsamında önerilen model BT'nin dış kaynaklardan elde edilen bilgilerle oluşturulduğunu ve sürekli güncellendiğini varsaymaktadır. Dış kaynaktan elde edilen bilgi *Conf(h, r, t)* güven değeri ile birlikte modele sunulmaktadır [153]. Sistem açık dünya varsayımı ile elde edilen bilgiyi BT'ye eklerken güven değerleri üzerinde yeniden hesaplama yapılmaktadır. Aynı zamanda kapalı dünya varsayımını kullanarak BT'ye eklenen yeni bilginin BT üzerinde yayılma etkisi göstererek tekrar değerlendirilmesi

yapılmaktadır. Yeni bilginin eklenmesi veya var olan bilginin güven değerinin güncellenmesi BT üzerinde bulunan mevcut olgular üzerinde de etki oluşturmaktadır. BT modelinin akış şeması Şekil 4.2'de gösterilmiştir.



Şekil 4.2. Önerilen BT'ye üçlü ekleme işlemi için akış şeması

4.3. BT Üzerinde Güven Değerlerinin Yayılması

Güven değeri tez kapsamında geliştirilen ve arıtılan BT'de en önemli veriyi ifade etmektedir. Bu değer bilginin elde edildiği kaynağın güvenilirliğini temel alarak farklı bir yaklaşım ortaya koymaktadır. Tez çalışmasında yapılan tüm araştırmalarda BT'ye eklenen veya güncellenen bilginin güven değerinin sadece mevcut üçlünün güven değerini etkilediği görülmüştür. Her ne kadar kural tabanlı oluşturulan bazı BT'lerde mevcut güven değerlerinden yeni üçlülerin güven değerlerinin hesaplanması önerilse de bu hesaplama sadece yeni üçlülerle sınırlı kalmaktadır. Bu çalışmalarda dikkat çeken bir diğer konu da mevcut güven değerlerinin farklı bilgi yerleştirme yöntemleri ile elde edilmesidir [27, 33,

149]. Dolayısıyla bu güven değerleri dış kaynaktan ziyade bilginin kendi ifade ediliş biçiminden kaynaklı bir değerdir.

Güven değeri başka BT'ler gibi yapılandırılmış veya Web kaynakları gibi yarı yapılandırılmış dış kaynaklardan elde edilebileceği gibi çevrimdışı yöntemler aracılığıyla BT üzerinden de elde edilebilir. Elde edilen güven değeri sadece mevcut üçlü ile sınırlı kalmadan BT'de bulunan diğer üçlüler açısından da anlamlı bilgi barındırır [154]. Bu durumu daha iyi ifade edebilmek açısından aşağıdaki örnek incelenebilir.

Örnek 1. Şekil 4.1'de gösterilen örnek çizge modelinde *Barış Manço*'nun *vatandaşlığı – Türkiye Cumhuriyeti* üçlüsüne ait güven değerinin güncellendiğini varsayalım. *Türkiye Cumhuriyeti* varlığı *T.C. Devlet Sanatçısı* varlığı ile *veren* ilişkisine sahip olduğundan güncellenen güven değerinin *Barış Manço*'nun *Unvanı* bilgisinin de güven değerini etkilemesi anlamlıdır.

Örnek 1'de tanımlanan etki BT üzerinde bağlantılı tüm üçlüler için tekrarlanarak devam edeceğinden güven değerinin yayılma etkisi olarak adlandırılabilir. Yayılma etkisinin oluşturulmasında ana fikir zayıf ve güçlü güven değerlerinin hem elde edilen üçlü üzerinde hem de bu üçlü ile bağlantılı diğer üçlüler üzerinde etki oluşturarak BT'nin güvenilirlik açısından istikrarlı duruma getirilmesidir. Bu anlamda dış kaynaktan elde edilen bilgi ilk değerinden bağımsız olarak BT'ye dahil edilir. BT'ye bilgi ekleme işlemi devam ettiği sürece yeni eklenen zayıf bağlantılar ilintili diğer zayıf bağlantıların da temizlenmesine ve sistemin saflaştırılmasına neden olacaktır. Bu durum güçlü güven değerleri için de sistemin güçlendirilmesi anlamına gelmektedir.

Kural tabanlı yöntemlerde öne çıkan başka bir özellik üçlüde bulunan tüm ilişkilerin ters ilişkiye de sahip olduğu varsayımıdır [113]. Bu özellik Şekil 4.1'deki örnek çizge yapısı üzerinden açıklanabilir.

Örnek 2. Aşağıdaki her iki üçlü çifti için ters ilişkilerin geçerliliği gösterilmektedir.

- *(Barış Manço, ölüm Nedeni, Kalp krizi) → (Kalp krizi, ölen, Barış Manço)*
- *(Barış Manço, memleketi, İstanbul) → (İstanbul, doğumlu, Barış Manço)*

Bu özellik E varlıklar kümesi için $\forall x, y \in E, \boldsymbol{v}(x, r_i, y) \Rightarrow \boldsymbol{v}(y, r_j, x)$ şeklinde genelleştirilir ve BT çizgesini yönsüz çizge olarak ele almayı olanaklı kılar.

Yayılma etkisinin doğru bir şekilde tanımlanabilmesi için aşağıda gösterilen bazı parametrelerin dikkate alınması gerekmektedir:

1. Yayılma kuralları: güven değerinin BT üzerinde hangi kurallar çerçevesinde yayılacağını tanımlar. Bu kurallar eklenen veya güncellenen bilgi ile ilişkili bilgileri belirlemektedir.
2. Yayılma hesaplaması: yayılma etkisinin mevcut güven değerlerini nasıl değiştireceğini ortaya koyar.
3. Yayılma genişliği: BT üzerinde yayılma etkisinin hangi genişlikte ve hangi yöntem uygulanarak sınırlandırılacağını belirler.

Bu parametrelerin tanımlanması için bazı ön bilgiler sunulacaktır. Bu bilgiler yayılma kurallarını, güven değerlerinin yeniden hesaplanması için önerilen yaklaşımları ve yayılma işleminin karakteristiğini ele almaktadır.

4.4. Yayılma Kuralları

Kural tabanlı çıkarım yöntemlerinde aşağıdaki kurallar kullanılmaktadır [113]. Bu kurallar güven değerinin ilişkili üçlüler üzerinde yayılma kuralları olarak kabul edilecektir.

1. Birleşme kuralı *(composition rule)*: Herhangi üç *x*, *y*, *z* varlığı için r_k ilişkisi r_i, r_j ilişkilerinin birleşimidir.

$$\forall x, y, z \in E, \boldsymbol{v}(x, r_i, y) \wedge \boldsymbol{v}(y, r_j, z) \Rightarrow \boldsymbol{v}(x, r_k, z) \quad (4.1)$$

2. Ters ilişki kuralı *(inverse rule)*: Herhangi iki *x* ve *y* varlığı arasında r_i kuralı bulunuyorsa *y* ve *x* arasında da r_j ilişkisi bulunmaktadır.

$$\forall x, y \in E, \boldsymbol{v}(x, r_i, y) \Rightarrow \boldsymbol{v}(y, r_j, x) \quad (4.2)$$

3. Simetri kuralı *(symmetric rule)*: Herhangi iki *x* ve *y* varlığı arasında r_i, ilişksi *y* ve *x* arasındaki r_j ilişkisi ile aynı ise bu varlıklar arasında ilişki simetrisi bulunmaktadır.

$$\forall\, x, y \in E, \boldsymbol{v}(x, r, y) \Rightarrow\ \boldsymbol{v}(y, r, x) \tag{4.3}$$

4. Alt ilişki kuralı *(subrelation rule)*: Herhangi iki *x* ve *y* varlığı arasında birbirinden farklı iki r_i, ve r_j ilişkileri bulunuyorsa r_i, ve r_j ilişkileri alt ilişkidir.

$$\forall\, x, y \in E, \boldsymbol{v}(x, r_i, y) \Rightarrow\ \boldsymbol{v}\big(x, r_j, y\big) \tag{4.4}$$

Açıkça görüldüğü gibi birleşme kuralı dışında diğer kurallar için güven değerinde yayılma söz konusu değildir. Sadece mevcut üçlünün değeri yeni gelen değer ile güncellenebilir. Birleşme kuralında ise üçgen yapısı üzerinden yayılma sağlanması mümkündür.

4.5. Güven Değerinin Hesaplanması

Yayılma hesaplaması tanımlanan yayılma kuralları için yeni güven değerlerinin hesaplanma yöntemini ortaya koymaktadır. Bu yöntem BT'ye yeni bilgi eklenmesi durumunda veya yayılma kuralı sonucunda güncellenen güven değerlerinin hesaplanması için de kullanılmaktadır.

Kural tabanlı çıkarım yöntemlerinde yeni kuralın doğruluk değerinin belirlenmesi için probabilistic soft logic [112], Markov mantık ağı (MLN) [114], Bayes ağı veya t-norm bulanık mantık hesaplamaları tercih edilmektedir. İncelenen çalışmalarda probabilistic soft logic yönteminin ağırlıklı olarak tüm bilginin dağılımı üzerinden sonuç elde edilmesi yaklaşımlarında tercih edildiği görülmüştür [112]. Bayes ağı ise yönlü çizgelerde kullanılmaktadır [115]. Her ne kadar oluşturulan BT yönlü bir çizge olsa da ters ilişki ve simetrik ilişki kuralları üçlüler arasında güven değerlerinin yönlü olmasını önemsiz hale getirmektedir.

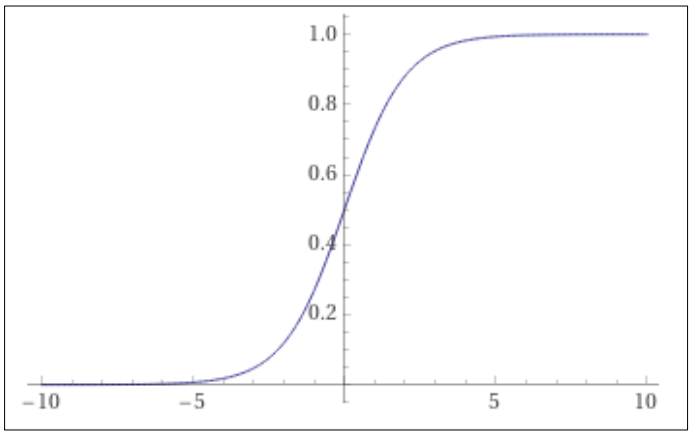
Kural çıkarımlarında tercih edilen bir başka yöntem de Markov mantık ağıdır. Bayes ağından farklı olarak Markov mantık ağı yönsüz çizgeyi temel almaktadır. Bu ağlarda çıkarım işlemi Markov zincirine dayanmaktadır. Çizge yapısının büyümesi durumunda yaklaşık çıkarım sonuçlarının elde edilmesi için Markov Zincirli Monte Carlo (Markov Chain Monte Carlo – MCMC) yöntemi tercih edilmektedir [113].

Tez kapsamında önerilen model güven değerinin güncellenmesi ve yayılmasını dikkate almaktadır. Bu nedenle çevrimdışı yaklaşımlarla hesaplanan veya çevrimiçi yaklaşımlarla

elde edilen güven değerinin BT'de mevcut olan güven değerini nasıl güncelleştireceğini hesaplamaktadır. Bu hesaplama yayılma sırasında güncellenecek güven değerleri için de geçerli olacaktır. Bu anlamda güven değerinin hesaplanmasının mevcut çalışmalarla ortak yönleri bulunmaktadır. Aynı zamanda yeni güven değerlerinin hesaplanması doğruluk değerlerinin hesaplanmasından daha farklı bir yaklaşıma ihtiyaç duymaktadır. Her ne kadar güven değeri bilginin doğruluk değeri ile ilintili olsa da bilginin doğası açısından bu iki değerin farklı anlam ifade ettiği açıktır.

Yeni güven değerinin güncellenmesi için BT'de bulunan üçlünün güven değeri ve yeni elde edilmiş güven değeri birleştirilerek sonuç elde edilebilir. Burada belirli bir olguya olan güvenin (x) dış kaynaktan bu olgu ile ilgili yeni elde edilmiş güven (y) karşısındaki değişimi referans alınabilir. Eski güven değeri yüksek ve yeni elde edilmiş değer de yüksekse mevcut olguya güven ivmeli artış gösterir. Eski güven değerinin yüksek, yeni güven değerinin düşük olması durumunda ise eski güven yavaş bir şekilde azalır. Tam tersi durumda; eski güven değeri düşük, yeni elde edilen değer yüksekse güven yavaş bir şekilde artar. Hem eski güven değerinin hem de yeni güven değerinin düşük olması durumunda güven ivmeli bir şekilde azalır. Güven değerlerinin düşük veya yüksek olması eşik değeri (λ) üzerinden tanımlanabilir.

Girdi ve çıktı arasındaki bu tarz ilişki *sigmoid* fonksiyonunu yansıtmaktadır. Şekil 4.3'de görüldüğü gibi sigmoid fonksiyonları düşük girdi değerlerini zayıflatma, yüksek değerleri ise güçlendirme eğilimi gösterir. Bu durum girdinin tanımlanmış eşik değerin üstünde veya altında olmasına göre değişir.



Şekil 4.3. Örnek sigmoid fonksiyonu

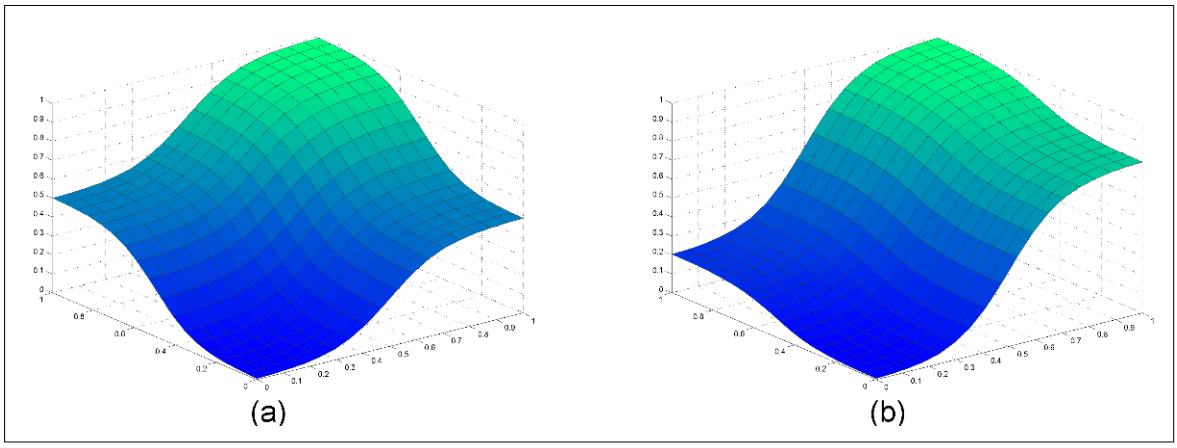
Sigmoid fonksiyonu ailesinden lojistik fonksiyonu, ters trigonometrik fonksiyonlar (*arctan*), hiperbolik fonksiyon vb. fonksiyonlar tercih edilebilir. Tez çalışmasında hesaplama işlemleri için lojistik fonksiyon (Eş. 4.5) tercih edilmiştir.

$$f(x) = \frac{1}{1+e^{-x}} \tag{4.5}$$

Güven değerlerinin {0, 1} aralığında olduğu kabul edilirse lojistik fonksiyonun da bu aralıkta olması beklenmektedir. Dönüşüm işlemi sonucunda lojistik fonksiyonu x ve y ekseninde {0, 1} aralığına daraltılarak istenen çıktı sağlanmaktadır.

Sigmoid fonksiyonun oluşturulmasında bir diğer parametre girdi sayısının iki eksen üzerinde tanımlı olmasıdır. Bu durum Eş. 4.6 şeklinde tanımlanır ve Şekil 4.4 (a)'da gösterilen düzlemi oluşturur.

$$f(x,y) = \frac{1}{1+e^{-x}} + \frac{1}{1+e^{-y}} \tag{4.6}$$



Şekil 4.4. Eşit ağırlıklara sahip (a) ve dikey parametre ve ağırlık çarpanları tanımlı (b) lojistik fonksiyon düzlemi

Ön tanımlı eşik değerine göre stabil sistemin sağlanması amacıyla sigmoid fonksiyonun dikey parametrelerinde değişiklik yapılarak istenen durum elde edilir. Çıktı değerlerinin tanımlanmasında bir diğer parametre girdi olarak kullanılan eski ve yeni güven değerlerinin ağırlık çarpanlarıdır. Örneğin; mevcut veya eski güven değerinin çıktı üzerindeki etkisi 0,8 ile çarpılarak ağırlık değeri değiştirilebilir. Dikey parametre ve ağırlık çarpanlarının tanımlandığı örnek lojistik fonksiyon Eş. 4.7'de gösterilmiştir.

$$f(x,y) = w_1 * \frac{1}{1+e^{-10x+5}} + w_2 * \frac{1}{1+e^{-10y+5}} \quad (4.7)$$

Eş. 4.7de tanımlı fonksiyon için $w_1 = 0{,}8$, $w_2 = 0{,}2$ olarak belirlenirse elde edilecek lojistik fonksiyona Şekil 4.4 (b)'deki düzleme dönüşecektir.

4.6. Çizge Üzerinde Yayılma İşlemi

Güven değerinin BT üzerinde yayılma işlemi güven değeri güncellenen üçlünün komşu (*adjacent*) ve karşı (*opposite*) üçlüleri ile belirlenmiş üçgenlerin tespit edilmesi ile başlatılır. Üçgen yapısı (*h, r_1, t*) üçlüsü için *(h, r_1, t) → (t, r_2, x) → (x, r_3, h)* şeklinde tanımlanır. Burada (*h, r_1, t*) en son güncellenen üçlüyü, *(x, r_3, h)* güncellenen üçlü ile ortak *head* varlığını paylaştığı için komşu (*adjacent*) üçlüyü, *(t, r_2, x)* ise *head* varlığının karşısında yer aldığı için karşı (*opposite*) üçlüyü ifade etmektedir. Bu üçlüler için güven değerleri hesaplanarak kuyruk üzerinde güncellenen her bir üçlünün komşu ve karşı üçlüleri bitinceye kadar güncelleme işlemi devam ettirilir. Tanımlanmış eşik değerin altında güven değerine sahip ilişkiler BT'den silinir. Yayılma işlemi için sözde kod "*GüvenYayılımı*" ve "*GüvenDeğişimi*" fonksiyonları olarak aşağıda gösterilmiştir.

```
Triple[] gezilenÜçlüler
Queue üçlüKuyruğu
Graph G

GüvenYayılımı(üçlü)
begin
      enqueue üçlü to üçlüKuyruğu
      add üçlü to gezilenÜçlüler

      while üçlüKuyruğu ≠ ∅ do
      begin
      t ← dequeue üçlüKuyruğu
      yeniGüven ← güven of t
      üçgenler ← t üçlüsü için üçgenleri bul
      if üçgenler ≠ ∅ then
            for all üçgen in üçgenler do
                  komşuÜçlü ← komşu Triple of üçgen
                  karşıÜçlü ← karşı Triple of üçgen
                  GüvenDeğişimi(komşuÜçlü, yeniGüven)
                  GüvenDeğişimi(karşıÜçlü, yeniGüven)
               end for
            end if
      end
end
```

```
GüvenDeğişimi(üçlü, yeniGüven)
begin
      if üçlü ∉ gezilenÜçlüler then
            hesaplananGüven ← yeniGüven değerini sigmoid fonksiyonu ile
hesapla
            if hesaplananGüven ≤ eşikDeğer then
                  remove üçlü from G
            else
                  güven of üçlü ← hesaplananGüven
                  enqueue üçlü to üçlüKuyruğu
                  add üçlü to gezilenÜçlüler
            end if
      end if
end
```

Güven değerlerinin BT üzerinde yayılma karakteristiği farklı yaklaşımlar ile belirlenebilir. Bu yaklaşımlar yayılmanın etkinliği, çapı ve eşzamanlılığı başlıkları altında incelenecektir.

Yayılmanın etkinliği

Bu kriter yeni gelen güven değerinin ilişkiler üzerinde hangi etkinlikle yayılacağını belirlemektedir. Güven değeri yayılmanın başladığı kaynakta sahip olduğu değeri yayıldığı tüm ilişkiler üzerinde sabit şekilde devam ettirebileceği gibi başlangıç noktasından uzaklaştıkça etkisini azaltarak da devam ettirebilir. Etkinliğin sabit tutulması BT üzerinde daha güçlü değişimlere yol açacaktır. Tam tersi durumda güven değerinin etkinliği sönümlenerek azalacaktır. Güven değerinin zayıflatılması sigmoid fonksiyonunda dikey parametrelerin değiştirilmesi ile yapılır.

Yayılmanın çapı

Yayılma işleminin çapı güven değerinin başlangıç noktasından kaç atlama (*hop*) uzaklığa etki edeceğini belirlemektedir. Bu kriter sınırsız olarak belirlendiğinde üçgen bağlantıya sahip tüm ilişkiler etkilenecektir. Yayılma çapının tanımlanması durumunda, çap başlangıç ilişkisinden yapılan atlama sayısı olarak belirlenir ve tanımlı sınıra gelindiğinde yayılma işlemi durdurulur. Her bir atlama komşu ve karşı ilişkilerin önceki üçgeni içine almayacak şekilde diğer üçgenlere geçişi olarak tanımlanır. Yayılmanın çapı işlem performansı ile doğrudan ilişkili olduğu için deneysel çalışmalarda farklı atlama sayısı için elde edilen sonuçlar paylaşılacaktır.

Yayılmanın eşzamanlılığı

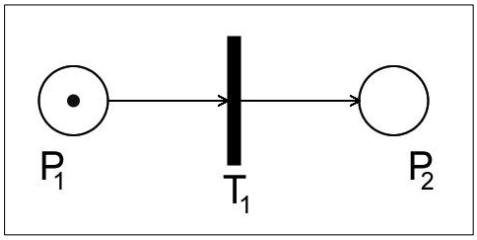
Yayılma karakteristiğini belirleyen bir diğer kriter güncelleme işlemlerinin eşzamanlılığına dayanmaktadır. Sıralı (*eşzamanlı olmayan*) güncelleme işlemi güncellenen her bir ilişkinin yayılmasını tekil işlem olarak ele alır. Eşzamanlı yayılma durumunda ise BT üzerindeki farklı ilişkiler senkron olarak güncellenir. Teorik olarak eşzamanlı yayılma işleminin BT'nin farklı ilişkilerinde güven değerlerinin sıralı işleme göre farklı sonuçlar üreteceği öngörülmektedir. Eşzamanlı güncelleme işlemi güven değerleri üzerinde girişime (*interference*) neden olarak güven değerlerinde sönümleme veya güçlendirme etkisi oluşturabilir.

4.7. Petri Ağları ile Yayılma İşlemi

Petri ağları (Petri Nets – PN) Carl Adam Petri tarafından 1962 yılında önerilmiştir [155]. PN dinamik sistemlerdeki olayların simüle edilmesinde başarılı matematiksel model ortaya koymaktadır. Petri ağları görsel olarak da yüksek ifade yeteneğine sahiptir.

Basit ağ modeli dörtlü şekilde ifade edilir: $N = (P, T, I, O)$. Burada $P = (P_1, P_2, \ldots, P_n)$ yerlerin (*places*), $T = (T_1, T_2, \ldots, T_n)$ ise geçişlerin (*transitions*) ayrık sonlu kümeleridir. $I \subseteq (P \times T)$ girişlerin, $O \subseteq (T \times P)$ ise çıkışların kümesidir.

Petri ağı basit ağın (N) genişletilmesi ile elde edilir. $PN = (N, M, W)$ şeklinde üçlü olarak tanımlanır. Burada N, daha önce tanımlanmış olan basit ağı, $M: P \rightarrow Z$ jetonların bulunduğu yerlere göre ağın mevcut durumunu, $W: I \cup O \rightarrow Z$ giriş ve çıkış ağırlıklarını ifade etmektedir. Şekil 4.5 temel Petri ağını göstermektedir.



Şekil 4.5. Temel Petri ağı

Petri ağı, bir yerde (P_1) bulunan jetonların (*token*) (Şekil 4.5'te P_1'de siyah nokta olarak gösterilmiştir) geçişin (T_1) etkinleştirilmesi (*firing*) ile bir sonraki yere taşınmasını

sağlamaktadır. Etkinleştirme işlemi atomiktir, tekil ve durdurulamaz işlem olarak tanımlanır.

Temel Petri ağı yerler, geçişler ve jetonlar ile sadece belirgin durumları ifade edebilmektedir. Belirsiz bilginin ortaya çıkması durumunda temel Petri ağı yetersiz kalmaktadır. Bu yetersizliği ortadan kaldırmak için Bulanık Petri ağları (Fuzzy Petri Nets – FPN) öne sürülmüştür [156, 157].

Bir FPN PN'i genişletir ve $\mathrm{FPN} = (\mathrm{N}, \mathrm{S}, \alpha, \beta, \lambda, \mathrm{M}_0)$, $\mathrm{P} \cap \mathrm{T} = \mathrm{P} \cap \mathrm{S} = \mathrm{T} \cap \mathrm{S} = \varnothing$ olarak gösterilir [158, 159]. Burada N, daha önce tanımlanmış basit ağı, $\mathrm{S} = (\mathrm{S}_1, \mathrm{S}_2, \ldots, \mathrm{S}_n)$ sonlu ayrık önermeler (*statement*) kümesini, α: P → S önerme birleştirme fonksiyonunu, β: T → [0, 1] doğruluk derecesi fonksiyonunu, λ: T → [0, 1] eşik değer fonksiyonunu, M_0: P → [0, 1] yerlerdeki değerlerin başlangıç durumunu göstermektedir. Etkinleştirme fonksiyonu Eş. 4.8'deki gibi tanımlanır [156].

$$F_E = \begin{cases} aktif, minM(t_j) \geq \lambda_{pj} \\ pasif, diğer \end{cases} \tag{4.8}$$

Etkinleştirme işlemi sonrasında jetonun (*değerin*) bir sonraki yere aktarımı seçilmiş moda (*min veya max*) göre Eş. 4.9 ile gerçekleştirilir.

$$M'(p) = \begin{cases} mode1 \min(M_{tj}) * \beta_{tj} \\ mode2 \max(M_{tj}) * \beta_{tj} \end{cases} \tag{4.9}$$

Bu tanım FPN için temel tanım olmakla birlikte literatürde farklı FPN tanımları ve varyasyonları mevcuttur [160–163]. Özellikle, FPN'in kontrol sistemleri, karar destek sistemleri, benzetim vb. farklı uygulama alanlarında çok farklı uyarlamaları söz konusudur. PN ve FPN'ler renklendirilmiş jetonlar ile zenginleştirilerek Renkli Petri Ağları (Coloured Petri Nets – CPN) [164] ve Bulanık Renkli Petri Ağları (Fuzzy Coloured Petri Nets – FCPN) [165] geliştirilmiştir. Renklendirilmiş jetonlar farklı durum türleri için farklı etkinleştirme fonksiyonlarını devreye sokmak amacıyla kullanılır.

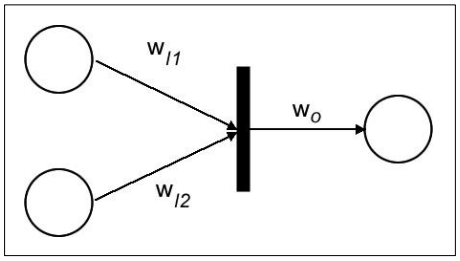
Tez kapsamında araştırılan PN ve FPN'in bilgi gösterimi ve bilgi çıkarımı için kullanıldığı çalışmalar da incelenmiştir [166, 167]. Bu çalışmalar mevcut FPN'lerin kapsamını genişleterek dinamik ve uyarlanabilir yeni ağ yapıları ortaya koymaktadır. FPN'lerin bilgi

gösterimi için kullanımlarında ağırlıklandırılmış bulanık üretme kuralları (*weighted fuzzy production rules*) öne çıkmaktadır. Bunlar kural tabanlı sistemlerde bilginin ifade edilmesi için uygulanan *IF a THEN b* yapılarının birleşim, kesişim gibi kombinasyonlarından oluşmaktadır. Bulanıklaştırma işlemi kuralların ağırlık değerlerine sahip olması ile elde edilmektedir. FPN'lerin bilgi gösteriminde önemli nokta kuralların taban (*ground*) olarak değil evrensel nicelik (*universal quantified*) olarak tanımlanmasıdır. Taban ve evrensel nicelik kurallarının tanımı aşağıdaki örnekle gösterilebilir [111].

Evrensel nicelik kuralı: $\forall\, x, y : (x, başkent, y) \Rightarrow\ (x, yerleşir, y)$

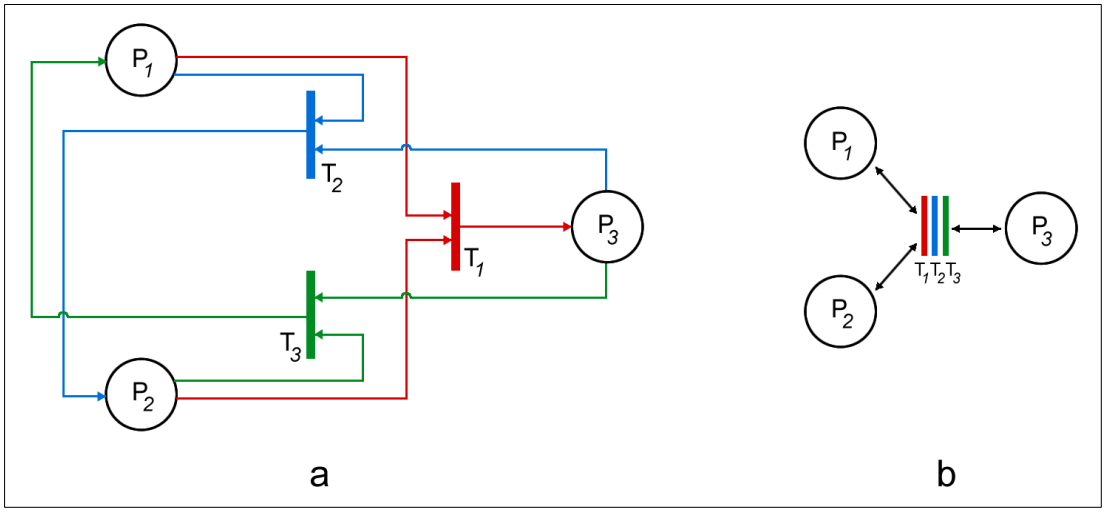
Taban kuralı: $\forall\, x, y : (Ankara, başkent, Türkiye) \Rightarrow\ (Ankara, yerleşir, Türkiye)$

Bir önceki başlıkta bahsedilen yayılma kuralları ve yayılma hesaplama yöntemleri göz önünde bulundurulduğunda FPN'lerin yayılma işlemleri için uygun yapı sağladığı görülebilir. Oluşturulan BT'nin doğası gereği her bir taban kuralının güven değerleri farklılık göstereceğinden FPN'ler evrensel nicelik kuralları olarak tanımlanamaz. Bir diğer önemli nokta ise FPN'lerin farklı güven değerleri için dinamik jeton yapısına sahip olmasıdır [161].



Şekil 4.6. Birleşme yayılma kuralı için FPN

Yayılma işlemi için oluşturulan FPN'lerde yerler (*places*) BT'deki üçlüyü temsil etmektedir. Birleşme kuralı için örnek FPN yapısı Şekil 4.6'da gösterilmiştir. Bu ağ, iki üçlünün güven değerlerini alarak çıktı üretmektedir. Birbiri ile üçgen ilişkisi oluşturan ve birleşme kuralını tamamlayan üçlüler kendi içinde üçgen yapısı oluşturmaktadır. Bu durumda birleşme yayılma kuralı bu üçlüler arasında döngüsel özellik kazanmakta ve birbiri ile ilişkilendirilmiş üç FPN yapısı ortaya çıkarmaktadır. Bu tarz bir FPN yapısının açık ve kompakt gösterimleri Şekil 4.7'de a) açık ve b) kompakt olacak şekilde gösterilmiştir.



Şekil 4.7. Üçgen yapılar için oluşturulan yayılma kuralı FPN yapısı

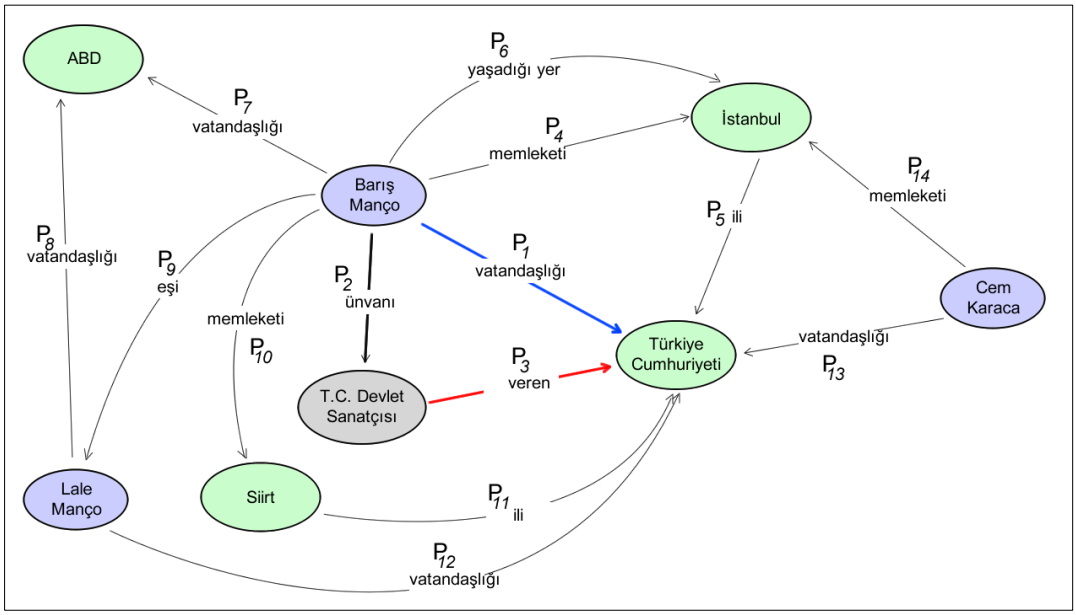
Şekil 4.7'den de görüldüğü gibi üç birleşme yayılım işleminin tek bir ağ üzerinde birleştirilmesi sonsuz yayılım işlemine neden olmaktadır. Bu sorunu ortadan kaldırmak için yayılım işleminin başladığı öncülden geri yayılmanın engellenmesi gerekmektedir. Geri yayılmanın engellenmesi öncüle bağlı geçişin yönlendiği çıktıya bağlı geçişlerde tekrar öncülün çıktı olarak bulunmaması durumunu kontrol ederek mümkün olmaktadır. Yayılma işleminin etkinleştirilebilmesi için bu durumun ilk yer (*place*) üzerinde devre dışı bırakılması zorunludur.

Tez kapsamında geliştirilen BT'nin dinamik yapısı göz önüne alındığında oluşturulacak FPN ağının da dinamik yapıda olacağı öngörülebilir. Bahsedilen FPN ağı yeni gelen üçlüler üzerinden oluşturulan veya güven değeri belli değerin altında kalan üçlülerin silinmesi sonucu çıkarılan yerlerin (*place*) dinamik şekilde denetimini destekleyecek esnekliğe sahip olmalıdır. Şekil 4.8'da örnek bir BT çizge yapısı her bir üçlünün karşılık geleceği yer numaraları ile birlikte gösterilmiştir. Bu BT'nin FPN ağına dönüştürülmüş diyagramı Şekil 4.9'da gösterilmiştir.

4.8. Petri Ağında Geçiş Hesaplaması

FPN ağının yapısal niteliği ile birlikte geçiş noktalarının tetiklenme durumları ve hesaplama fonksiyonları da ağın karakteristiğini ortaya koymaktadır. Önceki bölümlerde anlatılan sigmoid hesaplamalarından farklı olarak FPN'ler α ve β değerleri ile tetikleme durumlarını ve çıktı hesaplamalarını denetlemektedir. Ancak FPN'nin sunduğu esnek yapı geçiş tetiklenmesi ve hesaplama işlemlerini bununla sınırlı tutmamaktadır. Bu nedenle

tezde farklı hesaplama yaklaşımları dikkate alınarak BT'lerin FPN gösterimi için arıtma işlemlerinde elde edilen sonuçlar karşılaştırılmıştır. Tercih edilen hesaplama yöntemleri üç alt başlık olarak aşağıda sunulmuştur.



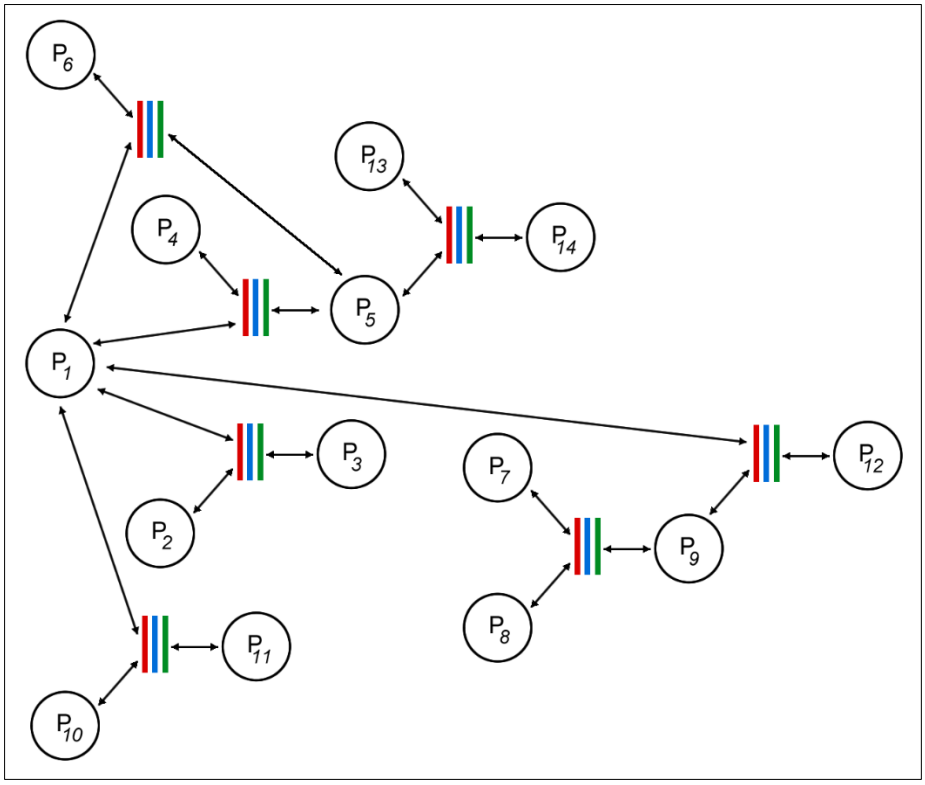
Şekil 4.8. Üçlülerin yerler olarak ifade edildiği örnek çizge

4.8.1. Öncül güven değerlerinin doğrudan etkisi

Bu durum mevcut FPN yaklaşımını olduğu gibi uygulamaktadır. Geçişin tetiklenmesi *mode2-max* moduna bağlı olarak sağlanmakta ve α değeri silme eşik değerini temsil etmektedir. β değeri ise her bir çıktının önceki güven değerini alarak dinamik olarak tanımlanmaktadır. Bu durumda çıktı fonksiyonu Eş. 4.10'da tanımlandığı şekliyle hesaplanmaktadır.

$$M'(p) = \max(M_{tj}) * \beta_{tj} \tag{4.10}$$

Bu şekilde çıktı fonksiyonunun tanımlanması eşik güven değerinin üzerinde tanımlı değere sahip öncüllerin çıktı güven değerini artırma veya azaltma davranışını modellemektedir.



Şekil 4.9. Çizge üzerinden oluşturulmuş FPN

4.8.2. Sigmoid fonksiyonu ile çıktı hesaplama

Bu yaklaşım çizge veri yapısı üzerinde daha önce uygulanan sigmoid fonksiyonunun FPN üzerinde de geçiş işlemlerinde çıktı üretme fonksiyonu olarak kullanılmasını temel almaktadır.

Bu yaklaşımda daha önce detayları açıklanmış olan sigmoid fonksiyonunun düşük ve yüksek güven değerlerine göre belirlenmiş çıktı üretme davranışı temel alındığından herhangi eşik değer sınırı gözetilmeksizin yayılma işlemini başlatan güven değerinin çıktı üzerindeki etkisi hesaplanmaktadır.

4.8.3. Bulanık mantık kümeleri ile çıktı hesaplama

Daha önce de ifade edildiği gibi BT üzerinde tanımlı üçgen yapıları üç farklı üçlüden oluşmaktadır. Her bir üçlü önermeler mantığı açısından birer önerme olarak kabul edilebilir. Bu durumda üçlüler arasında p, q ve r şeklinde öncüllere bağlı oluşan sonuç için $p \wedge q \rightarrow r$ şeklinde yeni bir önerme oluşturulabilir. Bir üçgen için bu tür EĞER-ÖYLE İSE *(IF-THEN)* önermesi 3 farklı şekilde tanımlanır. Üçgenin mantıksal doğruluğunun

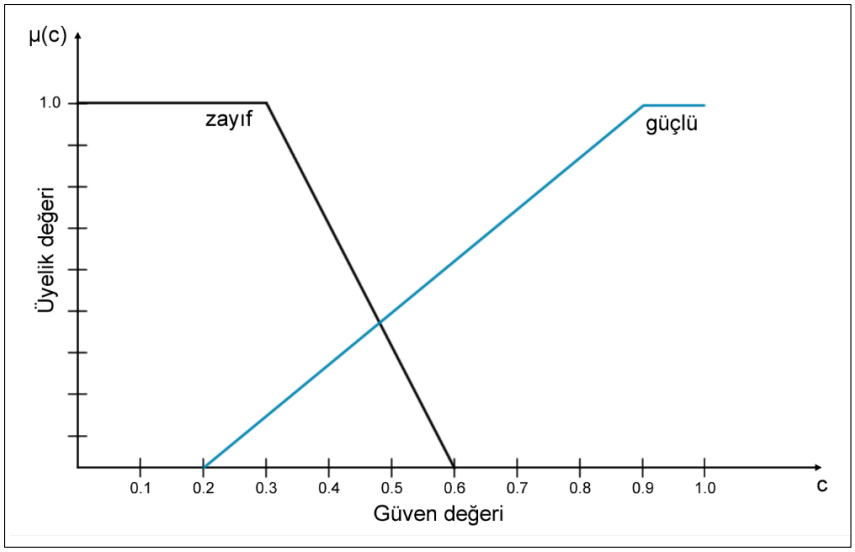
sağlanabilmesi açısından bu önermelerin eşdeğer olması gerekmektedir. Bu durumda ortaya çıkan üç önermenin eşdeğer olduğu durumlar dikkate alınarak öncüllere bağlı sonuç önermesinin doğruluk durumu öngörülebilir. Çizelge 4.1'de üçgen için oluşturulan üç farklı önerme için doğruluk çizelgesi gösterilmiştir. Çizelgede T – doğru (*true*) durumları F – yanlış (*false*) durumları ifade etmektedir.

Çizelge 4.1. Üçgen önermeleri için doğruluk tablosu

p	q	r	$p \wedge q \rightarrow r$	$q \wedge r \rightarrow p$	$r \wedge p \rightarrow q$
T	T	T	T	T	T
T	T	F	F	T	T
T	F	T	T	T	F
T	F	F	T	T	T
F	T	T	T	F	T
F	T	F	T	T	T
F	F	T	T	T	T
F	F	F	T	T	T

Çizelge 4.1'de görüldüğü gibi olası sekiz farklı durumdan beşinde (*koyu olarak işaretlenmiş durumlar*) üçlü önermeler aynı doğruluk değerini alır. Bu durumları ele alarak öncüllerin (*p, q*) doğru olması durumunda sonucun (*r*) da doğru olması gerektiği, öncüllerden herhangi birinin yanlış olması durumunda sonucun yanlış olduğu ve öncüllerin her ikisinin yanlış olması durumunda ise sonucun belirsiz olduğu gösterilebilir.

Öncüllerin güven değerleri ve belirtilen kurallar çerçevesinde bulanık sistem oluşturularak ilk güven değerleri üzerinden yeni güven değeri hesaplanabilir. Bu durum üçlülerin güven değerlerinin bulanıklaştırılarak doğruluk değerleri için tanımlanmış üyelik fonksiyonuna aktarılması ve elde edilen bulanık değerin durulaştırılarak yeni güven değerinin elde edilmesi şeklinde kurgulanır. Şekil 4.10'de güven ve doğruluk değerleri için üyelik fonksiyonları gösterilmiştir.



Şekil 4.10. Güven ve doğruluk değerleri için üyelik fonksiyonları

Çizelge 4.1'de gösterilen doğruluk tablosu temel alınarak bulanık sistem için oluşturulan kural tablosu da Çizelge 4.2'te gösterilmiştir.

Çizelge 4.2. Bulanık sistem için oluşturulmuş kural tablosu

	p - Zayıf	p - Güçlü
q - Zayıf		Zayıf
q - Güçlü	Zayıf	Güçlü

FPN üzerinden geçiş fonksiyonlarında tanımlı bulanık sistem için çıkarım yöntemi olarak Mamdani (Min-Max) [168], durulaştırma işlemi için ise ağırlık merkezi yöntemi tercih edilmiştir [169].

5. BİLGİ TABANLARINDA PERFORMANS OPTİMİZASYONU

Tez kapsamında ele alınan BT sistemleri dinamik, sürekli değişime uğrayan ve kendi içinde milyonlarca üçlüyü barındıran canlı yapıya sahip büyük veri sistemleridir. Özellikle, günümüzde kullanılan ticari ve akademik BT yapıları göz önünde bulundurulduğunda bu sistemlerin oluşturulma, arıtılma ve doğrulanma süreçleri ciddi işlem gücü gerektirmektedir. Bu nedenle BT işlemlerinin tüm aşamalarında performans optimizasyonu kaçınılmaz bir gereksinim olarak ortaya çıkmaktadır. Bu başlık altında BT yapılarının akan veri üzerinden oluşturulması, güncellenmesi ve sorgulama yapılması için performans optimizasyonundan bahsedilecektir.

5.1. Çizge Veri Yapısında Performans Kısıtları

Tez kapsamında önerilen BT yapısı çizge veri modelini temel alan ve sürekli güncellenen bir veri deposudur. Bu çizge, eklenen veriler ile güncellenmekte, çizgede daha önce bulunmayan veriler oluşturulmakta, önem ve güven parametrelerine bağlı olarak üçlüler silinmektedir. Dolayısıyla BT'ye yeni bilgi akışını sağlayan akan veri kaynağı ile (*stream*) depolanan veri arasında sürekli bir etkileşim söz konusudur. Bu nedenle yapılan deneysel çalışmalarda sabit veri deposu üzerinde akan veri kaynağından elde edilen varlıklar ile işlemler gerçekleştirilmiştir.

Mevcut BT'de yeni olguların ortaya çıkarılması ve farklı bilgi düzeylerinde olguların kapsüle edilmesi de veri manipülasyonlarına ihtiyaç duymaktadır. Son olarak oluşturulan yapının bilgi sorgulamalarında kullanılması BT'nin ana görevlerinden biridir. Bu nedenle deneylerde küme içinde üyelik araması (*membership search*) ve dönüş değerleri ile arama işlemleri de ayrı şekilde ele alınmıştır.

Çizge veri yapısını kullanan büyük veri uygulamaları performans optimizasyonuna ihtiyaç duymaktadır. Performansla ilgili darboğaz özellikle akan veri uygulamalarında öne çıkmaktadır. Akan verinin mevcut veri kümesi ile karşılaştırılması ekleme, güncelleme ve silme işlemlerinin performansını etkilemektedir. Dolayısıyla hafızada saklanan veri yapısı üzerinde gerçekleştirilen işlemlerin performansı kadar canlı verinin anlık işlenme kapasitesi de önemli parametredir.

Çizge veri tabanlarında optimizasyon konusunda literatürde birçok çalışma yapılmıştır. Bu çalışmalar aşağıdaki başlıklar altında ele alınabilir.

İndeks optimizasyonu

İndeksleme çizge veri yapısında efektif arama ve sıralama işlemleri için kullanılmaktadır. L. Zou ve ark. tarafından yapılan gStore [170] çalışmasında veri yapısındaki varlıkların temsil edilmesi için hash imzası kullanılmaktadır. Bu imza Bloom filtresi benzeri bit vektörlerinde tutulmaktadır. İndeks optimizasyonuna yönelik başka bir çalışma ise GraSS [171] motorudur. Bu çalışmada alt çizgeler için FDD-indeks yardımı ile imzalar oluşturulmaktadır. RIQ sistem [172] PV-index yaklaşımı ile Basic Graph Pattern (BGP) üzerinde Bloom filtresi uygulayan başka bir yöntemdir. Bu yöntemde benzer alt çizgeler kodlanarak (*encode*) hafızadan tasarruf sağlanmaktadır.

Birleştirme (*join*) optimizasyonu

Birleştirme işlemleri veri manipülasyonunda önemli yere sahiptir. Özellikle, BGP'ler üzerinde en temel birleştirme işlemleri için çizgedeki komşu düğümlerin çift birleştirme (*pairwise join)* optimizasyonu önem kazanmaktadır. Bloom filtresinin kullanıldığı böyle bir çalışmaya Neumann ve Weikum tarafından yapılmış Ubiquitous Sideways Information Passing (U-SIP) örnek gösterilebilir [173].

Sorgu optimizasyonu

Çizge veri tabanları standart sorgulama diline sahip değildir. Bununla birlikle SPARQL, Gremlin ve Cypher gibi farklı sorgulama dilleri bulunmaktadır. Veri modelinde tutulan verinin indekslenmesi kadar performans artışı için sorguların optimizasyonu da önemlidir. Bu konuda Bloom filtresi kullanılarak geliştirilen yöntemlere FERRARI [174] örnek gösterilebilir. Bu yöntem düğümler arası komşuluk ilişkilerinin Bloom filtresi üzerine kodlanmasını temel almaktadır. İndekslenmiş ilişkiler sorgularda *join* işlemleri için kullanılmaktadır. Dia ve ark. [175] tarafından yapılan başka bir çalışma da akan veri ve hafıza üzerindeki sorgularda Bloom filtresini kullanmaktadır.

Diğer çalışmalar

Üçlü veri yapıları ve çizge veri modeli üzerinden yaklaşık üyelik fonksiyonu kullanan diğer çalışmalar da mevcuttur. Sande ve ark. [176] yaptıkları çalışmada RDF IRI sorgularının optimizasyonu için Bloom filtresi ve Golomb-coded kümelerini tercih etmişlerdir. Benzer bir çalışma Taelman ve ark. [177] tarafından da yapılmıştır. Bu çalışmada SPARQL sorgularının istemci tarafında Bloom filtresi aracılığıyla optimize edilerek sunucu üzerindeki yükün azaltılması hedeflenmiştir.

5.2. Dinamik BT Yapısında Performans Optimizasyonu

Dinamik BT yapısında yazma, okuma ve arama işlemlerinin performans optimizasyonu için tez çalışması kapsamında indeksleme yöntemleri araştırılmış ve Ölçeklenebilir Bloom filtresi ile indekslemeye yönelik yöntem önerilmiştir. Önerilen Bloom filtresinin mevut yöntemlerden farkı üyelik arama fonksiyonu ile beraber veri indekslemesini de olanaklı hale getirmesidir. Bu başlık altında akan RDF veri yapısı, Bloom filtresi ve önerilen indeksleme yöntemi detaylandırılacaktır.

5.2.1. RDF üçlüsü ve akan veri

Bir RDF akan verisi *S*, sıralı ve zaman damgasına sahip üçlüler olarak ele alınır ve $S = (< s, p, o >_i,\ t_i)$ şeklinde gösterilir. Burada s ve o düğümleri, p düğümler arasındaki ilişkiyi, i akan veri üzerindeki sırayı, t_i ise zaman damgasını göstermektedir. <*s, p, o*> üçlüsü alt çizge olarak ifade edilirse $S = (G_i, t_i)$ olarak gösterilebilir. Akan veride önemli nokta i değerlerinin zaman içinde her zaman artıyor olmasıdır ($\forall i,\ t_i \leq t_i + 1$). Akan veri kayan pencere mantığı ile tamponlanarak işlenmektedir. Tampon boyutu sistemin işlem gücüne ve performans kriterlerine göre belirlenir [175].

5.2.2. Bloom filtresi

Bloom filtresi 1970 yılında H. Bloom tarafından küme içinde eleman sorgulama amacıyla olasılığa dayalı bir yöntem olarak öne sürülmüştür [178]. Son yıllarda büyük veri analizi çalışmaları ile birlikte Bloom filtresinin kullanımı da popülerlik kazanmaktadır.

Bloom filtresi en basit hali ile *M* bit alandan oluşan bir dizi içermektedir. Dizinin tüm elemanları başlangıçta 0 değeri alır. Filtre aynı zamanda *k* adet hash (*özet*) fonksiyonuna sahiptir. Her bir hash fonksiyonu girdi elemanından dizi üzerinde bir indekse karşılık gelecek çıktıyı üretir. Girdi olarak alınan *e* elemanının hash fonksiyonundan elde edilen çıktıları $h_1(e), h_2(e), \ldots, h_k(e)$ dizi üzerindeki *k* adet biti 1 olarak değiştirir [179, 180].

Bloom filtresinde sorgulama işlemi de ekleme işlemine benzer şekilde yapılmaktadır. Girdi olarak alınan *x* elemanının hash fonksiyonundan elde edilen indeks değerleri $h_1(x), h_2(x), \ldots, h_k(x)$ *M* dizisi üzerinde kontrol edilir. Herhangi indeksin 0 değeri döndürmesi durumunda *x* elemanının kümede kesinlikle bulunmadığına karar verilir. Tüm indekslerin 1 değerini döndürmesi durumunda ise *x* elemanının kümede olduğu varsayılır. Olasılığa dayalı bir yöntem olmasından kaynaklı Bloom filtresinde yanlış pozitif (*false positive*) sonuç üretme ihtimalinin göz önünde bulundurulması gerekmektedir. *k* hash fonksiyonu sayısının veya *M* filtre boyutunun artırılması Bloom filtresinde yanlış pozitif oranını düşürmektedir.

Bloom filtresinin literatürde birçok varyasyonu bulunmaktadır. En temel varyasyonlardan biri hash fonksiyon çıktılarının kesişimini önlemek amacıyla *M* bitlik filtre dizisini *k* bölüme ayırmaktır [181]. Bu durumda yeni filtre bölümlerinin boyutu *m* olarak ifade edilir. Verilen bir bölüm için herhangi bir indeksin 1 olma olasılığı bu bölümün boyutunun 1 olarak işaretlenen indekslerin sayısına oranı, bir diğer ifade ile doluluk oranı *p* ile gösterilir. *m* değeri büyüdükçe doluluk oranı *p* tüm bölümler için aynı orana yakınsayacaktır. Bu durumda parçalanmış filtrenin yanlış pozitif oranı Eş. 5.1 ile hesaplanır.

$$P = p^k \tag{5.1}$$

Bu varyasyon için yanlış pozitif oranının belirlenen aralığa sınırlandırılması için Eş. 5.2 kullanılmaktadır.

$$n \approx M \frac{(\ln 2)^2}{|\ln P|} \tag{5.2}$$

Bu eşitlikte *n* filtre üzerinde indekslenecek veri kümesinin boyutunu, *P* ise yanlış pozitif oranını göstermektedir. Eşitlikten görüldüğü gibi verilen *P* değeri için filtrede

indekslenecek eleman sayısı n ile toplam filtre boyutu M arasında doğrusal ilişki bulunmaktadır. p doluluk oranının optimal durumda $p = \frac{1}{2}$ olduğu [179] göz önünde bulundurulursa belirlenen yanlış pozitif oranı için hash fonksiyonu sayısı Eş. 5.3 ile belirlenebilir.

$$k = \log_2 \frac{1}{P} \tag{5.3}$$

Örnek olarak, %0,1 hata oranı ile sadece 32 KB filtre alanı ayırarak 18.232 eleman Bloom filtresinde indekslenebilir [179].

5.2.3. Ölçeklenebilir Bloom filtresi

Temel Bloom filtresi büyük veri kümelerinde düşük hafıza kullanımı ile başarılı sonuçlar üretmektedir. Yalnız bu yöntemin en önemli dezavantajı ölçeklenebilir olmamasıdır. İndekslenecek veri boyutunun önceden bilinmemesi durumunda filtre sürekli yeniden oluşturularak kullanılmalıdır. Bu sorunu ortadan kaldırmak için ölçeklenebilir Bloom filtresi (Scalable Bloom Filter - SBF) önerilmiştir [179, 182]. SBF iki ana fikri temel almaktadır:

- SBF bir veya daha fazla temel Bloom filtresinden oluşur. Bloom filtresinin doluluk oranına erişmesi durumunda yeni filtre tanımlanır.
- Her yeni oluşturulan Bloom filtresi tanımlanmış maksimum hata oranı sınırının üstüne çıkmayacak şekilde parametrik olarak eklenmelidir.

SBF k_0 bölümlü ve P_0 hata oranına sahip birincil Bloom filtre ile başlatılır. Bu filtre doluluk oranına ulaştığında yeni filtre k_1 ve $P_1 = P_0 r$ parametreleri ile eklenir. r katsayısı hatayı sıkıştırma oranı olarak nitelendirilir ve $0 < r < 1$ aralığındadır. l adet yeni filtre oluşturulması durumunda hata oranları P_0, $P_0 r$, $P_0 r^2$, …, $P_0 r^{l-1}$ değerlerini alır. SBF'nin toplam hata oranı ise Eş. 5.4 ile hesaplanır.

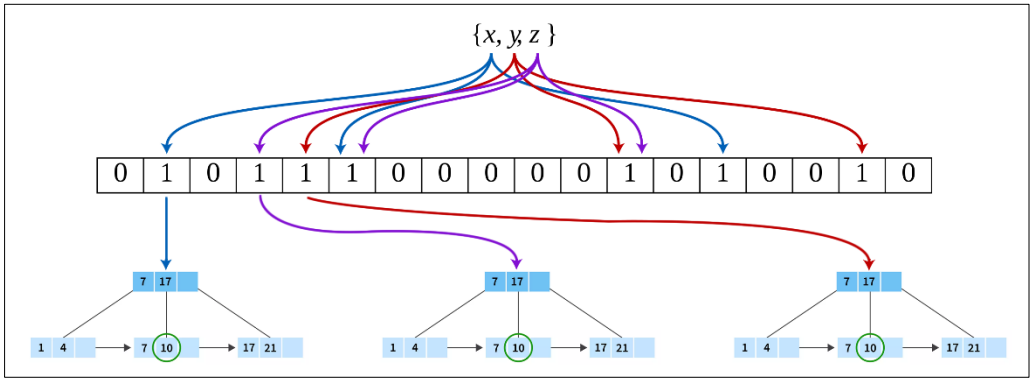
$$P = 1 - \prod_{i=0}^{l-1} \left(1 - P_0 r^i\right) \tag{5.4}$$

$1 - \prod_i (1 - P_i) \leq \sum_i P_i$ eşitsizliğinden P değeri için $\lim_{l \to \infty} \sum_{i=0}^{l-1} P_0 r^i$ üst limiti tanımlanabilir. Bu durumda $P \leq P_0 \frac{1}{1-r}$ eşitsizliği elde edilir. Eş. 5.3'te *i.* filtre için k_i hash fonksiyonunun sayısı Eş. 5.5 ile hesaplanır.

$$k_i = \log_2 P_i^{-1} = k_0 + i \log_2 r^{-1} \tag{5.5}$$

r katsayısının $r = \frac{1}{2}$ kabul edilmesi durumunda Eş. 3.5 $k_i = k_0 + i$ olarak basitleştirilebilir. Bu da SBF'ye eklenecek her yeni Bloom filtresi için hash fonksiyonu sayısını bir artırma anlamına gelir. Yapılan deneyler sonucunda r katsayısının 0,8 – 0,9 aralığında seçilmesi durumunun ortalama hafıza alanı üzerinde performans artışı sağladığı gözlemlenmiştir [183, 184].

M filtre boyutu ve buna bağlı m bölüm boyutları da SBF filtresi için önemli parametrelerdir. Bu parametrelerin belirlenmesi için deneysel çalışmalara yer verilmiştir. m_0, ilk filtredeki bölüm boyutu kabul edilirse yeni eklenen Bloom filtresi için bölümleme boyutları s katsayısı ile çarpılarak m_0, $m_0 s$, $m_0 s^2$, …, $m_0 s^{l-1}$ boyutları elde edilir. Yapılan deneyler $s = 2$ değeri için optimal hafıza kullanımını ortaya koymaktadır.



Şekil 5.1. Bloom filtresi ve B+ ağacı ile oluşturulmuş indeks yapısı

5.2.4. Bloom filtresi ile indeksleme

Bloom filtresi üyelik arama sonuçları sunduğundan küme içinde elemanın var olma durumunu sorgulayabilmektedir. Kayıt arama işlemlerinde ise kümedeki elemanın kayıt olarak döndürülmesi gerektiğinden üyelik arama yetersiz kalmaktadır. Bu nedenle ölçeklenebilir Bloom filtresinin kayıt döndürmeye yönelik kapsamı genişletilmiştir.

Bilindiği gibi B+ ağaçları arama indeksleme işlemlerinde sık tercih edilen veri yapılarıdır. B+ ağaçları arama, yazma ve silme işlemleri için *O(log n)* karmaşıklık sunmaktadır. Performans optimizasyonu için tez kapsamında Bloom filtresi ailesine kayıt arama özelliği kazandırma amacıyla B+ ağaçları ile entegrasyon önerilmiştir. Bu yapıda Bloom filtresinin ilk hash fonksiyonunun indeks değeri ilgili B+ ağacı için işaretçi (*pointer*) olarak kullanılmıştır. Bu sayede ilk hash fonksiyonu aynı indekse karşılık gelen kayıtlar aynı B+ ağacı yapısında tutulmaktadır. Bloom filtresi ve B+ ağaçlarından oluşan veri yapısı Şekil 5.1'te gösterilmiştir.

SBF filtresi arama ve ekleme işlemlerinde *k* adet hash fonksiyonu için *O(k)* karmaşıklığı sunmaktadır. Bölüm 4.2'de anlatılan üçlülerin çizge BT üzerinden aranması ve eklenmesi işlemleri SBF yardımı ile veri boyutundan bağımsız olarak sabit işlem sürelerine eşit olacaktır. Kayıt arama işlemlerinde ise bu karmaşıklık *O(k)* + *O(log n)* olarak hesaplanmaktadır.

6. DENEYSEL ÇALIŞMALAR

BT modelleri kendi içinde büyük veri kümelerini barındırmaktadır. Bu veri kümeleri özellikle ticari projelerde milyarlarca düğüm ve ilişkiye sahip olmakla birlikte akademik çalışmalarda sentetik olarak oluşturulmuş veya mevcut veri kümelerinden seçilerek elde edilmiş daha küçük kümeler kullanılmaktadır [62]. Tezin bu bölümünde yapılan çalışmalarda ele alınan veri kümeleri, bu verilerin saklanma şekli, deneysel çalışmalar için geliştirilmiş yazılım ortamından bahsedilecektir. Ayrıca güven değerlerinin yayılma işlemlerinde tez kapsamında önerilen yaklaşımlar temel alınarak uygulanan deneyler ve elde edilen sonuçlar sunularak değerlendirmeler yapılacaktır.

6.1. Geliştirilen Yazılım Ortamı

BT altyapısı oluşturma, tez kapsamında önerilen güven değerlerinin yayılmasına bağlı BT temizleme ve doğrulama çalışmalarının deneysel olarak sınanması için yazılım ortamı hazırlanmıştır. Yazılım ortamı hazırlanırken öncelikli olarak veri depolama sistemleri üzerinde araştırma yapılarak mevcut sistemler incelenmiştir.

Son dönemde mevcut BT'lerin birbiri ile birleştirilmesi de dikkate alınırsa ortaya çıkan yapıların ciddi anlamda büyük veriler içerdiği görülmektedir [185]. Çizelge 6.1 popüler BT'lerin içerdiği üçlülerin ve varlıkların (özne ve nesnelerin tekil toplamı şeklinde) istatistiğini göstermektedir.

Çizelge 6.1. Popüler BT'lerin üçlü ve varlık istatistikleri [54]

Bilgi Tabanı	Varlıklar	Üçlüler	Kaynak
WordNet [186]	117 597	207 016	https://wordnet.princeton.edu
OpenCyc [187]	47 000	306 000	https://www.cyc.com/opencyc/
Cyc [187]	~250 000	~2 200 000	https://www.cyc.com
YAGO [188]	1 056 638	~5 000 000	http://www.mpii.mpg.de/~suchanek/yago
DBpedia [9]	~1 950 000	~103 000 000	https://wiki.dbpedia.org/develop/datasets
Freebase [189]	-	~1,9 milyar	https://developers.google.com/freebase/
NELL [10]	-	242 453	http://rtw.ml.cmu.edu/rtw/
Wikidata [190]	14 449 300	-	https://www.wikidata.org/wiki
Probase IsA	12 501 527	85 101 174	https://concept.research.microsoft.com/home/download
Google KG	>500 milyon	>3,5 milyar	https://developers.google.com/knowledge-graph

Çizelge 6.1'de görüldüğü gibi en az bilgi barındıran BT'ler bile yüz binlerce üçlü ve varlık içermektedir. Diğer taraftan geliştirilmeye devam edilen BT'lerde bulunan veriler milyarları aşmaktadır.

Bu ölçekteki verilerin barındırılması ve kullanımı için mevcut veri tabanı modellerinden biri kullanılabileceği gibi veri yapısına özgü veri depolama yaklaşımı da hazırlanabilir. Yapılan çalışmalarda veri depolama amacıyla ağırlıklı olarak çizge veri tabanları veya üçlü depolama (*triple store*) modellerinin tercih edildiği görülmüştür. Bu anlamda doğru veri depolama modelinin belirlenmesi için BT'de üçlülerin ifade edilme şekli önem kazanmaktadır.

Veri depolama yaklaşımları

BT üzerinde bilgi salt üçlüler (nesne, özne ve ilişki) ve bunlara ait değerler (*literal*) veya RDF standardına uygun olarak barındırılabilir [191]. Salt üçlü ve RDF gösterimleri arasındaki fark "Ahmet Fatma'nın eşidir" bilgisi üzerinden aşağıdaki örnekle açıklanabilir.

Salt üçlü gösterimi:

```
(a:İnsan {adı: "Ahmet"})-[:EŞİ]->(b:İnsan {adı: "Fatma"})
```

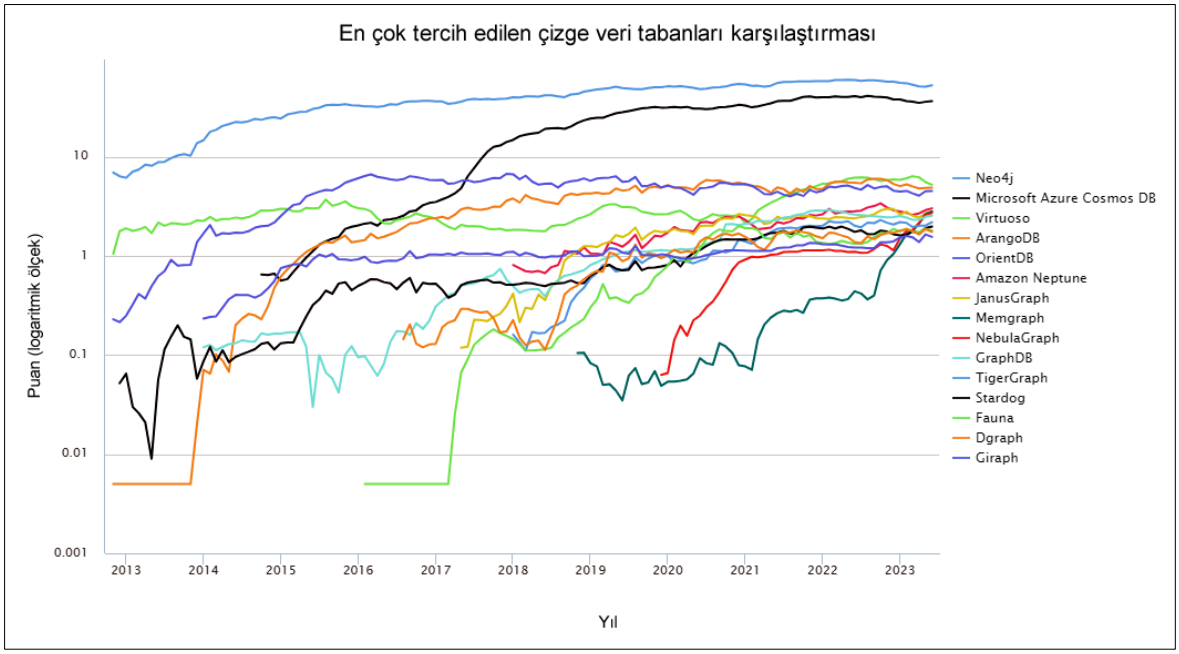
RDF gösterimi:

```
<http://domain.org/insan/1> :adı "Ahmet".
<http://domain.org/insan/1> foaf:eşi <http://domain.org/insan/2>.
<http://domain.org/insan/2> :adı "Fatma".
```

Bilginin salt üçlüler olarak ifade edilmesi durumunda çizge veri tabanı modelleri kullanılmaktadır. Üçlülerin RDF olarak ifade edilmesi durumunda ise üçlü depolama modelleri tercih edilmektedir.

Tez kapsamında geliştirilen yazılım ortamı için her iki yaklaşım ele alınarak karşılaştırılmıştır. Bu karşılaştırmada amaç mevcut platformların performansını değerlendirerek BT sistemi için yeni bir veri depolama sisteminin geliştirilmesine ihtiyacın olup olmadığını ortaya koymaktır. Bu amaçla en çok tercih edilen veri depolama sistemleri araştırılmıştır. Şekil 6.1'de en çok tercih edilen çizge veri tabanlarının karşılaştırılması gösterilmiştir.

Puanlama kriterleri olarak veri tabanının teknik değerlendirmeleri, profesyonel ağlar üzerinde bu veri tabanları ile çalışma istatistikleri, web arama istatistikleri, sosyal medya istatistikleri vb. kriterler dikkate alınmıştır [192].

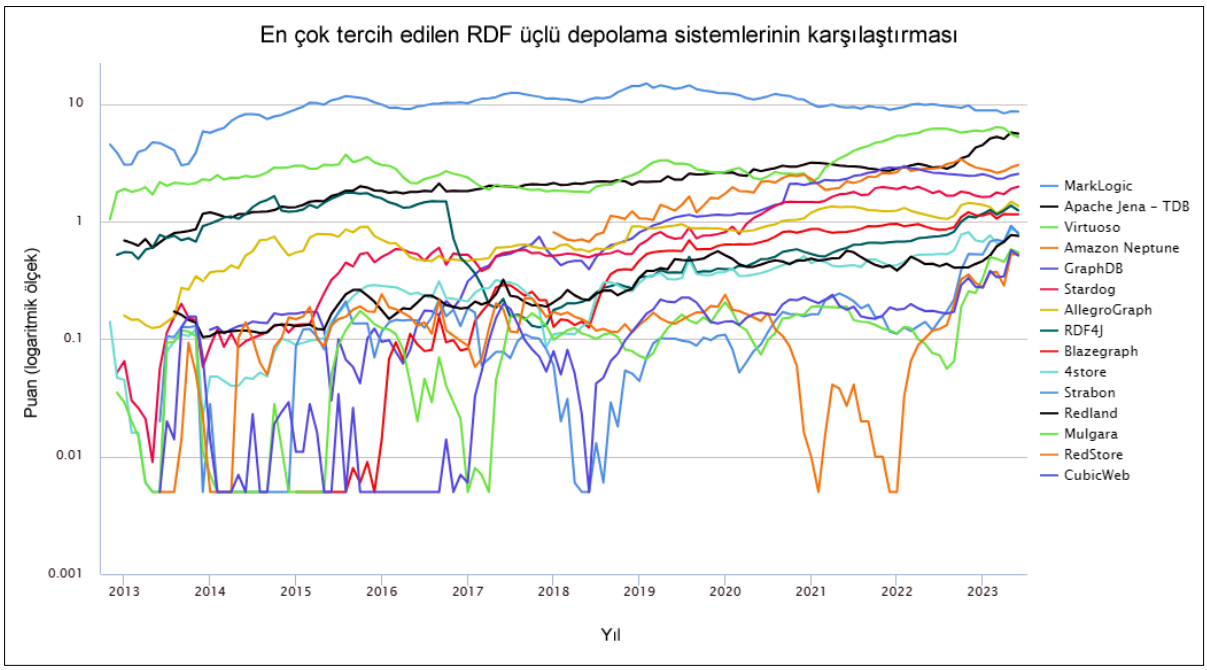


Şekil 6.1. En çok tercih edilen çizge veri tabanlarının karşılaştırması [193]

Benzer şekilde RDF üçlü depolama sistemlerinin karşılaştırması da Şekil 6.2'de gösterilmiştir. Karşılaştırma grafikleri dikkate alınarak çizge veri tabanlarında kullanılmak üzere "Neo4j" sistemi test için seçilmiştir [194]. RDF üçlü depolama testleri için ise "MarkLogic" [195] platformu sadece ticari kullanım sunduğundan "Apache Jena" [196] sistemi tercih edilmiştir.

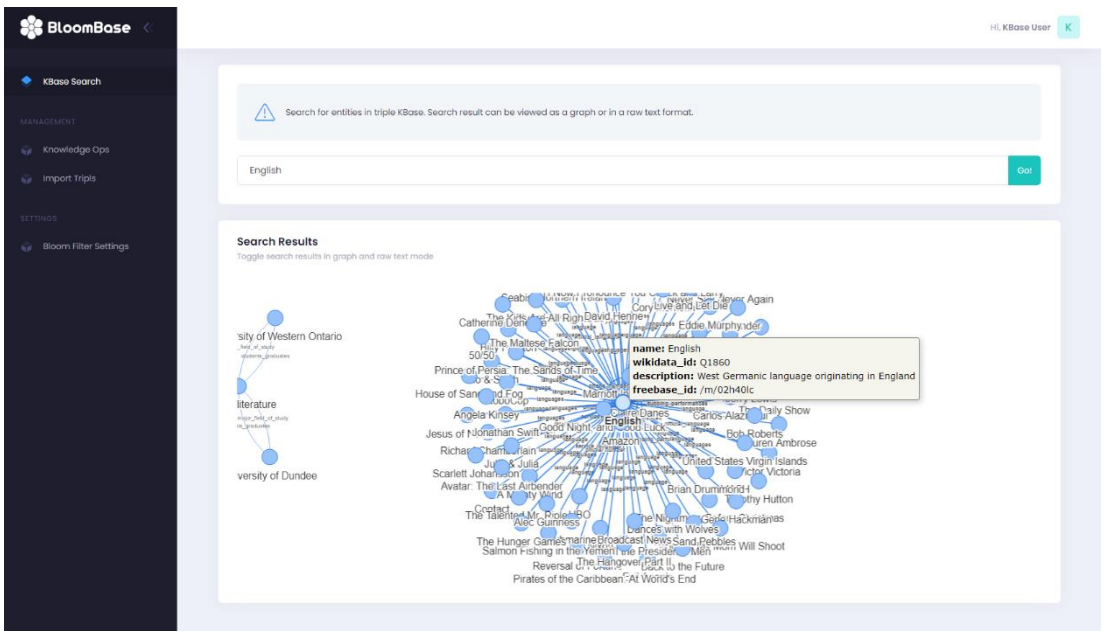
Yazılım ortamının işlevleri

Yazılım ortamı kendi içinde veri depolama sistemi ve bu sisteme bağlı olarak arama, ekleme, silme, güncelleme gibi "Create, Read, Update, Delete" (CRUD) işlemlerini, üçlü güven değerlerinin eklenmesi, güncellenmesi ve BT üzerinde yayılma etkisinin oluşması için gerekli yöntemleri içermektedir. Geliştirilen sistem kendi üzerinde BT'deki iş akışını ve işlem aşamalarını detaylandırmaktadır. İş akışı sırasında ortaya çıkan verilerin (yapılan işlemlerin süresi, üçlülerin sayısı ve dağılımı, işlemler için performans değerleri) istatistiği de yazılım ara yüzü üzerinden üretilmektedir.



Şekil 6.2. En çok tercih edilen RDF üçlü veri depolarının karşılaştırması [193]

Yazılım ortamının bir diğer işlevi oluşturulan BT'nin görselleştirilerek kullanıcı ara yüzünde gösterilmesidir. Görselleştirme işleminin performansı göz önünde bulundurulmakla birlikte bu işlem tez kapsamının dışında olduğu için ikinci planda değerlendirilmiştir. Yazılım ortamında hazırlanan arama ve çizge görselleştirme ara yüzünün ekran görüntüsü Resim 6.1'de gösterilmiştir.



Resim 6.1. Yazılım ortamının arama ve görselleştirme ara yüzü ekran görüntüsü

6.2. Veri Kümesi Seçimi

BT'lerde kullanılan veri kümeleri iki farklı kategoride değerlendirilebilir. İlk kategori ticari ve bazı akademik projelerde kullanılan büyük ölçekli veri kümeleridir [4]. Büyük ölçekli veri kümeleri bilgi gösterimi yaklaşımlarının modellenmesi, soru cevap uygulamaları, arama motorları ve dikey alana yönelik yapay zekâ uygulamalarında tercih edilmektedir. İkinci kategoride değerlendirilen veri kümeleri ise ağırlıklı olarak akademik çalışmalarda kullanılmaktadır. Bu veri kümeleri sentetik üretilmiş bilgilerden veya birinci kategorideki büyük BT'lere ait veri kümelerinin arıtılması ile oluşturulan küçük veri kümelerinden oluşmaktadır. Bu kategorideki veri kümeleri bilgi çıkarımı, Knowledge Representation Learning (KRL) ve Knowledge Graph Embedding (KGE) çalışmalarında tercih edilmektedir. İkinci kategoriye ait veri kümelerinin istatistikleri Çizelge 6.2'de gösterilmiştir. Bu çizelgede iki büyük veri kümesi kullanılarak arıtılmış veri kümeleri yer almaktadır. "WN" ön eki ile gösterilen veri kümeleri WordNet veri kümesinden, "FB" ön eki ile gösterilen veri kümeleri ise Freebase veri kümesinden arıtılarak elde edilmiştir.

Tez çalışması kapsamında güven değerlerinin yayılmasına yönelik yapılan deneylerde BT çalışmalarında popüler olarak tercih edilen dört farklı veri kümesi kullanılmıştır. Farklı veri kümeleri kullanmada amaç yayılma işleminin sadece hatalı üçlü sayısı ve küme boyutuna göre değil hem de çizge topolojisine göre sonuçlarını değerlendirmektir. Veri kümesi seçiminde dönüşüm tabanlı ve kural tabanlı bilgi yerleştirme çalışmalarında aktif kullanılan veri setleri ile beraber NELL gibi gerçek-dünya büyük veri seti de tercih edilmiştir.

Çizelge 6.2. Bilgi çıkarımı için kullanılan yaygın veri kümeleri [197]

Veri Kümesi	İlişkiler	Varlıklar	Eğitim Kümesi	Geçerli Kümesi	Test Kümesi
WN18 [65]	18	40 943	141 442	5 000	5 000
FB15K [65]	1 345	14 951	483 142	50 000	59 071
WN11 [84]	11	38 696	112 581	2 609	10 544
FB13 [84]	13	75 043	316 232	5 908	23 733
WN18RR [198]	11	40 943	86 835	3 034	3 134
FB15K-237 [199]	237	14 541	272 115	17 535	20 466
FB5M [66]	1192	5 385 322	19 193 556	50 000	59 071
FB40K [67]	1336	39 528	370 648	67 946	96 678

Güven değerlerinin güncellenmesi ve yayılma işleminin sınanması için seçilen ilk küme daha önce BT çalışmalarında kullanılan FB15K veri kümesidir. FB15K veri kümesi birçok

BT yerleştirme ve arıtma çalışmalarında tercih edilmiştir. FB15K kendi içinde yaklaşık 15000 varlık barındırmaktadır. Kullanılan diğer küme NELL [10], 2010 yılından beri geliştirilmeye devam etmektedir ve kendi içinde üçlüler için güven değerleri de barındırmaktadır. NELL veri setinde aday ve yüksek güven değerine sahip üçlüler yer almaktadır. Deneylerde yüksek güven değerine sahip küme kullanılmıştır. Tercih edilen üçüncü veri seti WN18, WordNet veri setinin alt kümesi olarak oluşturulmuştur ve eğitim seti 141442 üçlüden oluşmaktadır. Bu üçlüler 18 farklı ilişki türüne sahiptir. WN18 ve FB15K kümeleri Bordes ve ark. [65] tarafından geliştirilmiştir. Kullanılan son veri kümesi YAGO3-10 [198] YAGO3 veri kümesinin 123182 varlık ve 37 ilişkiden oluşan alt kümesi olarak hazırlanmıştır. Kümelere ilişkin istatistikler Çizelge 6.3'de gösterilmiştir. Seçilen tüm veri kümelerinde yönlü bağlantıların tekrarlı ilişkilere yol açmaması için aynı yükleme sahip nesne özne ilişkileri temizlenmiştir.

Çizelge 6.3. BT deneylerinde kullanılan veri kümeleri

Veri Kümesi	İlişkiler	Varlıklar	Üçlüler
WN18 [65]	18	40 943	141 442
FB15K [65]	1 345	14 951	483 142
YAGO3-10 [198]	37	123 182	1 079 040
NELL [10]	-	-	2 810 379

BT üzerinde yazma, arama ve silme işlemlerinin performans optimizasyonuna yönelik deneylerde ise veri kümesi olarak Freebase seçilmiştir. Freebase veri kümesi 2007 yılında Metaweb firması tarafından geliştirilen aynı adlı BT'nin altyapısında kullanılmıştır. Bu veri kümesi gönüllü topluluk üyeleri tarafından girilen yapısal verilerden oluşmaktadır. 2010 yılında Google tarafından satın alınan BT [200], 2014 yılında Google Knowledge Graph BT'sinin altyapısını oluşturmak için kullanılmıştır [7]. 2016 yılı itibariyle Google, Freebase sistemine erişimi durdurmakla birlikte akademik çalışmalar için veri kümesinin çevrim dışı olarak kullanılmasına olanak tanımıştır. 2014 yılı itibariyle Freebase 44 milyon başlık ve 2,4 milyar üçlü içermekte idi [12, 201]. Google tarafından çevrim dışı olarak sunulan veri kümesinde ise 1,9 milyar üçlü bulunmaktadır. Performans optimizasyonu ile ilgili deney tasarımı ve elde edilen sonuçlar Bölüm 6.7'de sunulmuştur.

6.3. Yanlış Üçlüler ve Güven Değerlerinin Oluşturulması

Deneysel çalışmaların yapılabilmesi için geçerleme kümesine yanlış üçlülerin eklenmesi gerekmektedir. Yanlış üçlülerin oluşturulmasında DSKRL [108] ve PTrustE'nin [116]

yaklaşımı göz önünde bulundurulmuştur. Yanlış üçlülerin oluşturulması için aşağıdaki adımlar uygulanmıştır. Önce veri kümesinden (h, r_1, t) ve (t, r_2, s) üçlüleri $h \neq s$ ve $r_1 \neq r_2$ olacak şekilde rastgele seçilir. $R' = \{r \mid r \neq r_1 \; ve \; r \neq r_2\}$ ilişki kümesinden rastgele r ilişkisi seçilerek (h, r, s) yanlış (*corrupted*) üçlüsü oluşturulur. Oluşturulan ilişki veri kümesine sıra gözetilmeksizin rastgele eklenir. Yanlış üçlü oluşturma oranı farklı deney tasarımları için geçerleme kümesinin %20, %40 ve %60 oranlarında uygulanarak veri seti zenginleştirilir. Farklı hatalı üçlü oranlarının üretilmesinde amaç hata oranının artışına göre yayılma işleminin başarısını ölçmek olmuştur.

NELL veri kümesi hariç diğer popüler veri kümeleri kendi içinde herhangi güven değerleri barındırmamaktadır. Bu nedenle deneysel çalışmalarda kullanılmak üzere üçlüler için tanımlı güven değerleri sentetik olarak üretilmiştir. Sentetik veri üretme işlemlerinde tarafsız yaklaşım sergilemek amacıyla verilerin dağılımı üçlülerin içeriğinden bağımsız hale getirilmiştir. Güven değerlerinin belirlenmesinde normal dağılım, üstel dağılım ve rastgele üretilmiş güven değerleri kullanılmıştır. Normal dağılımda doğru ve yanlış değerlerin standart sapmaları eşit; 0,1 olarak tanımlanmıştır. Normal dağılım için doğru ve yanlış üçlülerin güven değeri dağılım ortalamaları sırasıyla 0,7 ve 0,4 olarak belirlenmiştir. Yanlış değerlerin silinmesini amaçlayan deneylerde silme eşik değeri 0,3 olarak alınmıştır. 0,3 eşik değeri 0,4 ortalama ve 0,1 standart sapma değerlerine sahip yanlış üçlülerin en alt seviye tolerans eşiğine denk gelmektedir. Önerilen yöntemin güven değerlerinin dağılımından bağımsız olduğunu denemek için benzer deneyler üstel dağılım ve rastgele üretilmiş güven değerleri üzerinde de tekrarlanmıştır. Sentetik değerlerdeki bu varsayımlar tamamen değiştirilebilir veya dış kaynak üzerinden elde edilen güven değerleri ile oluşturulmuş BT'deki değerler kullanılabilir.

6.4. Deney Tasarımı

Deneyler FB15K, NELL, WN18 ve YAGO3-10 veri setlerinin eğitim kümelerinden 10K, 20K, 30K ve 40K şeklinde ayrıştırılmış altkümeleri üzerinde yapılmış ve bu sayede farklı küme boyutlarında önerilen yöntemin başarı oranlarındaki ve işlem performansındaki değişim incelenmiştir. Geçerli veri kümeleri yanlış üçlüler ile zenginleştirilirken nihai kümede üçlü sayısı belirtilen oranda artmaktadır. Sonuçların değerlendirilmesinde 10K, 20K, 30K ve 40K ifadeleri geçerli üçlü sayılarına atıfta bulunmak için kullanılacaktır. Gerçek veri kümesi boyutlarının hesaplanması için yanlış üçlü oranlarının bu sayılara

eklenmesi gerekmektedir. Örneğin, %60 oranında yanlış veri içeren 30K kümesi toplam 48000 üçlüye sahiptir.

Yayılma işlemi veri kümesinde bulunan ilk üçlüden başlayarak tüm üçlüler için eşzamanlı olarak çalıştırılmaktadır. BT'ye ekleme işlemlerinde üçlülerin sıralamasının deney sonuçlarına etkisini ortadan kaldırmak amacıyla üçlüler rastgele farklı sıralarda eklenmiştir. Farklı sıralama işlemlerinde elde edilen sonuçlar karşılaştırılmış ve nihai başarı oranlarının hesaplanmasında bu değerlerin ortalaması dikkate alınmıştır.

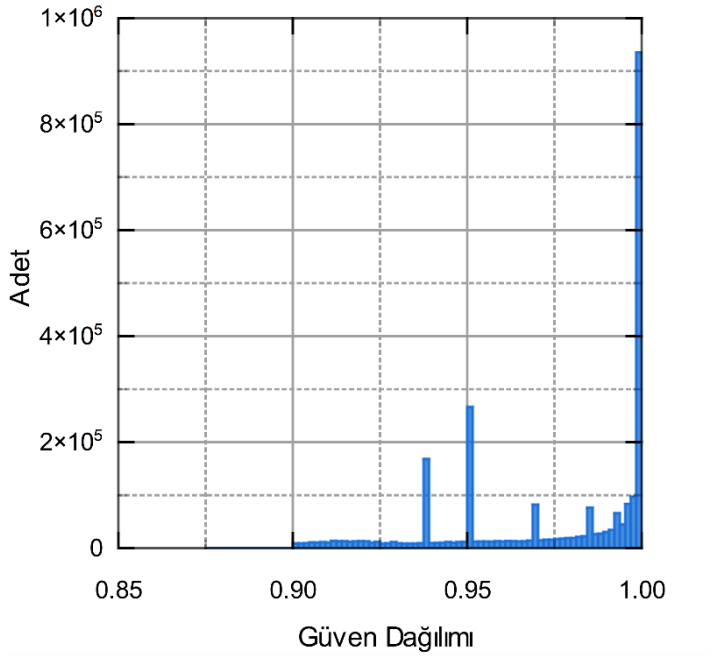
Deney tasarımlarında ayrıca Bölüm 4.3'te anlatılan yayılma kriterleri ele alınmış ve farklı kriterler için deneyler tekrarlanmıştır. Yapılan deneylerde dikkat edilen bir diğer nokta sigmoid yöntemlerinde güven değerlerinin hesaplanmasında farklı dikey parametrelerin ve ağırlık çarpanlarının etkisi, FPN yönteminde silme işleminin yayılmadan önce veya sonra yapılması, bulanık sistemde kural seti ve üyelik fonksiyonu parametreleri olmuştur.

Tasarlanan tüm deneyler 3 kez tekrarlanarak her bir deney için sonuç ortalamaları dikkate alınmıştır. Bu şekilde ekleme sıralarında, senkron ve asenkron ekleme işlemlerinde oluşabilecek yanlı değerler bertaraf edilmiştir.

6.5. Değerlendirme Ölçütleri

Yayılma işleminin değerlendirilmesi farklı yaklaşımlarla yapılabilir. İlk yaklaşım başarı oranının yanlış üçlülerin silinme sayısına göre belirlenmesidir. Bu durumda karışıklık matrisinde (*confusion matrix*) gerçek pozitif (*true positive - TP*) değer silinen yanlış üçlüleri gösterecektir. İkinci yaklaşım doğru üçlülerin mümkün olduğunca az silinmesini temel alır. Bu durumda silinmeyen geçerli üçlü sayısı gerçek pozitif değeri ifade edecektir. Bu çalışmada yanlış üçlülerin silinme sayısı gerçek pozitif değer olarak alınmıştır.

Yayılma işleminin başarısını gösteren başka bir ölçüt yayılma işlemi öncesi ve sonrasında güven değerlerinin dağılımlarıdır. Zayıf bağlantılar silindiği ve güçlü bağlantılar güçlendiği için güven değerlerinin ortalamasının yayılma işlemi sonrasında yükselmesi beklenmektedir. Bu nedenle deneylerde kullanılan veri kümelerinin işlem öncesi ve sonrası güven değerleri ortalamaları karşılaştırılmıştır.



Şekil 6.3. NELL veri kümesinde tanımlı doğru güven değerlerinin dağılımı

Değerlendirmede kullanılan üçüncü ölçüt elde edilen güven değeri dağılımının gerçek dünya güven değeri dağılımı ile karşılaştırma olmuştur. Deneylerin yapıldığı veri kümelerinden sadece NELL veri kümesinde doğru veriler için güven değerleri bulunmaktadır. Bu veri kümesinden alınmış 2766078 güven değerinin dağılımı Şekil 6.3'te gösterilmiştir. Yayılma işleminin etkinliğini test etmek için işlem sonucunda elde edilmiş doğru güven değerlerinin dağılımları bu dağılımla karşılaştırılmıştır.

Çizelge 6.4. Veri kümeleri için değerlendirme sonuçları

	Doğru Üçlü Sayısı	Doğruluk	Kesinlik	Duyarlılık
FB15K	10 000	0,87100	0,98783	0,51098
	20 000	0,88001	0,98344	0,54914
	30 000	0,88910	0,97794	0,58678
	40 000	0,89881	0,97386	0,63096
NELL	10 000	0,92909	0,99562	0,73605
	20 000	0,92769	0,99342	0,73359
	30 000	0,92899	0,99260	0,73649
	40 000	0,93105	0,99218	0,74426
WN18	10 000	0,86886	0,98808	0,50938
	20 000	0,86877	0,98806	0,51020
	30 000	0,87216	0,98400	0,52752
	40 000	0,87696	0,98324	0,54920
YAGO3-10	10 000	0,93626	0,99697	0,76157
	20 000	0,93338	0,99467	0,75277
	30 000	0,93065	0,99322	0,74206
	40 000	0,93094	0,99395	0,74303

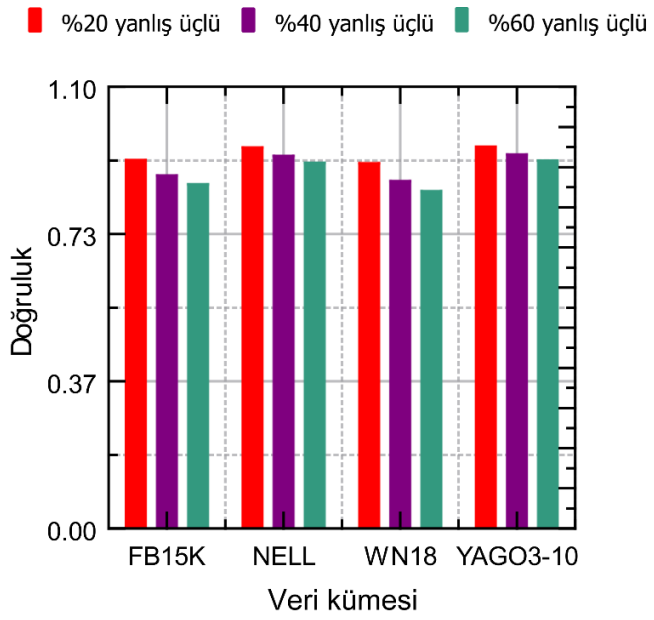
6.6. Deneysel Sonuçlar

Bu başlıkta yukarıda ele alınan deneysel çalışma ve değerlendirme kriterlerine göre elde edilen sonuçlar incelenecektir. Sonuçlar çizge üzerinde sigmoid fonksiyonu ile yayılma yöntemi ve FPN yöntemi için iki ayrı başlık şeklinde sunulmakla birlikte nihai sonuçlar karşılaştırılacaktır.

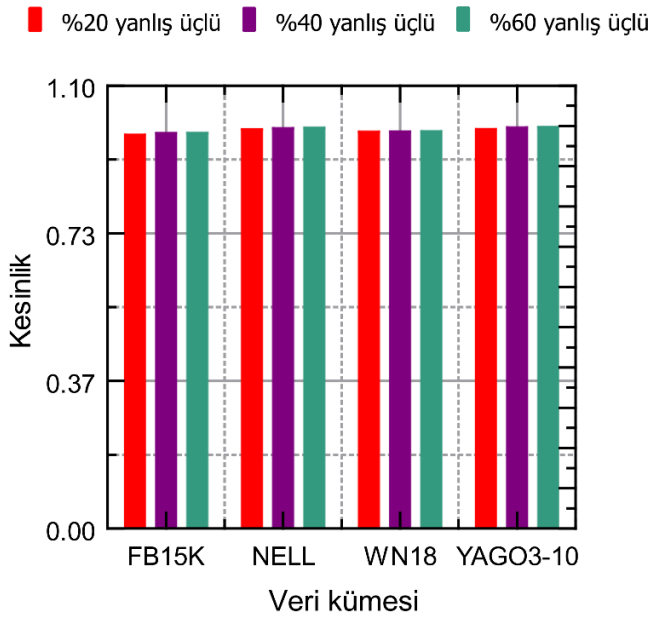
6.6.1. Çizge üzerinde sigmoid yöntemi ile elde edilen sonuçlar

Çizelge 6.4'te farklı veri setleri ve farklı küme boyutlarında normal dağılımla oluşturulmuş güven değerleri için değerlendirme sonuçları gösterilmiştir. Görüldüğü gibi veri kümesinin boyutu arttıkça doğruluk ve kesinlik değerlerinde küçük değişimler gözlemlenmektedir. Duyarlılık ise küçük veri kümelerinde düşük değerlerle başlamasına rağmen küme boyutu ile doğru orantıda yükselmektedir. Bu eğilim veri kümesinin büyümesi durumunda duyarlılık puanında da başarı oranlarının yükseldiğini göstermektedir.

Yanlış üçlü oranındaki değişim de sonuçları etkilemektedir. Şekil 6.4'te yanlış oranları için veri setlerinde doğruluk değişimi karşılaştırmalı olarak gösterilmiştir. Tüm veri kümeleri için yanlış üçlü oranında artışa rağmen doğruluk değerleri 0,8 bandının üzerinde seyretmektedir.

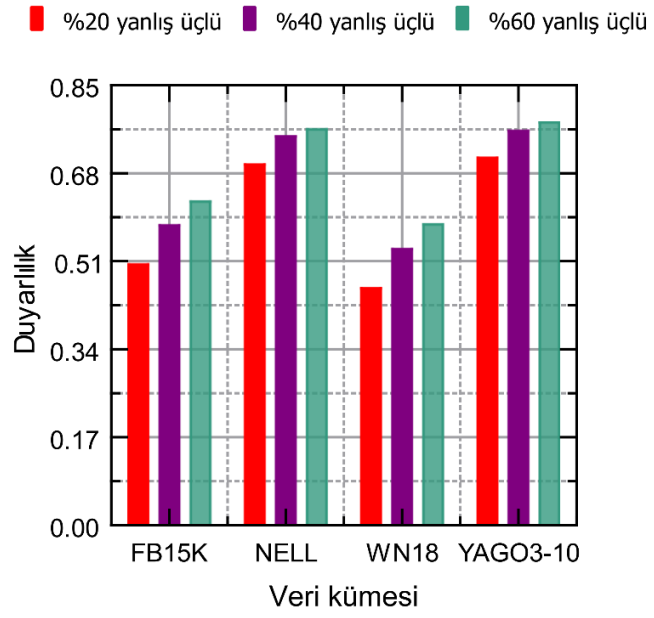


Şekil 6.4. Yanlış üçlü oranları için doğruluk değerleri



Şekil 6.5. Yanlış üçlü oranları için kesinlik değerleri

Şekil 6.5'te yanlış üçlü oranları için veri kümelerinin kesinlik değerlerindeki değişim gösterilmiştir. Kesinlik değerlerindeki değişim doğruluk değerlerine göre daha stabildir ve yanlış üçlü oranlarının artmasına rağmen silinen yanlış üçlülerin silinen doğru üçlülere oranla sabit kaldığını göstermektedir.



Şekil 6.6. Yanlış üçlü oranları için duyarlılık değerleri

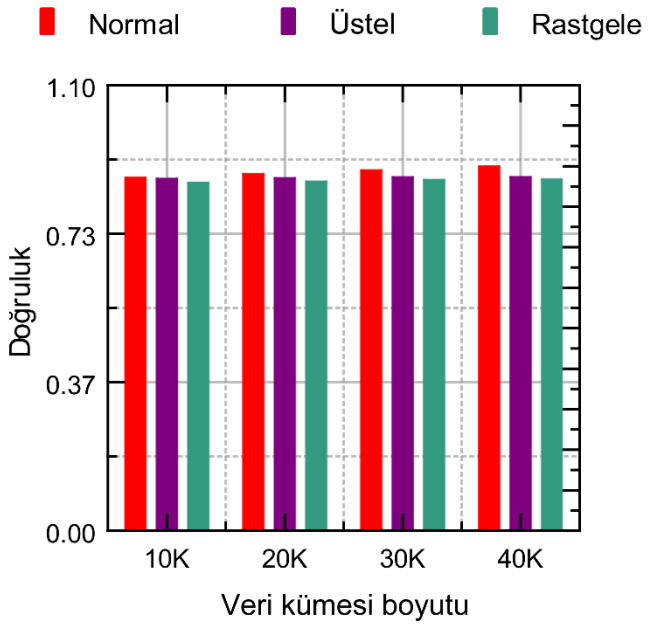
Şekil 6.6 yanlış üçlü oranları için duyarlılık değişimini göstermektedir. Görüldüğü gibi tüm veri kümelerinde yanlış oranlarının artması ile birlikte duyarlılık değeri de yükselmektedir.

Duyarlılık hesaplaması silinen yanlış üçlülerin toplam yanlış üçlülere oranı olarak ele alınmıştır.

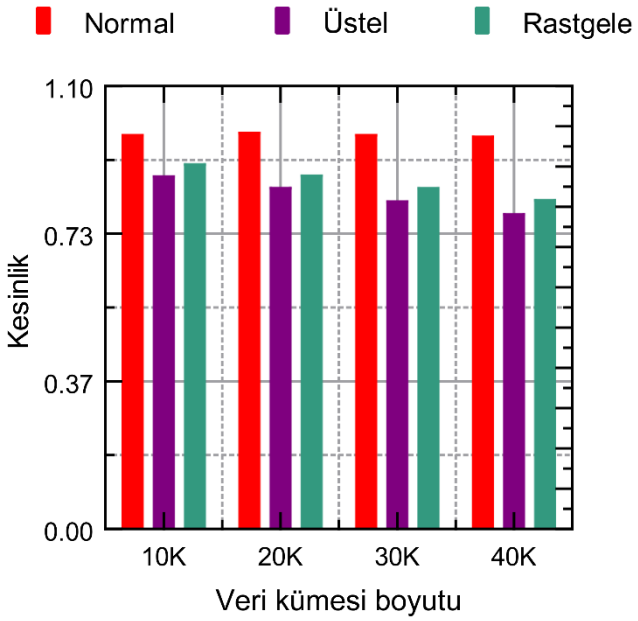
Deney sonuçlarının normal dağılımla üretilmiş güven değerlerinden bağımsız değerlendirilmesi için FB15K veri seti üzerinde deneyler üstel dağılım ve rastgele verilerle üretilmiş güven değerleri ile tekrarlanmıştır. Bu deneyler farklı veri seti boyutu ve farklı yanlış üçlü oranları için uygulanmıştır. Her bir veri seti boyutu için elde edilen değerler farklı yanlış oranlarının ortalaması olarak hesaplanmıştır.

Şekil 6.7'de dağılımlar için doğruluk sonuçları karşılaştırılmıştır. Bu sonuç dağılımdan bağımsız olarak %80'in üzerinde doğruluk oranı elde edildiğini göstermektedir. Doğruluk oranları için güven değerlerinin dağılım varyasyonu önemli farklılık oluşturmamaktadır.

Şekil 6.8'de farklı dağılım türleri için farklı veri seti boyutlarında elde edilmiş kesinlik değerlerindeki değişim gösterilmektedir. Burada da normal dağılımla elde edilmiş sonuçlar ≈%98, diğer dağılımlarda da ≈%80 üzerinde sonuç elde edilmiştir. Bu sonuçlar da dağılımın karakteristiğinden bağımsız olarak yayılma işleminin %80 üzerinde kesinlik sonucu elde ettiğini göstermektedir.

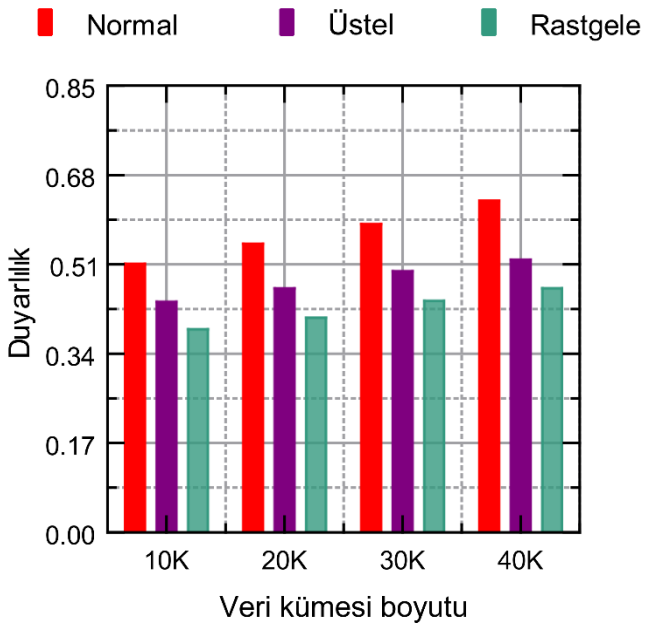


Şekil 6.7. Farklı dağılımlar için doğruluk değerleri



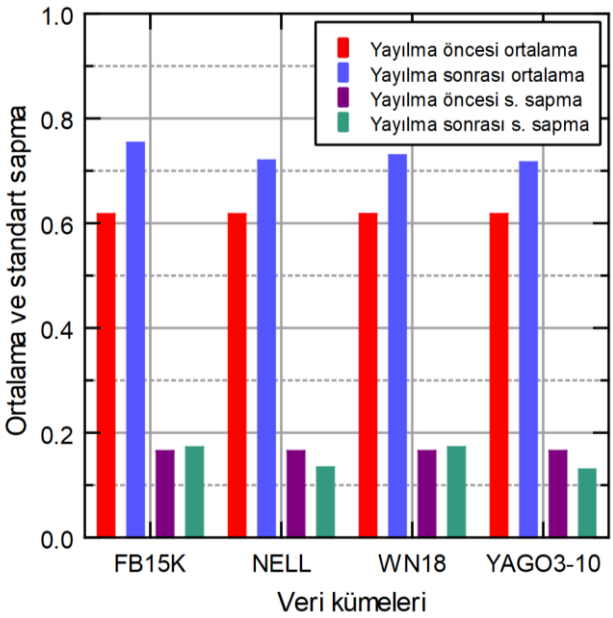
Şekil 6.8. Farklı dağılımlar için kesinlik değerleri

Şekil 6.9'da farklı dağılımlar için duyarlılık sonuçları karşılaştırılmıştır. Görüldüğü gibi farklı veri kümesi boyutlarında normal dağılım için duyarlılık sonuçları %50-%60 aralığında, diğer dağılımlar için ise %40-%50 aralığında değişmektedir. Bu deneylerde dikkat çeken nokta tüm dağılımlar için veri setinin boyutuna bağlı olarak kesinlik ve duyarlılık sonuçlarının yükselmesidir. Bu durum daha önce normal dağılımda elde edilen veri setinin boyutuna bağlı iyileşme sonucunu teyit etmektedir.



Şekil 6.9. Farklı dağılımlar için duyarlılık değerleri

Elde edilen sonuçlar önerilen modelde ileri sürüldüğü gibi güven değerlerinin yayılmasının sistemin stabil duruma ulaşmasına katkı sağladığını göstermektedir. Aynı zamanda zayıf güven değerlerindeki artışa rağmen duyarlılık azalmamakta tam tersi artmaktadır. Zayıf ve güçlü bağlantıların eşit dağılıma sahip olacağı varsayımı doğru kabul edilirse %100 gibi daha yüksek yanlış üçlü oranlarında zayıf üçlülerin daha etkin şekilde temizleneceği öngörülebilir.



Şekil 6.10. Yayılma işlemi sonrasında ortalama ve standart sapma değişimi

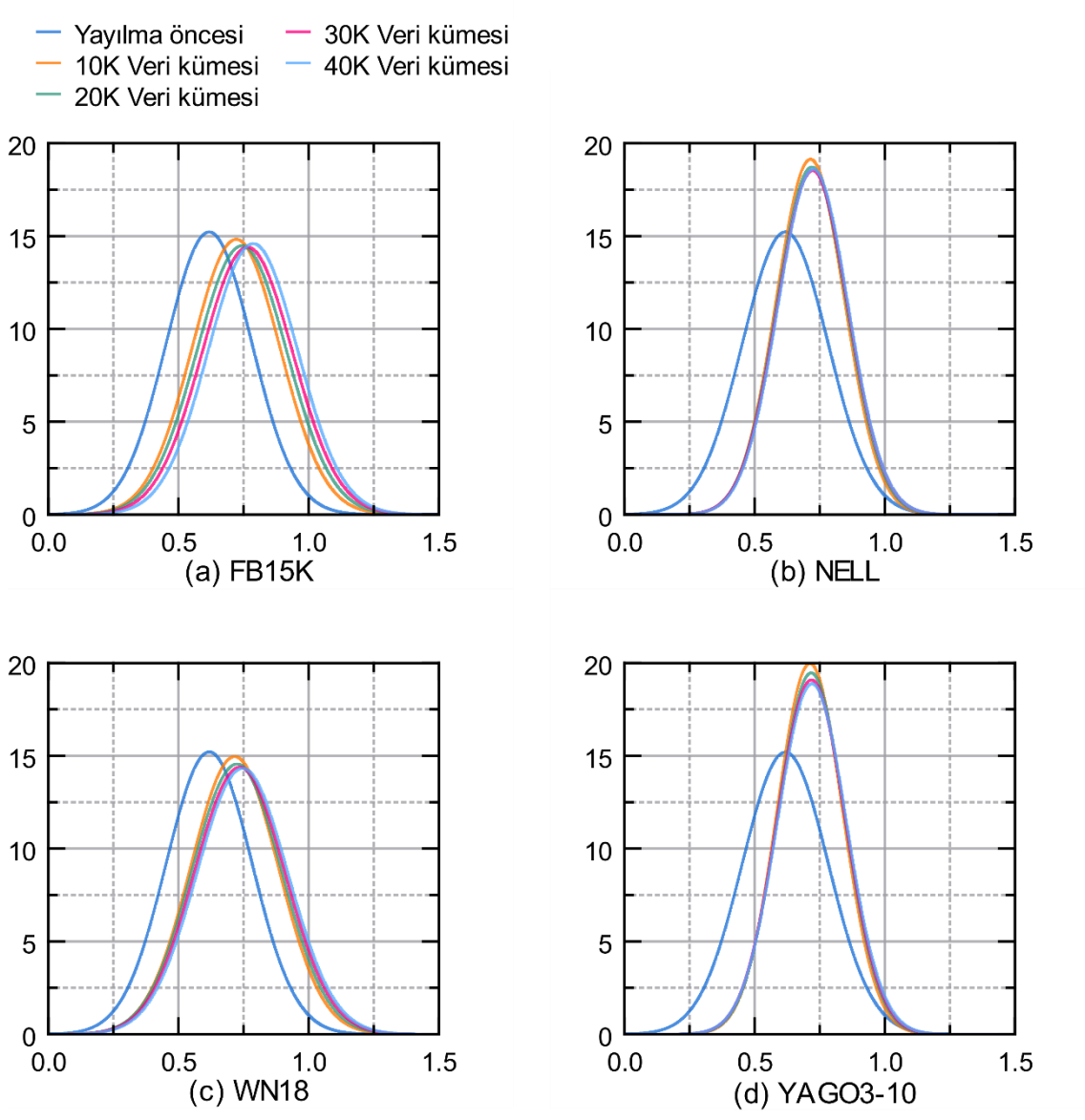
Deney sonuçları dağılımların ortalama ve standart sapma değerlerine göre de incelenmiştir. Şekil 6.10'de veri kümelerinin ilk ortalamaları ve yayılma işlemi sonrasında oluşan ortalamaları karşılaştırılmıştır.

Çizelge 6.5. Yayılma işlemlerinde ortalama ve standart sapma değişimi

Veri Kümesi	Yanlış Üçlü Oranı	İlk Ortalama	Son Ortalama	İlk Standart Sapma	Son Standart Sapma
FB15K	%20	0,64845	0,73470	0,14749	0,15675
	%40	0,65095	0,75521	0,14944	0,17517
	%60	0,65003	0,77027	0,14997	0,18404
NELL	%20	0,61303	0,71513	0,16822	0,12821
	%40	0,61428	0,72055	0,16806	0,13400
	%60	0,61478	0,72310	0,16789	0,13891
WN18	%20	0,58713	0,71603	0,17623	0,15332
	%40	0,58690	0,72983	0,17597	0,17566
	%60	0,58820	0,74280	0,17655	0,18771
YAGO3-10	%20	0,58713	0,71137	0,17623	0,12456
	%40	0,58690	0,71675	0,17597	0,12974
	%60	0,58820	0,72002	0,17655	0,13444

Yanlış üçlülerin veri kümesinde bulunması ilk durum için güven değeri ortalamasını düşürmektedir. Bu nedenle tüm veri kümelerinde ortalama güven değerleri yayılma işlemi öncesi duruma göre artış göstermiştir. Bu durum yayılma işleminde hem zayıf bağlantıların silinmesi hem de güçlü bağlantıların güçlendirilmesi ile açıklanır.

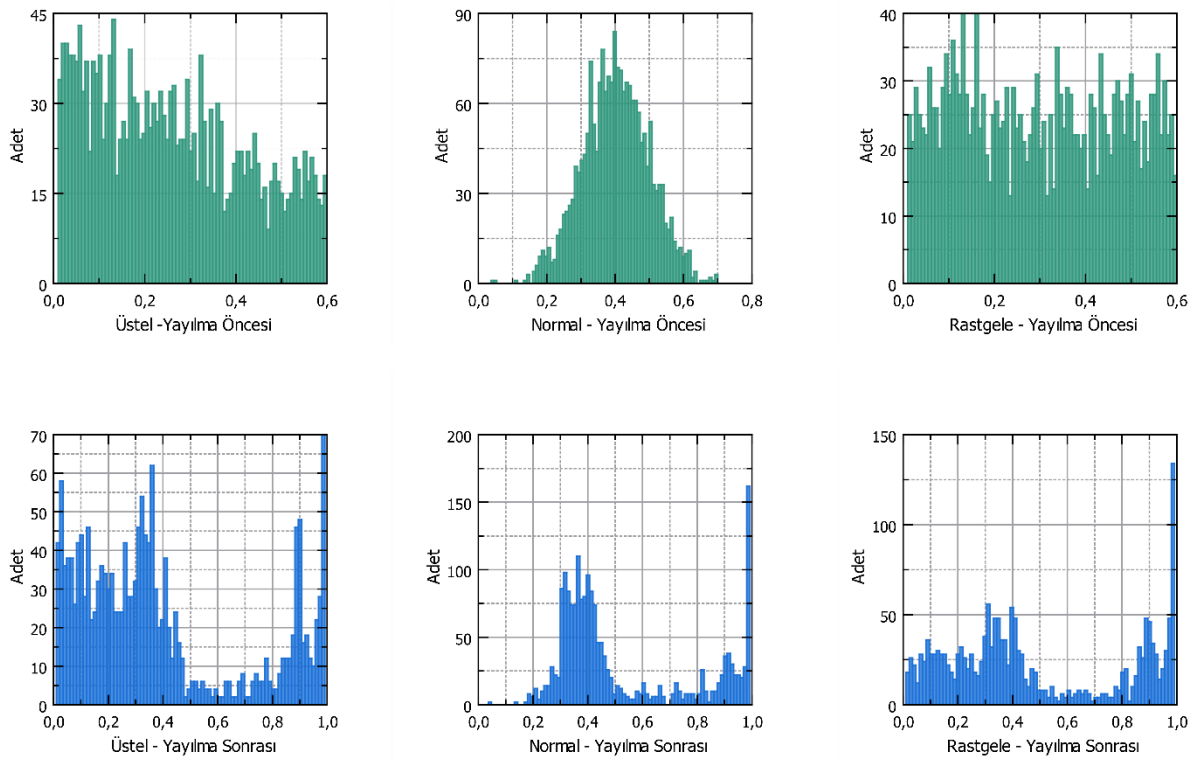
Küme ortalamasında beklenen yükselmeden farklı olarak standart sapma değerlerinin artmaması beklenir. Şekil 6.10'da yayılma işlemi sonrasında standart sapma değerlerinin değişimi gösterilmiştir. Grafikten de görüldüğü gibi yayılma işlemi sonrasında standart sapma değerlerinde artış gözlemlenmemiştir. Deneylerde elde dilen ilk ve son ortalama ve standart sapma değerleri Çizelge 6.5'te gösterilmiştir.



Şekil 6.11. Veri kümeleri için güven değeri değişimi

Ortalama ve standart sapmaya yönelik küme değerlendirmelerini daha iyi açıklayabilmek için yanlış üçlü oranlarının farklı değerlerinde çan eğrisi değişimleri incelenebilir. Bu grafikler tüm veri setleri için farklı küme boyutlarında yanlış üçlü oranlarının ortalaması dikkate alarak oluşturulmuştur. FB15K için çan eğrilerinde oluşan kayma Şekil 6.11 (a)'da gösterilmiştir. Görüldüğü gibi 10K, 20K, 30K ve 40K veri kümelerinde yayılma öncesi ve yayılma sonrası dağılımlarda sağa doğru kayma oluşmuştur. Ayrıca küme boyutunun büyüyerek üçlü sayısının artması sağa doğru kaymayı artırmaktadır.

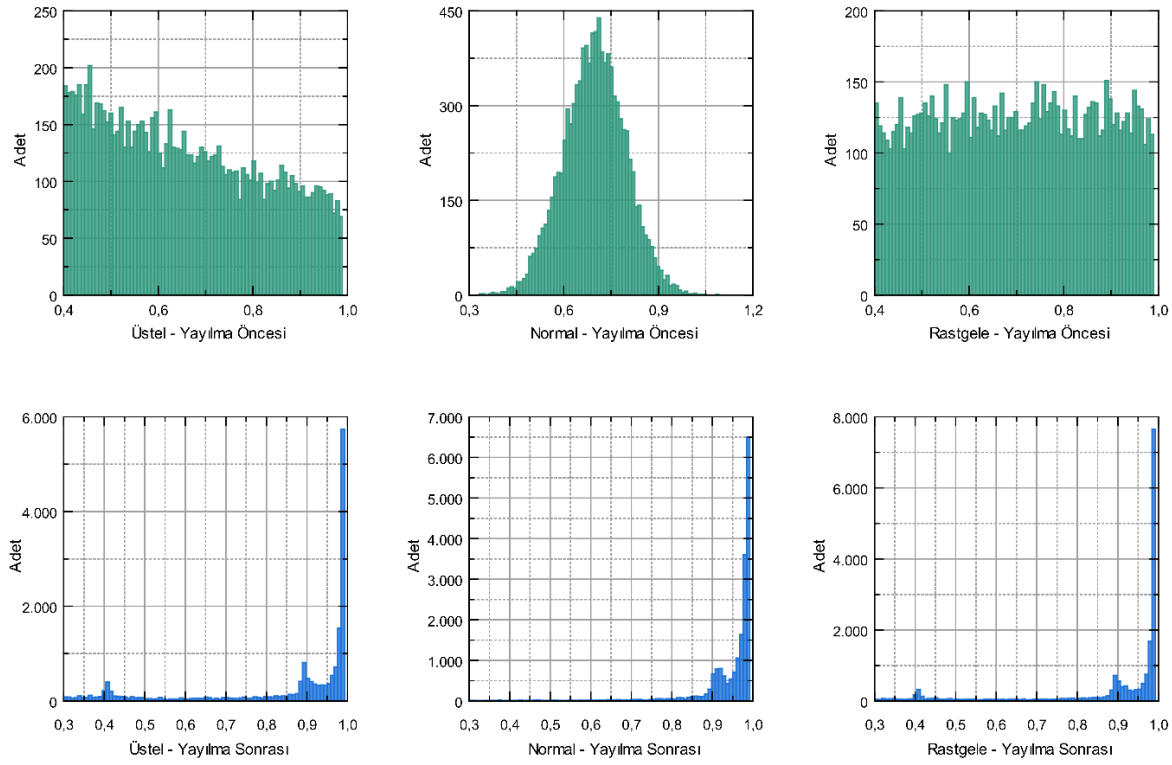
Şekil 6.11 (b) NELL veri seti için güven değerlerinde oluşan kaymayı gösterir. Burada da FB15K veri setine benzer sonuçlar elde edilmiştir. Benzer durum Şekil 6.11 (c)'de WN18 için gösterilmiştir. Ortalamaların değişiminden kaynaklı kayma durumu veri kümesinin boyutuna bağlı olarak artmaktadır. Bu durum yayılma işlemi sonrasında güven değerlerinin temizlenerek yükseldiğini göstermektedir. Şekil 6.11 (d)'de ise YAGO3-10 veri seti için farklı küme boyutlarında çan eğrisi değişimini göstermektedir. YAGO3-10 veri setinde de diğer veri setlerine benzer sonuçlar elde edilmiştir. Küme boyutlarındaki artış güven ortalamasının da artmasına neden olmaktadır.



Şekil 6.12. Yanlış üçlüler için başlangıç ve son güven değeri dağılımları

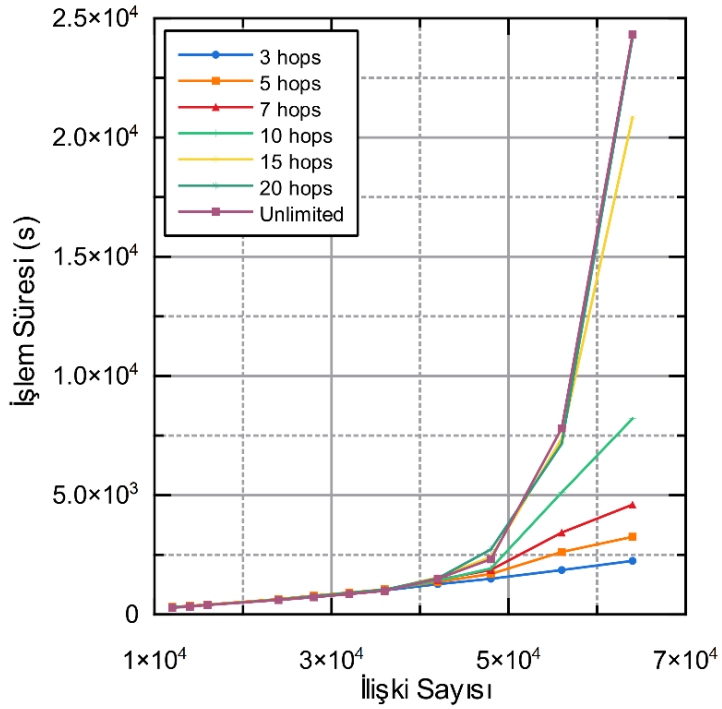
Sonuç olarak veri kümesinin boyutu arttıkça BT'nin daha iyi saflaştırıldığı önerisi deneysel olarak doğrulanmaktadır. Bu durum çizgeye eklenen yanlış üçlülerin oranı ile de ilintilidir ve oran arttıkça yanlış bilginin temizlenme oranı da doğru oranda artmaktadır. Ayrıca saflaştırma işleminin veri setinden bağımsız olduğu da söylenebilir.

Farklı güven değeri dağılımlarında yayılma işleminin performansını daha iyi ölçümlemek için doğru ve yanlış üçlülerin başlangıç ve son dağılımları da karşılaştırılmıştır. Ayrıca doğru üçlülerin son dağılımları gerçek dünya verisine sahip NELL veri setinin doğru üçlü dağılımı ile de karşılaştırılmıştır. Şekil 6.12 üstel, normal ve rastgele üretilmiş yanlış üçlülerin başlangıç ve son dağılımlarını göstermektedir. Görüldüğü gibi yayılma işlemi sonrasında yanlış güven değerlerinin negatif yönlü yoğunluğu artmaktadır. Benzer şekilde Şekil 6.13 üstel, normal ve rastgele üretilmiş doğru üçlülerin başlangıç ve son dağılımlarını karşılaştırmaktadır. Burada ise pozitif yönde güven değerlerinin yoğunluğu gözlemlenmektedir. Ayrıca doğru verilerin yayılma işlemi sonrası dağılımı NELL veri setinden elde edilmiş Şekil 6.3'deki doğru güven değerlerinin dağılım karakteristiğini yansıtmaktadır.



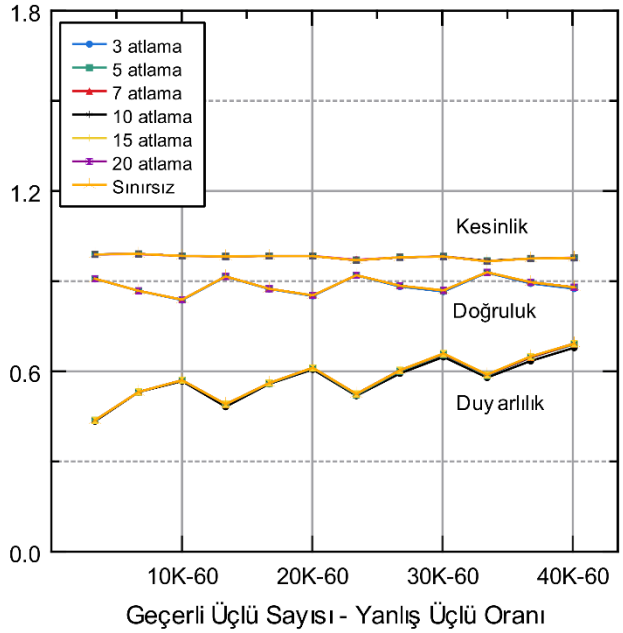
Şekil 6.13. Doğru üçlüler için başlangıç ve son güven değeri dağılımları

Deneyler başarı oranlarının değerlendirilmesi ile beraber yayılma işleminin performans değerlendirmelerini de göz önünde bulundurmuştur. Şekil 6.14'te FB15K veri seti için yayılma çapının işlem süresi üzerindeki etkisi gösterilmiştir. Görüldüğü gibi yayılma çapının sınırlandırılmadığı durumlarda toplam ilişki sayısının 40K üzerindeki değerlerinde yayılma işlemi için harcanan toplam süreler de hızlı bir şekilde artmaktadır. Buna karşılık yarıçapın, örneğin 3-atlama ile sınırlanması durumunda toplam işlem sürelerinde ciddi düşüş sağlanmıştır. 3-atlama yayılma çapının işlem süresi performansında iyileştirme sağlamakla birlikte saflaştırma başarı oranlarında da sınırsız yayılma işlemine çok yakın sonuçlar üretmektedir. Şekil 6.15'te 3-atlama yayılma çapı ve sınırsız yayılma çapı için başarı, duyarlılık ve kesinlik değerleri karşılaştırılmıştır.



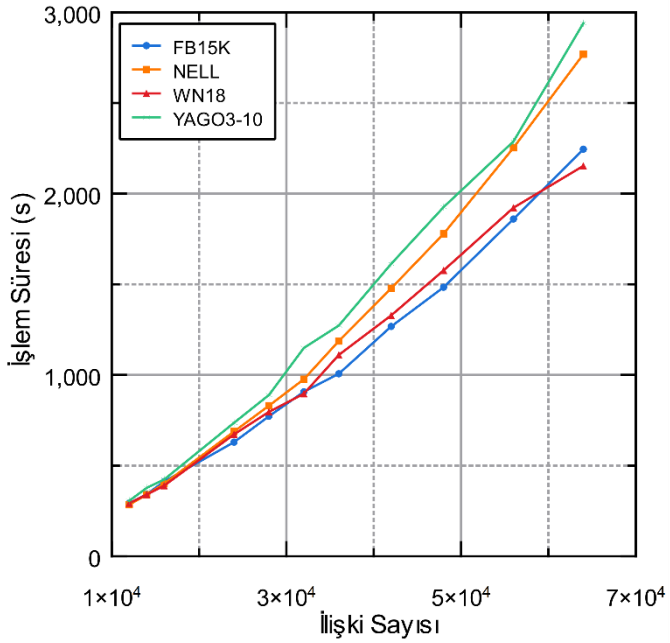
Şekil 6.14. FB15K için yayılma çapına göre işlem süresi değişimi

Deneylerde veri kümelerinin toplam işlem süreleri üzerinde etkisi de incelenmiştir. Şekil 6.16'te veri kümeleri için ilişki sayısına bağlı toplam işlem sürelerindeki değişim gösterilmiştir. Görüldüğü gibi veri kümeleri arasında toplam işlem süreleri değişimi anlamında büyük farklılık bulunmamaktadır. Oluşan farklılıklar veri kümelerinin düğüm sayısı ve barındırdığı üçgen sayısı gibi topolojik farklılıklardan kaynaklanmaktadır.



Şekil 6.15. FB15K için yayılma çapına göre başarı oranları

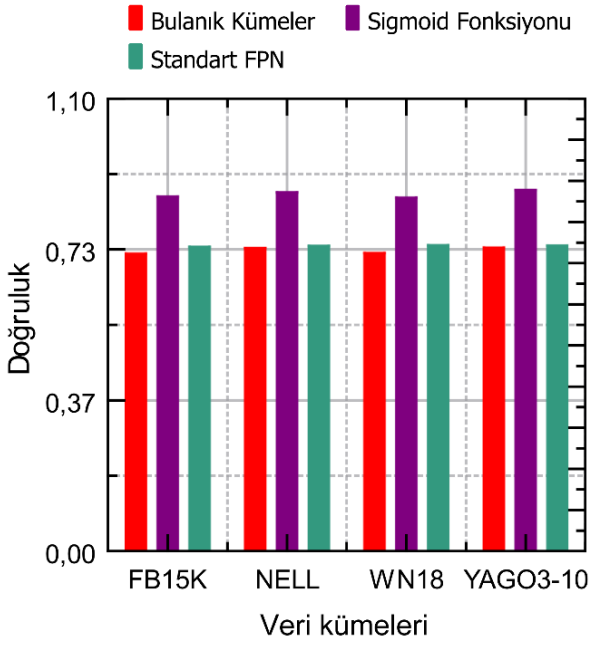
Sonuç olarak yayılma işleminin 3-atlama ile kısıtlanması hem başarı hem de işlem süresi performansı anlamında olumlu etki oluşturmuştur. Özellikle gerçek veri setlerinin daha büyük boyutlara ulaşacağı göz önünde bulundurulursa yayılma çapı kısıtlamasının başarı oranları anlamında bu sonuçları ortaya koyması işlem performansı açısından pozitif etki oluşturmaktadır.



Şekil 6.16. Veri kümeleri için ilişki sayısına bağlı işlem süresi değişimi

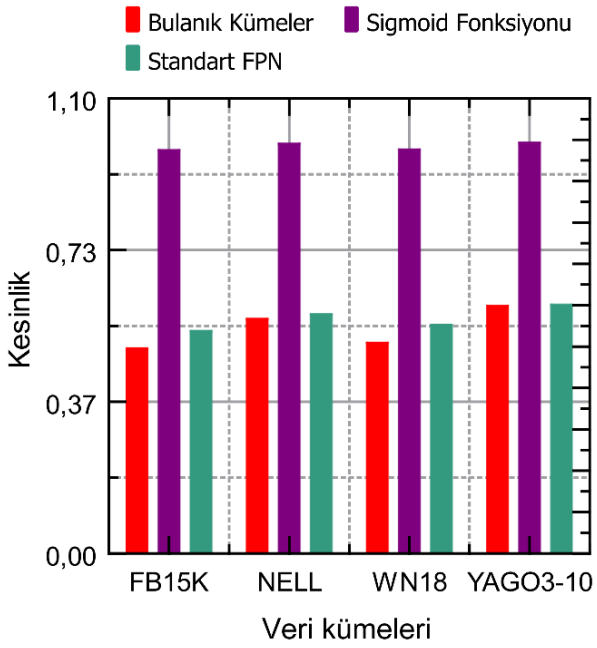
6.6.2. FPN yöntemleri için elde edilen sonuçlar

FPN yönteminde önceki bölümde bahsedildiği gibi standart, sigmoid ve bulanık kümeler hesaplama yöntemleri kullanılarak sonuçlar elde edilmiştir. Tüm sonuçlar çizge yönteminde olduğu gibi %20, %40 ve %60 hata oranlı veri setleri ve 10K, 20K, 30K ve 40K boyutundaki kümeler için üçer kez tekrarlanmıştır. Deneylerin tekrarlanmasında üçlülerin sıralarının karıştırılması özellikle dikkate alınmıştır. Sunulan sonuçlarda tekrarlanmış deneylerin sonuç ortalamaları gösterilmektedir. FPN yöntemi ile elde edilen sonuçlarda genel itibariyle sigmoid hesaplama metodunun daha başarılı sonuçlar ortaya koyduğu söylenebilir.



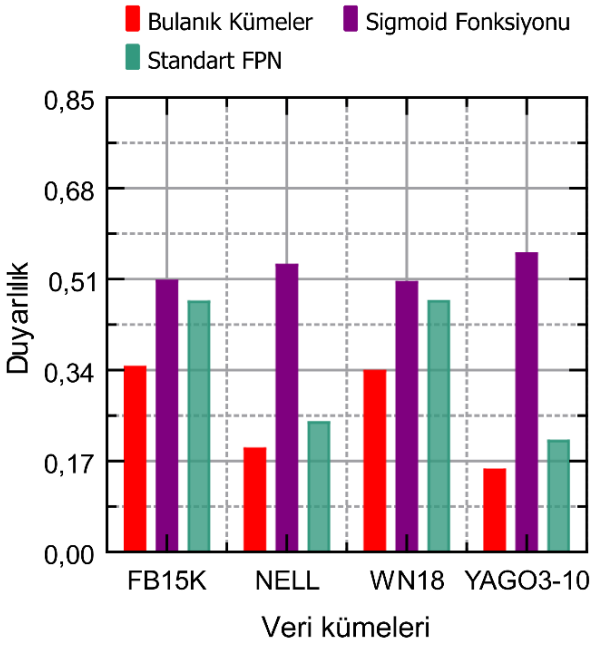
Şekil 6.17. FPN yönteminde hesaplama yöntemine bağlı başarı oranı değişimi

Şekil 6.17'te farklı veri kümeleri ve hesaplama yöntemleri için elde edilen doğruluk oranları gösterilmiştir. Sonuçlardan görüldüğü gibi veri setinden bağımsız olarak doğruluk oranları hesaplama yöntemine göre değişmektedir. Sigmoid yönteminde hesaplama sonuçları %85 bandında seyrederken diğer yöntemlerde sonuçlar %70 bandını korumaktadır.



Şekil 6.18. FPN yönteminde hesaplama yöntemine bağlı kesinlik değişimi

Şekil 6.18 veri kümeleri için hesaplama yöntemlerinden elde edilen sonuçların kesinlik açısından karşılaştırmasını göstermektedir. Başarı oranlarından farklı olarak sigmoid yöntemi burada daha da öne çıkarak %98 oranlarında sonuçlar ortaya çıkarmaktadır. Buna karşılık bulanık küme ve standart hesaplama yöntemleri %50-%55 bandında seyretmektedir.



Şekil 6.19. FPN yönteminde hesaplama yöntemine bağlı duyarlılık değişimi

Başarı oranları ve kesinlik değerleri ile birlikte FPN yönteminde elde edilen sonuçlar için duyarlılık ölçümleri de yapılmıştır. Duyarlılık sonuçları Şekil 6.19'te gösterilmiştir. Elde edilen veriler duyarlılık sonuçlarında da sigmoid yönteminin diğer iki yönteme oranla daha iyi sonuçlar ortaya çıkardığını kanıtlamaktadır. Duyarlılık değerleri özellikle varlık ve ilişki karmaşıklığının daha yüksek olduğu, dolayısıyla üçgen oluşma olasılığının daha düşük olduğu durumlarda daha kötü sonuçlar üretmektedir.

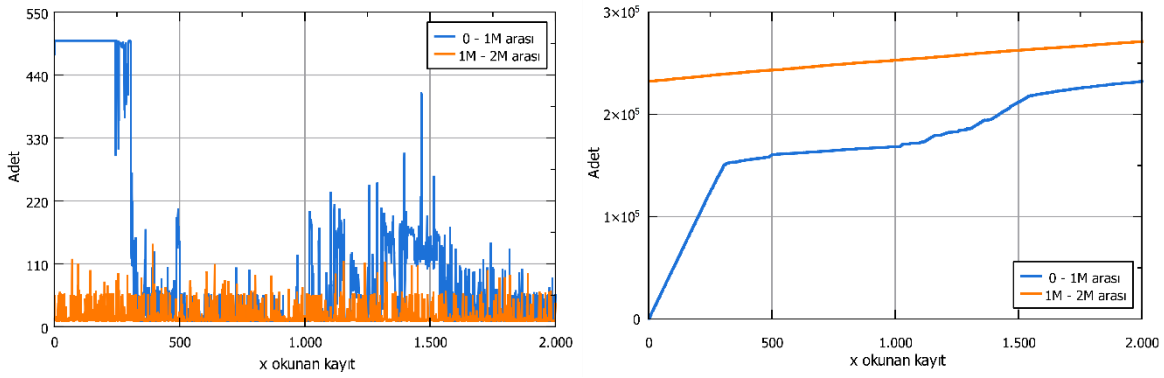
6.7. Performans Optimizasyonu için Deney Tasarımı

Freebase veri kümesi kullanılarak önerilen SBF'nin performans ölçümleri için Neo4j çizge veri tabanının 4.4.7 Desktop versiyonu tercih edilmiştir. Tüm deneyler AMD Ryzen 9 5950X 16 çekirdekli 3,4GHz işlemciye sahip 32 GB RAM ve SSD hafızalı bilgisayar üzerinde gerçekleştirilmiştir.

Deneysel çalışmada akan verinin sabit disk üzerindeki çizge veri tabanına yazma, okuma ve silme işlemlerinde performans değerlendirmesi yapılmıştır. Tüm deneyler Neo4j çizge veri tabanında bulunan aralık (*range*) indeks (R), metin (*text*) indeks (T) ve önerilen SBF indekslenmiş (B) verilerle gerçekleştirilmiştir. Neo4j kendi üzerinde range indeksleme yöntemi için BTREE indeksleme kullanmaktadır ve bu indeks 4096 byte veri boyutu sınırına sahiptir. Range indeksin üyelik ve eşitlik kontrollerinde kullanımı tavsiye edilmektedir. Text indeksleme ise daha çok metin içi aramalarda tercih edilmekle birlikte kümede bulunma (*list membership*) ve eşitlik kontrollerinde de kullanılabilir [202].

Bloom filtresi ile indeksleme deneylerinde performans ölçümleri yapmak amacıyla Freebase veri kümesinde bulunan ilk 2 milyon satır ele alınmıştır. Bu veri kümesinden ilk 1 milyon satır yazma, tekil değerleri tespit etme, arama ve silme işlemlerinde akan veri olarak kullanılmıştır. BT'de bulunmayan farklı değerlerle performans denemelerine tekrarlama amacıyla ikinci 1 milyon satırdan oluşan küme ile arama ve kayıt döndürme performans ölçümleri tekrarlanmıştır. Veri setinde her satır bir üçlüye karşılık gelmektedir. Bu nedenle her satırda *özne* ve *yüklem* olarak iki kaydın işlendiği göz önünde bulundurulmalıdır. Şekil 6.20 (a) ilk 1 milyon ve ikinci 1 milyon satır için okunan her 1000 kayıtta tekil değerlerin dağılımını göstermektedir. Benzer şekilde tekil değerlerin ilk 1 milyon ve ikinci 1 milyon satır için artış eğilimi de Şekil 6.20 (b)'de gösterilmiştir.

Görüldüğü gibi ilk 250000 satırda tekil değerlerin dağılımı ivmeli şekilde artarken sonradan daha yavaş bir artış sergilenmektedir.



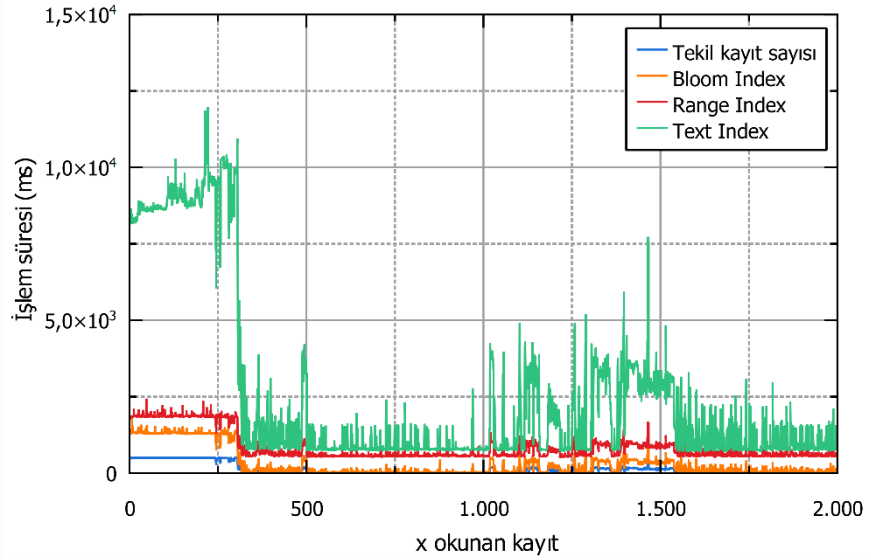
Şekil 6.20. Freebase veri seti için (a) tekil değerlerin her 1000 satır için dağılımı, (b) tekil değerlerin artış eğilimi

Tanımlanan SBF için n_0 = 10.000.000, P = 0,0001, r = 0,5 ve s = 2 parametreleri ile 1 Bloom filtresi oluşturulmuştur. Deneyler aşağıdaki işlemleri kapsayacak şekilde tekrarlanmıştır:

1. Ekleme: Ekleme işlemi boş çizge veri tabanına yapılmaktadır. Her bir ekleme işlemi öncesi varlıkların tekillik durumunu korumak için yeni gelen varlığın mevcut kümede bulunup bulunmadığı kontrol edilmektedir. Bu bir sonraki başlıkta ele alınacak üyelik arama işlemine karşılık gelmektedir. Aday varlığın kümede bulunmaması durumunda yeni varlık kümeye eklenmektedir.
2. Üyelik arama: Bu deneyin amacı aday varlığın kümede bulunma durumunu tespit etmektir. Varlığın kümede bulunması durumunda varlığın döndürülmesi söz konusu değildir.
3. Kayıt arama: Kayıt arama aday varlığın kümede bulunma durumunun ötesinde kaydın kendisini de döndürmektedir.
4. Silme: Silme deneyinde de akan veri üzerinden gelen verinin kümede bulunma durumuna göre işlem gerçekleştirilmiştir. Burada da Range indeks ve Text indeks deneyleri direkt silme işlemine tabi tutulurken SBF indeks deneylerinde sadece üyelik özelliğini sağlayan varlıklar silinmiştir.

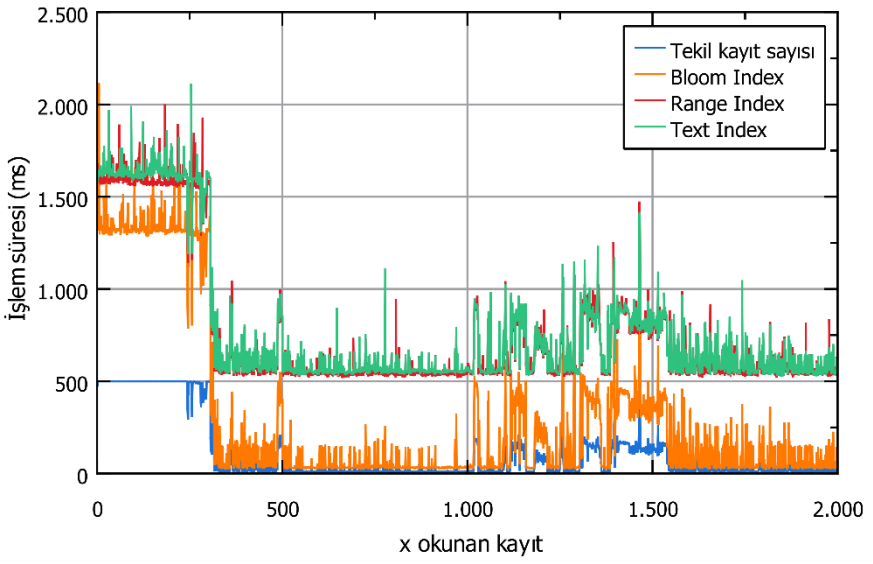
6.7.1. Performans ölçümleri

Deneylerin performans değerlendirmelerinde sabit satır sayısı için harcanan işlem süreleri karşılaştırılmıştır. İşlem zamanı karşılaştırmalarında her satırda iki varlık bulunduğundan 1000 kayıtlı işlem tekrarı oluşturabilme açısından satır sayısı 500 olarak belirlenmiştir. İşlem zamanlarındaki değişimler veri boyutundaki artışa bağlı olarak değerlendirilmiştir. Ayrıca işlemler için harcanan toplam ve ortalama işlem zamanları da karşılaştırılmıştır.



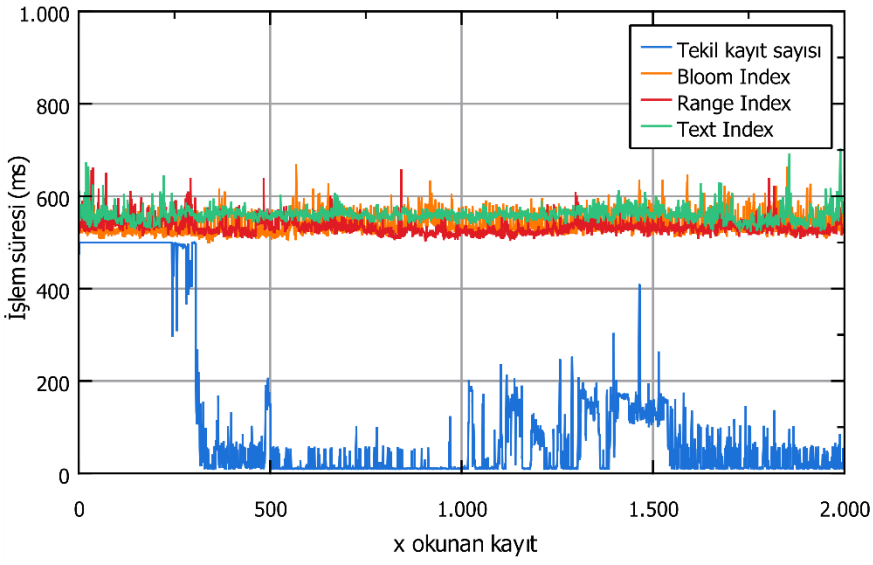
Şekil 6.21. Yazma işlemi için indeksleme yöntemlerinin işlem süresi dağılımları

İlk olarak farklı indeksleme yöntemleri için yazma işleminin işlem süreleri karşılaştırılmıştır. Bu deneyde yazma sürelerinin ilk 1 milyon satırdaki tekil değer dağılımına özdeş olarak seyretmesi beklenmektedir. Buna karşılık önerilen Bloom filtresi daha önce kaydedilmiş değerleri hash fonksiyonu yardımıyla daha hızlı sorgulayacağından önerilen yöntem için işlem sürelerinin diğer indeksleme yöntemlerine göre düşük çıkması gerekmektedir. Şekil 6.21'de indeksleme yöntemleri için yazma işleminin süre dağılımları gösterilmiştir. Görüldüğü gibi metin indeksleme yöntemi en verimsiz sonucu üretmesine karşılık, range indeksleme daha iyi sonuç üretmektedir. Ancak Bloom filtresi daha düşük işlem süreleri ile range indekslemeden de iyi sonuçlar üretmiştir.



Şekil 6.22. Silme işlemi için indeksleme yöntemlerinin işlem süresi dağılımları

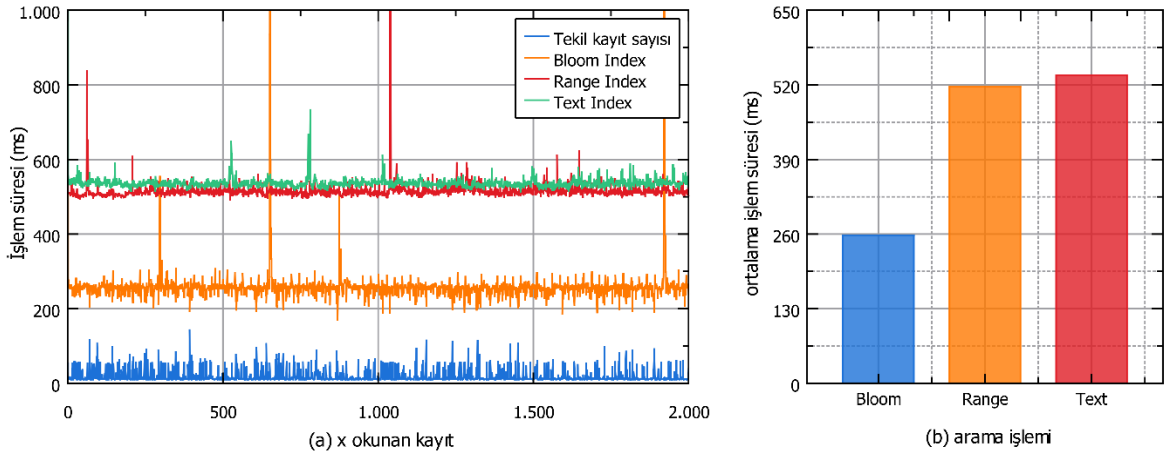
Silme deneylerinde daha önce BT'ye eklenmiş tüm veriler aynı sırada gelen tekil değerlerle karşılaştırılarak silme performansı sabit satır sayısının işlem süresi üzerinden değerlendirilmiştir. Silme işlemi için yapılan deneylerin işlem sürelerinin dağılımı da Şekil 6.22'de gösterilmiştir. Görüldüğü gibi bu grafikte de tüm indeksleme yöntemleri tekil değerlerin dağılımı ile uyumlu işlem süresi dağılımı sergilemektedir. Metin indeksleme yöntemi yazma işlemi sonucunda indeksleme sürecini tamamladığından range indeksleme ile yakın sonuçlar göstermektedir. Buna karşılık Bloom filtresi ile önerilen indeksleme diğer indeksleme yöntemlerinin ikisinden de daha iyi silme performansı ortaya koymaktadır.



Şekil 6.23. Kayıt arama işlemi için indeksleme yöntemlerinin işlem süresi dağılımları

Arama işlemi üyelik arama ve kayıt döndürme şeklinde iki farklı başlık altında ele alınarak performans değerlendirmeleri yapılmıştır. Kayıt arama işlemi yazma işlemi sonrasında ilk 1 milyon kaydın mevcut BT üzerinde aranması şeklinde düzenlenmiştir. Bu işlemin farklı indeksleme yöntemleri için süre dağılımları Şekil 6.23'te gösterilmiştir. Görüldüğü gibi kayıt arama işleminde tekil değerlerin dağılımından bağımsız olarak indeksleme yöntemleri yakın işlem sürelerine sahiptir. Bu arama işleminin mevcut BT'de var olan tüm kayıtlar için tekrarlanmasından kaynaklanmaktadır. Buna karşılık ikinci bir milyon satırda bulunan kayıtlar BT'ye yazma işleminde kullanılmış kayıtlardan farklı olacağından kayıt arama işlemlerinde de farklı sonuçların ortaya çıkacağı beklenir.

Şekil 6.24 (a) ikinci bir milyon satırda bulunan kayıtların arama işlem sürelerinin dağılımını göstermektedir. Görüldüğü gibi bu deneyde metin ve range indeksler birbirine yakın sonuçlar üretmekle birlikte önerilen Bloom indeksi daha düşük işlem süresi dağılımına sahiptir. Şekil 6.24 (b)'de aynı deney sonuçlarının indeksleme yöntemleri için ortalama işlem süreleri karşılaştırılmıştır. Bu grafikten de görüldüğü gibi Bloom indeksleme diğer indeksleme yöntemlerinden %50 daha düşük işlem süresi ortaya koymaktadır. Bu sonuç ikinci bir milyon satırlık veri kümesinde yazma işleminde bulunmayan kayıtların da arama sonucunu etkilemesinden kaynaklanmaktadır. Bloom indeksleme kayıt arama işlemini yapmadan önce üyelik araması yaptığından olmayan kayıtlar için ekstra disk okuma süreçleri devre dışı kalmakta, bu da işlem sürelerine performans artışı olarak yansımaktadır.

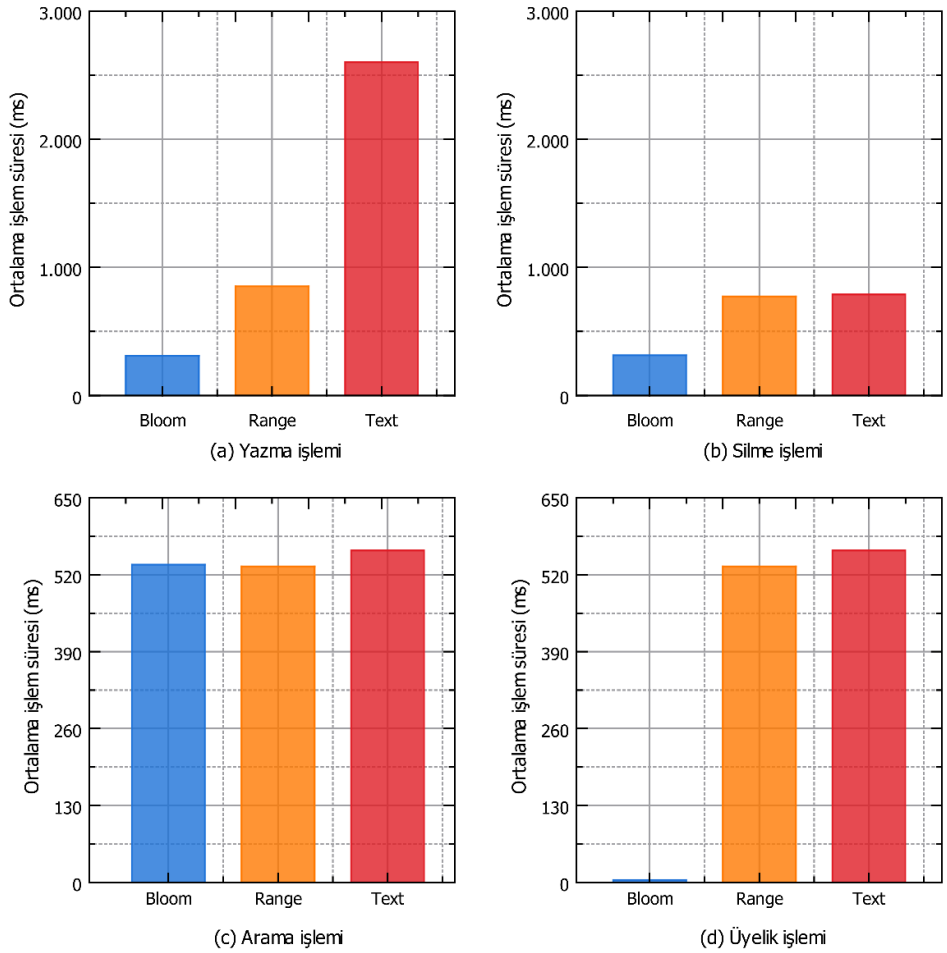


Şekil 6.24. Yazma deneylerinde kullanılmamış kayıtlar için arama işlem sürelerinin dağılımı (a) ve ortalama arama süreleri (b)

Çizelge 6.6. İndeksleme yöntemleri için ortalama işlem süreleri (ms)

İşlem	Bloom indeks	Range indeks	Metin indeks
Yazma	311,23	852,94	2600,79
Silme	314,49	772,96	789,22
Kayıt arama	537,34	533,95	560,90
Üyelik arama	3,36	533,95	560,90

Son olarak ilk 1 milyon satır için yapılmış tüm deneylerde indeksleme yöntemleri için ortalama işlem süreleri karşılaştırılmıştır. Karşılaştırma sonuçları Çizelge 6.6'da gösterilmiştir. Benzer şekilde bu sonuçlar Şekil 6.25'te görselleştirilmiştir. Görüldüğü gibi yazma işlemlerinde Bloom indeksi range indekslemeye göre ≈2,74 kat, metin indekslemeye göre ise ≈8,36 kat iyileştirme sağlamaktadır (Şekil 6.25 (a)). Silme işlemi için bu oran range indekslemeye göre ≈2,46 kat, metin indekslemeye göre ise ≈2,5 kat şeklindedir (Şekil 6.25 (b)). Tüm değerlerin BT'de bulunması deneylerinde Bloom indeksleme yakın sonuçlar üretmekle birlikte (Şekil 6.25 (c)) üyelik arama deneylerinde (Şekil 6.25 (d)) 3,36 ms ortalama işlem süresi ile açık ara en iyi performansı göstermiştir.



Şekil 6.25. İndeksleme yöntemleri için ortalama işlem süreleri karşılaştırması

7. SONUÇLAR VE ÖNERİLER

Bilgi tabanları 30 yıldan fazla süredir bilgisayar bilimlerinde çalışma konusu olarak aktifliğini korumaya devam etmektedir. Özellikle son dönem sayısallaşan veri boyutlarındaki inanılmaz artış, yapay zeka uygulamalarında kaydedilen ilerlemeler ve farklı üst katman uygulamaları için organize bilgiye olan talep bilgi tabanlarının popülerliğini daha da artırmıştır. Verinin sayısallaştırılması ve yapay zeka uygulamalarındaki ivme halen güncelliğini korumaktadır. Bu nedenle önümüzdeki süreçte de belirli alana özgü veya genel bilginin organize bir yapıda temsil edilmesine ihtiyacın artacağı öngörülebilir.

Bu ihtiyaç göz önünde bulundurularak tez çalışmasında bilgi tabanları, bilgi tabanlarının oluşturma, temizleme ve doğrulama yöntemleri ele alınmış, konuyla ilgili yapılmış akademik çalışmalar detaylı olarak incelenmiştir. Yapılan araştırmalarda BT'lerde bulunan bilginin niceliği kadar niteliğinin de önemi ortaya çıkmıştır. Özellikle, kapalı dünya varsayımını temel alan yaklaşımların sistemin kendisi ile sınırlı kalması, sadece belirli kaynaklardan toplanan verinin eksik bilgi çıkarımına neden olması, zamana bağlı bilgilerin sürekli güncellenme sorunu ve elde edilen bilgilerin doğrulanması ihtiyacı BT çalışmalarında karşılaşılan eksiklikler olarak sıralanabilir. Bu problemlere literatürde farklı çözüm önerileri sunulmakla birlikte BT sistemlerinin dinamik yapısının, esnek modelleme ihtiyacının ve ölçeklendirme zorunluluğunun da göz önünde bulundurulması gerekmektedir. BT sistemlerinde öne çıkan eksikler ve açık çalışma konuları ele alınarak tez çalışmasında bu konulara çözüm yolları aranmıştır. Bu anlamda BT oluşturma, doğrulama ve temizleme çalışmalarına yoğunlaşılarak mevcut çalışmalardaki eksik yönler tespit edilmiştir. Burada özellikle dikkat çeken nokta doğrulama ve temizleme çalışmalarının tek seferlik işlem olarak ele alınması ve dinamik BT yapısının göz ardı edilmesi olmuştur.

BT oluşturma çalışmaları doğal olarak veri kaynağına ihtiyaç duymaktadır. Her ne kadar erken dönem BT çalışmalarında insan emeğine dayanan yöntemler tercih edilse de günümüzde verinin ulaştığı boyutlar ve insan emeğinin maliyeti dikkate alındığında bu yöntemlerin verimsiz olduğu açıktır. Bu nedenle otomatik BT oluşturma yöntemleri geliştirilmekle birlikte veri kaynağı olarak da insan bilgisinin en geniş kapsamlı ve dinamik şekilde temsil edildiği Web ortamı tercih edilmektedir. Web ortamı sadece veri

kaynağı olarak da sınırlı kalmamakta, aynı zamanda kaynağın güvenilirliği ve elde edilmiş verinin farklı ve bağımsız kaynaklar üzerinden doğrulanması için de altyapı sunmaktadır. Web kaynağının güvenilirliği BT oluşturma çalışmalarından bağımsız olarak da istenmeyen sayfaların tespiti ve arama motorlarında sıralama işlemleri için önemli yer tutmaktadır. Web üzerinden bilgi çıkarımında kaynak güvenilirliği bilgiye olan güvenin, dolayısıyla bilginin doğru olma olasılığının bir göstergesi olarak kullanılabilir. BT oluşturma ve doğrulama üzerine yapılmış çalışmalarda bilginin güven değerini temel alan ve bu doğrultuda yeni bilgi çıkarımı ve doğrulama yapan yöntemler de bulunmaktadır. Her ne kadar bilgi yerleştirme yöntemlerinin yaygınlaşması ile birlikte güven değeri üzerine yapılan çalışmalar yaygınlık kazansa da bu tür çalışmalar genellikle elde edilen güven değerini mevcut bilgi ile sınırlı tutmaktadır. Bilginin, dolayısıyla bilgiyi ifade eden sistem olarak BT'lerin yapısı göz önünde bulundurulduğunda bilgi parçacıklarının bir biri ile ilişkili olduğu açıktır. Bu anlamda bilgiye ait güven değeri de sadece mevcut bilgi parçacığı ile sınırlı kalmadan tüm BT üzerinde artı veya eksi yönlü etki oluşturabilir.

Yukarıda bahsedilen eksikleri ortadan kaldırmak ve dinamik BT yapısı için ortaya çıkan güncelleme ve ölçeklendirme gereksinimlerini karşılamak amacıyla tez çalışmasında yeni bir BT oluşturma ve güncelleme yaklaşımı önerilmiştir. Bu yaklaşım Web kaynağından elde edilen bilgi üçlüsünü kaynağın güven değeri ile beraber ele almaktadır. Elde edilen üçlü güven değeri ile birlikte çizgeye sunulmakta ve güven değerinin değişim durumları kontrol edilmektedir. Güven değerinin hesaplanma durumları hem mevcut üçlünün güncellenmesi hem de ilgili üçlü ile bağlantılı diğer üçlülerde güven değeri değişimi şeklinde belirlenmektedir. Bu sayede belirli güven eşiğinin altındaki üçlüler BT'den silinerek sistem yanlış bilgiden arındırılmaktadır. Güven değerinin değişimi var olan güven değeri ile yeni elde edilen güven değerine bağlı sigmoid fonksiyonu üzerinden çıktı olarak hesaplanır. İlgili üçlü ile bağlantılı üçlülerin güven değerlerindeki döngüsel güncelleme işlemi BT üzerinde yayılma etkisi olarak tanımlanır. Yayılma etkisi BT arındırma işlemlerinden farklı ve yeni bir yaklaşım ortaya koymaktadır. Bu yaklaşım üçlüler için güncellenen güven değerlerini dikkate alarak güçlü ve zayıf üçlülerin birbirini etkilemesini tetiklemektedir. Sonuç olarak güçlü bilgilere güvenin daha da güçlenmesine, zayıf bilgilerin ise silinerek BT'nin temizlenmesine sebep olmaktadır. Yayılma etkisi kural tabanlı bilgi çıkarım yöntemlerinde tercih edilen birleşme kuralını temel alarak bilgi çizgesindeki üçgen yapıları üzerinde uygulanmaktadır. Tez çalışması kapsamında yayılma işleminin hem çizge veri yapısında hem de FPN ağlarında modellenmesi gerçekleştirilmiş

ve bu modeller için elde edilen sonuçlar karşılaştırılmıştır. Ayrıca yayılma etkisinin etkinliği, çapı ve eşzamanlılığı gibi karakteristik özellikleri de belirlenerek farklı deneysel çalışmalar ortaya koyulmuştur.

Deneysel çalışmalar birçok farklı parametrelerin yayılma işlemi üzerindeki etkisini değerlendirmek üzere tasarlanmıştır. Bu parametreler farklı veri kümeleri, değişken küme boyutları, yanlış üçlülerin dağılım oranları, güven değerlerinin dağılım eğilimleri, yayılma işleminin çapı, güven değerinin etkinliği ve farklı hesaplama fonksiyonları şeklinde sıralanabilir. Deneysel çalışmalar yayılma yönteminin yanlış üçlülerin silinmesinde ortalama %90 doğruluk oranı yakaladığını ortaya koymaktadır. Ayrıca veri kümesinin ve yanlış üçlülerin artması arıtma işleminin başarısını da olumlu etkilemektedir. Elde edilen sonuçlar çizge veri modeli ve FPN modelinden bağımsız olarak sigmoid fonksiyonunun diğer hesaplama fonksiyonlarına göre daha başarılı sonuçlar ürettiğini ortaya koymuştur. Yapılan çalışmada yayılma işleminin 3-atlama ile sınırlanması durumunda çok daha az zamanda aynı başarı oranlarının yakalandığı gösterilmiştir. Bu nedenle önerilen yöntem performans olarak da büyük BT'ler için uygulanabilir ve ölçeklenebilirdir. Gerçek dünya dağılımı ile yapılan karşılaştırmada da yayılma yönteminin işlem sonunda güven değerleri için benzer dağılım karakteristiği gösterdiği doğrulanmıştır.

Yayılma deneyleri ve BT oluşturma çalışmaları sırasında ve bilgi tabanlarının mevcut boyutları dikkate alındığında BT oluşturma, güncelleme ve arama performansında da bazı yetersizliklerin olduğu tespit edilmiştir. Bu sorunlara çözüm geliştirme adına Bloom filtresi ile B+ ağaçlarının birleşiminden oluşan tümleşik indeksleme çözümü sunularak veri depolama sistemlerinin yazma ve silme performansında yaklaşık 2,5 kat, arama performansında ise yaklaşık 2 kat iyileştirme sağlanmıştır.

BT üzerinde yayılma işlemi yeni bir yaklaşım olması nedeniyle çok farklı yeni çalışmalara kapı açabilir. Yayılma karakteristiğini belirleyen kriterler geliştirilerek farklı yayılma türevleri tasarlanabilir. Güven değeri hesaplaması için de farklı sigmoid fonksiyonları ve dikey katsayı parametreleri ile sonuçlar değerlendirilebilir. Bir diğer gelecek çalışma konusu olarak silme eşik değerinin tanımlanması için sınıflandırma çalışması öngörülebilir. Tez kapsamında önerilen BT modeli açık dünya varsayımına dayandığı ve Web kaynaklarından güven değerleri ile desteklendiği için oluşan BT'nin kaynak güvenilirliği anlamında çift taraflı etki oluşturması beklenir. Dolayısıyla elde edilen

bilginin doğruluk katsayısına bağlı olarak kaynağa olan güvenin de güncellenmesi farklı bir gelecek çalışma konusu olarak ele alınabilir. Ayrıca kaynakların elde edilen bilginin kategorisine göre yetkilendirilmesi ve bir kaynak üzerinden her bilginin aynı güven değeri ile elde edilmemesi sağlanabilir. Bu sayede yetkili kaynaklar sadece uzmanlık alanı dahilinde BT yapısının güncellenmesinde etkin role sahip olabilir.

Önerilen BT sisteminin dinamik ve doğrusal olmayan yapısı göz önünde bulundurulduğunda bu sistemin istikrarlılığı, gözlemlenebilirliği ve kontrol edilebilirliği de önem kazanmaktadır. Bu anlamda sistemin dış kaynakların manipülasyonuna karşı dayanıklı hale getirilmesi, ayrıca güven değerlerine bağlı denge ve kararlılık durumlarının belirlenmesi de gelecek çalışma konuları arasında sıralanabilir.

KAYNAKLAR

1. İnternet: SINTEF. (2013). Big Data, for better or worse: 90% of world's data generated over last two years. *ScienceDaily*. Web: https://www.sciencedaily.com/releases/2013/05/130522085217.htm, Son Erişim Tarihi: 14.08.2023.

2. Weikum, G., Hoffart, J., and Suchanek, F. (2016). Ten years of knowledge harvesting: Lessons and challenges. *Data Engineering*, *5*, 41–50.

3. Bonatti, P. A., Decker, S., Polleres, A., and Presutti, V. (2019). Knowledge Graphs: New Directions for Knowledge Representation on the Semantic Web (Dagstuhl Seminar 18371). *Report from Dagstuhl Seminar*, *8*(9), 29–111.

4. Ehrlinger, L., and Wöß, W. (2016). Towards a definition of knowledge graphs. *CEUR Workshop Proceedings*, *1695*, 1–5.

5. Li, X., Taheri, A., Tu, L. L. L., and Gimpel, K. (2016). *Commonsense knowledge base completion.* In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, (3), 1445–1455.

6. Zang, Liang-Jun and Cao, Cong and Cao, Ya-Nan and Wu, Yu-Ming and Cao, C.-G., Zang, L. J., Cao, C. G. C., Cao, Y. N., Wu, Y. M., and Cao, C. G. C. (2013). A survey of commonsense knowledge acquisition. *Journal of Computer Science and Technology*, *28*(4), 689–719.

7. İnternet: Singhal, A. (2012). Official Google Blog: Introducing the Knowledge Graph: things, not strings. *Official Google Blog*. Web: https://blog.google/products/search/introducing-knowledge-graph-things-not/ Son Erişim Tarihi: 14.08.2023.

8. Hao, J., Chen, M., Yu, W., Sun, Y., and Wang, W. (2019). *Universal representation learning of knowledge bases by jointly embedding instances and ontological concepts.* Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1709–1719.

9. Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., and Ives, Z. (2007). *Dbpedia: A nucleus for a web of open data*. In international semantic web conference 722-735.

10. Mitchell, T., Cohen, W., Hruschka, E., Talukdar, P., Yang, B. B., Betteridge, J., (2018). Never-ending learning. *Communications of the ACM*, *61*(5), 103–115.

11. Niu, F., Zhang, C., Ré, C., and Shavlik, J. (2012). Elementary: Large-scale knowledge-base construction via machine learning and statistical inference. *International Journal on Semantic Web and Information Systems (IJSWIS)*, *8*(3), 42–73.

12. İnternet: Google Freebase. (2012). Freebase Documentation. *Freebase Documentation*. Web: https://developers.google.com/freebase. Son Erişim Tarihi: 14.08.2023.

13. Speer, R., Chin, J., and Havasi, C. (2017). *Conceptnet 5.5: An open multilingual graph of general knowledge*. In Proceedings of the AAAI conference on artificial intelligence, 31(1).

14. Dong, X., Gabrilovich, E., Heitz, G., Horn, W., Lao, N., Murphy, K., and Zhang, W. (2014). *Knowledge vault: A web-scale approach to probabilistic knowledge fusion.* In Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining, 601–610.

15. Yu, D. (2017). *Unsupervised Graph-Based Relation Extraction and Validation for Knowledge Base Population*, Doktora Tezi, Rensselaer Polytechnic Institute, New York, 32-55.

16. İnternet: Meta. (2023). List of Wikipedias-Discussion about Wikimedia projects. Web: https://meta.wikimedia.org/wiki/List_of_Wikipedias, Son Erişim Tarihi: 14.08.2023.

17. İnternet: Eberhard, David M., Gary F. Simons, and C. D. F. (2019). Ethnologue: Languages of the World. *Dallas, Texas: SIL International.* Web: https://www.ethnologue.com/guides/ethnologue200 Son Erişim Tarihi: 14.08.2023.

18. Popat, K., Mukherjee, S., Strötgen, J., and Weikum, G. (2019). Where the truth lies: Explaining the credibility of emerging claims on the web and social media. *Proceedings of the 26th International Conference on World Wide Web Companion*, 1003–1012.

19. Thorne, J., & Vlachos, A. (2018). *Automated Fact Checking: Task Formulations, Methods and Future Directions*. In Proceedings of the 27th International Conference on Computational Linguistics, 3346-3359.

20. Shiralkar, P. (2017). *Computational Fact Checking by Mining Knowledge Graphs.* Doktora Tezi, Indiana University, Bloomington, 48-99.

21. Padia, A., and Others. (2017). *Cleaning noisy knowledge graphs*. In Proceedings of the Doctoral Consortium at the 16th International Semantic Web Conference (1962).

22. Labra-Gayo, J. E., García-González, H., Fernández-Alvarez, D., and Prud'hommeaux, E. (2019). Challenges in RDF Validation. In *Current Trends in Semantic Web Technologies: Theory and Practice*, (815), 121–151.

23. Borgwardt, S., Ceylan, I. I., and Lukasiewicz, T. (2018). *Recent advances in querying probabilistic knowledge bases*. In IJCAI International Joint Conference on Artificial Intelligence, 5420–5426.

24. Suchanek, F. M., and Weikum, G. (2014). Knowledge bases in the age of big data analytics. *PVLDB*, *7*(13), 1713–1714.

25. Tandon, N., De Melo, G., Suchanek, F., and Weikum, G. (2014). *Webchild: Harvesting and organizing commonsense knowledge from the web*. In Proceedings of the 7th ACM international conference on Web search and data mining, 523–532.

26. Chen, J., Tandon, N., Hariman, C. D., & de Melo, G. (2016). *WebBrain: Joint Neural Learning of Large-Scale Commonsense Knowledge*. In 15th International Semantic Web Conference, 102-118.

27. Paulheim, H., Heiko Paulheim, and Paulheim, H. (2017). Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic Web*, *8*(3), 489–508.

28. Moniruzzaman, A. B. M. M. M., Tang, M., Nayak, R., Balasubramaniam, T., Tang, M., and Balasubramaniam, T. (2019). *Fine-grained type inference in knowledge graphs via probabilistic and tensor factorization methods.* In The Web Conference 2019 - Proceedings of the World Wide Web Conference, *WWW 2019*, 3093–3100.

29. Vlachos, A., and Riedel, S. (2015). *Identification and verification of simple claims about statistical properties.* In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2596–2601.

30. Hanselowski, A., and Gurevych, I. (2017). *A Framework for Automated Fact-Checking for Real-Time Validation of Emerging Claims on the Web.* In NIPS Workshop on Prioritising Online Content (WPOC2017), 80–82.

31. Wang, W. Y. (2017). *"Liar, liar pants on fire": A new benchmark dataset for fake news detection.* In ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference.

32. Hogan, A., Blomqvist, E., Cochez, M., D'Amato, C., Melo, G. De, Gutierrez, C., Zimmermann, A. (2021). Knowledge graphs. *ACM Computing Surveys*. 54(4), 1-37.

33. Wang, Q., Mao, Z., Wang, B., and Guo, L. (2017). Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering*, 29(12), 2724-2743.

34. Gerber, D., Esteves, D., Lehmann, J., Bühmann, L., Usbeck, R., Ngomo, A. C. N., & Speck, R. (2015). Defacto—temporal and multilingual deep fact validation. *Journal of Web Semantics*, 35, 85-101.

35. Distiawan, B., Weikum, G., Qi, J., Zhang, R., Trisedya, B. D., Weikum, G., Zhang, R. (2019). *Neural Relation Extraction for Knowledge Base Enrichment.* In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 229–240.

36. Fatemi, B., Taslakian, P., Vazquez, D., and Poole, D. (2021). *Knowledge hypergraphs: prediction beyond binary relations.* In Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence, 2191-2197.

37. Kristiadi, A., Khan, M. A., Lukovnikov, D., Lehmann, J., and Fischer, A. (2019). *Incorporating Literals into Knowledge Graph Embeddings*. In International Semantic Web Conference, 347–363.

38. Sap, M., Le Bras, R., Allaway, E., Bhagavatula, C., Lourie, N., Rashkin, H., Choi, Y. (2019). *Atomic: An atlas of machine commonsense for if-then reasoning.* In Proceedings of the AAAI Conference on Artificial Intelligence, 33, 3027–3035.

39. Bosselut, A., Rashkin, H., Sap, M., Malaviya, C., Celikyilmaz, A., & Choi, Y. (2019). *COMET: Commonsense Transformers for Automatic Knowledge Graph Construction*. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics.

40. Bergman, M. K., Bergman, M. K., and Lagerstrom-Fife. (2018). *Knowledge Representation Practionary*. Springer International Publishing, 45-167.

41. Brachman, R. J., and Levesque, H. J. (2004). *Knowledge Representation and Reasoning*, Morgan Kaufmann Publishers, San Francisco, 15-182.

42. İnternet: Scharei, K., Heidecker, F., and Bieshaar, M. (2020). Knowledge Representations in Technical Systems--A Taxonomy, Web: https://arxiv.org/abs/2001.04835, Son Erişim Tarihi: 14.08.2023.

43. Van Harmelen, F., Lifschitz, V., and Porter, B. (2008). *Handbook of Knowledge Representation*. Elsevier Science, San Diego, 135-203.

44. Corea, F. (2019). *An Introduction to Data Everything You Need to Know About AI, Big Data and Data Science*, Springer International Publishing, Venice, 25-29.

45. Chein, M., and Mugnier, M.-L. (2008). *Graph-based knowledge representation: computational foundations of conceptual graphs*. Springer Science and Business Media, London, 21-80.

46. Davis, R., Shrobe, H., and Szolovits, P. (1993). What is a knowledge representation? *AI Magazine*, *14*(1), 17.

47. Croitoru, M. (2014). *Graph Based Knowledge Representation and Reasoning: Practical AI Applications*, Doktora Tezi, Université Montpellier 2, Montpellier, 13-21.

48. Mylopoulos, J. (1980). An overview of knowledge representation. *ACM SIGART Bulletin*, (74), 5-12.

49. Yao, L. (2015). *Universal Schema for Knowledge Representation from Text and Structured Data*, Doktora Tezi, University of Massachusetts Amherst, Amherst, 12-30.

50. Pavlić, M., Meštrović, A., Jakupović, A., Panlic, M., Niesirovic, N., and Jakupovi, M. (2013). *Graph-based formalisms for knowledge representation.* In Proceedings of the 17th world multi-conference on systemics cybernetics and informatics (WMSCI 2013), (2), 200–204.

51. Nematzadeh, A., Fazly, A., and Stevenson, S. (2014). *A cognitive model of semantic network learning*. EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, 244–254.

52. Helbig, H., and Gnörlich, C. (2002). *Multilayered extended semantic networks as a language for meaning representation in NLP systems.* In International Conference on Intelligent Text Processing and Computational Linguistics, Berlin, 69-85.

53. Cruz-Cunha, M. M., Oliveira, E. F., Tavares, A. J., & Ferreira, L. G. (2009). *Handbook of research on social dimensions of semantic technologies and web services*. IGI Global, New York, 1-68.

54. Ji, S., Pan, S., Cambria, E., Marttinen, P., and Yu, P. S. (2020). A Survey on Knowledge Graphs: Representation, Acquisition, and Applications. *IEEE Transactions on Neural Networks and Learning Systems*, 1–25.

55. Wu, T., Qi, G., Li, C., and Wang, M. (2018). A survey of techniques for constructing Chinese knowledge graphs and their applications. *Sustainability (Switzerland)*, *10*(9), 1–26.

56. Wang, Q., Mao, Z., Wang, B., and Guo, L. (2017). Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering*, *29*(12), 2724–2743.

57. Gesese, G. A., Biswas, R., and Sack, H. (2019). *A Comprehensive Survey of Knowledge Graph Embeddings with Literals: Techniques and Applications.* In Proceedings of DL4KG2019-Workshop on Deep Learning for Knowledge Graphs, 31-40.

58. Kolda, T. G., and Bader, B. W. (2009). Tensor decompositions and applications. *SIAM Review*, 51(3), 455-500.

59. Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). *Efficient estimation of word representations in vector space*. In 1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings.

60. Pennington, J., Socher, R., and Manning, C. D. (2014). *GloVe: Global vectors for word representation*. In EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference, 1532-1543.

61. Peters, M. E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., and Zettlemoyer, L. (2018). *Deep contextualized word representations*. In NAACL HLT 2018 - 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference.

62. Dai, Y., Wang, S., Xiong, N. N., and Guo, W. (2020). A Survey on Knowledge Graph Embedding: Approaches, Applications and Benchmarks. *Electronics*, *9*(5), 750.

63. Rosso, P., Yang, D., and Cudré-Mauroux, P. (2019). *Knowledge Graph Embeddings*, In Encyclopedia of Big Data Technologies, 1-7.

64. Wu, T., Qi, G., Li, C., Wang, M., Scharei, K., Heidecker, F., … Yu, P. S. (2020). Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering*, *29*(12), 2724–2743.

65. Bordes, A., Usunier, N., Garcia-Durán, A., Weston, J., and Yakhnenko, O. (2013). *Translating embeddings for modeling multi-relational data*. In Proceedings of the 26th International Conference on Neural Information Processing Systems, (2), 2787-2795.

66. Wang, Z., Zhang, J., Feng, J., and Chen, Z. (2014). *Knowledge graph embedding by translating on hyperplanes*. In Proceedings of the National Conference on Artificial Intelligence, (2), 1112–1119.

67. Lin, Y., Liu, Z., Sun, M., Liu, Y., and Zhu, X. (2015). *Learning entity and relation embeddings for knowledge graph completion.* In Proceedings of the National Conference on Artificial Intelligence, (3), 2181–2187.

68. İnternet: Xiao, H., Huang, M., Hao, Y., and Zhu, X. (2015). TransA: An adaptive approach for knowledge graph embedding, Web: https://arxiv.org/pdf/1509.05490.pdf, Son Erişim Tarihi: 14.08.2023.

69. Ji, G., Liu, K., He, S., and Zhao, J. (2016, February). *Knowledge graph completion with adaptive sparse transfer matrix*. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, 1(30), 985-991.

70. Nguyen, D. Q., Sirts, K., Qu, L., and Johnson, M. (2016). *STransE: a novel embedding model of entities and relationships in knowledge bases*. In Proceedings of NAACL-HLT, 460-466.

71. Ji, G., He, S., Xu, L., Liu, K., and Zhao, J. (2015). *Knowledge graph embedding via dynamic mapping matrix*. In Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing, (1), 687-696.

72. He, S., Liu, K., Ji, G., and Zhao, J. (2015). *Learning to represent knowledge graphs with Gaussian embedding.* In International Conference on Information and Knowledge Management, Proceedings, 623-632.

73. Xiao, H., Huang, M., and Zhu, X. (2016). *TransG: A generative model for knowledge graph embedding.* In 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016 - Long Papers, 2316-2325.

74. Nickel, M., Tresp, V., and Kriegel, H. P. (2011). *A three-way model for collective learning on multi-relational data*. In Proceedings of the 28th International Conference on Machine Learning, ICML 2011, 11(10.5555), 3104482-3104584.

75. Yang, B., tau Yih, W., He, X., Gao, J., and Deng, L. (2015). *Embedding entities and relations for learning and inference in knowledge bases.* In 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings.

76. Trouillon, T., Welbl, J., Riedel, S., Ciaussier, E., and Bouchard, G. (2016). *Complex embeddings for simple link prediction*. In 33rd International Conference on Machine Learning, ICML 2016, 2071-2080.

77. Yao, J., and Zhao, Y. (2019). *Knowledge Graph Embedding Bi-vector Models for Symmetric Relation*. In Chinese Intelligent Systems Conference, 27–36.

78. İnternet: Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., and Yakhnenko, O. (2013). Irreflexive and hierarchical relations as translations. Web: https://arxiv.org/pdf/1304.7158.pdf, Son Erişim Tarihi: 14.08.2023.

79. Kiers, H. A. (2000). Towards a standardized notation and terminology in multiway analysis. Journal of Chemometrics: *A Journal of the Chemometrics Society*, 14(3), 105-122.

80. Kazemi, S. M., and Poole, D. (2018). *Simple embedding for link prediction in knowledge graphs*. In Advances in Neural Information Processing Systems, (31).

81. Sun, Z., Deng, Z. H., Nie, J. Y., and Tang, J. (2018). *RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space*. In International Conference on Learning Representations.

82. Zhang, S., Tay, Y., Yao, L., and Liu, Q. (2019). *Quaternion knowledge graph embeddings.* In Advances in Neural Information Processing Systems, 2735–2745.

83. Bordes, A., Glorot, X., Weston, J., and Bengio, Y. (2014). A semantic matching energy function for learning with multi-relational data: Application to word-sense disambiguation. *Machine Learning*, 94, 233-259.

84. Socher, R., Chen, D., Manning, C. D., and Ng, A. Y. (2013). *Reasoning with neural tensor networks for knowledge base completion*. In Advances in Neural Information Processing Systems, (26).

85. Nguyen, D. Q., Nguyen, T. D., Nguyen, D. Q., and Phung, D. (2018). *A novel embedding model for knowledge base completion based on convolutional neural network*. In NAACL HLT 2018 - 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference, 327-333.

86. Schlichtkrull, M., Kipf, T. N., Bloem, P., van den Berg, R., Titov, I., and Welling, M. (2018). *Modeling Relational Data with Graph Convolutional Networks.* In Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 593-607.

87. Cai, L., and Wang, W. Y. (2018). *KBGAN: Adversarial learning for knowledge graph embeddings*. In NAACL HLT 2018 - 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference, 1470-1480.

88. Lin, Y., Liu, Z., Luan, H., Sun, M., Rao, S., and Liu, S. (2015). *Modeling relation paths for representation learning of knowledge bases.* In Conference Proceedings - EMNLP 2015: Conference on Empirical Methods in Natural Language Processing.

89. Leblay, J., and Chekol, M. W. (2018). *Deriving Validity Time in Knowledge Graph*. In The Web Conference 2018 - Companion of the World Wide Web Conference, WWW 2018, 1771-1776.

90. Jiang, T., Liu, T., Ge, T., Sha, L., Chang, B., Li, S., and Sui, Z. (2016). *Towards time-aware knowledge graph completion*. In COLING 2016 - 26th International Conference on Computational Linguistics, Proceedings of COLING 2016: Technical Papers, 1715-1724.

91. Jiang, T., Liu, T., Ge, T., Sha, L., Li, S., Chang, B., and Sui, Z. (2016). *Encoding temporal information for time-aware link prediction*. In EMNLP 2016 - Conference on Empirical Methods in Natural Language Processing, Proceedings, 2350-2354.

92. Ghor, T. A., Agrawal, E., Alam, M., Alqawasmeh, O., D'amato, C., Annane, A., Zocholl, M. (2019). Linked Open Data Validity, *ISWS 2018*, Bertinoro, 45-62.

93. Ciampaglia, G. L., Shiralkar, P., Rocha, L. M., Bollen, J., Menczer, F., and Flammini, A. (2015). Computational fact checking from knowledge networks. *PLOS One*, 10(6), 1-13.

94. Du, J., Pan, J. Z., Wang, S., Qi, K., Shen, Y., and Deng, Y. (2019). *Validation of growing knowledge graphs by abductive text evidences*. In Proceedings of the AAAI Conference on Artificial Intelligence, (33), 2784–2791.

95. Lin, P., Song, Q., and Wu, Y. (2018). Fact checking in knowledge graphs with ontological subgraph patterns. *Data Science and Engineering*, 3(4), 341–358.

96. Shi, B., and Weninger, T. (2016). Discriminative predicate path mining for fact checking in knowledge graphs. *Knowledge-based systems*, 104, 123–133.

97. Shi, B., and Weninger, T. (2016). *Fact checking in heterogeneous information networks.* In Proceedings of the 25th International Conference Companion on World Wide Web, 101–102.

98. Lin, P., Song, Q., Wu, Y., and Pi, J. (2019). Discovering patterns for fact checking in knowledge graphs. *Journal of Data and Information Quality*, 11(3).

99. Huynh, V.-P., and Papotti, P. (2018). *Towards a benchmark for fact checking with knowledge bases*. In Companion Proceedings of the The Web Conference 2018, 1595–1598.

100. Gad-Elrab, M. H., Stepanova, D., Urbani, J., Weikum, G., Stepanova, D., and Weikum, G. (2019). *Tracy: Tracing facts over knowledge graphs and text*. In The World Wide Web Conference, 3516–3520.

101. Dagan, I., Dolan, B., Magnini, B., and Roth, D. (2010). Recognizing textual entailment: Rational, evaluation and approaches--erratum. *Natural Language Engineering*, 16(1), 105.

102. Trouillon, T., Dance, C. R., Gaussier, E., Welbl, J., Riedel, S., and Bouchard, G. (2017). Knowledge Graph Completion via Complex Tensor Factorization. *Journal of Machine Learning Research*, 18(130), 1-38.

103. Wang, Y., Zhao, E., and Wang, W. (2022). A knowledge graph completion method based on fusing association information. *IEEE Access*, 10, 50500-50507.

104. Alam, M. M., Rony, M. R. A. H., Nayyeri, M., Mohiuddin, K., Akter, M. M., Vahdati, S., and Lehmann, J. (2022). Language model guided knowledge graph embeddings. *IEEE Access*, 10, 76008-76020.

105. Melo, A., and Paulheim, H. (2017). *Detection of relation assertion errors in knowledge graphs.* In Proceedings of the Knowledge Capture Conference, K-CAP 2017, 1-8.

106. Chen, X., Chen, M., Shi, W., Sun, Y., and Zaniolo, C. (2019). *Embedding uncertain knowledge graphs*. In Proceedings of the AAAI conference on artificial intelligence, 33(01), 3363-3370.

107. Shan, Y., Bu, C., Liu, X., Ji, S., and Li, L. (2018). *Confidence-aware negative sampling method for noisy knowledge graph embedding*. In 2018 IEEE International Conference on Big Knowledge (ICBK), 33-40.

108. Shao, T., Li, X., Zhao, X., Xu, H., and Xiao, W. (2021). DSKRL: A dissimilarity-support-aware knowledge representation learning framework on noisy knowledge graph. *Neurocomputing*, 461, 608-617.

109. Liu, S., Grau, B., Horrocks, I., and Kostylev, E. (2021). Indigo: Gnn-based inductive knowledge graph completion using pair-wise encoding. *Advances in Neural Information Processing Systems*, 34, 2034-2045.

110. Abedini, F., Keyvanpour, M. R., and Menhaj, M. B. (2020). Correction Tower: A general embedding method of the error recognition for the knowledge graph correction. *International Journal of Pattern Recognition and Artificial Intelligence*, 34(10), 2059034.

111. Guo, S., Wang, Q., Wang, L., Wang, B., and Guo, L. (2016). *Jointly embedding knowledge graphs and logical rules*. In EMNLP 2016 - Conference on Empirical Methods in Natural Language Processing, Proceedings, 192-202.

112. Jeyaraj, M. N., Perera, S., Jayasinghe, M., and Jihan, N. (2022). Probabilistic error detection model for knowledge graph refinement. *Computación y Sistemas*, 26(3), 1243-1257.

113. Qu, M., and Tang, J. (2019). Probabilistic logic neural networks for reasoning. In *Advances in Neural Information Processing Systems*, 32.

114. Richardson, M., and Domingos, P. (2006). Markov logic networks. *Machine learning*, 62, 107-136.

115. Murphy, K. P., Weiss, Y., and Jordan, M. I. (1999). *Loopy belief propagation for approximate inference: an empirical study.* In Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence, 467-475.

116. Ma, J., Zhou, C., Wang, Y., Guo, Y., Hu, G., Qiao, Y., and Wang, Y. (2022). PTrustE: A high-accuracy knowledge graph noise detection method based on path trustworthiness and triple embedding. *Knowledge-Based Systems*, 256, 109688.

117. Drumond, L., Rendle, S., and Schmidt-Thieme, L. (2012). *Predicting RDF triples in incomplete knowledge bases with tensor factorization*. In Proceedings of the ACM Symposium on Applied Computing, 326-331.

118. Syed, Z. H., Röder, M., Ngonga Ngomo, A.-C., R der, M., Ngomo, A. C. N., Röder, M., Ngonga Ngomo, A.-C. (2018). *FactCheck: Validating RDF triples using textual evidence*. In Proceedings of the 27th ACM International Conference on Information and Knowledge Management, 1599–1602.

119. Levine, B. (2016). Data Extractor Diffbot Wants To Turn The Web Into The Semantic Web. Web: https://martech.org/data-extractor-diffbot-wants-to-turn-the-web-into-the-semantic-web/ Son Erişim Tarihi: 14.08.2023.

120. Tang, J., Hong, M., Zhang, D. L., Li, J., Liang, B., Li, J., Li, J. (2008). *Information extraction: Methodologies and applications*. In Emerging Technologies of Text Mining: Techniques and Applications, 1–33.

121. Nguyen, D. Q., and Verspoor, K. (2019). *End-to-end neural relation extraction using deep biaffine attention*. In Advances in Information Retrieval: 41st European Conference on IR Research, ECIR 2019, Cologne, 1(41), 729-738.

122. İnternet: Eryiğit, G. (2019). Uygulamalı Türkçe Doğal Dil İşleme Evreleri. Web: http://byoyo.cmpe.boun.edu.tr/sunumlar/gulseneryigit-byoyo18.pdf, Son Erişim Tarihi: 14.08.2023.

123. Yates, A., Cafarella, M., Banko, M., Etzioni, O., Broadhead, M., and Soderland, S. (2007). *Textrunner: open information extraction on the web*. In Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations, 25–26.

124. Fader, A., Soderland, S., and Etzioni, O. (2011). *Identifying relations for open information extraction.* In Proceedings of the conference on empirical methods in natural language processing, 1535–1545.

125. İnternet: Moreau, L., and Groth, P. (2013). PROV-Overview: An Overview of the PROV Family of Documents. *W3C Note*, Web: http://www.w3.org/TR/2013/NOTE-prov-overview-20130430/ Son Erişim Tarihi: 14.08.2023.

126. Page, L., Brin, S., Motwani, R., and Winograd, T. (1998). The PageRank Citation Ranking: Bringing Order to the Web. *World Wide Web Internet And Web Information Systems*.

127. Gyöngyi, Z., Garcia-Molina, H., and Pedersen, J. (2004). *Combating web spam with trustrank*. In Proceedings of the Thirtieth international conference on Very large data bases, (30), 576–587.

128. Kleinberg, J. M. (1999). Authoritative sources in a hyperlinked environment. *Journal of the ACM (JACM)*, 46(5), 604-632.

129. Zhou, X., and Zafarani, R. (2018). Fake News: a survey of research, Detection Methods, and Opportunities. *ACM Computing Surveys*.

130. Ntoulas, A., Najork, M., Manasse, M., and Fetterly, D. (2006). *Detecting spam web pages through content analysis*. In Proceedings of the 15th International Conference on World Wide Web, 83-92.

131. Nakamura, S., Konishi, S., Jatowt, A., Ohshima, H., Kondo, H., Tezuka, T., and Tanaka, K. (2007). *Trustworthiness analysis of web search results.* In Research and Advanced Technology for Digital Libraries: 11th European Conference, ECDL 2007, Budapest, (11), 38-49.

132. Wawer, A., Nielek, R., and Wierzbicki, A. (2014). *Predicting webpage credibility using linguistic features*. In WWW 2014 Companion - Proceedings of the 23rd International Conference on World Wide Web, 1135-1140.

133. Alberts, W. A., and Van Der Geest, T. M. (2011). Color matters: Color as trustworthiness cue in web sites. *Technical communication*, 58(2), 149-160.

134. Lindgaard, G., Dudek, C., Sen, D., Sumegi, L., and Noonan, P. (2011). An exploration of relations between visual appeal, trustworthiness and perceived usability of homepages. *ACM Transactions on Computer-Human Interaction (TOCHI)*, 18(1), 1-30.

135. Jessen, J., and Jørgensen, A. H. (2012). Aggregated trustworthiness: Redefining online credibility through social validation. *First Monday*, (17), 1-2.

136. Dou, Z., Song, R., Yuan, X., and Wen, J. R. (2008). *Are click-through data adequate for learning web search rankings?* In International Conference on Information and Knowledge Management, Proceedings, 73-82.

137. Liu, X., Nielek, R., Adamska, P., Wierzbicki, A., and Aberer, K. (2015). Towards a highly effective and robust web credibility evaluation system. *Decision Support Systems*, 79, 99-108.

138. Esteves, D., Reddy, A. J., Chawla, P., and Lehmann, J. (2018). *Belittling the Source: Trustworthiness Indicators to Obfuscate Fake News on the Web*. EMNLP 2018, 50.

139. Dong, X. L., Gabrilovich, E., Murphy, K., Dang, V., Horn, W., Lugaresi, C., and Zhang, W. (2015). *Knowledge-Based Trust: Estimating the Trustworthiness of Web Sources*. Proceedings of the VLDB Endowment, 8(9).

140. Erkan, G., and Radev, D. R. (2004). Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research*, 22, 457-479.

141. Steinberger, J., and Jezek, K. (2004). *Using latent semantic analysis in text summarization and summary evaluation*. Proc. ISIM, 4(8), 93-100.

142. Si, L., and Callan, J. (2001). *A statistical model for scientific readability.* In Proceedings of the tenth international conference on Information and knowledge management, 574-576.

143. Zhang, Y., Jin, R., and Zhou, Z. H. (2010). Understanding bag-of-words model: a statistical framework. *International journal of machine learning and cybernetics*, 1, 43-52.

144. Ansari, S., and Gadge, J. (2012). Architecture for checking trustworthiness of websites. *International Journal of Computer Applications*, 44(14), 22-26.

145. Haas, A., and Unkel, J. (2017). Ranking versus reputation: perception and effects of search result credibility. *Behaviour & Information Technology*, 36(12), 1285-1298.

146. İnternet: Kim, J. S., & Choi, K. S. (2021). Fact Checking in Knowledge Graphs by Logical Consistency, Web: https://www.semantic-web-journal.net/system/files/swj2721.pdf. Son Erişim Tarihi: 14.08.2023.

147. Del Vicario, M., Bessi, A., Zollo, F., Petroni, F., Scala, A., Caldarelli, G., and Quattrociocchi, W. (2016). *The spreading of misinformation online.* Proceedings of the national academy of Sciences, 113(3), 554-559.

148. Li, Y., Gao, J., Meng, C., Li, Q., Su, L., Zhao, B., and Han, J. (2016). A survey on truth discovery. *ACM Sigkdd Explorations Newsletter*, 17(2), 1-16.

149. Sais, F. (2019). Knowledge Graph Refinement: Link Detection, Link Invalidation, Key Discovery and Data Enrichment. Doktora Tezi, Université Paris Sud, Orsay Cedex, 13-71.

150. Huaman, E., Tauqeer, A., and Fensel, A. (2021). *Towards knowledge graphs validation through weighted knowledge sources*. In Iberoamerican Knowledge Graphs and Semantic Web Conference, 47-60.

151. Trummer, I. (2021). WebChecker: Towards an Infrastructure for Efficient Misinformation Detection at Web Scale. *IEEE Data Eng. Bull.*, 44(3), 66-77.

152. de Souza, J. V., Assis, E. C., Mendonça, F. M., and de Souza, J. F. (2021). *ReVera Framework: A Framework for Fact Checking Traceability*. In Proceedings of the Brazilian Symposium on Multimedia and the Web, 137-140.

153. Gerber, D., Esteves, D., Lehmann, J., Bühmann, L., Usbeck, R., Ngonga Ngomo, A. C., and Speck, R. (2015). DeFacto - Temporal and multilingual deep fact validation. *Journal of Web Semantics*.

154. Huseynli, A., and Akcayol, M. A. (2023). Continuous Knowledge Graph Refinement with Confidence Propagation. *IEEE Access*, (11), 59226-59237.

155. Petri, C. A., and Reisig, W. (2008). Petri net. *Scholarpedia*, 3(4), 6477.

156. Yakrangi, O., Pazmiño, R. J. S., Cely, J. S., Rodríguez, A., Cena, C. E. G., Carrillo, P. S. S., Shapiro, A. (2021). An intelligent algorithm for decision making system and control of the gemma guide paradigm using the fuzzy petri nets approach. *Electronics*, 10(4), 489.

157. İnternet: Virtanen, H. (2007). A Study in Fuzzy Petri Nets and the Relationship to Fuzzy Logic Programming, Web: https://www.semanticscholar.org/paper/A-Study-in-Fuzzy-Petri-Nets-and-the-Relationship-to-Virtanen/b524a233b5d2ee4ae487b4aa84c81e6503eb0ccb, Son Erişim Tarihi: 14.08.2023.

158. Cardoso, J., Valette, R., and Dubois, D. (1996). Fuzzy Petri nets: an overview. *IFAC Proceedings Volumes*, 29(1), 4866-4871.

159. Liu, H. C., You, J. X., Li, Z., and Tian, G. (2017). Fuzzy Petri nets for knowledge representation and reasoning: A literature review. *Engineering Applications of Artificial Intelligence*, 60, 45-56.

160. Chiachío, M., Chiachío, J., Prescott, D., and Andrews, J. (2018). A new paradigm for uncertain knowledge representation by Plausible Petri nets. *Information Sciences*, 453, 323-345.

161. Li, X., and Lara-Rosano, F. (2000). Adaptive fuzzy Petri nets for dynamic knowledge representation and inference. *Expert Systems with Applications*, 19(3), 235-241.

162. Liu, H. C., Lin, Q. L., Mao, L. X., and Zhang, Z. Y. (2013). Dynamic adaptive fuzzy Petri nets for knowledge representation and reasoning. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 43(6), 1399-1410.

163. Liu, H. C., Xu, D. H., Duan, C. Y., and Xiong, Y. (2019). Pythagorean fuzzy Petri nets for knowledge representation and reasoning in large group context. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 51(8), 5261-5271.

164. Jensen, K. (1997). *A brief introduction to coloured petri nets*. In International Workshop on Tools and Algorithms for the Construction and Analysis of Systems, Berlin, 203-208.

165. Jensen, K., Kristensen, L. M., and Wells, L. (2007). *Coloured Petri Nets and CPN Tools for modelling and validation of concurrent systems*. International Journal on Software Tools for Technology Transfer, 9, 213-254.

166. Chen, S. M., Ke, J. S., and Chang, J. F. (1990). Knowledge Representation Using Fuzzy Petri Nets. *IEEE Transactions on Knowledge and Data Engineering*, 2(3), 311-319.

167. Manoj, T. V., Leena, J., and Soney, R. B. (1998). Knowledge representation using fuzzy Petri nets-revisited. *IEEE transactions on knowledge and data engineering*, 10(4), 666-667.

168. Mamdani, E. H. (1974). *Application of fuzzy algorithms for control of simple dynamic plant*. In Proceedings of the institution of electrical engineers, 121(12), 1585-1588.

169. Akcayol, M. A. (2004). Application of adaptive neuro-fuzzy controller for SRM. *Advances in Engineering Software*, 35(3-4), 129-137.

170. Zou, L., Mo, J., Chen, L., Özsu, M. T., and Zhao, D. (2011). *gStore: answering SPARQL queries via subgraph matching*. Proceedings of the VLDB Endowment, 4(8), 482-493.

171. Lyu, X., Wang, X., Li, Y. F., Feng, Z., and Wang, J. (2015). *GraSS: An efficient method for RDF subgraph matching*. In Web Information Systems Engineering–WISE 2015: 16th International Conference, Miami, 108-122.

172. Katib, A., Slavov, V., and Rao, P. (2016). RIQ: Fast processing of SPARQL queries on RDF quadruples. *Journal of Web Semantics*, 37, 90-111.

173. Neumann, T., and Weikum, G. (2009). *Scalable join processing on very large RDF graphs*. In Proceedings of the 2009 ACM SIGMOD International Conference on Management of data, 627-640.

174. Seufert, S., Anand, A., Bedathur, S., and Weikum, G. (2013). *Ferrari: Flexible and efficient reachability range assignment for graph indexing*. In 2013 IEEE 29th International Conference on Data Engineering (ICDE), 1009-1020.

175. Dia, A. F., Aoul, Z. K., Boly, A., and Métais, E. (2018). *Fast SPARQL join processing between distributed streams and stored RDF graphs using bloom filters*. In 2018 12th International Conference on Research Challenges in Information Science (RCIS), 1-12.

176. Vander Sande, M., Verborgh, R., Van Herwegen, J., Mannens, E., and Van de Walle, R. (2015). *Opportunistic Linked Data querying through approximate membership metadata*. In The Semantic Web-ISWC 2015: 14th International Semantic Web Conference, Bethlehem, 92-110.

177. Taelman, R., Van Herwegen, J., Vander Sande, M., and Verborgh, R. (2020). *Optimizing approximate membership metadata in triple pattern fragments for clients and servers*. In SSWS2020, the 13th International Workshop on Scalable Semantic Web Knowledge Base Systems, (2757), 1-16.

178. Bloom, B. H. (1970). Space/time trade-offs in hash coding with allowable errors. *Communications of the ACM*, 13(7), 422-426.

179. Almeida, P. S., Baquero, C., Preguiça, N., and Hutchison, D. (2007). Scalable bloom filters. *Information Processing Letters*, 101(6), 255-261.

180. Rajaraman, A., and Ullman, J. D. (2011). *Mining of massive datasets*. Cambridge University Press, Cambridge, 146-147.

181. Chang, F., Feng, W. C., and Li, K. (2004). *Approximate caches for packet classification*. In IEEE INFOCOM 2004, (4), 2196-2207.

182. Xie, K., Min, Y., Zhang, D., Wen, J., and Xie, G. (2007). *A scalable bloom filter for membership queries*. In IEEE GLOBECOM 2007-IEEE Global Telecommunications Conference, 543-547.

183. Almeida, P. S. (2023). A Case for Partitioned Bloom Filters. *IEEE Transactions on Computers*, 72(6), 1681-1691.

184. Hüseynli, A., and Akcayol, M. A. (2021). *Bloom Filter Based Graph Database CRUD Optimization for Stream Data*. In 2021 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), (2), 1056-1061.

185. Tanon, T. P., Vrandečíc, D., Schaffert, S., Steiner, T., and Pintscher, L. (2016). *From freebase to wikidata: The great migration.* In 25th International World Wide Web Conference, WWW 2016, 1419-1428.

186. Fellbaum, C. (Ed.). (1998). *WordNet: An electronic lexical database.* MIT press, Cambridge, 69-105.

187. Matuszek, C., Witbrock, M., Cabral, J., and DeOliveira, J. (2006). An introduction to the syntax and content of Cyc. *UMBC Computer Science and Electrical Engineering Department Collection*.

188. Rebele, T., Suchanek, F., Hoffart, J., Biega, J., Kuzey, E., and Weikum, G. (2016). *YAGO: A multilingual knowledge base from wikipedia, wordnet, and geonames*. In International Semantic Web Conference, 177–185.

189. Bollacker, K., Evans, C., Paritosh, P., Sturge, T., and Taylor, J. (2008). *Freebase: a collaboratively created graph database for structuring human knowledge.* In Proceedings of the 2008 ACM SIGMOD international conference on Management of data, 1247-1250.

190. Vrandečić, D., and Krötzsch, M. (2014). Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10), 78-85.

191. Jie, L., Jianfeng, C., Zhihong, R., Zhi, S., Hui, Y., and Rui, X. (2018). *A massive RDF storage approach based on graph database.* In Proceedings of the International Conference on Geoinformatics and Data Analysis, 169-173.

192. Li, Y., Manoharan, S., Jatana, N., Puri, S., Ahuja, M., Kathuria, I., Battersby, S. (2015). *Method of calculating the scores of the DB-Engines Ranking*. Proceedings - 3rd International Conference on Emerging Intelligent Data and Web Technologies, EIDWT 2012.

193. İnternet: DB-Engines. (2023). DB-Engines Ranking per database model category. DBMS popularity broken down by database model. Web: https://db-engines.com/en/ranking_trend/graph+dbms, Son Erişim Tarihi: 14.08.2023.

194. İnternet: Neo4j. (2023). Neo4j Graph Platform – The Leader in Graph Databases. Neo4j Graph Platform, Web: https://neo4j.com/ Son Erişim Tarihi: 14.08.2023.

195. İnternet: Green, A., and Fowler, A. (2013). MarkLogic Enterprise NoSQL Database Solution Definition for GCloud V What MarkLogic Server provides. MarkLogic Corporation. Web: https://silo.tips/download/marklogic-enterprise-nosql-database-solution-definition-for-gcloud-iv. Son Erişim Tarihi: 14.08.2023.

196. İnternet: APACHE_JENA_FUSEKI. (2018). Apache Jena - Apache Jena Fuseki. The Apache Software Foundation We: https://jena.apache.org/documentation/fuseki2/ Son Erişim Tarihi: 14.08.2023.

197. Akrami, F., Saeef, M. S., Zhang, Q., Hu, W., and Li, C. (2020). *Realistic Re-evaluation of Knowledge Graph Completion Methods: An Experimental Study*. In Proceedings of the ACM SIGMOD International Conference on Management of Data, 1995-2010.

198. Dettmers, T., Minervini, P., Stenetorp, P., and Riedel, S. (2018). *Convolutional 2D knowledge graph embeddings*. In 32nd AAAI Conference on Artificial Intelligence, AAAI 2018, 32(1), 1811-1818.

199. Toutanova, K., and Chen, D. (2015). *Observed versus latent features for knowledge base and text inference*. In Proceedings of the 3rd workshop on continuous vector space models and their compositionality, 57-66.

200. İnternet: Menzel, J. (2010). Deeper understanding with Metaweb. Deeper understanding with Metaweb (Google Official Blog), Web: https://googleblog.blogspot.com/2010/07/deeper-understanding-with-metaweb.html Son Erişim Tarihi: 14.08.2023.

201. İnternet: Chah, N. (2018). OK Google, What Is Your Ontology? Or: Exploring Freebase Classification to Understand Google's Knowledge Graph. Web: https://arxiv.org/pdf/1805.03885.pdf. Son Erişim Tarihi: 14.08.2023.

202. İnternet: Neo4j. (2023). The Use of Indexes. Web: https://neo4j.com/docs/cypher-manual/current/query-tuning/indexes/#administration-indexes-equality-check-using-where-single-property-index, Son Erişim Tarihi: 14.08.2023.

Gazili olmak ayrıcalıktır