



**T.C. İSTANBUL T CARET
 NİVERSİTESİ**

FEN BİLİMLERİ ENSTİT S 

SOSYAL A LARDA TOPLULUK ARAMA

Tu  e KA NICI

**Danışman
Dr.   r.  yesi Arzu KAKIŐIM**

**Y KSEK LİSANS TEZİ
BİLGİSAYAR M HENDİSLİ İ ANABİLİM DALI
İSTANBUL - 2021**

KABUL VE ONAY SAYFASI

Tuğçe KAĞNICI tarafından hazırlanan "**Sosyal Ağlarda Topluluk Arama**" adlı tez çalışması 02/09/2021 tarihinde aşağıdaki jüri üyeleri önünde başarı ile savunularak, İstanbul Ticaret Üniversitesi Fen Bilimleri Enstitüsü **Bilgisayar Mühendisliği Anabilim Dalı**'nda **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Danışman

Dr. Öğr. Üyesi, Arzu KAKIŞIM
İstanbul Ticaret Üniversitesi

Jüri Üyesi

Doç. Dr. M.Cem KASAPBAŞI
İstanbul Ticaret Üniversitesi

Jüri Üyesi

Doç. Dr. Buket DOĞAN
Marmara Üniversitesi

Onay Tarihi : 27.09.2021

İstanbul Ticaret Üniversitesi, Fen Bilimleri Enstitüsünün 27.09.2021 tarih ve 2021/322 numaralı Yönetim Kurulu Kararının 1. maddesi gereğince, ders yükümlülüğü ve tez yükümlülüğünü yerine getirdiği belirlenen "Tuğçe KAĞNICI" adlı öğrencinin mezun olmasına oy birliği ile karar verilmiştir.

Prof. Dr. Necip ŞİMŞEK
Enstitü Müdürü

AKADEMİK VE ETİK KURALLARA UYGUNLUK BEYANI

İstanbul Ticaret Üniversitesi, Fen Bilimleri Enstitüsü, tez yazım kurallarına uygun olarak hazırladığım bu tez çalışmada,

- tez içindeki bütün bilgi ve belgeleri akademik kurallar çerçevesinde elde ettiğimi,
- görsel, işitsel ve yazılı tüm bilgi ve sonuçları bilimsel ahlak kurallarına uygun olarak sunduğumu,
- başkalarının eserlerinden yararlanılması durumunda ilgili eserlere bilimsel normlara uygun olarak atıfta bulunduğumu,
- atıfta bulunduğum eserlerin tümünü kaynak olarak gösterdiğimi,
- kullanılan verilerde herhangi bir tahrifat yapmadığımı,
- ve bu tezin herhangi bir bölümünü bu üniversitede veya başka bir üniversitede başka bir tez çalışması olarak sunmadığımı

beyan ederim.

27/09/2021

Tuğçe KAĞNICI

İÇİNDEKİLER

	Sayfa
İÇİNDEKİLER.....	i
ÖZET	ii
ABSTRACT	iii
TEŞEKKÜR.....	iv
ŞEKİLLER DİZİNİ	v
ÇİZELGELER DİZİNİ	vi
SİMGELER VE KISALTMALAR DİZİNİ	vii
1. GİRİŞ.....	1
2. LİTERATÜR ÖZETİ.....	6
2.1. Özniteliksiz Ağlarda Topluluk Tespiti	6
2.2. Öznitelikli Ağlarda Topluluk Tespiti	7
2.3. Özniteliksiz Ağlarda Topluluk Arama.....	9
2.4. Öznitelikli Ağlarda Topluluk Arama.....	11
3. İLGİLİ KAVRAMLAR VE TOPLULUK ARAMA ALGORTİTMALARI	14
3.1. Graflar ve Topluluklar.....	14
3.1.1. Graflar	14
3.1.2. Topluluklar	15
3.2. Topluluk Arama Problemi	16
3.3. Öznitelikli Ağlarda Topluluk Arama.....	17
3.4. Topluluk Arama Algoritmaları	18
3.4.1. k -çekirdek tabanlı topluluk arama	19
3.4.2. k -tras tabanlı topluluk arama.....	23
3.4.3. k -klik tabanlı topluluk arama	26
3.4.4. k -kenar tabanlı topluluk arama	27
3.5. Topluluk Arama Algoritmalarının Karşılaştırılması	28
3.6. Topluluk Arama Problemi ve Topluluk Tespit Etme Problemi Arasındaki Farklar	29
4. ARAŞTIRMA BULGULARI VE TARTIŞMA.....	31
4.1. Önerilen Graf Yapısı	31
4.2. Veri Kümeleri ve Deneysel Kurulum.....	33
4.3. Performansın Değerlendirilmesi	35
4.4. Diğer Çalışmalar ile Karşılaştırma	37
5. SONUÇ VE ÖNERİLER.....	39
KAYNAKLAR	40
ÖZGEÇMİŞ.....	44

ÖZET

Yüksek Lisans Tezi

SOSYAL AĞLARDA TOPLULUK ARAMA

Tuğçe KAĞNICI

İstanbul Ticaret Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Dr. Öğr. Üyesi Arzu KAKIŞIM

2021, 44 sayfa

Bu çalışmada, topluluk arama problemi adreslenerek öznitelikli ağlar için öznitelik bilgisini ağ topolojisine entegre etme fikrine dayanan bir yaklaşım sunuyoruz. Gelişen teknoloji ile birlikte sosyal ağların yoğun bir şekilde kullanımı, bir dizi sosyal nesne ve bu nesnelerin arasındaki yoğun ilişkilerden oluşan karmaşık ağ yapılarını meydana getirmektedir. Bu tür karmaşık ağ yapılarından anlamlı bilgilerin çıkarılabilmesi için ağları topluluk seviyesinde analiz eden yaklaşımlar önerilmektedir. Bir topluluk, ağdaki nesneler arasındaki bağlantıların ağın geri kalanıyla olan bağlantılardan daha yoğun olduğu şekilde gözlendiği veya belirli özellikler açısından birbirleri ile benzerlik gösteren nesnelerin bir alt kümesi olarak tanımlanır. Son zamanlarda araştırmacılar, topluluk arama yaklaşımı olarak adlandırılan, belirli sorgu nesnesi veya nesneleri kullanılarak, karmaşık ağ yapılarından yüksek kaliteli toplulukları gerçek zamanlı olarak ortaya çıkaran yaklaşımlar önermektedirler. Bu yaklaşımlar, ağ yapılarını, ağ nesnelerinin düğümlere ve nesneler arasındaki ilişkilerin ise kenarlara karşılık düştüğü graf yapısı ile temsil etmektedirler. Var olan topluluk arama algoritmaları, çoğunlukla grafin bu topolojik yapısını temel alır. Öznitelikli ağlar ise topolojik graf bilgisinin yanı sıra, ağda bulunan düğümleri karakterize eden öznitelik verisini de içermektedir. Geleneksel topluluk arama yaklaşımları, düğümü karakterize eden zengin öznitelik verisini göz ardı ederek sadece ağ yapısı üzerinden sonuç üretmektedir. Bu çalışmada öznitelik verisinin ağ topolojisine entegre edilmesi sağlanarak topluluk arama yaklaşımlarına içeriksel bir boyut kazandırılır. Bu yaklaşım ile mevcut arama algoritmalarının öznitelikli ağlar üzerinde uygulanabilir hale getirilmesi amaçlanmaktadır. Böylelikle öznitelik verisinin sorgu verisi olarak kullanılması sağlanmaktadır. Ve böylece kullanılan algoritmalarının başarısının artırılması hedeflenmektedir. Önerilen yöntem gerçek dünya veri setleri kullanılarak denenmiştir. Sonuçların temel yöntemler ile karşılaştırıldığında daha yüksek doğruluk değerine sahip olduğu gözlemlenmiştir.

Anahtar Kelimeler: K-çekirdek, k-tras, öznitelikli ağlar, topluluk arama.

ABSTRACT

M.Sc. Thesis

COMMUNITY SEARCH ON SOCIAL NETWORKS

Tuğçe KAĞNICI

**İstanbul Commerce University
Graduate School of Applied and Natural Sciences
Department of Computer Engineering**

Supervisor: Asst. Prof. Dr. Arzu KAKIŞIM

2021, 44 pages

In this study, we present an approach based on the idea of integrating feature information into the network topology for attributed networks by addressing the community search problem. With the intensive usage of social networks, complex network structures are naturally formed, consisting of a series of social objects and the intense relations between these objects. In order to extract meaningful information from such complex networks, some existing approaches analyze the networks at the community level. A community is defined as a subset of objects in which the connections between objects in the network are observed to be more intense than the connections to the rest of the network, or that are similar to each other in certain properties. Recently, researchers propose new approaches, called community search, that aim to reveal high-quality communities from complex network structures in real time using specific query object or objects. These approaches represent the network structure as a graph structure in which network objects correspond to nodes and relationships between objects to edges. Existing community search algorithms are mostly based on this topological structure of the graph. On the other hand, attributed networks consist of attribute data characterizing the nodes in the network as well as topological graph information. Traditional search approaches ignore the rich attribute information and produce results only through the network structure. In this study, we aim to enhance community search approaches by integrating the attribute data into the network topology. Thus, we provide a contextual dimension for topology-based community search algorithms. With our approach, it is aimed to make existing search algorithms applicable on attributed networks. At the same time, it is ensured that the attribute information is used as query data and it is aimed to increase the accuracy of the existing algorithms. The proposed method has been evaluated using real-world datasets. The results demonstrate that our proposed method achieves higher accuracy than the baseline methods.

Keywords: Attributed networks, community search, k-core, k-truss.

TEŞEKKÜR

Bu araştırma için beni yönlendiren, karşılaştığım zorlukları bilgi ve tecrübesi ile aşmamda yardımcı olan değerli Danışman Hocam Dr. Öğr. Üyesi Arzu KAKIŞIM'a teşekkürlerimi sunarım.

Tezimin her aşamasında beni yalnız bırakmayan aileme sonsuz sevgi ve saygılarımı sunarım.

Tuğçe KAĞNICI
İSTANBUL, 2021



ŞEKİLLER

	Sayfa
Şekil 1.1. Topluluk yapısına örnek bir ağ yapısı	2
Şekil 1.2. Öznitelikli ağlarda topluluk arama yapısı için bir örnek.....	4
Şekil 3.1. Örnek bir graf yapısı.....	14
Şekil 3.2. Örnek bir graf yapısı üzerinde yer alan toplulukların gösterimi...	15
Şekil 3.3. Topluluk arama problemi için örnek bir sorgu düğümü	16
Şekil 3.4. Öznitelikli ağlar için örnek bir homojen alt topluluk	18
Şekil 3.5. Çekirdek Ayırıştırma Algoritması	21
Şekil 3.6. k -çekirdek tabanlı arama algoritması için örnek.....	22
Şekil 3.7. Tras Ayırıştırma Algoritması.....	24
Şekil 3.8. k -tras tabanlı arama algoritması için örnek	25
Şekil 3.9. k -klik tabanlı arama için örnek.....	26
Şekil 3.10. k -kenar tabanlı arama için örnek.....	28
Şekil 4.1. Önerilen graf yapısı için örnek çizim	32
Şekil 4.2. Farklı veri setleri için karşılaştırılan yöntemlerin keskinlik (<i>Prec</i>) ve duyarlılık (<i>Rec</i>) değerleri.....	36

ÇİZELGELER

	Sayfa
Çizelge 4.1. Çalışmada kullanılan beş farklı veri kümesine ait istatistikler..	33
Çizelge 4.2. Farklı veri setleri üzerinde yöntemlere ait F-ölçüm sonuçları..	35



SİMGELER VE KISALTMALAR

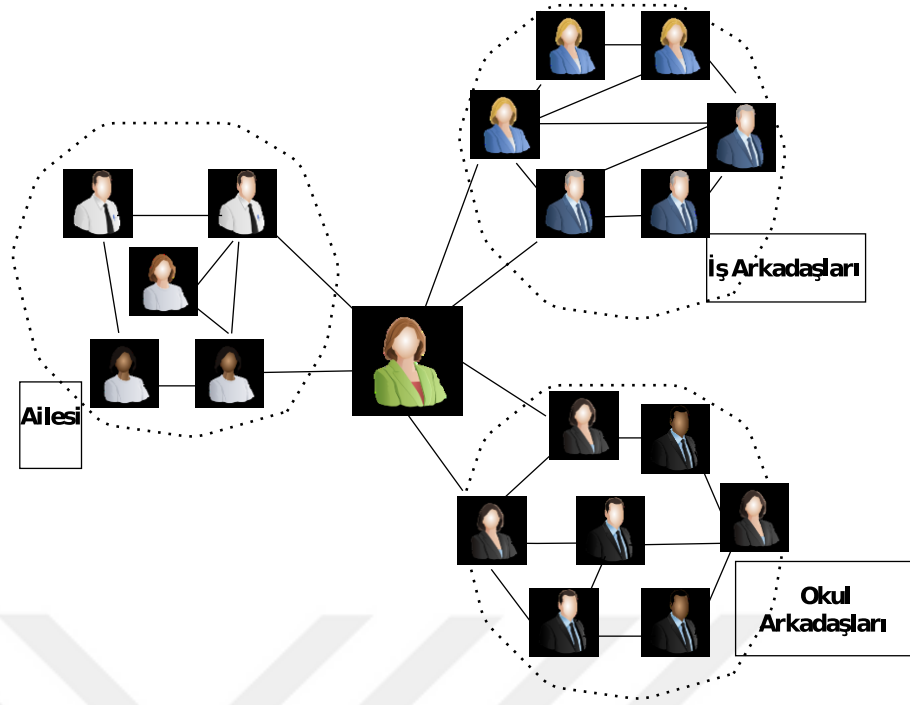
ATC	Attributed Truss Communities
e	Grafda bulunan kenar
E	Grafta bulunan kenar kümesi
F	Öznitelik Kümesi
G	Graf
S	En Büyük Alt Graf
VAC	Vertex-Centric Attributed
V	Grafda bulunan düğüm kümesi
q	Sorgu düğümü
v	Grafda bulunan düğüm
$\tau(e)$	Tras Derecesi



1. GİRİŞ

Son yıllarda teknolojiadaki gelişmeler ile birlikte sosyal ağlar, iş birliği ağları, protein-protein etkileşim ağları gibi birçok bileşen ve bileşenler arasındaki etkileşimden oluşan karmaşık ağ yapıları ortaya çıkmaktadır. Karmaşık ağ yapılarındaki bileşenleri ve aralarındaki ilişkileri temsil etmede graflar güçlü bir model haline gelmiştir. Oluşturulan graf temsili sayesinde ağ yapılarını formülize etmek ve bu sayede ağdaki bileşenleri ve bileşenler arasındaki etkileşimi analiz etmek kolaylaşmaktadır. Graf yapılarındaki düğümler ağdaki nesneleri, kenarlar ise nesneler arasındaki bağlantıları, etkileşimleri temsil etmektedir. Son yıllarda, ağın yorumlanabilen bir özetini sunabilmek için araştırmacılar büyük ve karmaşık ağ yapılarında gözlenen toplulukların tespiti problemini çalışmaktadırlar (Lancichinetti ve Fortunato, 2009; Chunaev, 2020). Topluluk tespiti, global bir ağ analizi sağlamak için ağdaki tüm toplulukları bulmayı hedefleyen bir yaklaşımdır. Topluluk kavramı ise benzer özellikler gösteren ve birbirleri ile yoğun ilişkiler içerisinde olan varlıklar/nesneler kümesi olarak tanımlanmaktadır. Bu durumda benzer ilgi alanlarına sahip kişiler bir sosyal ağda birleşme eğilimi göstererek düğüm kümelerin veya diğer deyişle toplulukların oluşmasına sebep olurlar. Şekil 1.1’de bir bireyin dâhil olduğu okul, iş arkadaşı ve aile gruplarının oluşturduğu topluluklar ve bu topluluklar arasındaki ilişkiler örneklendirilmiştir.

Son zamanlarda, sosyal ağların yoğun olarak kullanılması ile birlikte, Facebook, Twitter vb. birçok sosyal ağ sitesinde kullanıcılar ve kullanıcılar arasındaki etkileşimlerin doğal olarak oluşturduğu ağ yapıları ortaya çıkmaktadır. Etkileşim sayısı her geçen gün artan bu ağlar geniş ölçekli karmaşık ağ yapıları olarak tanımlanmaktadır. Bu karmaşık ağ yapısının anlamlı ve verimli bir şekilde analiz edilebilmesi, tüketicilere hizmet sunan öneri sistemleri veya reklam için doğru alıcıları bulmayı amaçlayan içerik sağlayıcıları gibi birçok gerçek zamanlı çözümler üreten uygulamalar için önemli bir hale gelmektedir. Bu aşamada örneğin; ürün önerisi yapmak için anlamlı bir hedef grubun seçilmesi problemi esasen bir topluluk tespiti problemi olarak karşımıza çıkmaktadır (Bedi ve Sharma, 2016).



Şekil 1.1. Topluluk yapısına örnek bir ağ yapısı

Bir sosyal ağdaki toplulukların keşfedilmesi pazarlama, sosyoloji, güvenlik, biyoloji gibi pek çok alanda insanlar arasındaki ilişkilerin ya da nesneler arasında bulunan etkileşimlerin analiz edilebilmesi için ya da hedeflenen hiyerarşik yapıların bulunabilmesi için oldukça önemlidir (Kakisim ve Sogukpınar, 2015). Örneğin konum tabanlı bir sosyal ağ yapısında keşfedilen sosyal çevre sayesinde yakınlardaki arkadaş buluşması kullanıcılara önerilebilir ya da bu arkadaş grubuna yakın konumdaki restoranların kupon, reklam gibi paylaşımları iletilebilir (Huang vd. , 2017). Biyoloji alanı için örnek verecek olursak protein-protein etkileşimi gibi ağ yapısında birbiri ile yüksek düzeylerde etkileşime giren protein topluluklarının tespit edilmesi belirli moleküler yapıların ortaya çıkarılması için önemli olmaktadır (Huang vd. , 2019).

Son zamanlarda ise bazı araştırmacılar daha yeni bir konu olan “topluluk arama” problemi üzerine yoğunlaşmaktadırlar. Topluluk arama, ağ yapıları üzerinde kişiselleştirilmiş sorgu yapma imkânı sağlayarak ağ yapısından hızlı ve verimli bir şekilde anlamlı ve yoğun ilişkili alt graf yapılarının diğer bir deyişle topluluklarının ortaya çıkarılması problemidir. Problemin ortaya çıkış nedeni ağ

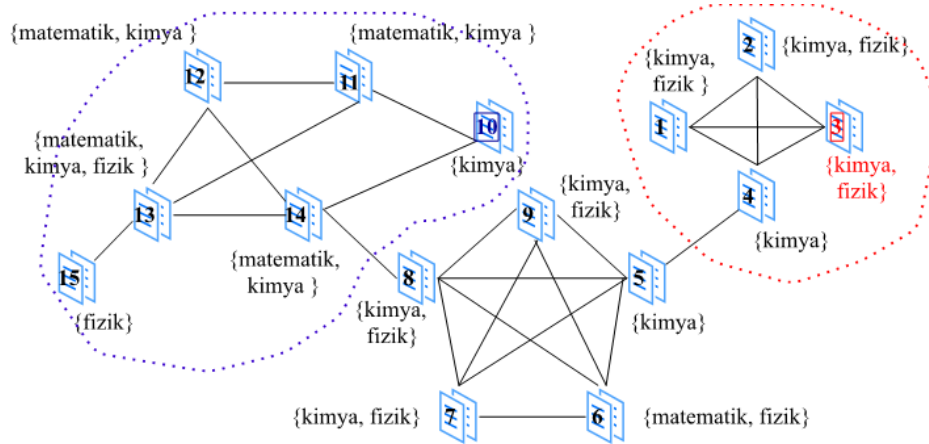
yapılarından daha çok kişiselleştirilmiş ve bir ya da birkaç anlamlı gruba odaklı cevapları bulma ihtiyacıdır. Topluluk arama yaklaşımları, belli özelliklere sahip sorgu düğümleri kullanarak, bu sorgu düğümleri ile ilişkili toplulukları gerçek zamanlı olarak ortaya çıkarmayı hedefler (Huang ve Lankshmanan, 2017).

Son zamanlarda, k -çekirdek (k -core) (Barbier vd. , 2015), k -klik (k -klick) (Yuan vd. , 2015), k -tras (k -truss) (Huang vd. , 2014) ve k -kenar (k -edge) (Chang vd. , 2015) olmak üzere literatürde birçok graf tabanlı topluluk arama yöntemi önerilmiştir. Bu yöntemlerin ortak özelliği topluluk arama problemi için literatürde bulunan graf altyapılarını keşfeden yaklaşımları kullanmalarıdır. Ağ yapısında topluluk arama yaklaşımları, sorgu düğüm ya da düğümlerini temel alarak k -çekirdek alt graftaki her bir düğümün en az bir k komşusu olduğu graf yapısını keşfederken , k -tras her bir düğümün en az $(k - 2)$ üçgen içinde yer aldığı bir alt grafi elde eder (Huang ve Lakshmanan, 2017). K -klik, k tane düğümün birbirine benzersiz bir kenarla bağlı olduğu bir alt grafi, k -kenar ise $k - 1$ sayıda kenar çıkarıldığında bağlılığını kaybetmeyen bir alt grafi bulmaktadır. (Zhang vd. , 2020).

Facebook, Twitter vb. gibi ağlarda ağdaki düğümler genellikle o topluluğu anlamlandırmak adına bazı özelliklere sahiptir. Örneğin iş birliği ağlarında yazarlar benzer uzmanlık alanlarına sahiptirler. Protein ağlarında, proteinler belirli moleküler yapılara ve işlevselliklere ya da benzer hücresel içeriğe sahiptirler. Sosyal ağlar da ise kullanıcılar ortak ilgi alanlarına sahiptirler. Ağdaki her bir nesneyi ya da düğümü tanımlayan bu karakteristik özellikler, öznitelik verisi olarak adlandırılır. Hem graf yapısına sahip hem de öznitelik verisine sahip ağ yapıları öznitelikli ağlar olarak isimlendirilirler. Graf temsil modeli, ağ nesnesinin ağdaki bütünsel ve yerel konumunu temsil ederken, öznitelik verisi düğümü tanımlayan içeriksel veriyi temsil etmektedir. Var olan graf tabanlı topluluk arama yaklaşımları, yalnızca ağ topolojisini dikkate alarak düğümlere ait öznitelik verisini göz ardı etmektedir.

Fakat son zamanlarda düğümlere ait özniteliklerinde topluluk oluşumları üzerinde etkisi olduğu düşünülerek öznitelik verisini de topluluk arama

sürecine dâhil eden yeni yöntemler önerilmektedir (Huang ve Lankshmanan, 2017; Sun vd. , 2020). Böylece özniteliklerinde sorgu olarak kullanılabilmesi sağlanarak topluluk arama yönteminde, içeriksel olarak benzer yapıya sahip düğümlerin ortaya çıkarılması önem kazanmaktadır. Şekil 1.2’de öznitelikli bir graf yapısı sunulmuştur. Şekilde, bir örnek ağ yapısı içerisinde gözlenen üyelerin (düğümlerin) uzmanlık alanlarına (fizik, kimya ve matematik) ve aralarındaki ilişkilere yer verilmiştir. Bu graf temsili üzerinde “3.” üye sorgulanır ise öznitelik homojenliği bakımından yüksek bir alt graf olarak 1. , 2. , 3. ve 4. üyenin oluşturduğu topluluk yapısı keşfedilir. Ağdaki düğümlerin sahip olduğu öznitelik verisi dikkate alınmadığında ağdaki sorgu düğümlerinin heterojenliği yüzünden, önceki arama metotları ile yapılan topluluk arama modellerinin ürettiği toplulukları yorumlamak zor olabilir. Örneğin; sadece graf yapısı temel alınarak, 10. düğüm ile ilgili sorgulama yapıldığında, 11, 12, 13, 14, 15 düğümlerinin hepsi sonuç olarak dönebilir. Bu graf topolojisi açısından birbiri ile bağlantılı düğümlerden oluşan bir alt graf yapısıdır. Graf yapısı ile birlikte öznitelik bilgisi temel alınarak sorgu yapıldığında ise, 10. düğüm ile ortak bir özniteliğe sahip olmayan ve 10. düğümün 3. dereceden komşusu olması nedeni ile sonuç olarak dönen topluluk içerisinde 15. düğüm olmayabilir. Sonuç olarak edilen topluluk öznitelik açısından daha homojen bir alt graftır. Öznitelik homojenliğinin yüksek olması daha doğru sonuçların keşfedilmesine yardımcı olabilir. Sadece graf yapısını temel alarak arama yapan yöntemler, öznitelik açısından daha homojen alt grafları gözden kaçırabilmektedir.



Şekil 1.2. Öznitelikli ağlarda topluluk arama yapısı için bir örnek

Özetlemek gerekirse, içeriksel olarak homojen gözlenen ve topolojik olarak benzer olan düğüm alt kümelerinin keşfedilmesi, öznitelikli graflar için daha iyi sonuçlar üretebileceği ön görülmektedir. Bu çalışmada, zengin öznitelik verisinden faydalanmak için öznitelikli ağlarda topluluk arama problemine yönelik düğümlerin sahip olduğu öznitelik bilgilerini ağ topolojisine entegre eden yeni bir graf temsil yaklaşımı sunuyoruz. Önerdiğimiz yöntem, öznitelikleri içeriksel bir düğüm olarak ele alır ve her düğümün yalnızca kendi öznitelikleriyle bağlantılı olduğu bir graf temsili oluşturarak, ağ topolojisine içeriksel bir boyut ekler. Bu yaklaşımı ile mevcut topluluk arama algoritmalarını öznitelikli ağlar üzerinde de uygulanabilir hale getirmek amaçlanmaktadır. Aynı zamanda çalışmamızın amacı olarak ağ yapısına içerik verisi dâhil edilerek topoloji tabanlı algoritmaların başarısının artırılması, öznitelik verilerinin sorgu düğümleri olarak kullanılabilirliğinin sağlanması hedeflenmiştir. Çalışmamızda gerçek dünya veri setleri kullanarak elde ettiğimiz deneysel sonuçlar, önerilen yöntemin, var olan temel yöntemler ile kıyaslandığında daha yüksek düzeyde doğruluk değerlerine sahip olduğunu göstermektedir.

Bu çalışmanın geri kalan kısmında, Bölüm 2’ de konu ile ilgili daha önceden yapılan yöntemler hakkında literatür çalışması yer almaktadır. Bölüm 3’ de ise konunun anlaşılabilirliğini sağlamak için ilgili kavramları ve topluluk arama algoritmalarına yer verilmiştir. Bölüm 4’ de önerilen graf yapısı, kullanılan veri setleri ile performansın değerlendirilmesi, araştırma bulguları ve tartışma başlığı altında sunulmuştur. Bölüm 5’ de sonuç ve öneriler paylaşılarak çalışma tamamlanmıştır.

2. LİTERATÜR ÖZETİ

2.1.Özniteliksiz Ağlarda Topluluk Tespiti

Topluluk tespiti problemi, bir graf yapısına ait tüm toplulukları keşfetmeyi veya tespit etmeyi amaçlar. Newman ve Girvan (Newman ve Girvan, 2004) problem için ağ düğümlerinin yoğun bir şekilde bağlı alt gruplara bölünebilmesi için bir dizi algoritma önermiştir. Önerilen algoritmaların tümü iki ortak özelliği taşımaktadır. İlk olarak, ağ yapısındaki toplulukları bölebilmek için en ilgisiz kenarların yinelemeli olarak kaldırılmasını içerir. Kaldırılacak kenarlar tanımlanan bir dizi olasılık ölçüsünden herhangi biri kullanılarak tespit edilir. İkincisi kaldırılan her kenardan sonra kullanılan ölçüler yeniden hesaplanır. Aynı zamanda ağın topluluklara bölünebilmesi için nesnel bir ölçüm sağlayan yapı önermişlerdir. Fortunato yaptığı çalışmada (Fortunato, 2010) bu alanda yapılan çalışmaları ayrıntılı olarak ele almış: Hiyerarşik kümeleme, parçalı kümeleme, spektral kümeleme gibi geleneksel metotlara, modülerlik tabanlı yöntemlere, spektral ve dinamik algoritma yöntemleri kullanılarak yapılan topluluk tespiti yöntemlerini çalışmasında kapsamlı bir şekilde sunmuştur. Böylece istatistiksel fizikçiler tarafından tasarlanan tekniklere özel olarak odaklanarak, problemin ana unsurlarının tanımından, geliştirilen çoğu yöntemin sunumuna kadar konuyu kapsamlı bir şekilde sunmuşlardır. Kümeleme yöntemlerinin nasıl test edilmesi ve birbirleri ile karşılaştırılması gerektiğinden bahsederek çalışmalarında gerçek dünya ağlarındaki uygulamalardan bahsetmişlerdir. Perozzi ve arkadaşları (Perozzi vd. , 2014) bir ağdaki sorgu düğümünün gizli temsillerini öğrenmek için yeni bir yaklaşım olarak çevrimiçi bir öğrenme algoritması olan derin yürüyüş (Deepwalk) yöntemini sunmuşlardır. Ağdaki gizli temsiller, istatistiksel modeller tarafından kolaylıkla kullanılan sürekli bir vektör uzayındaki sosyal ilişkileri kodlarlar. Derin yürüyüş yöntemi derin öğrenme ile beraber dil modellemede kullanılmaya başlanmıştır. Bu yöntemi kullanarak BlogCatalog, Flickr ve YouTube gibi sosyal ağlar için çeşitli çok etiketli ağ sınıflandırma görevlerindeki gizli temsilleri sunmuşlardır. Kullanılan derin yürüyüş algoritması çevrimiçi bir öğrenme algoritmasıdır. Çalışmalarında yöntemlerini rakip yöntemler ile

karşılaştırmışlar ve etiketlenmiş verilerin seyrek olması ve yoğunluğuna bağlı olarak sonuçlar arasında farklı başarımlar elde etmişlerdir. Grover ve Leskovec (Grover ve Leskovec, 2016) ise ağlardaki düğümler için sürekli özellik temsillerini öğrenmek için algoritmik bir çerçeve olan node2vec yöntemi önermişlerdir. Bu yöntemde düğümlerin ağ komşuluğunu koruma olasılığını en üst düzeye çıkaran düşük boyutlu bir özellik uzayına düğümlerin eşlemesini önermişlerdir. Bir düğümün ağ komşuluğunu tanımlamışlar ve bir rastgele yürüyüş prosedürü tasarlamışlardır. Verimli bir şekilde ağ yapısını araştırmışlardır. Daha önce ağ yapıları için yapılan komşuluk tanımlarından ziyade daha esnek bir komşuluk tanımı sunmuşlardır. Node2vec yönteminin çok etiketli sınıflandırma ve çeşitli alanlardan birçok gerçek dünya ağları üzerinde bağlantı tahminlerinin etkinliğini göstermişlerdir.

2.2.Öznitelikli Ağlarda Topluluk Tespiti

Öznitelikli ağlarda topluluk tespiti, öznitelik bakımından homojenliği yüksek olan alt graf yapılarını tespit edebilmeyi amaçlar. Bu problemi çözme yöntemleri kabaca üç kategoriye ayrılabilir: İlk kategori karmaşık ağ yapısı üzerinde klasik yöntemleri uygulamaktır. İkinci kategori topluluğu topolojik yapısına uygun olarak bölümlene yapılarak algoritmaların ağ üzerinde topluluk tespit edilmesi sağlanır. Üçüncü olarak ise veri madenciliği yöntemleri kullanılabilir.

Graf kümelemenin amacı, düğüm bağlantısı ve ya komşuluk benzerliği gibi çeşitli kriterlere dayalı olarak büyük hacimli graf yapısındaki düğümleri farklı kümelerle bölmektir. Graf kümeleme tekniği büyük graf yapılarında yoğun bağlantılı grupları tespit edebilmek için yaygın olarak kullanılmaktadır. Mevcuttaki yaklaşımların topolojik olarak ağ yapısına odaklanması ve heterojen olan düğüm özelliklerini göz ardı etmesi nedeni ile Zhou ve arkadaşları (Zhou vd. , 2009) birleşik bir mesafe ölçütü aracılığı ile hem yapısal hem de özniteliksel benzerliklere dayanan SA-Cluster adını verdikleri yeni bir kümlere algoritması önermişlerdir. Önerilen yöntem niteliklerle ilişkili k 'yı bölerek elde edilen her kümenin homojen öznitelik değerine sahip yoğun bağlantılı alt graf

yapısını verir. Böylece yapısal benzerlik ve öznitelik benzerliğinin katkı derecesini öğrenmek için etkili bir yöntem önermiş ve önerilen yöntemi teorik olarak analiz etmişlerdir. Combe ve arkadaşları (Combe vd. , 2012) yaptıkları çalışmada düğümler arasındaki öznitelik benzerliğini elde etmek için vektör benzerlik hesaplama yöntemini kullanmış ve global benzerliği elde etmek için yapısal benzerliğin doğal kombinasyonunu birleştirmiştir.

Büyük graf yapıları için bir başka yaklaşımda graf yapısının yanı sıra, veri kaynakları genellikle özellik vektörleri aracılığı ile nesneler hakkında bilgi sağlar. Bilgi çıkarmak içinde son yaklaşımlar her iki bilgi türünü de aynı anda dikkate alır. Bu sebeple kümele modelleri bir ağ içerisindeki bir birine yoğun şekilde bağlı bir yapı olarak oldukça karmaşıktır ve mevcut yöntemler belirli şekilde yürütülemez ve büyük hacimli graflara uygulanamaz. Bu duruma çözüm olarak Günnemann ve arkadaşları (Günnemann vd. , 2013) birleştirilmiş veri kaynaklarının verimli bir şekilde kümelenebilmesi için öncelikle verimliliği etkileyen kritik kısımları belirleyerek yerleşik GRASP (Greedy Randomized Adaptive Search) ilkesini kullanarak optimal kümeleme çözümüne yaklaşan EDCAR algoritmasını geliştirmişlerdir. Geliştirilen yöntemi çalışma zamanı açısından diğer yaklaşımlarla kıyaslamışlar ve daha iyi performans göstererek verimli kümeler elde ettiğini çalışmalarında sunmuşlardır. Fakat önerilen bu algoritma yüksek zaman karmaşıklığına sahiptir. Bello ve arkadaşları (Bello vd. , 2013) graf yapısının yapısal özellikleri ile birlikte hem zamansal hem de öznitelik bilgisini içeren bir topluluk tespiti algoritması önermişlerdir. Önerilen yöntem heterojen özniteliğine sahip grafların yanı sıra, gerçek dünya ağlarına uygulanabilirliği için gerekli olan yapı ve öznitelik bilgisindeki değişiklikleri ele alır. Chunaev 2020 yılında yaptığı çalışmada (Chunaev, 2020) ağıın hem yapısını hem de özniteliklerini kullanan topluluk tespit yöntemlerinin kapsamlı bir araştırmasını çalışmıştır. Çalışmasında özniteliklerin ne zaman ve nasıl birleştiğine bağlı olarak bir sınıflandırma önermiştir. Her sınıfın tanımını yapıp, genel fikre değinerek hangi yöntemin diğerinden daha iyi bir performans gösterdiğine ve performansın değerlendirilmesi için hangi veri setlerinin ve kalite ölçütlerinin kullanıldığına dair mevcut bilgileri inceleyerek sunmuşlardır.

Aynı zamanda problemin çözümü için mevcut durum hakkında sonuçlar çıkararak gelecekte çözülmesi önemli görünen diğer sorunları açıklamışlardır.

2.3.Özniteliksiz Ağlarda Topluluk Arama

Topluluk ara problemi Sozio ve Gionis (Sozio ve Gionis, 2010), tarafından ilk kez tanımlanmıştır. Topluluk arama sorunu, topluluk tespiti sorununun sorguya bağımlı varyantı olup bir G grafi ve sorgu düğümü q verildiğinde, q 'nın da içerisinde bulunduğu birbirine yoğun bağlantılı düğümlerden oluşan G' nın alt grafini bulmayı amaçlar. Bu amaçla çalışmalarında minimum derece ve mesafeye dayalı bir yoğunluk ölçüsü belirleyerek bu ölçü için açgözlü bir algoritma geliştirmişlerdir. Daha sonrasında algoritmalarını uygun değer çözümleri verebilmesi için genişletmişlerdir. Çalışmalarında belirli bir üst sınırdan büyük olmayan toplulukları bulan algoritmalar sunmuşlardır.

Topluluk arama probleminin tanımın yapılmasından beri topluluk arama ile ilgili çalışmalar kapsamlı bir şekilde devam etmektedir. Yapılan çalışmalar genel olarak graf türüne göre sınıflandırılabilir: Basit yapıdaki graflarda topluluk arama ve karmaşık yapıdaki graflarda topluluk arama. Basit yapıli graflarda topluluk arama yöntemleri olarak, çekirdek tabanlı model (core-based model), tras tabanlı model (truss-based model), klik tabanlı model (clique-based model) ile kenar bağlantı modeli(edge connectivity component (ECC)-based model) ve sorgu yoğunluğuna bağıli model (query-biased density model) olmak üzere çeşitli yöntemler önerilmiştir.

Huang ve arkadaşları(Huang vd. , 2014) pratik olarak yararlı olan ancak literatürde az çalışılmış çevrim içi bir topluluk arama problemi incelemişlerdir. Çalışmalarında k -tras yapısına dayalı yeni bir topluluk modeli önermişlerdir. Aynı zamanda k -tras topluluğı arama yöntemini graftaki kenar ve düğümlerin sık eklendiğı ve kaldırıldığı dinamik bir graf ortamında araştırmışlardır. Topluluk boyutu ile doğru orantılı bir maliyetle k -tras topluluklarının verimli bir şekilde aranmasını destekleyen kompakt bir dizin yapısı sunmuşlardır.

Barbier ve arkadaşları(Barbier vd. , 2015) min-derece tabanlı topluluk aramasına yönelik mevcut yaklaşımların graf yapısının neredeyse tamamına yakın bir kısma bakması gerektiğinden dolayı oluşan verimlilikle ilgili kısıtlamalara alternatif olarak, k -çekirdek tabanlı yeni bir arama yöntemi önermişlerdir. Önerilen yaklaşım, sorgu sırasında alınan ve nihai çözümü üretmek için önceden hesaplanmış verilerin kullanılmasına dayanmaktadır. Bunun için iki yaklaşım sunmuşlardır. Birincisi sorgu zamanında gerekli olan verilerin daha hızlı alınmasını garanti eder, diğeri ise düşük düzeyde depolama alanı gerektirir. Çevrimiçi sorgu işleme için iki aşamalı bir yöntem tasarlamışlardır. İlki önceden hesaplanmış bilgi olarak döndürülen alt graf indirgemek için ağgözlülük stratejisi kullanır. İkinci adım, ilk adımda elde edilen indirgenmiş alt grafi kullanarak sıralı bir şekilde bağlantı ve kısıtlamaları hedefleyen nihai çözümü verir.

Chang ve arkadaşları(Chang vd. , 2015) verilen bir k için G grafının tüm k -kenar bağlantı bileşenlerini maksimum bağlantı ile verimli bir şekilde hesaplamak için çalışmışlar ve çevrimiçi sorgu yapabilmek için algoritmanın sonuç boyutuna göre doğrusal zamanda çalıştığı yeni bir dizine dayalı verimli bir algoritma önermişlerdir.

Büyük hacimli graflardaki maksimum klik numaralandırma işlemi zaman açısından performanslı olmadığından ötürü Yuan ve arkadaşları(Yuan vd. , 2016), graftaki en çok sayıda düğümü kapsayabilen üst- k klikleri bulmak üzere pek çok çeşitli k -klik topluluk arama problemlerini incelemişlerdir. Bu yöntem, ağ motifin keşfedilmesi, büyük graf yapılarında anormallik tespiti dâhil olmak üzere pek çok uygulamada yaygın olarak kullanılabilir.

Yoğun bağlantılı topluluk arama yaklaşımları için Huang ve arkadaşları (Huang vd. , 2017) sosyal çevrelerin keşfedilmesi ve coğrafi sosyal grupların sorgulandığı farklı uygulama alanlarındaki çeşitli ağ türlerinde kullanılan topluluk arama yöntemlerini kapsamlı olarak ele almış ve yöntemlerin detaylı olarak karşılaştırmalarını sunmuşlardır.

Ebadian ve Huang (Ebadian ve Huang, 2019) özel bir graf yapısı örneğın Facebook'taki gizli arkadaş listeleri ve Sina Weibo'daki gizli takipler gibi yalnızca sahibi tarafından görüntülenebilen yapıları veya da kamu, resmi kurum ve kuruluşlara ait ağ yapıları üzerinde ağ yapısındaki her bir kenarın en az $k-2$ üçgen içersinde yer aldığı graf yapısındaki k -tras topluluklarını bulmak için hibrit bir stratejiye dayanan yeni bir algoritma önermişlerdir.

2.4. Öznitelikli Ağlarda Topluluk Arama

Öznitelikli graflarda topluluk araması, ağ yapısındaki düğümlerin sahip olduğu karakteristik özelliğe sahip olmasından dolayı ağın hem yapısal bütünlüğünü hem de nitelik bütünlüğünü dikkate alır. Sorgu parametrelerine göre, öznitelikli ağlar üzerinde topluluk araması iki kategoride sınıflandırılabilir. Birinci kategoride sorgu parametresi olarak hem öznitelik verisi ya da verilerini ve sorgu düğümü ya da düğümlerini alır. İkinci kategorideki yöntemler ise sadece öznitelik verisini, sorgu parametresi olarak alarak sorgulanan, öznitelik verisini içeren alt grafi sonuç olarak vermektedir.

Öznitelikli graflarda öznitelikli bir alt grafi sonuç olarak döndüren topluluk arama sorgusu çalışmışlardır. Sorgu sonucu dönen alt graf, verilen G grafının hem yapısal tutarlılığı (düğümlerinin bağılılığı) hem de düğümlerin ortak özniteliklerini karşılamaktadır. Alt graflar bir topluluğun nasıl ve neden oluştuğunun daha iyi anlaşılmasını sağlar. Bu yapılarda kullanıcıların kişiselleştirilmiş sorgu yapabilmesi mümkündür. Örneğın topluluk yapısında üyeler hobilerine göre birliktelik gösterebilir. Bu nedenle sorgu parametresi olarak aranılan hobiye göre girdi verilerek topluluk ağ yapısında aranabilir. Bu yaklaşımla Fang ve arkadaşları (Fang vd. , 2017) etkin bir arama için CL-ağaç dizin yapısı (CL-tree index structure) ve buna dayalı olarak da üç algoritma geliştirmişler ve önerilen algoritmaları Flickr, DBLP gibi büyük hacimdeki graflar üzerinde denemişlerdir. Huang ve arkadaşları (Huang vd. , 2017) ise son yıllarda araştırmacıların ilgi odağı olmaya başlayan topluluk arama problemi için ağların yapısını, problemin kavramları tanıtarak en son yapılan araştırmalara genel bir bakış sunmuşlardır. Aynı zamanda farklı topluluk

modellerinde kullanılan arama tekniklerinin kapsamlı bir karşılaştırmasını paylaşmışlardır. Huang'ın Lakshmanan (Huang ve Lakshmanan, 2017) ile aynı yıl yaptığı bir başka çalışmada problemin NP-zor olduğunu kanıtlayarak öznitelikli ağ yapısında öznitelikli k -tras topluluklarını (attributed truss communities (ATC)) bulmak için çalışmışlardır. Önerdikleri yöntemde bir skorelama fonksiyonu kullanarak en az popüler niteliğe sahip olan düğümler ATC' yi küçülmek için çıkartılır. Bu yaklaşımla ağgözlü bir algoritma geliştirmişlerdir. Zhang ve arkadaşları(Zhang vd. , 2018) gerçek hayatta sosyal ağlardaki düğümlerin ve özniteliklerin değişimini göz önünde bulundurarak dinamik graflar üzerinde öznitelik içeren yeni bir topluluk tanımlamış ve dizin yapısı kullanmadan graf üzerinde yerel arama yapan yeni bir yaklaşımda bulunmuşlardır. Daha sonraki yıllarda Huang ve arkadaşları(Huang ve Lakshmanan, 2019) LocATC algoritmasını önermişlerdir. Bu algoritma öncelikli olarak ilişkili düğümler için korelasyon hesaplar ve sorgu düğümünü içeren en yüksek ilişkili düğümlerin oluşturduğu alt grafi bulmaya çalışır. Fakat bu algoritma bazı ağ yapıları için yetersiz kalabilir. Sun ve arkadaşları (Sun vd. , 2020) ise birden fazla sorgu düğümü için arama yapabilen bir algoritma önermişlerdir. Arama yalnızca belli kriterleri taşıyan topluluklarla ilgilenir. Önerilen algoritma öncelikli olarak sorgu düğümünü içeren aday alt graflarda maksimum k -kenar bağlantı bileşenini bulan, ardından öznitelik işlevselliği en az katkısı olan düğümü aday alt graftan iteratif olarak silmektedir. Algoritmanın aynı zamanda tasarlanan steiner uzaklığını kullanarak mesafeye dayalı olarak arama yapması sağlanmıştır. Liu ve arkadaşları(Liu vd. , 2020) ise kullanıcıların sorgu özniteliklerine ait koşulları belirlemek durumunda kalması ve tek bir öznitelik türüne göre sorgu yapabilmesi gibi kısıtlamalara alternatif olarak VAC (Vertex-Centric Attributed) isimli yeni bir algoritma önermişlerdir. Algoritma öznitelik puanına sahip olan ve birbirlerine yoğun bir şekilde bağlı sorgu düğümünü içeren k -tras modeline uygun olan alt grafi döndürmektedir. Arama yaklaşımı olarak ise iki yöntem kullanmışlardır. Birinci yöntem de topluluğu derinlemesine arayan derinlik öncelikli arama (depth-first manner) yaklaşımını, ikinci yöntem de ise diğer yöntem ise en iyi öncelikli (best-first manner) arama yaklaşımını kullanmışlardır.

Topluluk arama problemi her ne kadar hem akademisyenlerin hem de endüstrinin dikkatini çekse de öznitelikli graf yapıları üzerine çok sayıda topluluk arama yöntemi bulunmamaktadır. Mevcuttaki yöntemlerin çoğu girdi parametresi olarak sadece öznitelik ya da sorgu düğümü alabilmektedir ki buda kullanıcıyı kısıtlamaktadır. Aynı zamanda çoğu yöntem çevrimiçi arama yapabilme olanağını sağlamamaktadır. Önerdiğimiz yöntemde her düğümün sadece kendi öznitelikleriyle bağlı olduğu yeni bir graf temsili oluşturularak ağ topolojisine içeriksel boyut getirmekteyiz. Böylelikle öznitelik verisini göz ardı eden graf topoloji tabanlı topluluk arama algoritmalarının da öznitelikli ağlarda kullanılabilen temsilini önermekteyiz. Önerdiğimiz yöntemde, kullanıcılara hem öznitelik verisi hem de sorgu düğümünü ya da hem öznitelik hem de sorgu düğümünü beraber sorgulayabilme imkânını sunulmaktadır.

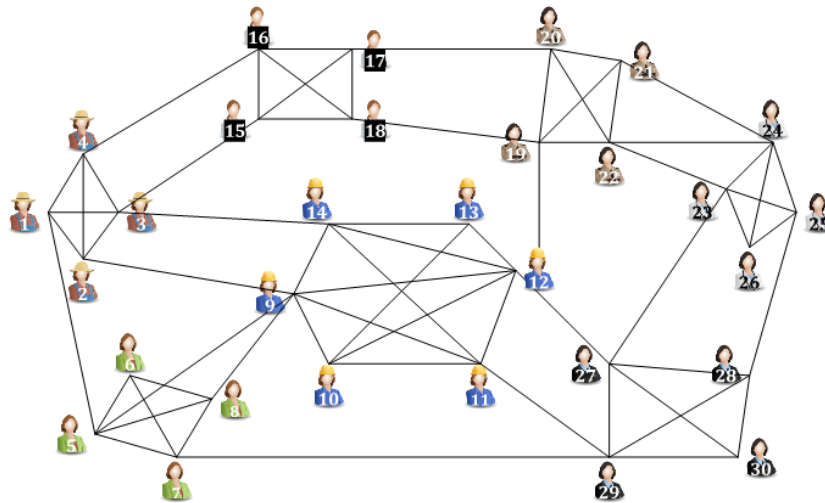
3. İLGİLİ KAVRAMLAR VE TOPLULUK ARAMA ALGORTİTMALARI

3.1. Graflar ve Topluluklar

3.1.1. Graflar

Doğadaki ve toplumdaki birçok karmaşık sistemi oluşturan nesne ve nesnelerin arasındaki bağlantılar graf (ağ) olarak tanımlanabilir. Graflar farklı türdeki veriyi temsil edebilmekte kullanılan güçlü modellerdir. Örneğin metin belgeleri gibi yapılandırılmamış veriler, XML ve veri tabanı gibi yarı yapılandırılmış veriler ve ilişkilendirilmiş veri tabanları yapılandırılmış veriler için örnektir. Bu örnekler için ağ yapılarındaki nesneler graf yapısındaki düğümlere $v \in V$ karşılık gelecek şekilde temsil edilebilirler. Bu temsilde nesnelerin arasındaki bağlantı ve ilişkiler ise graftaki kenarlar $e \in E$ ile modellenenerek $G = (V, E)$ grafi oluşturulur. (Huang vd. , 2019).

Sosyal ağların gerçek hayattaki uygulamalarına örnek olarak arkadaşlık tabanlı uygulamaların oluşturduğu ağlar, e-posta ve işbirliği ağları örnek olarak verilebilir (Bedi ve Sharma. , 2016). Şekil 3.1.'de örnek bir graf yapısı verilmiştir. Bu yapıda farklı meslek grubundaki bireylerin arasındaki bağlantılar görülmektedir.

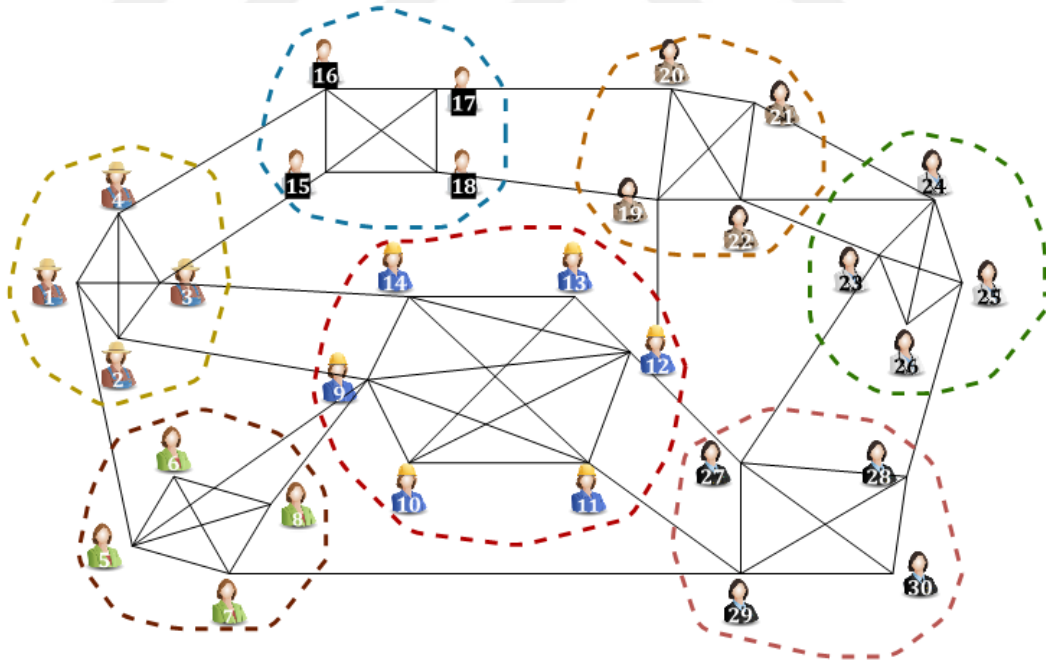


Şekil 3.1. Örnek bir graf yapısı

Bir sosyal ağda, ağ yapısında bulunan n sayıdaki kullanıcı temsil eden n sayıdaki düğümden oluşan yapı $G(V, E)$ grafi olarak temsil edilir. G grafında düğümler V , kenarlar ise E ile ifade edilir. a ($a \in V$) düğümü ile b ($b \in V$) düğümü arasında eğer bir bağlantı varsa bu bağlantı grafın e ($e \in E$) kenarı olarak temsil edilir. G grafındaki a ve b düğümleri arasında bir kenar olması durumunda e kenarının değeri 1, diğer durumda 0 olmaktadır.

3.1.2. Topluluklar

Bir topluluk bulunduğu veri seti içerisinde diğer varlıklara kıyasla birbirine daha yakın bir varlık grubu olarak tanımlanır. Esasen bir topluluk aslında bir gruptur ve bir toplulukta grup içindekiler, grup dışındakilere göre birbiri ile daha sıkı bir etkileşim içindedirler. Ya da başka bir ifade ile birbirleri ile daha çok benzer özellik göstermektedirler. Bir grubun varlıkları arasındaki yakınlık veya benzerlik uzaklık ölçüleri ile ölçülebilmektedir.



Şekil 3.2. Örnek bir graf yapısı üzerinde yer alan toplulukların gösterimi

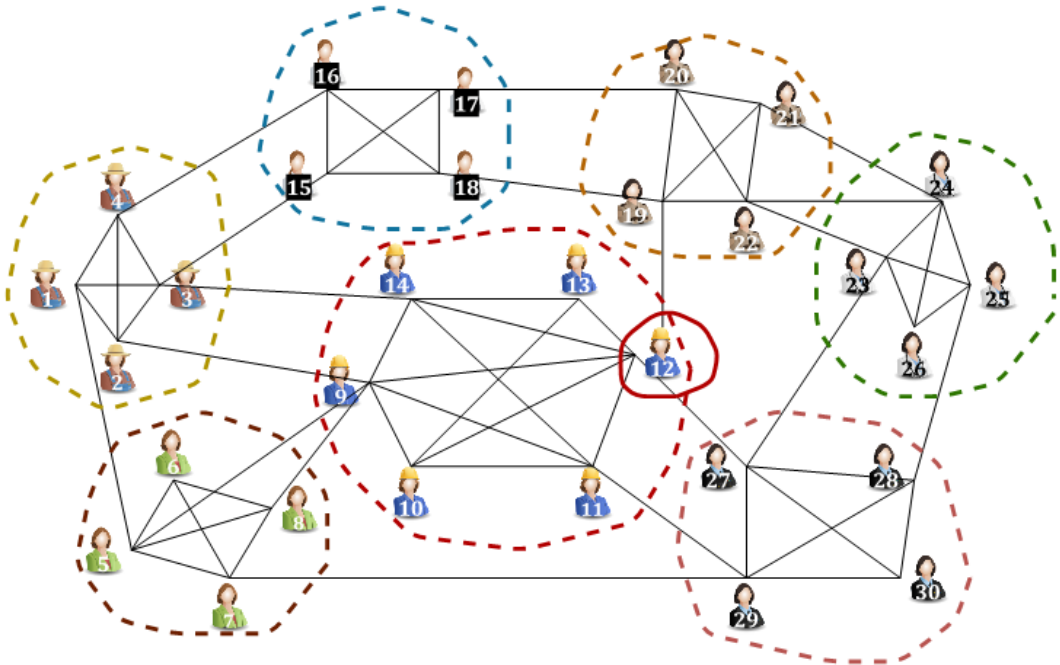
Sosyal ağlardaki topluluklar, ağlardaki düğüm kümelerine benzemektedirler. Örneğin Şekil 3.2.'de bir $G(V, E)$ grafındaki bir v ($v \in V$) düğümü tarafından

temsil edilen bireyler ve yer aldıkları topluluklar örneklendirilmiştir. Esasen her düğüm kümesi, G grafında bulunan bir alt grafi temsil etmektedir (Bedi ve Sharma, 2016).

3.2. Topluluk Arama Problemi

Topluluk arama problemi Np-zor problemlerdendir. Problemin çözümü için ağ yapıları anlaşılabilirliği ve analizi kolaylaştırmak için graflar ile temsil edilmektedir. Bir $G = (V, E)$ grafi için bir ve ya birden fazla sorgu düğümü q ($q \in V$) verildiğinde, topluluk arama algoritması q düğümünde içinde yer aldığı toplulukları bulmayı amaçlar.

Şekil 3.3.'de topluluk arama problemi için örnek bir sorgu düğümü olarak düğüm 12 gösterilmiştir. Topluluk arama algoritması şekilde de görüldüğü gibi, sorgu düğümü olan 12. düğümü de içeren ve düğümün yoğun bağlantılı olduğu diğer beş düğüm olan 9. ,10. , 11. , 13. ve 14. düğümleri içeren tek bir topluluğu sonuç olarak keşfedebilir.



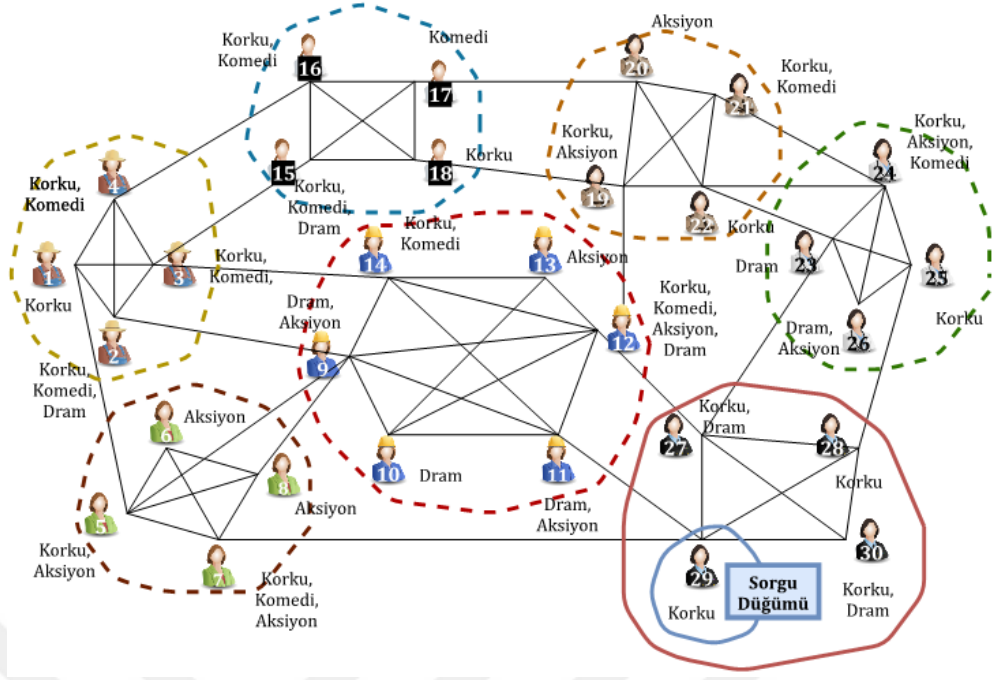
Şekil 3.3. Topluluk arama problemi için örnek bir sorgu düğümü

Bir ağdaki farklı düğümler tarafından tanımlanan topluluklar oldukça farklı yapıya sahip olacağından dolayı sorgu düğümü ile topluluk arama, sorgulama yapan kullanıcıya anlamlı cevaplar üretmesinden dolayı kullanıcı merkezli ve kişiselleştirilmiş arama olanağı sunmaktadır (Huang vd. , 2019).

3.3. Öznitelikli Ağlarda Topluluk Arama

Birçok gerçek dünya ağı, ağdaki nesneyi ya da düğümü karakterize eden öznitelik verisine sahiptir. Örneğin bir sosyal ağ yapısındaki düğümler kullanıcıya ait isim, ilgi alanları, okuduğu okul bilgisi, iş yeri bilgisi, becerileri gibi bilgileri içeren öznitelik verisine sahip olabilir. Graf yapısındaki ağın topolojisine ek olarak sorgulama yapan kullanıcılar özniteliklerle ilgili toplulukları aramak isteyebilir. Öznitelikli topluluğun ağ yapısı homojen özniteliklere sahip bir birleri ile yoğun bağlantılı bir grup düğümden oluşmaktadır.

Şekil 3.4’de öznitelikli ağlar için örnek bir homojen yapıya sahip olan alt topluluklar gösterilmiştir. Burada düğümlerinin ilişkili olduğu öznitelik verisi graf yapısı üzerinde sunulmuştur. Graf yapısı üzerinde üyelerin(düğümelerin) ilgi alanları (korku, komedi, aksiyon ve dram) olan film türleri ve aralarındaki ilişkiler verilmiştir. Bu graf temsili üzerinde yer alan üye “29” sorgulanır ise “korku” özneliğine sahip homojenlik bakımından yüksek bir alt graf topluluğu olarak 29. düğümünde içerisinde yer aldığı 27. , 28. ve 30. düğümün oluşturduğu topluluk keşfedilmesi beklenmektedir. Bu alt graf yapısı şekildeki ağ yapısı üzerinde daire içerisinde gösterilmiştir.



Şekil 3.4. Öznitelikli ağlar için örnek bir homojen alt topluluk

3.4. Topluluk Arama Algoritmaları

Toplulukların önemi nedeniyle, büyük ölçekli ağ yapılarında toplulukların nasıl etkili ve verimli bir şekilde keşfedileceği, büyük veri çağında araştırmacılar için önemli bir araştırma konusudur. Topluluk arama problemi NP-zorlukta olan bir problemdir ve problemin çözümü için önerilen yöntemlerin yüksek verime sahip olması gerekmektedir (Huang ve Lakshmanan, 2017).

Birçok uygulamada yapılan sorgulamalar gerçek zamanlı olarak yapılabilir. Bu nedenle yapılan sorgulamalara gerçek zamanlı yanıt verebilir olması gerekmektedir. Büyük ağ yapılarında çok fazla sayıda sorgu düğümü bulunmakta, bu nedenle de önerilen çözümün yüksek düzeyde ölçeklenebilir olması gerekmektedir. Büyük graf yapılarında kullanıcılar genellikle ağ yapısındaki tüm topluluklardan ziyade bazı belirli kullanıcı toplulukları ile ilgilenmektedir. Bu nedenle kullanıcılara sorgu düğümü belirleyebilme imkânı sağlanarak kişiselleştirilmiş arama olanağı verilmelidir. Arama sonucunda bulunan topluluk yapılarında bulunan düğümlerin birbirine uyumlu bir şekilde bağlı olması gerekmektedir. Bu durum bulunan tüm topluluk yapılarının

kalitesini belirtir. Gerçek dünya ağlarının pek çoğu dinamik değişimler göstermektedir. Önerilen yöntemin dinamik ağ yapılarını destekleyebilir olması gerekir.

Yukarıda belirtildiği gibi topluluklarının etkili ve verimli bir şekilde keşfedilebilmesi amacıyla araştırmacılar tarafından topluluk arama adı verilen problem önerilmiştir. Genel olarak topluluk aramanın amacı, yüksek düzeyde kaliteli yapıya sahip olan topluluk yapılarını çevrimiçi olarak aramaktır. Pek çok topluluk arama problemini adresleyen çözüm yöntemleri önerilmiştir. Bu yöntemler arasındaki en temel farklar, farklı graf yapıları ile ilgilenir olmaları ve ağ yapısındaki toplulukları farklı şekilde formülize etmeleridir. Önerilen yöntemleri, graf yapılarına göre, graflar üzerinde arama yapan topluluk arama yöntemleri ve öznitelikli ağlar üzerinde arama yapan topluluk arama yöntemleri olarak ayırabiliriz. Bu bölümde, çalışmada ele alınan öznitelikli ağlar üzerinde kullandığımız k -çekirdek ve k -tras tabanlı topluluk arama yöntemlerini detaylı olarak açıklanacaktır. Ardından diğer topluluk arama yöntemlerinden k -klik ve k -kenar yöntemlerine değinilecektir.

3.4.1 k -çekirdek tabanlı topluluk arama

Bir ağ yapısının yapısal özelliklerini ortaya çıkarmamıza yardımcı olacak yöntemlere ve algoritmalara sahip olmak çok önemlidir. Son yıllarda birçok farklı alandan bilim adamları çeşitli algoritmalar önermektedir. Bunlardan en yaygın kullanılan algoritmalarından biriside k -çekirdek tabanlı topluluk arama algoritmasıdır. Algoritma ilk olarak 1983 yılında Seidman (Seidman,1983) tarafından sunulmuş ve kısa sürede yalınlığı ve geniş uygulanabilirliğinden dolayı ağ yapısının tespitinde kullanılan en popüler uygulamalardan biri haline gelmiştir. Bu algoritma, biyoloji, ağ bilimi, bilgisayar bilimi ve sosyal bilimler gibi pek çok alandaki ağ yapılarının en yoğun bağlantıya sahip bölümlerini bulmak veya da bağlantı yerlerindeki bilgileri edinmek için yaygın olarak kullanılmaktadır (Kong vd. , 2019).

k -çekirdek yöntemi, ağ yapısındaki bağlantı yoğunluğunu bulma konusunda k -tras, k -kenar ve k -klik tabanlı topluluk arama yöntemlerine kıyasla zayıf kalmakta iken etkinlik açısından bu yöntemlerle kıyaslandığında daha yüksek etkinliğe sahiptir. Algoritmanın en büyük avantajlarından biri ise doğrusal zaman karmaşıklığına sahip olmasıdır (Fang vd. , 2019).

Verilen bir G grafi için k -çekirdek topluluk arama yöntemi, sorgu girdisi olarak kullanıcı tarafından belirlenebilen bir sorgu düğümü q almaktadır. Sorgu çıktı olarak sorgu düğümü olan q' nunda içerisinde bulunduğu her bir düğüm derecesinin minimum k tam sayısı kadar olduğu yani düğümlerin her birinin en az k tamsayısı kadar bağlantıya sahip olduğu en büyük alt graf yapısını $S \subseteq G$ bulmayı amaçlar. Matematiksel olarak ifadesi ise $\forall$ düğüm $v \in V(S)$ için $deg_S(v) \geq k$ olmalıdır. Bir düğüm $v \in V'$ 'ye ait ise ve çekirdek numarası $\varphi(v)$ ise v düğümünü içeren en yüksek çekirdek dereceli alt grafın derecesine eşittir. Bu durumda, eğer bir düğüm bir k -çekirdek alt grafının içerisinde bulunuyor fakat bir $(k+1)$ -çekirdek alt grafının içinde bulunmuyor ise, $\varphi(v) = k$ olur.

k -çekirdek topluluk arama yönteminde (Barbier vd. , 2015; Fang vd. , 2020), bir G grafindaki toplulukları aramak için sorgu parametresi olarak k ve q parametreleri ile birlikte G grafında bulunan tüm düğümler için k -çekirdek numarasını hesaplayan bir çekirdek ayrıştırma (core decomposition) işlemi kullanır. Şekil 3.5.'de k -çekirdek ayrıştırma işlemine dair algoritma sunulmaktadır. Algoritmanın zaman ve alan karmaşıklığı $O(m+n)$ 'dir. k -çekirdek ayrıştırma işlemine göre algoritma ilk olarak graftaki tüm düğümlerin düğüm derecelerini hesaplar.

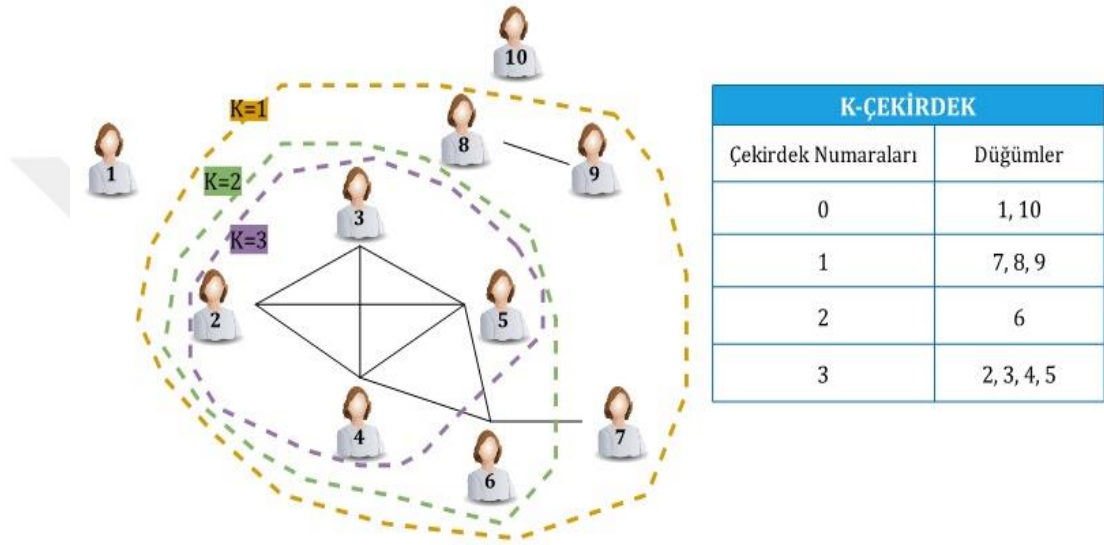
Çekirdek Ayırıştırma Algoritma
Sorgu Girdisi: $G = (V, E)$ Sorgu Çıktısı: $\varphi(v)$ her biri için $v \in V$ Adım 1: G grafindaki tüm düğümler derecelerine göre artan sırada sıralanır. Adım 2: Eğer G grafindaki tüm düğümler 2.1. : d, G'deki minimum düğüm derecesi olsun; 2.2. : derecesi en fazla olan düğüm d bulunmakta ise 2.2.1. : d değeri $\varphi(v)$ olarak atanır ($\varphi(v) \leftarrow d$); 2.2.2. : v ve v'ye bağlı tüm kenarlar G grafindan kaldırılır. 2.2.3 : G grafinda kalan tüm düğümlerin dereceleri hesaplanır ve artan sırada sıralanır. Adım 3: Döndür : $\varphi(v)$ her biri için $v \in V$

Şekil 3.5. Çekirdek Ayırıştırma Algoritması

Düğüm derecesi düğüme bağlı olan komşu düğümlerin sayısıdır. Graftaki tüm düğümlerin düğüm derecesi hesaplandıktan sonra düğümler derecelerine göre artan sırada sıralanır. Sıralamaya göre en düşük düğüm derecesine sahip olan düğüm, örneğin d , düğüm v 'nin çekirdek numarası $\varphi(v) = d$ olarak belirlenir. Ve düğüm v ve düğüm v 'ye bağlı kenarlar G grafindan kaldırılır. Daha sonra grafta kalan tüm düğümler için tekrardan düğüm derecesi hesaplanarak düğümler, derecelerine göre artan sırada tekrardan sıralanırlar. Bu işlem yinelemeli olarak G grafindaki tüm düğümlere ait çekirdek numaraları hesaplanıncaya kadar devam eder. Bu işlem sayesinde düğüm v 'ye bağlı olan tüm düğümler içinde çekirdek numarası işaretlenmiş olur. Bir q düğümü sorgulandığında, q' ya bağlı olan her bir düğümün $v \in V$ için $\varphi(v) \geq k$ koşulunu sağlanıp sağlanmadığı kontrol edilir. Koşul sağlanıyor ise q 'yu da içeren bir dizi k -çekirdek toplulukları G grafinın alt grafları olarak keşfedilir.

Şekil 3.6.'da verilen k -çekirdek tabanlı arama algoritması için örnek graf yapısı üzerindeki çekirdek toplulukları k derecesine göre farklı renklerdeki elipslerle işaretlenerek gösterilmektedir. Bu yöntemde k -çekirdek topluluğunun

bulunabilmesi için sorgulanan düğüm ile bu düğüme bağlı olan düğümlerin k -çekirdek derecelerinin en az k derecede bağlantıya sahip olması gerekmektedir. Verilen graf yapısında düğüm 1 ve 10' un dereceleri 0 eşittir çünkü bu düğümlerin hiçbir düğüm ile kenar bağlantıları bulunmamaktadır. Bu nedenle eğer bu düğümler için k değerine 1 veya 1'den büyük olan bir tam sayı değeri verilerek sorgulama yapılmak istenir ise bu düğümlerin çekirdek numarası 0' a eşit olduğundan dolayı arama sonucunda hiçbir k -çekirdek topluluğu bulunamaz.



Şekil 3.6. k -çekirdek tabanlı arama algoritması için örnek

Yine başka bir örnek verecek olursak düğüm 7, $k = 3$ için sorgulanır ise bu düğümün derecesi 1 eşittir ve çekirdek numarası 1 olduğu için algoritma hiç bir k -çekirdek topluluğu bulamayacaktır. Fakat $k = 3$ için düğüm 2 sorgulanır ise, k -çekirdek topluluk arama algoritması düğüm 2'nin de içinde bulunduğu çekirdek numarası en az 3'e eşit olan birbirlerine bağlı düğümlerden oluşan bir alt graf bulmaya çalışacaktır. Graf yapısında bu koşulu sağlayan alt graf topluluğu ise 2, 3, 4 ve 5 düğümlerini içeren alt graf yapısıdır. k -çekirdek tabanlı topluluk arama yöntemi düşük hesaplama karmaşıklığına sahip olup sosyal ağ analizlerini ve etkileşimlerini analiz etmede etkili bir yöntemdir. Büyük ağ yapılarına uygulanabilir olmasından dolayı büyük hacimli grafların anlaşılabilirliği için tercih edilmektedir. Bu uygulamanın temel hedefi ağdaki düğümlerin ve

bağlantılar hakkında bilgi edinilmesini sağlayarak ağ yapısındaki yoğun bağlantılı alt topluluk yapılarını ortaya çıkarmaktır.

3.4.2 k -tras tabanlı topluluk arama

Bir ağdaki yoğun bağlantılı yapıları keşfeden topluluk tespiti konusunda önerilen bir diğer algoritma ise k -tras tabanlı topluluk arama yöntemidir. Düğümlerin özniteliklerle ilişkili olduğu yönsüz bir G grafi ve sorgu düğümü q verildiğinde sorgu düğümünün de içerisinde bulunduğu düğümlerin birbirlerine benzer özellikler taşıdığı ve ilişkili olduğu alt topluluk yapıları keşfedilir. Yapı bütünlüğü açısından k -tras, k -çekirdek tabanlı topluluk arama yöntemine kıyasla daha anlamlı topluluklar üretebilmektedir.

k -tras tabanlı topluluk arama yaklaşımında her kenarın en az $k - 2$ üçgen yapısı içerisinde yer aldığı en büyük alt grafi $S \subseteq G$ keşfetmek amaçlanır. Matematiksel olarak ifade edilecek olursa $\forall$ kenar $e \in E(S)$ için $\sup(e, S) \geq (k - 2)$ olmalıdır. Bu ifade de yer alan $\sup(e, S)$ alt graf yapısındaki e kenar desteği olarak tanımlanır. S ifadesi ise alt graftaki e kenarını içeren üçgen sayısına karşılık gelmektedir.

Verilen bir $G(V, E)$ grafindaki üçgen yapısı $\{(u, v), (u, w), (v, w)\} \in E$ ve $\{u, v, w\} \in V$ olmak üzere üç düğümü birbirine bağlayan üç kenardan oluşan üçgen biçimdeki alt graf T_{uvw} olarak ifade edilir. Bu durumda, örneğin ve T_{qvw} ve T_{quv} olmak üzere iki farklı üçgende e kenarını içermekte ise kenar desteğinin değeri $\sup(e(q, v)) = 2$ olur. Bir alt graftaki S 'in tras derecesi $\tau(S)$, içerdiği en düşük kenar destek değerine sahip olan kenarlar tarafından belirlenmektedir. Bir kenar e 'nin tras derecesi $\tau(e)$ ise e kenarını içeren en yüksek tras derecesine sahip olan S alt grafının tras derecesine $\tau(S)$ eşittir.

k -tras tabanlı topluluk arama yöntemi (Huang vd. , 2014; Shang vd. , 2017), k ve q parametreleri verildiğinde, her bir kenarın tras derecelerinin hesaplayan, bir tras ayrıştırma (truss decomposition) işlemi uygulanır. Algoritma 3.4.2.1.' de verilmektedir. Bu işlem süresince graftaki her bir kenar $e \in E$ için öncelikli

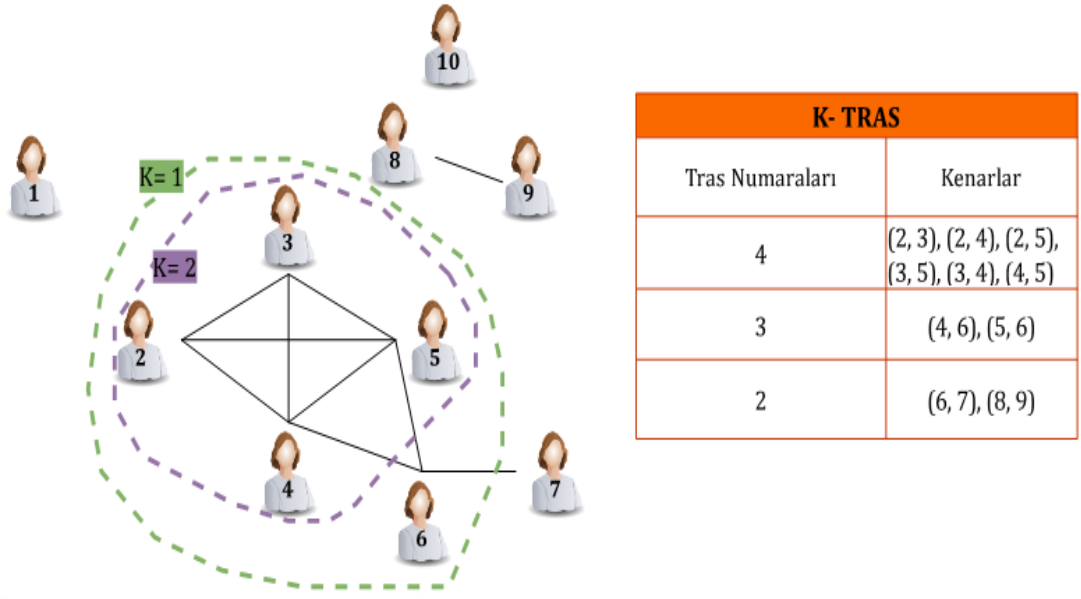
olarak destek değeri $sup(e)$ bulunur. Sonraki süreçte $k=2'$ den başlanarak yinelemeli olarak $sup(e) \leq k - 2$ koşulunun sağlanıp sağlanmadığı kontrol edilerek koşulun sağlanmadığı durumlarda en düşük destek değerine sahip olan kenarlar graftan kaldırılır. Ardından çıkarılan kenarlara ait maksimum tras derecesi hesabı yapılır ve bu değer k olarak belirlenir. Çıkarılan kenarları içeren üçgen yapılarındaki diğer kenarlara ait olan kenar destek değerleri de bir azaltılır. Bu yinelemeli işlem, $k - 2$ değerinden daha küçük ve eşit kenar destek değerlerine sahip kenarlar kaldırılincaya kadar devam etmektedir.

Tras Ayırıştırma Algoritması
Sorgu Girdisi: $G = (V, E)$ Sorgu Çıktısı: $\tau(e)$ her biri için $e \in E$ Adım 1: $k=2$ başlangıç değeri olarak atanır; Adım 2: Graftaki her bir kenar $e \in E$ için destek değeri $sup(e)$ hesaplanır; Adım 3: Tüm kenarlar kenar destek değerleri artan düzende sıralanır; Adım 4: Her bir kenar $e \in E$ için $sup(e) \leq k - 2$ koşulu kontrol edilir; 4.1. : En düşük destek değerine sahip $e(u, v)$ kenarı kaldırılır; 4.2. : Çıkartılan kenarla bağlantılı olan üçgen yapılarındaki diğer kenarların kenar destek değerleri 1 azaltılır: $(u, v, w) \in E$; $sup(u, w) \leftarrow sup(u, w) - 1$; $sup(v, w) \leftarrow sup(v, w) - 1$; 4.3: G grafından kaldırılan e kenarından sonra $\tau(e) = k$ olarak belirlenir; Adım 5: Eğer G deki tüm kenarlar kaldırılmadı ise; Adım 6: $k \leftarrow k + 1$; Adım 7: 4. Basamağa ger dönülür; Adım 8: Döndür $\{\tau(e) e \in E\}$;

Şekil 3.7. Tras Ayırıştırma Algoritması

Algoritma sonuç olarak graftaki her bir e kenarı için bağ derecesi $\tau(e)$ değerini hesaplamış olur. Ardından her bir düğüm $v \in V$ için, düğüm v 'nin komşuları kenar tras değerine göre azalan bir sıralama işlemi uygulanarak kenar tras değeri k 'ya eşit olan işaretlenir. Böylelikle düğüm v 'ye bağlı olan diğer kenarlar tras dereceleri belirlenmiş olur. Sorgu parametresi olarak verilen bir q düğümü sorgulanır ise, algoritma verilen sorgu düğümünden başlayarak q' ya bağlı her bir kenar $(q, v) \in E$ için $\tau(q, v) \geq k$ koşulunu sağlayıp sağlamadığına bakılır. Bir başka ifadeyle sorgulanan q düğümünün de içinde yer aldığı üçgen yapıya sahip topluluklar aranır. Arama işlemi q' yu içeren tüm bağlı kenarlar için koşulların sağlanıp sağlanmadığı kontrol edilene kadar devam eder. Arama işlemi sonucunda sorgu düğümünün de içinde bulunduğu bir dizi k -tras topluluklar bulunur.

Şekil 3.8.' de verilen graf örneği üzerinde k -tras toplulukları örneklendirilmiştir. Şekil üzerindeki düğüm 1 ve 10 sorgulandığında, algoritma düğüm 1 ve düğüm 10'un içerisinde bulunduğu toplulukları aramaya başlayacaktır fakat bu düğümler hiçbir üçgen yapı içerisinde bulunmamaktadır ve kenar destek değerleri sıfır olduğu için algoritma hiçbir k -tras topluluğu döndürmeyecektir. Düğüm 2 sorgulanır ise bu düğüme bağlı olan kenarlar iki farklı üçgen içerisinde yer aldığı için bu kenarların kenar destek değeri iki ve tras numaraları dört olur. Algoritma bu düğüme bağlı diğer kenarlarında kenar destek değerlerini hesaplar. Düğüme bağlı olan kenarlar 3, 4, 5' in de tras değeri 4 olduğu için algoritma $k=4$ için 2, 3, 4 ve 5'in de içinde yer aldığı tras topluluğu ortaya çıkarır.



Şekil 3.8. k -tras tabanlı arama algoritması için örnek

3.4.3 k -klik tabanlı topluluk arama

Bir k -klik, ağ yapısındaki her düğüm çifti arasında bir kenar bağlantısının olduğu alt graflara verilen isimdir. NP-zor bir problem olarak bilinir. Klik problemi için geliştirilmiş arama yaklaşımları sosyal ağlarda ve sosyal medya da topluluk arama, biyoinformatikte gen ifadesi ve motif keşfi gibi alanlarda kullanılmaktadırlar (Lu vd. , 2017). k -klik yapıları topluluklarda keşfedilen düğümler, birden fazla topluluğa ait olabilir. Bu nedenle bu model örtüşen toplulukların tespiti için yaygın olarak kullanılmaktadır. Problemin çözümü için indeks tabanlı yaklaşımlar önerilmiştir (Yuan vd. , 2019). Bir G grafi verildiğinde alt graf $S \subseteq G$ için yapı G 'deki her düğümün bir k -klik olduğu ve iki düğüm arasında bir kenarın olduğu yapıdır. k -klik topluluğu olan C , düğümler tarafından temsil edilen tüm k -kliklerin birleşimidir. k -klik düğümler arasında yoğun bağlantılı bulunan bir yapıdan oluştuğu için graf yapısı içerisinde k -klik topluluklarını tespit etmek zordur. Bu nedenle γ -yarı- k -klik (γ -quasi- k -clique) gibi k -klik topluluk aramasına daha esnek yaklaşımlar önerilmiştir.



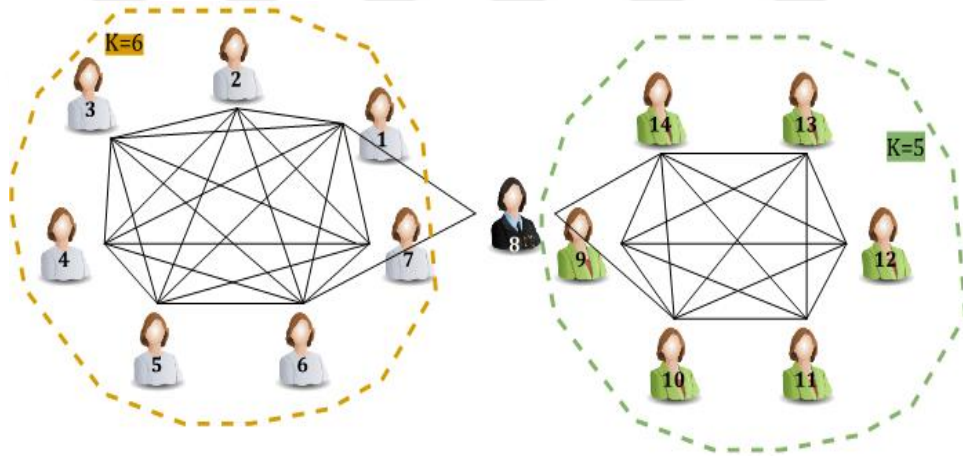
3.4.4 k -kenar tabanlı topluluk arama

27

topluluk tespiti (Zhao vd. , 2012) gibi çeşitli uygulamalarda kullanılabilir.

k -kenar hesaplamasının birçok uygulaması vardır. Örneğin k -kenarlar, sosyal ağ analizlerinde tutarlılık için kullanılabilir (Örneğin facebooktaki engellenen topluluklar). Bileşen yapılarının yüksek bağlantıya sahip hesaplamaları ve tanımlamaları için kullanılabilir. Gen dizilimlerinin analiz edilmesi için biyoloji alanında kullanılabilir. DBLP ağı gibi ağ yapısında benzer araştırmalara sahip araştırmacı gruplarını bulmak için de kullanılabilir.

Örneğin; Şekil 3.10.' da k -kenar tabanlı topluluk arama için bir örnek gösterilmiştir. k -kenar alt grafi olarak gösterilen topluluklara bakıldığında, $k=5$ ve $k=6$ için yapıdan bir kenar çıkarıldıktan sonra yapı uyumluluğun hala korunduğu gözlemlenmektedir.



Şekil 3.10. k -kenar tabanlı arama için örnek

Kenar ayrıştırma için basit bir çözüm, tüm k -kenar 'ı hesaplamak için mevcut k -kenar hesaplama algoritmasını uygulamaktır. Ancak bu yaklaşım büyük graf yapılarına uygulanamaz. Çünkü mevcut tüm k -kenar hesaplama algoritmaları, algoritmalarda kullanılan karmaşık veri yapıları nedeniyle yüksek oranda bellek kullanmaktadır. Büyük graf yapılarına uygulanamamasının bir diğer nedeni ise olası k değerlerinin sayısı çok büyük olabilir ve bu da her k değerini ayrı ayrı

düşündüğümüzde yüksek bir hesaplama maliyetine neden olur (Yuan vd. , 2017).

3.5. Topluluk Arama Algoritmalarının Karşılaştırılması

Bu bölümde Bölüm 3.4.'de incelenen algoritmalar için literatürde yapılan karşılaştırmaları sunuyoruz. Fang ve arkadaşlarının (Fang vd. , 2020) yaptıkları çalışmada tutarlılık ve hesaplama verimliliği açısından önerilen topluluk arama yöntemleri karşılaştırılmış ve yapı bütünlüğü açısından en iyi sonuçları k -klik topluluk arama yönteminin verdiği belirtilmiştir. Bunun nedeni ise k -klik yapısında bulunan topluluklardaki her düğüm graftaki diğer tüm $(k-1)$ düğümlere bağlıdır. Böylelikle k -klik topluluklarının topluluk içerisindeki kenar yoğunluğu yüksektir. Bu yöntemi k -tras yöntemi takip etmektedir. K -tras yönteminde ise her düğüm çifti ortak $(k-2)$ kenara sahiptir. Bu yöntemleri ise sırası ile k -kenar ve k -çekirdek yöntemleri takip etmektedir (Akiba vd. , 2013). Hesaplama verimliliği açısından ise tam tersi durum söz konusudur. Fang ve arkadaşları yaptıkları çalışmada (Fang vd. , 2020) zaman ve uzay karmaşıklığı açısından yöntemleri karşılaştırmıştır ve düğümlerin komşuluklarını dikkate alan k -çekirdek yönteminin $O(m+n)$ zaman ve $O(m+n)$ uzay karmaşıklığına sahip olduğunu belirtmişlerdir. Bu nedenle yöntemler arasında en verimli k -çekirdek yönteminin olduğunu görmüş ve bu yöntemi sırası ile k -kenar, k -tras ve k -klik yöntemlerinin takip ettiklerini tespit etmişlerdir. Bir başka ifade ile yapısal uyumluluk açısından daha iyi olan bir yöntem hesaplama verimliliği açısından daha maliyetlidir. Buna istinaden çalışılan graf yapılarında; küçük ve orta ölçekli graf yapısı ile çalışılıyorsa k -klik ve k -tras yöntemlerinin yüksek başarı gösterebileceğini bunun yanı sıra makul düzeyde hesaplama maliyeti ile yapı tutarlılığı olduğunu tespit etmişlerdir. Büyük ölçekli graf yapılarında ise k -kenar ve k -çekirdek yöntemleri daha doğru seçim olabilir. Çünkü hesaplama verimlilikleri diğer yöntemlere kıyasla daha iyidir. Eğer sonuç olarak daha alt graf olarak kümelenmiş topluluklar keşfedilmek istenir ise yapı bütünlüğü nedeni ile k -tras yöntemi tercih edilebilir. Fakat bu yöntem bipartite graf yapıları gibi ikili yapıya sahip özel graflar da kullanılamayabilir. Çünkü bu graf yapıları topolojik özellikleri itibari ile üçgen yapıya sahip değildir.

Çevrimiçi sorgu algoritmaları için, sorgu süresi açısından karmaşıklığı karşılaştırmış ve artan şekilde sırası ile k -çekirdek, k -kenar, k -tras ve k -klik yöntemlerini sırlamışlardır. Sonuç olarak k -çekirdek tabanlı algoritma en yüksek ölçeklenebilirliğe sahip olup k -klik en düşük ölçeklenebilirliğe sahiptir. Önerilen tüm algoritmalar dinamik graf yapılarını desteklemektedir (Batagelj ve Zaversnik, 2003; Wang ve Cheng, 2012; Fang vd. , 2020) .

3.6. Topluluk Arama Problemi ve Topluluk Tespit Etme Problemi Arasındaki Farklar

Topluluk arama problemi ve topluluk tespiti problemi benzer hedeflere sahip olmaları nedeniyle literatürde oldukça ilişkilidirler. Ancak bazı temel farklılıklar bulunmaktadır. Topluluk arama probleminde topluluk tanımları kullanıcı tarafından verilen sorgu parametrelerine dayanırken topluluk tespitinde graftaki tüm topluluklar için aynı küresel kriterler kullanılmaktadır. Topluluk arama da kullanıcı özelinde kişiselleştirilmiş arama yaparak hedeflenen topluluk yapıları bulunurken topluluk tespitinde graf yapısındaki tüm topluluklar tespit edilir. Topluluk arama yöntemi ile çevrimiçi verimli bir sorgulama yapılabilirken topluluk tespiti çözümleri genellikle zaman alıcıdır ve büyük graf yapılarında ölçeklenebilme problemine sahip olabilir (Fang vd. , 2020). Topluluk arama problemi için çözüm yaklaşımları genellikle dizin yapıları ile desteklenerek dinamik graf yapılarının ele alınabilmesi için imkân sunmaktadır. Böylece, topluluk tespiti problemi belirtilen faktörlere göre topluluk arama problemi istenilen çözümleri daha iyi sağlayabilir (Porter vd. , 2009).

4. ARAŞTIRMA BULGULARI VE TARTIŞMA

4.1. Önerilen Graf Yapısı

Ağ yapısı, nesnelerin graf yapısındaki düğümlere, nesneler arasındaki ilişkilerin de kenarlara denk geldiği bir $G(V, E)$ graf yapısı ile temsil edilir. Öznitelikli ağlar, ağ topolojisini temsil eden graf yapısının yanı sıra aynı zamanda düğümün karakteristik özelliğini tanımlayan öznitelik verisine de sahiptir. Örneğin bir blogun web sayfası ağında düğümler web sayfalarına denk gelirken yönlendirilmiş kenarlar web sayfaları arasındaki hiper bağlantılara denk gelmektedir. Web sayfasında bulunan kelimeler, başlıklar, konular, etiketler ağın içeriksel öznitelik verisini temsil etmektedir. $G(V, E)$ graf yapısı $V = \{v_1, v_2, \dots, v_n\}$ düğümler kümesi ve $E = (V \times V)$ kenarlar kümesine sahiptir. Düğüm 1 ve 2 arasında bir bağlantılı ilişki durumu söz konusu ise e_{12} kenar değeri 1, diğer durumda 0'dır. Öznitelikli ağ yapısında ise G grafi ile birlikte, her bir düğümün sahip olduğu öznitelikleri temsil eden bir $F = \{f_1, f_2, \dots, f_n\}$ öznitelik kümesi bulunmaktadır. Bu temsilde, f_v, v düğümüne ait öznitelikleri içeren alt bir öznitelik kümesidir. Graf ve öznitelik verisi ayrı bir veri yapısına sahip olduğundan, geleneksel topluluk arama yöntemleri ağın öznitelik verisi olan F veri yapısına girdi olarak alamamakta ve bu bilgiyi göz ardı etmektedir.

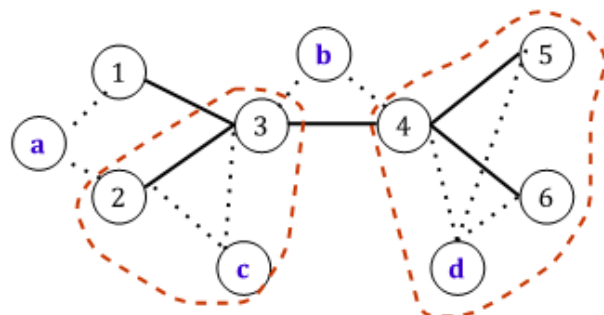
Bu çalışmada, ağın içeriksel verisi olan F öznitelik kümesinin ağın yapısal verisi olan G grafına entegre edildiği yeni bir graf temsili önerilmektedir. Bu doğrultuda, F öznitelik kümesinde bulunan her bir öznitelik bir düğüm olarak ele alınır ve G graf yapısı içeriksel düğüm olarak isimlendirdiğimiz öznitelikler ile genişletilir. Önerilen bu graf yapısı $G' = (V', E')$ olarak temsil edilmektedir. Düğüm kümesi V' yapısal düğümler ve içeriksel düğümler olmak üzere iki farklı kümeden gelen düğümlerin birleşiminden $V \cup F$ oluşmaktadır. Kenar kümesi olan E' ise yapısal graftan gelen E kümesinin yanı sıra, yapısal düğümler ve o düğümlere ait öznitelikleri temsil eden içeriksel düğümler arasında oluşturulmuş yeni bağlantıları da içermektedir. Bir v düğümü eğer u özniteliğini içermekte ise v yapısal düğümü ile u öznitelik verisi arasında bir bağlantı oluşturulur. Oluşturulan bu bağlantı yapısal düğüm ile içeriksel düğüm

Şekil 4.1’de düğümler ve öznitelikler listelenerek ağın orijinal yapısı ve önerilen graf yapısı örneklendirilmiştir. Şekil 4.1 (a)’da 6 adet düğüme karşılık gelen öznitelik verisi liste olarak verilirken Şekil 4.1 (b)’de bu altı düğüm ve öznitelikleri arasındaki bağlantı orijinal graf temsili olarak gösterilmiştir. Şekil 4.1 (c)’de ise altı adet yapısal düğümü temsil eden graf nesnesi ile bu nesnelere ait dört farklı özniteliği temsil eden içeriksel düğümleri içeren yeni bir graf temsili sunulmaktadır. Yapısal düğümler ile düğümlerin sahip olduğu öznitelikleri arasında doğrudan bir bağlantı(kenar) oluşturulmuştur. Örnek olarak yapısal düğüm 2 öznitelikleri(içeriksel düğümler) a ve c ile yapısal düğüm 3 ise özniteliksel düğümler b ve c ile doğrudan kenar bağlantısına sahiptir. Önerilen graf yapısı ile mevcut yapıdaki orijinal graf yapısına yeni alt graf yapıları dâhil edilmektedir. Örneğin şekil 4.1 (c)’de görüldüğü gibi eklenen yeni içeriksel düğüm c düğümden dolayı düğüm 2 ve düğüm 3’ü içeren yeni bir üçgen yapı grafa eklenmiştir.

(a) - Döğümler ve Öz nitelikler

Düğümler	Öz Nitelikler
1	a
2	a, c
3	b, c
4	b, d
5	d
6	d

(c) - Önerilen Graf



32

Yukarıdaki terminolojiler kullanılarak, topluluk arama problemi, önerilen öznitelikli graf yapısı $G' = (V', E')$ ve sorgu parametresi olarak q düğümü verildiğinde, q düğümüne ya da özneliğine ait $C'(q)$ topluluğunu arama olarak tanımlanır. Topluluk arama probleminde, ağ topolojisi yapısı üzerinde çalışan geleneksel topluluk arama yaklaşımları önerilen $G' = (V', E')$ graf yapısına uygulanacaktır.

4.2. Veri Kümeleri ve Deneysel Kurulum

Deneyler, beş farklı karmaşık ağ yapısına sahip olan Cora, Citeseer, Blogcatalog, Pubmed ve Facebook veri kümeleri kullanılarak gerçekleştirilmiştir (LINQS, 2021). Çizelge 4.1.'de çalışılan veri setleri için düğüm sayısı $|V|$, kenar sayısı $|E|$ ve öznitelik sayısı $|A|$ 'na ait istatistikler verilmektedir.

Çizelge 4.1. Çalışmada kullanılan beş farklı veri kümesine ait istatistikler

Veri Kümesi	$ V $	$ E $	$ A $	Topluluk Sayısı	Test Edilen Düğüm Sayısı
Cora	2708	5429	1433	7	700
Citeseer	3312	4598	3703	6	600
Blogcatalog	5196	171743	8189	6	600
Pubmed	19717	44324	500	3	300
Facebook	22470	11235	4714	4	400

Cora, Citeseer ve Pubmed atıf ağı olup, her düğümün bir makaleye karşılık geldiği ve kenarların yayınlar arasındaki referansını temsil eden graf yapısı ile modellenmişlerdir. Düğümlerin niteliklerini yayınların anahtar kelimeleri temsil eder. Bu yayınlar ilgili oldukları araştırma konularına göre sınıflandırılmaktadır. Örneğin Cora veri kümesi, vaka tabanlı (case based)", "genetik algoritmalar (genetic algorithms)", "sinir ağları (neural networks), "olasılıksal yöntemler (probabilistic methods) , "pekiştirmeli öğrenme (reinforcement learning)" , "kural öğrenme (rule learning)" ve "teori (theory)" olmak üzere 7 kategoriye sahip topluluk yapısına sahiptir. Pubmedeki yayınlar 500 benzersiz kelimedenden oluşan bir sözlükten alınan Terim Frekans-Ters Belge sıklığı (Term Frequency-Inverse Document Frequency (TF-IDF) vectors) vektörü ile tanımlanan

özelliklere sahiptir. Bu nedenle her alıntı ağı için, bir düğümün özellik boyutu olan D , sözlük boyutuna göre belirlenir. Blogcatalog, blog yazarlarının birbirlerini takip ettikleri çevrimiçi bir topluluk platformudur. Blog yazarları graftaki düğümlere karşılık gelirken yazarlar arasındaki sosyal ilişkilerde kenarlar ile temsilenmektedir. Her bir blog yazarının özellik vektörü, ilgili bloğun açıklamasına göre elde edilir. Blog yazarları, ilgi alanlarına göre önceden tanımlanmış altı topluluk sınıfının birinde sınıflandırılırlar (Xiao vd. , 2021). Facebook ise bir sosyal ağıdır. Bu ağda düğümler sosyal medya kullanıcılarına karşı gelirken kenarlar ise kullanıcılar arasındaki arkadaşlık ağına karşılık gelen graf yapısı ile modellenir.

Tüm topluluk arama yöntemleri tarafından bulunan toplulukların kalitesini değerlendirmek için her bir ağa ait her bir sınıftan rastgele 100 düğüm seçilerek sorgu düğümü olarak kullanılmıştır. Sorgu düğümüne ait olan topluluk C , topluluk arama işlemlerinden sonra elde edilen topluluk C' olmak üzere, elde edilen her bir topluluk için kesinlik ($Prec$) ve duyarlılık (Rec) değerleri hesaplanarak, $F1$ ölçüm değerleri elde edilmiştir. Denklem 1 kullanılarak sorgulanacak düğüm kümesi olan Q ve sorgu düğümü olan q olmak üzere kesinlik ve duyarlılık değerleri hesaplanmaktadır. Her bir veri seti için $F1$ değeri kesinlik ve duyarlılık değerlerinin ortalaması hesaplanarak Denklem 2' de gösterildiği şekilde elde edilmektedir (Huang ve Lakshmanan, 2017).

$$Prec(C_q, C'_q) = \frac{|C_q \cap C'_q|}{|C'_q|}, Rec(C_q, C'_q) = \frac{|C_q \cap C'_q|}{|C_q|} \quad (1)$$

$$Prec_{ort} = \frac{\sum_{q \in Q} Prec(C_q, C'_q)}{|Q|}, Rec_{ort} = \frac{\sum_{q \in Q} Rec(C_q, C'_q)}{|Q|}, \quad (2)$$

$$F1 = \frac{2 * Prec_{ort}(C_q, C'_q) * Rec_{ort}(C_q, C'_q)}{Prec_{ort}(C_q, C'_q) + Rec_{ort}(C_q, C'_q)}$$

4.3. Performansın Değerlendirilmesi

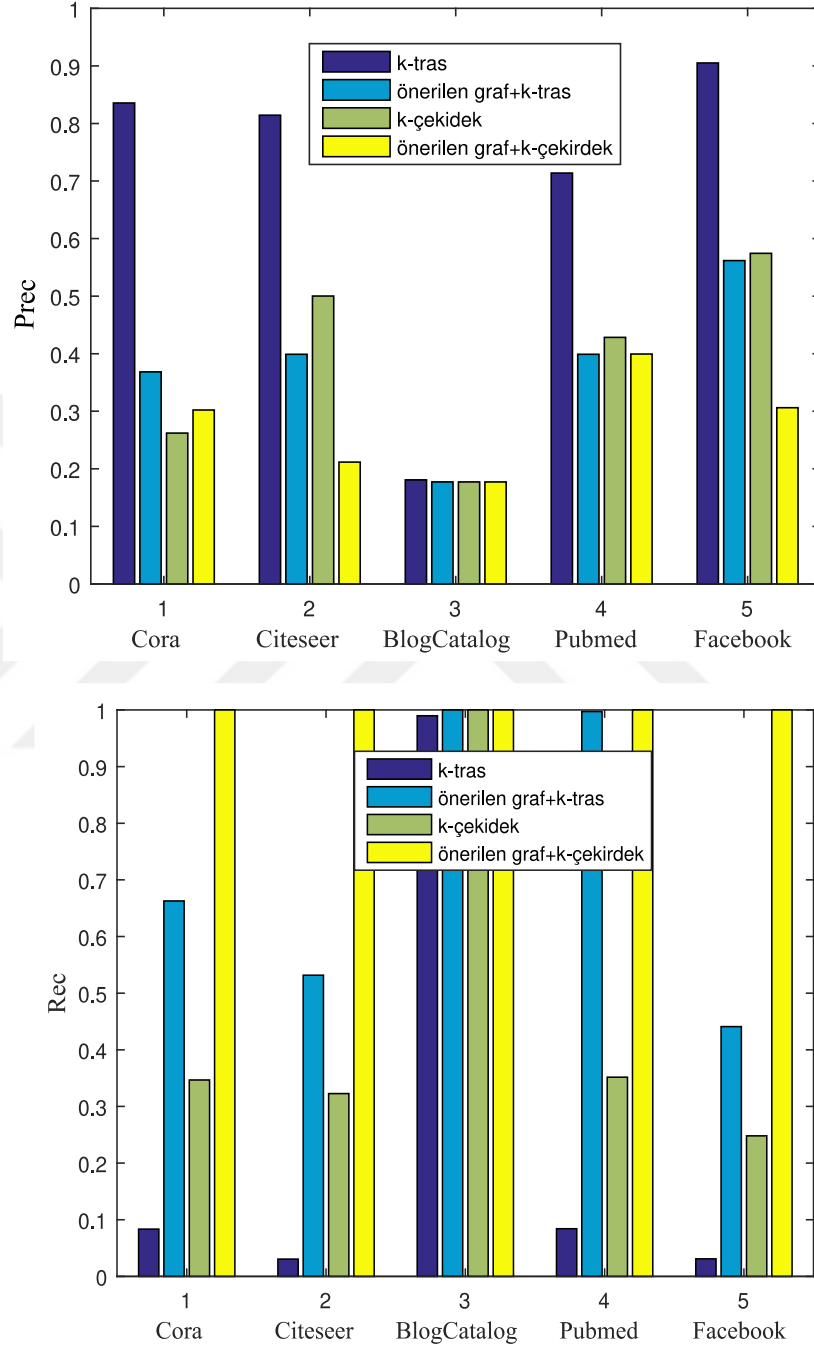
Bu bölümde, bölüm 3.4’de detaylı olarak anlatılan k -tras ve k -çekirdek topluluk arama yöntemlerinin performansı orijinal graf yapısı ve önerilen graf yapısı için analiz edilmektedir. Çizelge 4.3.’de beş farklı veri seti üzerinde çalıştırılan topluluk arama yöntemleri için F-ölçüm sonuçları $k=3$ için sunulmuştur. Bu sonuçlara göre, önerilen graf yapısı üzerinden elde edilmiş olan k -tras sonuçlarının, tüm veri setleri için, orijinal graf yapısı üzerinden elde edilen k -tras performans sonuçlarından daha yüksek olduğu gözlemlenmektedir. Benzer şekilde, k -çekirdek yöntemi önerilen graf yapısı ile çalıştırıldığında da Citeseer veri seti haricinde tüm veri setleri için daha yüksek doğruluk değerleri elde edilmiştir.

Çizelge 4.2. Farklı veri setleri üzerinde yöntemlere ait F-ölçüm sonuçları ($k=3$)

Yöntem	Cora	Citeseer	BlogCatalog	Facebook	Pubmed
k -tras	0.11450	0.05434	0.29812	0.05304	0.12964
Önerilen Graf + k -tras	0.36054	0.29166	0.30113	0.37883	0.56986
k -core	0.27851	0.38035	0.30113	0.34574	0.38623
Önerilen Graf + k -core	0.46411	0.34936	0.30113	0.46882	0.57082

Şekil 4.2’de orijinal graf yapısı ve önerilen graf yapısında kullanılan yöntemlere ait kesinlik (Prec) ve duyarlılık (Rec) değerleri gösterilmektedir. k -tras ve k -çekirdek algoritmaları için orijinal graf ve önerilen graf yapısı için elde edilen kesinlik sonuçları karşılaştırıldığında, neredeyse önerilen graf yaklaşımının çalışılan tüm veri setleri içerisinde daha iyi performansa sahip olduğu görülmektedir. Bu durumun en temel nedeni ise önerilen graf yapısının öznel (içeriksel) düğümler için aynı grupta bulunmayan pek çok düğümle arasında doğrudan kenar bağlantısı oluşturmamasından kaynaklıdır. Önerilen graf yapısı yaklaşımında bir düğüm sorgulandığında, orijinal grafa kıyasla, daha fazla sayıda düğüm içeren topluluklar keşfedilir. Şekil 4.2’de sunulan duyarlılık sonuçlarına göre, k -tras ve k -çekirdek yöntemleri ile yapılan topluluk aramalarında önerilen graf yapısı ile orijinal graf yapısı karşılaştırıldığında önerilen graf yapısının daha yüksek performansa sahip olduğu

gözelemlenmektedir. Önerilen graf yaklaşımı özellikle k-çekirdek yöntemi ile çalıştırıldığında kullanılan tüm veri setleri içerisinde test edilen tüm sorgu düğümleri için %100 doğrulukla toplulukları bulduğu görülmektedir.



Şekil 4.2. Farklı veri setleri için karşılaştırılan yöntemlerin keskinlik (*Prec*) ve duyarlılık (*Rec*) değerleri

Önerilen graf yapısı ve orijinal graf yapısı üzerinde uygulanan topluluk arama yöntemi sonuçlarına göre, önerilen graf yapısının orijinal graf yapısına göre daha geniş bir ölçekte arama yapabileceği görülmektedir. Bu doğrultuda, önerilen yaklaşımın, tek bir sorgu düğümünü topluluk arama parametresi olarak kullanarak bir topluluğu ait olan birçok düğümü bulabilme eğiliminde olduğu analiz edilmiştir.

4.4. Diğer Çalışmalar ile Karşılaştırma

Literatürde, öznetelik ağlarda topluluk tanıma problemi için henüz az sayıda çalışma bulunmaktadır (Huang ve Lakshmanan, 2017), (Zhang vd. , 2018) (Sun vd. , 2020), (Liu vd. , 2020). Bu çalışmalardan öne çıkan çalışmalardan biri Huang ve Lakshmanan'ın 2017 yılında önerdiği çalışmadır (Huang ve Lakshmanan, 2017). Önerdikleri yöntemle ait deneysel sonuçları 9 farklı veri seti üzerinden sunmuşlardır. Çalışmalarında, bizim çalışmamızla ortak veri seti olarak sadece Facebook veri seti kullanılmıştır. Kullandıkları veri setindeki düğüm ve kenar sayısı bizim çalışmamızdaki düğüm ve kenar sayısından farklı olup farklı parametre değerleri ile testlerini gerçekleştirmişlerdir. Çalışmada elde edilen f-ölçüm değeri yaklaşık olarak 0,8'lere yakın yüksek bir değerdir. Bizim deneylerimizde ise $k=3$ için önerilen graf yapısındaki k -tras sonucumuz 0.37883, k -core için 0.46882 olarak bulunmuştur. (Zhang vd. , 2018) çalışmasında ise ortak veri seti olarak Facebook veri seti bulunmakta olup kullanılan veri seti kaynağı farklıdır. Burada kullanılan Facebook veri seti ile de kenar ve düğüm sayılarımız farklılık göstermektedir. Çalışmada maksimum keyword frekansı için ACS, Lokal, Global ve ACQ yöntemler uygulanarak çalışma süreleri paylaşılmıştır. (Sun vd. , 2020) çalışmasında ise çalışmamızla ortak olan veri setleri Cora ve Citeseer'dır. Benzer şekilde, kullanılan veri setlerindeki kenar ve düğüm sayıları bizim kullandığımız veri setlerindeki kenar ve düğüm sayısından farklıdır. Yaptıkları deneyde ağdaki düğümlerden rastgele seçilerek belirlenen her düğüm için Basic, DASC ve LocATC algoritmalarını kullanarak 10 kez deneyi tekrar etmişlerdir. Cora için f-ölçüm değeri 1'e yakın bulunurken Citeseer için 0,9'a yakındır. Bizim çalışmamızda ise Cora için k -tras 0.36054 iken k -core 0.46411, Citeseer için k -tras 0.29166 iken k -core 0.34936'dır. (Liu

vd. , 2020) çalışmasında ise ortak veri seti Facebook olup kenar ve düğüm sayıları bizim kullandığımızdan farklıdır. Çalışmada topluluk yapısı benzerliği (community structure similarity (CSS)) ve topluluk öznitelik benzerliği (community attribute similarity (CAS)) sunularak, $k = 8$ 'den $k = 20$ ' ye kadar değişen değerleri için toplam çalışma süresi ile uzay maliyeti (Space Cost) değerleri çalışmada sunulmuştur. Sonuç olarak, mevcut öznitelikli ağlarda topluluk tanıma yöntemlerinin başarısı, kullanılan veri setlerinin aynı olmaması, aynı veri setlerinde test edilen düğüm sayısının ve kenar sayısının farklılıklar göstermesi nedeni ile oldukça farklılık göstermektedir.



5. SONUÇ VE ÖNERİLER

Topluluk arama problemi için yapılan çalışmaların birçoğunda öznitelik verisi göz ardı edilerek sadece graf topolojisi tabanlı topluluk arama yöntemleri kullanılmaktadır. Bu çalışmada, öznitelik verisi dikkate alınarak graf topolojisi tabanlı topluluk arama yöntemlerini öznitelikli ağ yapıları içinde kullanılabilir hale getiren yeni bir graf temsil modeli önerdik. Ağ yapısından bulunan öznitelik verisini bir graf yapısına dönüştürerek, mevcuttaki orijinal graf yapısına içeriksel boyut kazandırmayı amaçladık. Deneyler sırasında kullandığımız veri setleri üzerinden deneysel sonuçlar, önerilen yaklaşım ile k-tras ve k-çekirdek topluluk arama algoritmalarının daha yüksek başarı değerleri elde edilebildiğini göstermektedir.

Gelecek çalışmamızda, öncelikle karşılaştırılan topluluk arama algoritmalarının sayısı ile veri setlerinin sayısını artırarak deneysel sonuçlarımızı genişletmeyi hedefliyoruz. Önerdiğimiz graf yaklaşımına ait topluluk arama sonuçlarımızdaki duyarlılığı artırmak için önerdiğimiz graf yaklaşımını kullanarak keşfettiğimiz topluluklardaki ilgisiz düğümleri filtreleyen yeni bir yaklaşım geliştirmeyi hedefliyoruz. Bu doğrultuda keşfettiğimiz k-tras ve k-çekirdek alt toplulukları için sorgu düğümünü içeren daha anlamlı bir alt topluluk araması yapmayı amaçlıyoruz. Bir diğer gelecek çalışmamız ise, önerilen yaklaşım üzerine daha hızlı ve daha yüksek başarılar üretebilecek, ağın hem yerel hem de küresel komşuluklarını dikkate alarak topluluk arama süreçlerini kullanan yeni bir yaklaşım önermektir.

KAYNAKLAR

- Akiba, T., Iwata, Y., Yoshida, Y., 2013. Linear-time Enumeration of Maximal k-edge-Connected Subgraphs in Large Networks by Random Contraction. In Proceedings of the 22nd ACM International Conference on Information & Knowledge Management , 909-918.
- Barbieri, N., Bonchi, F., Galimberti, E., Gullo, F., 2015. Efficient and Effective Community Search. Data Mining and Knowledge Discovery, 29(5), 1406-1433.
- Batagelj, V., Zaversnik, M., 2003. An $O(m)$ Algorithm for Cores Decomposition of Networks. ArXiv Preprint cs/0310049.
- Bedi, P., Sharma, C., 2016. Community Detection in Social Networks. Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 6(3), 115-135.
- Bello, G. A., Harenberg, S., Agrawal, A., Samatova, N. F., 2016. Community Detection in Dynamic Attributed Graphs. In International Conference on Advanced Data Mining and Applications, 329-344.
- Chang, L., Lin, X., Qin, L., Yu, J. X., Zhang, W., 2015. Index-based Optimal Algorithms for Computing Steiner Components with Maximum Connectivity. In Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, 459-474.
- Chunaev, P., 2020. Community Detection in Node-Attributed Social Networks: A Survey. Computer Science Review, 37, 100286.
- Combe, D., Largeron, C., Egyed-Zsigmond, E., Géry, M., 2012. Combining Relations and Text in Scientific Network Clustering. In 2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, 1248-1253.
- Ebadian, S., Huang, X., 2019. Fast Algorithm for k-truss Discovery on Public-private Graphs. ArXiv Preprint ArXiv:1906.00140.
- Fang, Y., Cheng, R., Chen, Y., Luo, S., Hu, J., 2017. Effective and Efficient Attributed Community Search. The VLDB Journal, 26(6), 803-828.
- Fang, Y., Huang, X., Qin, L., Zhang, Y., Zhang, W., Cheng, R., Lin, X., 2020. A Survey of Community Search Over Big Graphs. The VLDB Journal, 29(1), 353-392.
- Fang, Y., Yu, K., Cheng, R., Lakshmanan, L. V., Lin, X., 2019. Efficient Algorithms for Densest Subgraph Discovery. arXiv preprint arXiv:1906.00341.

- Fortunato, S., 2010. Community Detection in Graphs. *Physics reports*, 486(3-5), 75-174.
- Grover, A., Leskovec, J., 2016. node2vec: Scalable Feature Learning for Networks. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 855-864.
- Günemann, S., Boden, B., Färber, I., Seidl, T., 2013. Efficient Mining of Combined Subspace and Subgraph Clusters in Graphs with Feature Vectors. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 261-275.
- Huang, X., Cheng, H., Qin, L., Tian, W., Yu, J. X., 2014. Querying k-truss Community in Large and Dynamic Graphs. In *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, 1311-1322.
- Huang, X., Lakshmanan, L. V., Xu, J., 2017. Community Search Over Big Graphs: Models, Algorithms, and Opportunities. In *2017 IEEE 33rd International Conference on Data Engineering (ICDE)*, 1451-1454.
- Huang, X., Lakshmanan, L. V., Xu, J., 2019. Community Search Over Big Graphs. *Synthesis Lectures on Data Management*, 14(6), 1-206.
- Huang, X., Lakshmanan, L. V., 2017. Attribute-driven Community Search. *Proceedings of the VLDB Endowment*, 10(9), 949-960.
- Kakisim, A., Sogukpinar, I., 2015. Community Detection in Social Networks Using Content and link Analysis. In *2015 23rd Signal Processing and Communications Applications Conference (SIU)*, 1521-1524.
- Kong, Y. X., Shi, G. Y., Wu, R. J., Zhang, Y. C., 2019. K-core: Theories and Applications. *Physics Reports*, 832, 1-32.
- Lancichinetti, A., Fortunato, S., 2009. Community Detection Algorithms: a Comparative Analysis. *Physical Review E*, 80(5), 056117.
- Liu, Q., Zhu, Y., Zhao, M., Huang, X., Xu, J., Gao, Y., 2020. VAC: Vertex-centric Attributed Community Search. In *2020 IEEE 36th International Conference on Data Engineering (ICDE)*, 937-948.
- Lu, C., Yu, J. X., Wei, H., Zhang, Y., 2017. Finding The Maximum Clique in Massive Graphs. *Proceedings of the VLDB Endowment*, 10(11), 1538-1549.
- LNQS, (2021). Statistical Relational Learning Group, (<https://linqs.soe.ucsc.edu/data>), (Erişim Tarihi: 07.05.2021).
- Newman, M. E., Girvan, M., 2004. Finding and Evaluating Community Structure in Networks. *Physical Review E*, 69(2), 026113.

- Perozzi, B., Al-Rfou, R., Skiena, S., 2014. Deepwalk: Online Learning of Social representations. In Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 701-710.
- Porter, M. A., Onnela, J. P., Mucha, P. J., 2009. Communities in Networks. Notices of the AMS, 56(9), 1082-1097.
- Seidman, S. B., 1983. Network Structure and Minimum Degree. Social Networks, 5(3), 269-287.
- Sozio, M., Gionis, A., 2010. The Community-search Problem and How to Plan a Successful Cocktail Party. In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 939-948.
- Sun, H., Huang, R., Jia, X., He, L., Sun, M., Wang, P., Sun, Z., Huang, J., 2020. Community Search for Multiple Nodes on Attribute Graphs. Knowledge-Based Systems, 193, 105393.
- Traud, A. L., Frost, C., Mucha, P. J., Porter, M. A., 2009. Visualization of Communities in Networks. Chaos: An Interdisciplinary Journal of Nonlinear Science, 19(4), 041104.
- Wang, J., Cheng, J., 2012. Truss Decomposition in Massive Networks. ArXiv Preprint ArXiv:1205.6693.
- Xiao, J., Dai, Q., Xie, X., Lam, J., Kwok, K. W., 2021. Adversarially Regularized Graph Attention Networks for Inductive Learning on Partially Labeled Graphs. ArXiv Preprint ArXiv:2106.03393.
- Yuan, L., Qin, L., Lin, X., Chang, L., Zhang, W., 2016. Diversified Top-k Clique Search. The VLDB Journal, 25(2), 171-196.
- Yuan, L., Qin, L., Lin, X., Chang, L., Zhang, W., 2017. I/O Efficient ECC Graph Decomposition via Graph Reduction. The VLDB Journal, 26(2), 275-300.
- Yuan, L., Qin, L., Zhang, W., Chang, L., Yang, J., 2019. Index-based Densest Clique Percolation Community Search in Networks. In 2019 IEEE 35th International Conference on Data Engineering (ICDE), 2161-2162.
- Zhang, J., Han, K., Han, J., Li, F., Li, S., 2018. Attributed Community Search in Dynamic Networks. In 2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI), 893-900.

- Zhang, Q., Li, R. H., Yang, Q., Wang, G., Qin, L., 2020. Efficient Top-k Edge Structural Diversity Search. In 2020 IEEE 36th International Conference on Data Engineering (ICDE), 205-216.
- Zhao, Z., Feng, S., Wang, Q., Huang, J. Z., Williams, G. J., Fan, J., 2012. Topic Oriented Community Detection Through Social Objects and Link Analysis in Social Networks. Knowledge-Based Systems, 26, 164-173.
- Zhou, Y., Cheng, H., Yu, J. X., 2009. Graph Clustering Based on Structural/Attribute Similarities. Proceedings of the VLDB Endowment, 2(1), 718-729.



ÖZGEÇMİŞ

Adı Soyadı : Tuğçe KAĞNICI

Eğitim Durumu

Lise : Henza Akın Çolakoğlu Lisesi, 2011

Lisans : Dokuz Eylül Üniversitesi, Fen Fakültesi, Kimya Bölümü, 2015

Yüksek Lisans : İstanbul Ticaret Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı, 2021

Yayınları

Kağnıcı, T., Kakişim, A., 2021. Öznitelikli Ağlarda Topluluk Arama. Avrupa Bilim ve Teknoloji Dergisi, 26, 332-337.