T.C.
GEBZE YÜKSEK TEKNOLOJİ ENSTİTÜSÜ
MÜHENDİSLİK VE FEN BİLİMLERİ
ENSTİTÜSÜ

E-POSTA LİSTELERİNDE METİN KÜMELEME VE SOSYAL AĞ ANALİZİ UYUMU

HAYATİ GÖNÜLTAŞ
YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

GEBZE
2010

T.C.
GEBZE YÜKSEK TEKNOLOJİ ENSTİTÜSÜ
MÜHENDİSLİK VE FEN BİLİMLERİ
ENSTİTÜSÜ

E-POSTA LİSTELERİNDE METİN KÜMELEME VE SOSYAL AĞ ANALİZİ UYUMU

HAYATİ GÖNÜLTAŞ
YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

DANIŞMANI
YRD. DOÇ.DR. MEHMET GÖKTÜRK

GEBZE
2010

GEBZE YÜKSEK TEKNOLOJİ ENSTİTÜSÜ	YÜKSEK LİSANS JÜRİ ONAY FORMU

G.Y.T.E. Mühendislik ve Fen Bilimleri Enstitüsü Yönetim Kurulu'nun …….. tarih ve …… sayılı kararıyla oluşturulan jüri tarafından ……/……/…… tarihinde tez savunma sınavı yapılan ……..'ın tez çalışması …….. Anabilim Dalında Yüksek Lisans tezi olarak kabul edilmiştir.

JÜRİ

ÜYE (TEZ DANIŞMANI) :

ÜYE :

ÜYE :

ONAY

G.Y.T.E. Mühendislik ve Fen Bilimleri Enstitüsü Yönetim Kurulu'nun ……. tarih ve ……/…….. sayılı kararı.

İMZA/MÜHÜR

ÖZET

TEZ BAŞLIĞI : E-POSTA LİSTELERİNDE METİN KÜMELEME VE SOSYAL AĞ ANALİZİ UYUMU

TEZ YAZARI : HAYATİ GÖNÜLTAŞ

E-posta listeleri birbirleriyle herkese açık içerikler aracılığıyla iletişim kuran kişiler tarafından ve açık kaynak kod topluluğunca popüler olarak kullanılmaktadır. Bu listeler veri madenciliği için büyük miktarda veri sunmaktadır. Sadece e-postaların içeriklerinin değil, bunun yanında e-posta gönderen bireylerin birbirleriyle bağlantılarının (gönderen alan ilişkisi içerisinde) yapısı da ilgili e-posta alt gruplarının daha önceden gözlemlenmemiş ancak önemli olan bilgilerini ortaya çıkarabilir.

Metin kümeleme yapılırken e-posta listelerindeki metinler doküman verisi gibi ele alınarak alt kümeler tespit edilmiştir. Vektör uzay modelinde kelimelerin normalize edilmiş frekansları kullanılarak k-means algoritması ile kümeleme yapılmıştır. Sosyal ağ analizinde ise CONCUR algoritması ile alt kümeler bulunmuştur. Metin madenciliği ile tespit edilen alt gruplar ile sosyal ağ analizinin tespit ettiği alt gruplar arasında aynı sosyal yapıyı dokudukları için benzer küme varlıklarına sahip olmaları beklenmektedir. Bu sosyal yapıda, e-posta göndericilerin genellikle benzer konularda yazdıklarını ve benzer konuların genellikle benzer kişiler tarafından cevaplandığını öne sürmekteyiz. Metin madenciliği ve sosyal ağ analizi metotları teker teker uygulandığında (ör. sosyal ağ analizi için CONCUR, metin madenciliği için k-means) oluşan kümelerin uyumu, kullanılan algoritmalara ve oluşması beklenen küme sayısına göre değiştiği gözlemlenmiştir. Çalışmamızda sosyal ağ analizinin ve metin madenciliğinin tespit ettiği alt gruplar yaklaşık %60 birbiriyle aynı sonucu verdi. Ek olarak uygun olmayan algoritma ve/veya küme sayısı seçildiğinde ise uyumun etkileyici şekilde düştüğü gözlemlenmiştir.

SUMMARY

THESIS TITLE : COHERENCE BETWEEN TEXT CLUSTERING AND SOCIAL NETWORK ANALYSIS IN E-MAIL LISTS

THESIS AUTHOR : HAYATİ GÖNÜLTAŞ

E-mail listings are popularly used in open source community and by individuals that contact each other with publicly reachable contexts. This listings serve vast amounts of data that could be used for data mining. Not only textual context in e-mails but also structure of links between e-mail senders (individuals) could reveal previously unobserved and also important information about related e-mail subgroups.

In order to find clusters in e-mail lists, e-mail contents threated as document data. Normalized values of frequences of terms used in vector space model, and k-means algorithm used for clustering the e-mails. CONCUR algorithm is used to find clusters while analysing social network of e-mail lists. It is expected to appear a relation between entities of clusters found by text mining and social network analysis as they weave the similar social subgroupings. At this social structure, we propose that senders mostly write about similar topics, and similar topics are mostly responded by similar senders. When text mining and social network analysis methots are taken one by one (such as CONCUR for SNA and k-means for text mining), it is observed that coherence of resulting clusters depend on used algorithms and number of clusters expected. In our work, coherence of social network analysis and text mining subgroups are matched about %60. In addition, when inappropriate algorithms and/or number of clusters are chosen, results will decrease dramatically.

TEŞEKKÜR

En kritik zamanlarda yanımda olduğunu bildiğim tez danışmanım Yrd. Doç. Dr. Mehmet GÖKTÜRK'e yardım ve katkılarından dolayı teşekkür ederim. Ayrıca bütün Gebze Yüksek Teknoloji Enstitüsü çalışanlarına ve özellikle bilgisayar mühendisliği öğretim üyelerine öğrettiklerinin bir değil bin harf olduğunu bilerek teşekkürü bir borç bilirim.

Sabırla çalışmalarımı bitirmemi bekleyen aileme, okuluma gideceğim günler bana izin veren işyerime de ayrıca teşekkürlerimi sunarım.

İÇİNDEKİLER DİZİNİ

SİMGELER VE KISALTMALAR DİZİNİ

SNA	: Social Network Analysis
SAA	: Sosyal Ağ Analizi
KDD	: Knowledge Discovery in Databases
GUI	: Graphical User Interface
WEKA	: Waikato Environment for Knowledge Analysis
VM	: Veri Madenciliği
VTYS	: Veri Tabanı Yönetim Sistemi
YSA	: Yapay Sinir Ağı
DVM	: Destek Vektör Makineleri
KNN	: K-Nearest Neighbor Algoritması
DY	: Denetimsiz Yöntemler
BHK	: Birleştirici Hiyerarşik Kümeleme
KDA	: Küçük Dünya Ağları
AOM	: Arada Olma Merkeziliği

ŞEKİLLER DİZİNİ

ÇİZELGELER DİZİNİ

1. GİRİŞ

E-posta listeleri toplulukların haberleşmesi için kullanılan forum benzeri bir yapıdır. Bu yapı kimi zaman işletim sistemindeki hataların gönderildiği bir liste olurken kimi zaman ise güncel bir konu hakkındaki tartışmayı barındırabilir. Ancak e-posta listeleri genel itibariyle belirli bir konuda yazışmaların olduğu bir mesajlaşma tahtası gibidir. Standart mesajlaşmalardan farklı olarak yazılanlar normal bir e-posta gibi bir kullanıcının e-posta adresinden gönderilir ancak standart olmayan şey bu e-postanın listedeki herkese gönderilmesidir. Dolayısıyla listeye gönderilen bir e-postayı bütün liste aboneleri görecek ve bu postaya cevap yazabileceklerdir. Tahmin edilebileceği gibi verilen cevap da bütün abonelere gönderilecektir.

E-posta listelerindeki bu yazışmalar belirli bir metin verisi sunmaktadır. Bu metin verisi, sınıflandırma ve kümeleme gibi veri madenciliği işlemlerinde kullanılabilir. E-postaların metni üzerinde yapılacak kümeleme doküman kümeleme ile büyük benzerlik göstermektedir. Veri madenciliği tekniklerinin metin üzerinde kullanılması ve metin üzerinde kullanılabilmesi için bu tekniklerin uyarlanması işlemine metin madenciliği denmektedir. Bu haliyle e-postalar üzerinde metin madenciliği herhangi bir doküman kümesi için metin madenciliği yapmaktan farksızdır.

Kişiler genellikle ilgi duydukları konularda e-posta yazmaktadırlar. Bir takımın taraftarının tuttuğu takımı ilgilendiren durumlarda tepki verdiği gibi e-posta listelerine üye olan kullanıcıların da kendilerini ilgilendiren durumlarda yazı yazdıkları düşünülebilir. Bu durumda kişiler genellikle kendi konu kümelerinde e-posta yazmaktadır denilebilir. Dolayısıyla aslında metin madenciliği ile e-postalar konu kümelerine ayrılırken aslında şahıslarında ilgili olduğu konular belirginleşme başlamaktadır.

E-posta listesinde postalar konu temelli ayrılmaktadırlar. Belirli bir konuya cevap vermiş bütün kullanıcılar arasında bir bağ bulunmaktadır. A kişinin B kişisine verdiği e-posta cevabı A ile B arasında bir ilişki kurmaktadır. Aslında bu durum da metin madenciliğindeki gibidir. Çünkü kişiler genellikle ilgili oldukları konularda

yazacaklardır. Zaten aynı ilgiye sahip kişiler aynı konularda yazacaklardır. Dolayısıyla aralarındaki bu bağ kuvvetlenecektir.

Çalışmamızda metin madenciliğinden elde edilen kümelerle, sosyal ağ analizinden elde edilen kümeler arasındaki benzerlik ölçülmüştür.

Tezin Organizasyonu

Bu tez çalışması dokuz bölümden oluşmaktadır. İkinci bölümde e-posta listeleri hakkındaki açıklamalar, kavramlar ve örnek e-posta listeleri gösterilmektedir.

Üçüncü bölümde veri madenciliğinin tanımı, diğer disiplinlerle olan ilişkisi, veri madenciliği işlem süreci ve görevlerinden bahsedilmektedir. Burada ayrıca veri önişleme de açıklanacaktır.

Dördüncü bölümde sosyal ağ analizi anlatılacak ve graf teorisi ile ilişkisi açıklanacaktır. Sosyal ağ analizi için önemli kavramlar olan yoğunluk ve merkezilik de bu bölümde açılacaktır.

Metin kümelemenin anlatıldığı beşinci bölümünün ilk kısmında e-posta listelerinde metin kümeleye hazırlık aşaması olan veri önişlemenin aşamaları açıklandıktan sonra ikinci kısmında ise metin kümeleme için kullanılan algoritma ve sonuçları ifade edilecektir.

Altıncı bölümde sosyal ağ analizi algoritmalarından CONCUR'un elimizdeki veriye uygulanması anlatıldıktan sonra e-posta isim alanından kişi isim alanına geçiş için kullanılan yöntem açıklanacaktır.

Sonuçların açıklandığı yedinci bölümde metin kümeleme ve sosyal ağ analizi ile elde edilen sonuçlar değerlendirilecektir. Gelecek çalışmalar için değerlendirmeler

sekizinci kısımda, çalışmanın daha iyi sonuç vermesi için yapılabilecek öneriler ise dokuzuncu kısımda açıklanmıştır.

2. E-POSTA LİSTELERİ

2.1. E-posta Listeleri ile ilgili Kavramlar

E-posta listeleri bir e-postanın aynı andan birden fazla kişiye gönderilmesine olanak veren bir sistem sunmaktadır. E-posta listeleri yarı ya da tam otomatik olarak çalışıyor olabilir. Bu durum kullanılan yazılımın özelliklerine göre değişebilmektedir.

Genellikle e-posta listelerine üye olmak veya üyelikten çıkmak için ayrı adresler kullanılmaktadır. Kullanılan e-posta listesi uygulaması bu adreslere gönderilen komutları işleyerek gerekli işlemleri yapabilmektedir. E-posta listeleri genel olarak iki çeşittir:

- ***Haber grubu:*** Bu şekildeki listeye sadece belirli kullanıcılar e-posta gönderebilir.
- ***Tartışma grubu:*** Bu şekildeki listeye bütün kullanıcılar mesaj gönderebilir.

E-posta listelerine üye olan kullanıcılar ya yeni bir konu açarak o konu hakkında e-posta gönderebilir ya da daha önceki bir e-postaya cevap yazabilir. Her iki durumda da ileti bütün gruba gönderilmektedir.

E-posta listelerine web arayüzünden ya da eğer POP, IMAP gibi protokolleri destekliyorsa çeşitli e-posta uygulamalarıyla erişilebilmektedir. Ancak bu şekilde erişildiğinde maalesef kullanılan uygulamanın eklediği ek mesajlar çeşitlilik gösterebilmektedir. Bunun yanında farklı kullanıcıların farklı yerelleşme ayarları da bu standardı bozabilmektedir.

Listeler tek bir konu üzerine odaklanabileceği gibi genel bir konuda yazmaya olanak vermektedir. Örneğin programlama dalı altında tek bir grup olabileceği gibi,

“linux-programlama” programlama listesi altındaki daha alt bir grup olarak ta bulunabilir.

2.2. Kullanıldığı Alanlar ve Benzer Kullanımlar

Bu sistemde kullanıcılar genellikle hata takibi (bug fix), topluluk haberleşmeleri, bir konu hakkında toplulukta bilgisi olanlardan yardım alma gibi amaçlarla kullanılabilmektedir. Örneğin biz çalışmamızda ***alt.politics.bush*** e-posta listesini inceledik. Türkçe e-posta listesi olarak ***linux-programlama*** listesi de örnek verilebilir.

Forumlar web sayfalarında sıklıkla karşımıza çıkmaktadır. Forum konuların daha önceden ayrıldığı ve kullanıcıların ilgili konuya yazmalarının beklendiği bir yapıdadır. Bu haliyle e-posta listeleri forumların bir alt konusu gibidirler. Ancak e-posta listeleri konuya biraz daha geniş bakabilmektedir. Bunun yanında forumlarda genellikle konu hatalı bir bölümde olduğunda yönetici tarafında uygun bölüme çekilirken e-posta listelerinde böyle bir uygulama yoktur.

2.3. Örnek E-Posta Listesi

Qt bir GUI kütüphanesidir. Bu kütüphaneyi kullanan ancak gerek öğrenmek istediği bir konuda sorusu olan, gerekse takıldığı bir problemi olan kişiler bu kütüphanenin e-posta listesine (lists.trolltech.com) üye olarak ve e-posta göndererek yardım alabilmektedirler. Şekil 2.1’de örnek bir qt e-posta listesi haberleşmesi gösterilmektedir.

- [Qt4-preview-feedback] Qt installation instructions *x*
  - [Qt4-preview-feedback] Qt installation instructions y
    - [Qt4-preview-feedback] Qt installation instructions x
      - [Qt4-preview-feedback] Qt installation instructions x
- [Qt4-preview-feedback] Symbian build error x
  - [Qt4-preview-feedback] Symbian build error y
    - [Qt4-preview-feedback] Symbian build error x

Şekil 2.1: Örnek e-posta listesi

3. VERİ MADENCİLİĞİ

3.1. Veri Madenciliği Nedir?

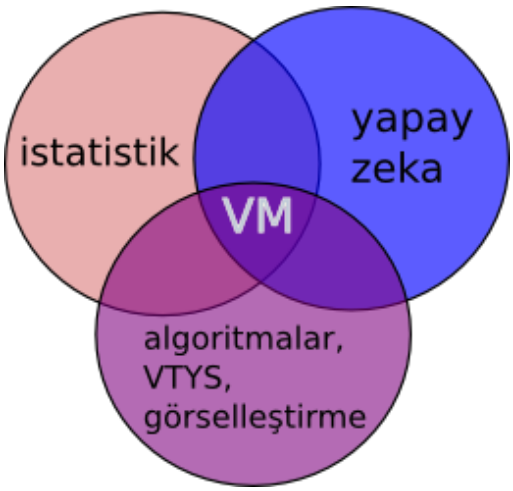
Verinin her bir kaynaktan bolca geldiği günümüzde eldeki veriyi standart metotlarla işlemek ve bundan faydalı bilgileri elde etmek pek mümkün değildir. Elimizdeki bu büyük miktardaki veriden faydalanmak için yeni ve etkin metotlar ortaya koyan veri madenciliği en basit haliyle belirli daha önceden tahmin edilemeyen desenlerin, istatistiklerin, değişimlerin, yeni ilişki ve eğilimlerin büyük miktardaki veriden çıkarılması için kullanılan süreçlerin tümü olarak tanımlanabilir. Veri Madenciliği; büyük veri tabanlarından ve veri ambarlarından kullanışlı, önceden bilinmeyen, önemli ve sonuç olarak anlaşılabilir örüntülerin keşfedimesini ve çıkarılmasını sağlar [U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy. Advances in Knowledge Discovery and Data Mining. AAAI/MIT Press, 1996]. Geniş içeriğe sahip veri kaynaklarından, anlamlı ve kullanılabilir bilginin otomatik şekilde elde edilmesini hedefler. Veri Madenciliği; matematik, istatistik, yapay zeka ve veritabanı tekniklerini kullanarak, veriler hakkında kurallar, eğilimler ve benzerlikler ortaya çıkartıp karar verme aşamasında kullanıcıları bilgilendirerek destek olmayı amaçlar. Veri Madenciliği, veri tabanlarından ve veri ambarlarından toplanan veriye değer katmak ve bilgi keşfetmek için çok geniş veri havuzlarını tarar [Morzy, Mikolaj, "Advanced database structures for effective association rule mining", phD, Pozna´n University of Technology, 2004], [J. Han and M. Kamber. Data Mining: Concepts and Techniques. Morgan Kaufmann, 2000].

Veri madenciliği, kendi başına bir çözüm değil, çözüme ulaşmak için verilecek karar sürecini destekleyen, problemi çözmek için gerekli olan bilgileri sağlamaya yarayan bir araçtır.[S.K Madria, et al., “Research Issues in Web Data Mining”. In Proceedings of Data Warehousing and Knowledge Discovery”, First International Conference. DaWak1999. ss 303-312.]

VM, eldeki verinin örüntülerini, ilişkilerini, istatistiksel bilgilerini kullanarak gelecekteki veri için çıkarımlarda bulunur. VM, çıkarımları ile verinin anlaşılmasına

yardımcı olur. VM istatistik, yapay zeka, uzman sistemler, veritabanı sistemleri, yapay sinir ağları, veri görselleştirme gibi pek çok alt disiplinden faydalanmaktadır.

Veri madenciliği istatistikten farklı bir disiplindir. İstatistikçiler daha önceden ilişkisi bilinen faktörler arasındaki korelasyonu incelerken veri madenciliği ise ilişkisi daha önceden bilinmeyen faktörler arasındaki ilişkileri araştırır. Örneğin "X ürününden alan müşterilerden %30'u Y de satın alır." ilişkisi VM ile bulunabilir. [Tug E. Genetik Algoritmalar ile Tıbbi Veri Madenciligi, Yüksek Lisans Tezi, 2005].



Şekil 3.1: Veri madenciliğinin kullandığı disiplinler

3.2. Veri Madenciliği'nin Kullanıldığı Alanlar

VM verideki örüntülerin insan denetiminde zor tespit edilebildiği durumlarda kullanıldığında faydalı olmaktadır. Veri madenciliğinin kullanıldığı alanlar aşağıda verilmiştir.

3.2.1. Pazarlama

Müşterilerinin davranışlarını incelemek isteyen işyerleri ellerindeki veriler yardımıyla müşteriler hakkında daha fazla bilgiye sahip olabilirler. Pazarlama alanında veri madenciliği aşağıdaki amaçlar için kullanılabilir.

- Müşteri ayrıştırmada,

- Çeşitli ürünleri müşterilere pazarlama kampanyalarında,
- Müşteri memnuniyeti çalışmalarında,
- Pazar sepeti analizinde,
- Müşteri ilişkileri incelemesinde,
- Satış tahminlerinde,

3.2.2. Sağlık Sektörü

Sağlık sektörü verinin oldukça fazla olduğu ve suiistimallerin görülebildiği bir alandır. Bu alanda veri madenciliği suiistimallerin tespitinde kullanılabileceği gibi aşağıdaki amaçlarla da kullanılabilir.

- Stok tespiti,
- Teşhis,
- Standart dışı ve tedaviler

3.2.3. Finans ve Bankacılık

Kredi kartlarının sıklıkla kullanıldığı günümüzde ilk akla gelen amaçlardan birisi kredi kartı dolandırıcılığıdır. VM bu amaçla kullanılabileceği gibi müşterilerin kredi verilebilirliğini de ortaya koyabilir. Finans ve bankacılık sektöründe VM amaçlarından bazıları aşağıda verilmiştir.

- Kredi kartı dolandırıcılıklarının tespiti,
- Müşteri karakteristiklerini tespit etme,
- Kredi taleplerinin değerlendirilmesinde,
- Sahtekarlık tespiti,
- Risk analizleri

3.2.4. Diğer Sektörler

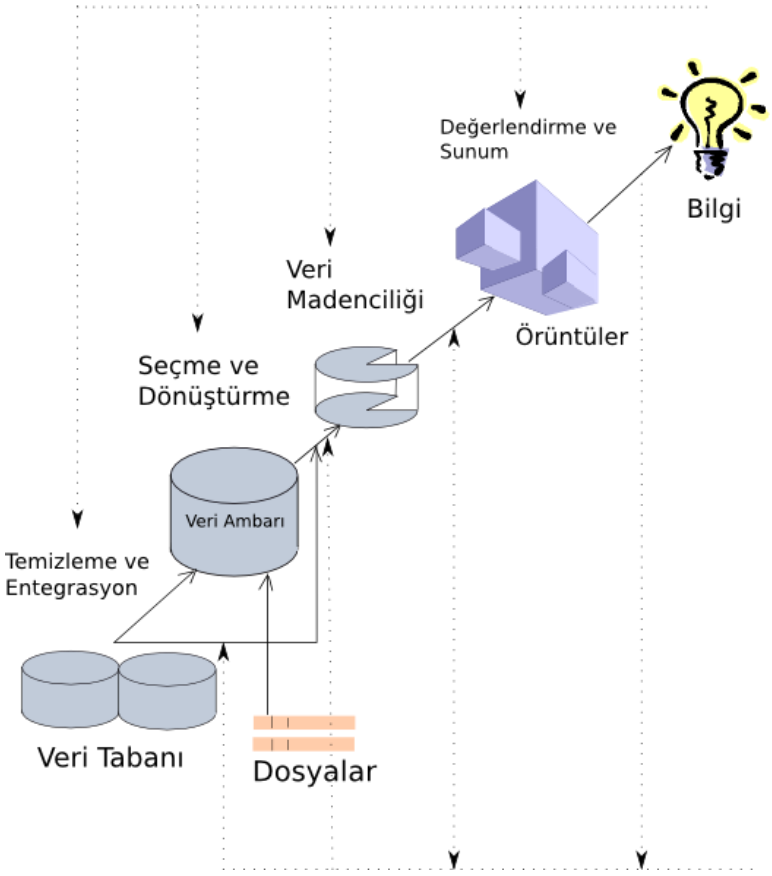
Sigortacılıktan borsaya, endüstriden haberleşmeye pek çok alanda VM'den faydalanılarak değerli bilgi ham bilgiden ayıklanabilir. Aşağıda VM'nin diğer sektörlerdeki ilgi alanları sıralanmıştır.

- Sigorta poliçesini yenilemek isteyebilecek müşterilerin tespitinde,
- Sigorta dolandırıcılığında,
- Sigorta risk değerlendirmesinde,
- Satış noktası memnuniyet ölçümlerinde,
- Alış-veriş sepeti analizleri,
- Alış-veriş sepeti analizi ve alış-veriş alışkanlıkları tespitine göre mağaza düzenlenmesinde,
- Hisse senedi fiyat tahmini,
- Hatların yoğunluk tahminlerinde,
- Lojistik ve iletim için tahmin ve iyileştirmelerde,
- Üretim süreçlerinin iyileştirilmesinde,

kullanılabilir. [Sinan Ataseven, Yüksek Lisans Tezi, 2008]

3.3. Veri Madenciliği Adımları

Veri madenciliğinin sunduğu veri doğrudan geliştirilen algoritmalara genellikle uygulanabilir değildir. Bunun en temel sebebi pek çok veri madenciliği algoritmasının sayısal veriler üzerinde istatistiksel hesaplamalar yaparak işlem görmesidir. Bunun yanında verideki gürültüyü de unutmamak gerekir. Ayrıca verinin normalleştirilmesi de çoğu zaman gerekmektedir. Bu gibi nedenler sebebiyle veri üzerindeki adımlar belirli standart işlemlerden geçirilmelidir. Veri madenciliği temel olarak 7 adımdan oluşmaktadır. (Bkz. Şekil 3.2)



Şekil 3.2: Veri madenciliğinin adımları

3.3.1. Veri Temizleme

Veri tabanından yada veri ambarından elde edilen verinin hepsi amaçlanan VM görevi için uygun olmayabilir. Örneğin alışveriş sepet analizi için müşterilerin yaş bilgisi önemli olabilirken, müşteri hesap numarası önemli değildir. Ayrıca verinin özellik boyutunun büyük olması da VM algoritmalarının çalışma süresini uzatacağından veri temizleme oldukça önemli bir adımdır.

Kullanılacak veride gürültü de bulunabilir. Gürültü VM sonuçlarını etkileyebilir. Bu durumda veri gürültüsü mümkün ise veri temizleme aşamasında ortadan kaldırılmalıdır. Ancak bunun mümkün olmadığı durumlarda VM için kullanılacak algoritmaların seçiminde bu etken göz önünde bulundurularak işlem yapılmalıdır.

3.3.2. Veri Bütünleştirme

VM için kullanılacak veri birden fazla kaynaktan elde edilmiş olabilir. Bu durumda VM gerek performansı gerekse doğru sonuçları üretmesi için veri bütünleştirme yapılmalıdır. Örneğin bir kaynaktan alınan veri dolar cinsinden diğerinden gelen veri ise TL cinsinden olduğunu düşünelim, burada veri ya dolar cinsine ya da TL cinsine çevrilmelidir. Böylelikle her iki kaynaktan gelen veri bütünlüğü bozulmamış olur.

3.3.3. Veri Seçme

Veri gürültü ve hatalı ifadelerden ayrılsa bile verinin tamamı yapacağımız işlem için uygun olmayabilir. Bu nedenle veri seçme aşamasında VM görevimize uygun olan veri bütün veri kümesinden seçilerek VM'ne hazırlanmalıdır. Örneğin metin kümeleme ile sosyal ağ analizinin bulduğu kümeler arasındaki uyumu incelediğimiz çalışmamızda kullandığımız verinin tamamı belki metin kümeleme için uygundu fakat sosyal ağ analizi için uygun olmayan veriler metin kümelemede de kullanılmadı. Böylelikle veri kümesi içerisinden belirli bir küme seçilmiş oldu.

3.3.4. Veri Dönüşümü

VM görevlerini gerçekleştirmek için çeşitli metotlar ve algoritmalar kullanmaktadır. Bu algoritmalar kimi zaman sınıflandırılmış üzerinde çalışırken kimi zamansa sürekli veriler üzerinde çalışmaktadır. Bu nedenle araştırılacak amaca yönelik olarak kullanılacak algoritmalarla uyumlu çalışacak veri türüne dönüşüm sağlanmalıdır.

3.3.5.Veri Madenciliği Metotlarının Uygulanması

Bu adımda çeşitli veri madenciliği algoritmalarıyla kümeleme, sınıflandırma gibi işlemler veri üzerinde yapılabilir.

3.3.6. Örüntü Değerlendirme

Elde edilen örüntüler üzerinden ilginç olanları değerlendirme işleme safhasıdır. Her örüntü önem arz etmeyebilir. Bir örüntünün önemli olması için;

- İnsanlar tarafından kolaylıkla anlaşılabilir olması
- Belirli bir kesinlik ve doğruluk derecesinde yeni veriye ya da test verisine uygulanabilir olması
- Yeni ve faydalı olması

Şartlarını sağlaması gerekmektedir.

3.3.7.Veri Görüntüleme

Verinin görselleştirilmesi aşamasıdır. Bu aşamada veri daha anlaşılır bir şekilde sunularak örüntüler göz önüne serilmeye çalışılmaktadır. Bazı veri kümelerinde görselleştirme uygulandığında örüntüler kolaylıkla gözlemlenebilmektedir. Bu şekliyle veri görüntüleme örüntü çıkarımında da kullanılabilir.

VM adımlarından ilk 4 tanesi veri ön işleme olarak değerlendirilebilir. Son iki tanesi ise çıktıları değerlendirme kısmıdır. Aslında veri ön işleme buradaki en önemli adımlardan birisidir. Bununla birlikte verinin elde edilmesi işlemi de veri ön işleme kısmına dahil edilebilir.

3.4. Veri Madenciliği Yöntemleri

VM ön işlenmesi yapılmış veriyi belirli yöntemler kullanarak inceler ve örüntüleri keşfetmeye çalışır. Temel olarak VM dört görevi gerçeklemektedir. Bunlar;

- Kümeleme

- Sınıflandırma
- Regresyon
- Birliktelik kuralı analizi

Bunlardan kümeleme denetimsiz, sınıflandırma ise denetimli yöntemlerdir.

3.4.1. Denetimli Yöntemler

Denetimli yöntemler eğitim verisinden sonuç fonksiyonu çıkarmaya çalışan makine öğrenmesi tekniğidir. Eğitim verisi etiketi(sınıfı) daha önceden tespit edilmiş veri demektir. Bu veride verinin kendisi ve atandığı sınıf genellikle bir vektör olarak tutulmaktadır. Bu yöntemde amaç daha önceden görülmemiş veriyi eğitim verisinden elde ettiği sonuç fonksiyonu ile tahmin etmeye çalışmaktır. [en.wikipedia.org]

Denetimli yöntemler için bir başka terim ise sınıflandırmadır. Sınıflandırmada aşağıdaki konular göz önünde bulundurulmalıdır.

- Veri kümesinin özelliği
- Veri kümesinin elde edilmesi
- Verideki her bir varlığın özelliklerinin tespit edilmesi. Eğer gerekiyorsa boyut düşürme tekniklerinin kullanılması
- VM için kullanılacak algoritmanın tespit edilmesi
- Çeşitli ölçüm yöntemleriyle test verisi üzerinde modelin sonuçlarının yorumlanması

Denetimli yöntemlerde en sık kullanılan sınıflayıcılar yapay sinir ağları, destek vektör makineleri, k-nearest neighbor algoritması, naive bayes, karar ağaçları yöntemleridir.

3.4.1.1. Yapay Sinir Ağları

Yapay zeka insandaki sinir sistemindeki sinirlerin bazı özelliklerini yapay sinir sistemiyle (YSA) ile gerçeklemeye çalışmıştır. Bu gerçekleme ses tanıma, resim

analizi gibi alanlarda başarılı sonuçlar vermiştir. YSA yapay sinir hücrelerinin birkaç katman sinir hücresinin birbirleriyle bağlanması ile meydana gelmektedir. Şekil 3.3'te gösterilen örnek bir YSA nöronunda diğer hücrelerden veya dış ortamdan alınan girişler ve bu girişlerin ağırlıkları ile aktivasyon ve toplama fonksiyonları da gösterilmektedir. Burada çıkış değeri;

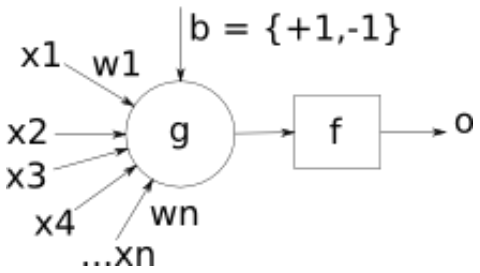
$$W=w_1,w_2,w_3,\ldots w_n \qquad (3.1)$$

$$X=x_1,x_2,x_3,\ldots.x_n \qquad (3.2)$$

3.1 nörondaki giriş ağırlıklarını, 3.2. ise nöron girişlerini ifade etmek üzere çıkış fonksiyonu;

$$O = f(\sum w_i x_i + b) \qquad (3.3)$$

Şeklinde ifade edilebilir. f aktivasyon fonksiyonudur. Aktivasyon fonksiyonu doğrusal olabileceği gibi sigmoid gibi farklı türde de olabilir. Birden fazla katmana sahip YSA modelleri bulunabilir. Burada ara katmanlara gizli katman denilmektedir.



Şekil 3.3: Yapay Sinir Ağı Şablonu

YSA'lar bilginin iletimine göre ileri beslemeli YSA ve geri beslemeli YSA olarak iki sınıfa ayrılırlar. Burada yön bilginin ve YSA'nın ağırlıklandırılmasının yönünü ifade etmektedir.

3.4.1.2. Destek Vektör Makineleri

Destek vektör makineleri (DVM) sınıflandırma ve regresyon için kullanılan denetimli öğrenme yöntemlerinin kümesidir. Verilen bir örneğin iki sınıftan hangisine ait olduğunu tahmin eden bir yöntemdir. Vapnik tarafından öne sürülmüştür [Vapnik 1995,1998]

DVM el yazısı tanıma, ses tanıma, konuşmacı tanıma, metin sınıflandırma gibi pek çok alanda kullanılmıştır. DVM sınıflandırma için yüksek yada sonsuz boyut uzayında aralığı maksimum yapan bir ara-düzlem oluşturmaya çalışır.

DVM doğrusal ve doğrusal olmayan sınıflayıcılar olarak ikiye ayrılmaktadır. Doğrusal sınıflayıcılar veri kümesini doğrusal olarak ayırabildiği durumlarda kullanılabilir. Ancak pek çok veri doğrusal olarak ayırmaya elverişli değildir bu durumda veri doğrusal olarak ayrılabilecekleri farklı bir uzaya taşınarak sınıflandırma bu uzayda yapılabilmektedir.

3.4.1.3. K-nearest Neighbor (KNN) Algoritması

k-nearest neighbor (KNN) algoritması tembel öğrenici bir algoritmadır. En basit ifadeyle çevresindeki “k” adet en yakın komşusunun sınıfına, sınıfı tahmin edilecek veriyi atayan bir algoritmadır. Diğer varlıkların mesafesine göre komşularını ağırlıklandırmak mümkündür. Ancak KNN algoritması verinin yerel yapısına karşı hassastır [Duda, Hart et al Pattern Classification, 2000].

KNN algoritmasındaki eğitim YSA’daki gibi değildir. YSA’da ağırlıkların tespit edilmesi işlemi eğitim esnasında yapılırken KNN’de sadece verinin belleğe sınıf etiketiyle çekilmesi söz konusudur. Uzaklık (benzemezlik) ölçütü olarak euclidean mesafe kullanılabileceği gibi metin kümeleme gibi çalışmalarda hamming mesafesi de kullanılabilir. Bunun yanında KNN algoritması gürültüden de kolaylıkla etkilenebilir.

Şekil 3.4 yıldız şeklindeki verinin sınıfını KNN tespit etmek için aşağıdaki adımları izler:

- K = 2 değeri seçilirse;

Yıldız herhangi bir sınıfa atanamaz çünkü kendisine en yakın olan kare ve daire sınıflarıdır. Baskın olan sınıf olmadığından sınıflandırma hatası oluşur.

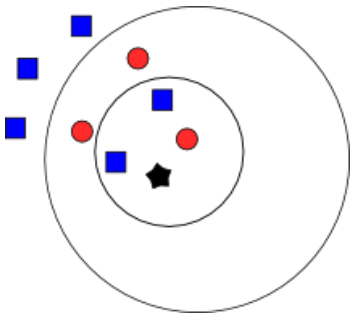
- K = 3 seçilirse;

Yıldıza en yakın 3 komşudan 2 tanesi kare bir tanesi dairedir. Yıldız kare ile etiketlenir.

- K = 5 seçilirse;

Yıldıza en yakın 5 komşudan 3 tanesi daire 2 tanesi karedir. Yıldız daire ile etiketlenir.

Görüldüğü gibi KNN algoritması hem seçilecek K değerine hem verinin yapısına da bağlı olarak çalışmaktadır.



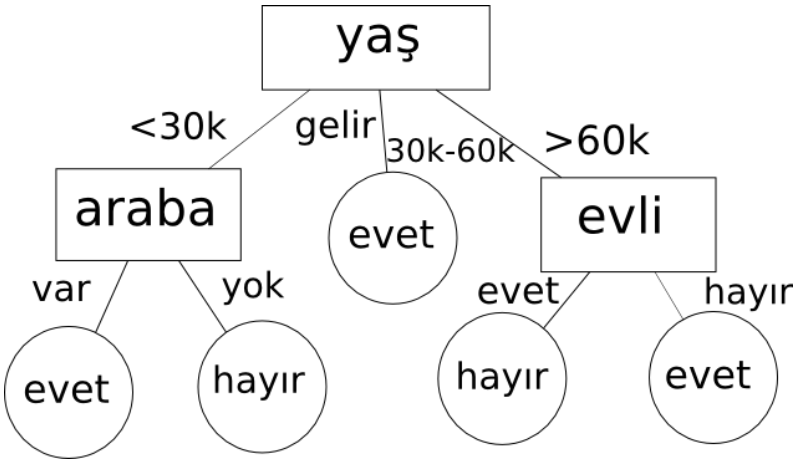
Şekil 3.4: KNN sınıflandırması

3.4.1.4. Naive Bayes Sınıflandırıcı

Naive Bayes sınıflandırıcı veri özelliklerinin birinin olma olasılığının diğerinden bağımsız olduğu kabulünü yapan Bayes teoremi temelli bir olasılıksal sınıflayıcıdır. Bayes inanç ağları ise yine sınıflandırma için kullanılabilecek grafiksel modellerden birisidir.

3.4.1.5. Karar Ağaçları Öğrenimi

Karar ağaçları öğrenimi etiketli sınıf varlıklarından öğrenimdir. Her ara-düğüm test özelliklerinden herhangi birisini, yaprak düğümü testin sonucunu, en üstteki düğümse kök düğümü ifade etmektedir. Şekil 3.5'te örnek bir karar ağacı verilmiştir.



Şekil 3.5: Kredi vermek için karar ağacı

Şekil 3.5'te bir banka müşterisinin kredi verilebilirliği için karar ağacı gösterilmektedir. Ara-düğümler dikdörtgenlerle yaprak düğümleri ise dairelerle gösterilmiştir. Bu şekilde oluşturulmuş bir karar ağacına kredi verilip verilmeyeceği tespit edilmek istenen "Öğren" adlı veri getirildiğinde öğren verisinin özellikleri karar ağacına uygulanarak kredi sonucu öğrenilebilir.

Görüldüğü gibi öğren kişisinin ilk olarak geliri ardından gelirine göre arabasının olup olmaması veya evli olup olmamasına göre kredi sonucu tespit edilebilmektedir.

3.4.2. Denetimsiz Yöntemler

Denetimsiz yöntemler (DY) daha önceden sınıf etiketinin belirli olmadığı durumlarda kümelemenin verinin kendisi kullanılarak yapıldığı yöntemlerdir. DY istatistikteki yoğunluk (density estimation) ile yakından ilgilidir. Çünkü kümeleme yapılırken veriler arasında benzerlik benzemezlik ilişkisi için belirli bir ölçüte göre mesafe ölçümü yapılır. Kümelemenin hedefi küme içi mesafenin minimize edilmesi ve kümeler arası mesafenin maksimize edilmesidir.

Hiyerarşik olarak çalışan kümeleme algoritmaları sıklıkla karşımıza çıkmaktadır. Bu algoritmalar ardı ardına kümeyi daha alt kümelere bölerek (divisive) kümeleme yapabildiği gibi başlangıçta bütün verinin ayrık olduğu ve birleştire birleştire kümelerin tespit edildiği birleştirici (agglomerative) bir tazla da kümelemeyi yapabilmektedir.

Kümeleme için söylenebilecek problemlerin başında kaç tane küme oluşmasının beklendiği temel bir problemdir. Pek çok kümeleme algoritması bu küme sayısının önceden tespit edilmesini gerekli kılmaktadır.

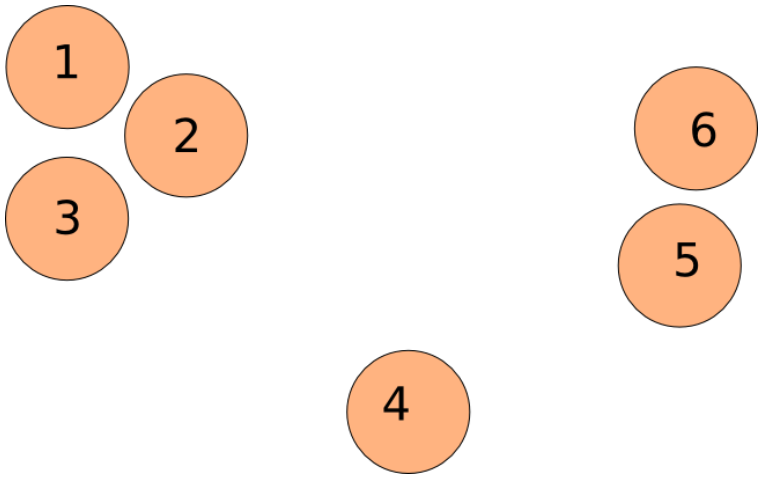
Kümelemede bir diğer zorluk ise mesafe ölçütünün tespit edilmesinde karşımıza çıkmaktadır. Mesafe ölçütü diğer deyişle benzerlik kümelerin gerek şekillerini gerekse barındırdığı üyeleri etkilemektedir. Aşağıda en sık kullanılan mesafe ölçütleri verilmiştir.

- ***Euclidean Mesafe:*** Kuş uçuşu mesafe olarak ta bilinmektedir. Geometrideki iki nokta arasındaki uzaklığı veren ifadedir. Benzerlik ölçütü olarak Euclidean mesafenin karesi de kullanılabilir.

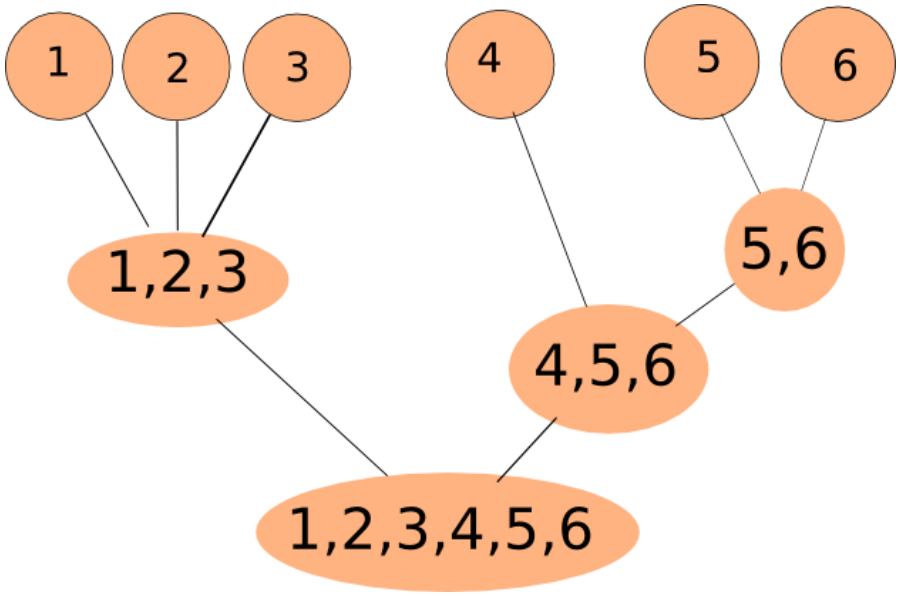
- ***Manhattan Mesafe:*** Bir aracın bloklar arasında gittiğinde bir noktadan diğerine ne kadar mesafe ile gideceğini ölçer.
- ***Hamming Mesafe:*** Bir varlığı diğerine çevirmek için gereken ekleme ve çıkarmaları ölçer. Metin kümelemede bir kelimenin diğerine olan uzaklığı için hamming mesafe kullanılabilir. Bu durumda "elim" ile "seli" kelimesi için "s" harfinin eklenmesi "m" harfinin çıkarılması için belirli değerler (ağırlıklandırma) ile bu iki kelime arasındaki mesafe ölçülebilir.

3.4.2.1. Birleştirici Hiyerarşik Kümeleme

Birleştirici hiyerarşik kümeleme (BHK), başlangıçta bütün veri varlıklarının ayrık olduğu ancak her adımda mesafe ölçütüne göre (bu mesafe ölçütü herhangi bir ölçüt olarak kullanılabilir) kümeler arası mesafeyi maksimum edecek (inter-cluster distance), ancak küme içi mesafeyi minumuma getirecek (in-cluster) şekilde birleştirmeleri yapacak bir yöntemdir. Şekil 3.6'da başlangıçta her biri ayrı olan verinin şekil 3.7'de BHK ile kümelerin nasıl oluşturduğu gösterilmektedir.



Şekil 3.6: Birleştirici hiyerarşik öncesi veri (mesafe ölçütü euclidean)



Şekil 3.7: BHK ile verinin kümelenmesi

4. SOSYAL AĞ ANALİZİ KAVRAMI

4.1. Sosyal Ağ Analizi Nedir?

Bireylerin birbirleriyle iletişimi ve etkileşimi günümüzde her geçmişle kıyas edilemeyecek kadar artmıştır. Gerek telekomünikasyon imkanlarının artması gerekse kolay yer değiştirme olanağının yaygınlaşması bu iletişimin temel dinamiklerini oluşturmaktadır. Artık insanlar birkaç saat içerisinde ülkeler arası yolculuklar yaparak yeni insanlarla yüz yüze iletişime geçebilmekte ya da dünyanın öteki ucuyla telefon görüşmesi yapabilmektedir.

Sosyal ağ en temel tarifiyle birden fazla varlığın birbiriyle etkileşimidir. Sosyal ağ insanlar arasındaki etkileşimi ifade edebildiği gibi elektrik sisteminin, ticari ilişkilerin, gen ilişkilerinin yapısını da ifade edebilir. Örneğin web sayfalarının birbirlerine bağlantı vermesi ve web sayfaları arasındaki bu ilişkiler bir sosyal ağ oluşturmaktadır.

Sosyal ağlar pek çok alanda karşımıza çıkmaktadır. Örneğin çalışmamızı da oluşturan haber gruplarına gönderilen e-postalarda da bir sosyal ağ mevcuttur. Benzer şekilde akademik çalışmada bulunan kişilerin referansları da bir sosyal ağ sunar. Sohbet odalarındaki kişilerin mesajlaşmaları ve mesajların kimden kime gönderildiği bilgisi, e-postaların bir kişiden diğerine gönderilmesi gibi konular hep sosyal ağ analizine güzel örneklerdir. Hatta günümüzde sıklıkla karşılaşılan facebook gibi siteler temelinde sosyal ağın gücünü kullanmaktadırlar.

4.2. Küçük Dünya Ağları (Small World Networks)

Küçük dünya ağları (KDA) bireylerin birbirleriyle oluşturduğu ve son günlerde üzerinde yoğun çalışmaların yapıldığı bir alandır. KDA'da bir birey diğerine herhangi bir ilişkiyle bağlı olsun. Örneğin bu ilişki tanıyor ilişkisi ise "X" kişisi "Y" kişisini tanıyor diyebiliriz. Bu durumda birbirini tanıyan bireylerin oluşturduğu ağ bir KDA'dır. 1967 yılında Stanley Milgram ve arkadaşları tarafından küçük dünya ağını

keşfetmek için bir çalışma yapılmıştır. Bu çalışmada Kansan ve Nebraska'dan Boston'daki kişilere bu kişileri tanımayan kişiler tarafından mektup gönderilecektir. Ancak bu gönderi sadece Boston'daki kişileri tanıyor olabilecek arkadaşlarına mektubu iletmek şeklinde olacaktır. Bu çalışmanın sonucunda görülmüştür ki mektupların yarısı 6 aracı kişiden daha az kişi ile başarılı şekilde hedef kişilere ulaştırılmıştır. Bu ise birbirisini hiç tanımayan kişiler arasındaki kısa yolu ortaya koymaktadır.

4.3. Neden Sosyal Network Analizi?

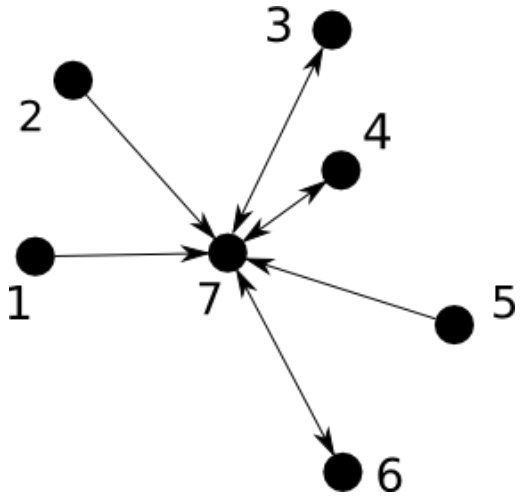
VM daha önceden kolaylıkla görülemeyen ilişkileri çıkarabilmekteydi. Örneğin kullanıcıların alışveriş alışkanlıklarına göre birbirine yakın olduğu zaman daha çok satılabilecek ürünlerin birbirlerine yakın konulması bu şekildeki ilişkilerin çıkarılması durumunda sağlanacak kazanıma güzel bir örnektir. Benzer şekilde sosyal ağlardaki yapı da buna benzer kazanımlar verebilmektedir. Sosyal ağlar gerek görsel yöntemlerle gerekse matematiksel yaklaşımlarla analiz edildiğinde oldukça ilginç ve bir o kadar faydalı sonuçlar elde edilebilir. Örneğin bir topluluk davranışı ortaya çıkarılabilir.

4.4. Sosyal Ağ Analizinin Gelişimi

Sosyal ağ analizi (SAA) günümüze kadar pek çok aşamadan geçmiştir. Wolfgang Köhler'den etkilenen alman göçmenler 1930 yılında kavramsal ve sosyal psikoloji üzerinde çalıştılar ve bu çalışmalar grup dinamikleri üzerinde çalışmalara oldukça faydalı oldu. Laboratuar teknikleri yardımıyla sosyal gruplar arasındaki bilgi akışını incelediler.

1937'de Moreno sociogram'ı buldu. Sociogram bireylerin nokta ile bireyler arasındaki ilişkilerin ise bağlarla gösterimiydi. Bu her ne kadar günümüzde çok büyük bir fikir olarak algılanmasa da 1930'larda hiç kimse toplumun yapısını analitik olarak göstermeyi bulamamıştı.

Moreno'ya göre sociogram incelendiğinde gruptaki liderler, ilişkinin akışı gibi bilgiler ortaya çıkarılabilir. Sociometrik yaklaşım bir gruptaki lider ya da star yapısı şekil 4.1'de gösterilmiştir.



Şekil 4.1: Sociometrik star

Şekil 4.1'deki sociogramda 1,2,3,4,5,6 numaralı noktalar 7 numaralı nokta üzerinde birbirleriyle bağlıdırlar. Bunun yanında 7 numaralı nokta sahip olabileceği maksimum bağlantı sayısına (n-1: n nokta sayısı) sahiptir. Görüldüğü gibi bazı noktalardan 7'ye tek yönlü bağlantı varken bazılarında ise durum çift yönlüdür.

Heider ise grup dinamikleri üzerinde denge kavramını geliştirmiştir. 1946 yılında Heider dengenin olması için en basit haliyle pozitif ve negatif olan işaretlerin benzer olması gerektiğini ifade etmiştir. Örneğin "X" kişisi "Y" kişisini seviyor olsun. "Y" ise "Z"yi seviyor olsun. Dengenin olabilmesi için "X"in "Z"yi de seviyor olması gerekmektedir demiştir.

Harary 1953 yılında graf teorisini grup dinamikleri üzerinde uygulamaya başlamıştır.

4.5. İlişkili Veri ve Getirdikleri

SAA'de varlıkların birbirleriyle olan ilişkileri incelenmektedir. VM verisi başka bir deyişle özellik verisi bu bakımdan ilişkili veriden farklıdır. Örneğin bir kişinin yaşı bir özellik verisidir. Yaş kişinin diğer bireylerden bağımsız bir özelliğidir. Ancak ilişkili veride ise durum daha farklıdır. İlişki sosyal ağdaki bireylerin bir özelliği değildir. Bireylerin oluşturduğu sosyal ağın bir özelliğidir.

İlişkili veri pek çok kaynaktan elde edilebilir. Örneğin anketler, web sayfaları, e-postalar bunlardan ilk akla gelenlerdir. Ancak verinin saklanması için veri matrisi [Galtung, 1967, Theory and Methods of Social Research] kullanılmalıdır. Veri matrisi en basit haliyle basit bir exel dökümanıdır.

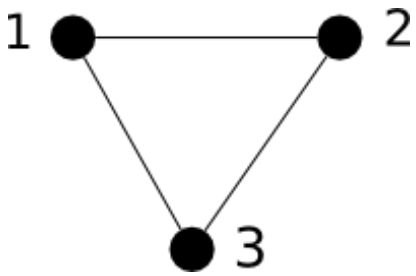
Özellik verisi de veri matrisinde saklanabilir. Bu matrisi durum-değişken matrisi adını verecek olursak (bkz. çizelge 4.1.) bu veri matrisi ilişkili veri ile kullanılamaz. İlişkili veri için durum-ilişki matrisi (bkz. çizelge 4.2, şekil 4.2) matrisi kullanılmalıdır.

Çizelge 4.1. durum-değişken matrisi

		Değişkenler		
Durumlar		Gelir	Yaş	Cinsiyet
	1			
	2			
	…			
	n			

Çizelge 4.2. durum-ilişki matrisi

	Benzerlikler			
Durumlar		X	Y	Z
	1	1	0	0
	2	1	0	0
	3	1	0	0



Şekil 4.2: Çizelge 4.2.'deki sosyal ağ için sociogram

Çizelge 4.2.'de X, Y ve Z etkinliğine 1,2,3 kişilerinin katılımıyla ilgili durum-ilişki matrisi verilmiştir. Eğer bir kişi bir etkinliğe katılmış ise matriste ilgili hücre 1 olarak yazılacak katılmaması durumunda 0 olarak yazılacaktır.

Küçük boyutlardaki ağlardan elde edilen durum-ilişki matrislerinden bile sociogram oluşturmak kolay değildir. Çünkü pek çok bağ bir düğümden diğerine rastgele açılarda ve karmakarışık şekilde gitmektedir. Bu durum nedeniyle durum-ilişki matrisindeki yapıyı temsil eden iki yeni matris elde edilmiştir.

4.5.1. Incidence ve Adjacency Matrisleri

Durum-ilişki matrisi çoğunlukla farklı sayıda satır ve sütuna sahip matrislerdir. Satırlar durumları sütunlar ise ilişkileri ifade etmektedir. Adjacency matrisi ise incidence matrisinden elde edilen kare matristir. Çizelge 4.3. deki incidence matrisi için durum-durum matrisi ve ilişki-ilişki matrisi olarak 2 yeni adjacency matris elde edilebilir (bkz çizelge 4.4. ve çizelge 4.5).

Çizelge 4.3. Incidence Matrisi (durum-ilişki matrisi)

	İlişkiler			
Durumlar		A	B	C
	1			
	2			

Çizelge 4.4. ve Çizelge 4.5. durum-durum matrisi ve ilişki-ilişki matrisi

	Durumlar	
Durumlar	1	2
	2	

	İlişkiler		
İlişkiler	A	B	C
	B		
	C		

Incidence matrisinde genel pratik olarak satırlarda durumlar sütünlarda ise ilişki bulunmaktadır. Bu yapıyı m satır ve n sütun boyutunda bir incidence matrisi için düşünecek olursak matrisin herhangi bir elemanını a(i,j) şeklide ifade edebiliriz.

Her ne kadar verinin bu şekilde tutulması gerçeklenecek görevleri kolaylaştırsa da verinin türüne göre matrisin sadece köşegeninden aşağısını saklamanın yeterli olduğu durumlar da bulunabilir.

4.6. Graf Teorisi

Matris işlemleri matematiksel işlemlerin sosyal algoritmalarla daha etkin kullanılmasına olanak verse de graf teorisi SAA için oldukça önemlidir. Sociogramların graflarla yakın benzerliği zaten görülmektedir.

Incidence matrisindeki her bir satır yada sütun için grafta bir nokta çizilir. Elde edilen adjacency matrislerinden de bu noktalar arasındaki doğrular çizilir. Adjacency matrisindeki hücrelerde bir değerin 0'dan farklı olması ilgili iki nokta arasında bir ilişki olduğunu ve bir doğru çizilmesi gerektiğini ifade eder.

Burada SAA yapılırken dikkat edilecek husus noktaların birbirine göre konumu ve doğruların uzunlukları gibi konular değil ağdaki örüntülerdir.

4.6.1. Genel Kavramlar

Yönlü ve yönsüz olmak üzere iki çeşit bağ vardır. İlişkinin bir noktadan diğerine olduğu durumlarda yönlü bağ, ilişkinin yönünün önemsiz olduğu yada olmadığı durumlarda ise yönsüz bağ kullanılır. Örneğin ***X*** kişisi ***Y*** kişisinin arkadaşı ise bu ilişki yönsüz bağla ifade edilebilirken, ***X***, ***Y***'yi seviyor dediğimizde burada yönlü bağ kullanılmalıdır. Çünkü ***Y***, ***X***'i sevmeyebilir. Şekil 4.1'de yönlü bağa şekil 4.2'de ise yönsüz bağa örnek verilmiştir.

İki nokta arasındaki bağın sadece varlık yokluk durumu dışında belirli bir ağırlık değeri de olabilir. Örneğin e-posta gönderilerinde bir kişi diğerine birden fazla e-posta göndermiş ise bu durumda iki kişi arasındaki bağ gönderilen e-posta sayısı kadar ağırlıklandırılabilir. Bu şekildeki graflara ağırlıklandırılmış graf denir.

Birbirisine bir bağ ile bağlanmış iki nokta birbirisinin komşusudur. Bir noktanın komşularının toplam sayısı o noktanın derecesini vermektedir. Eğer graf yönlü ise iç-derece ve dış-derece kavramı ortaya çıkmaktadır. İç-derece ilgili noktaya yönlenmiş bağların sayısıdır. Dış-derece ise noktadan yönlenmiş bağların sayısıdır.

Noktalar birbirleriyle doğrudan ya da diğer bağlar aracılığıyla bağlı olabilirler. Birbiri ardına sıralanan bağlar dizisine yürüyüş denir. Bir yürüyüşteki her bağ ve nokta sadece bir kez kullanılıyorsa buna yol denir. Yolun uzunluğu geçilen bağ sayısı kadardır. Ancak mesafe bu yollar arasından ilgili iki noktayı bağlayan en kısa yolun uzunluğudur. Yönlenmiş bir graftaki yol ise bütün yönlerin aynı olduğu bağlar dizisidir.

Ağ yarıçapı iki düğüm arasındaki en uzak mesafedir. Etkin yarıçap ise belirli bir oranda ağın büyük çoğunluğuna erişilebilen mesafedir.

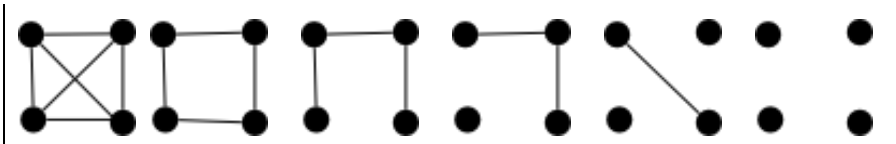
4.6.2. Ağ Yoğunluğu

Yoğunluk bir ağdaki düğümler arasındaki bağlanma düzeyini tanımlar. Bir ağdaki bütün düğümlerin birbirleriyle doğrudan bağlantılı olduğu durumdaki graf ***tam graf***tır. Yoğunluk hesaplanırken iki önemli unsur dikkate alınmalıdır. Bunlardan ilki dahil-edicilik ikincisi ise düğümlerin derecesidir. ***Dahil-edicilik*** graftaki bağlı düğüm sayısını ifade etmektedir. Bir grafta olabilecek maksimum bağ sayısı belirlidir. Dolayısıyla bir grafın yoğunluğu kendi bağ sayısının olabilecek maksimum bağ sayısına oranıdır.

$$d = \frac{\mathrm{L}}{\mathrm{n(n-1)/2}} \qquad (6.1)$$

d yoğunluk, L graftaki bağ sayısı, n ise toplam düğüm sayısıdır. Çizelge 4.6.'da dahil-edicilik ve yoğunluk örnekleri verilmiştir.

Çizelge 4.6. Dahil-edicilik ve yoğunluk

						
Bağlı nokta sayısı	4	4	4	3	2	0
Dahil-edicilik	1.0	1.0	1.0	0.7	0.5	0
Dereceler toplamı	12	8	6	4	2	0
Bağ sayısı	6	4	3	2	1	0
Yoğunluk	1.0	0.7	0.5	0.3	0.1	0

[John Scott, Social Network Analysis a Handbook, s.71]

4.6.3. Bağ Madenciliği Hedefleri

Graflar daha önce de ifade edildiği gibi birbirlerine bağlarla bağlanmış düğümlerdir. SAA yapılırken ağdaki düğümler ve bağlar birlikte ele alınarak işlenir. Bu analizin hedefinde ağdaki gizli örüntüleri ortaya çıkarmak vardır. Bu örüntüler aşağıda sıralanan hedefleri kapsamaktadır. [Lise Getoor et al., Link Mining A Survey, SIGKDD Explorations, Vol.7 Issue 2]

4.6.3. 1. Nesne Tipi Tahmini

Nesnenin ve nesnenin bağlarının özelliklerinden nesnenin türünü tahmin etmeye çalışır. Nesneye bağlı diğer düğümlerin özelliklerine de bakılabilir. Buna örnek olarak akademik bir yazının konferans yada sunum olup olmadığını bulmaya çalışır.

4.6.3. 2. Bağ Tipi Tahmini

Bağın ilgili olduğu düğümlerin özelliklerine ve ağın özelliğine bakarak bağın türünü tahmin etmeye çalışır. Örneğin tanışıklık ilişkisi verilen bir ağ için bağın aile üyelerine mi yoksa arkadaşlık bağı mı olduğunu tahmin eder.

4.6.3. 3. Bağ Temelli Nesne Sınıflandırma

Nesnenin hem özelliklerine hem de bağlarının özelliklerine göre nesneyi sınıflandırmaya çalışır. Örneğin web sayfası sınıflandırmayı düşünecek olursak, web sayfasının hem içeriği hem de web sayfasına verilen ve web sayfasının dışarı verdiği bağlar kullanılarak web sayfasının işlevi sınıflandırılabilir.

4.6.3. 4. Grup Tespiti

Çalışmamızda da hedeflediğimiz grup tespiti aslında VM'de kullanılan kümelemeyle oldukça benzerdir. Kümeleme için düğümlerin ve düğümlerin bağlarının özellikleri kullanılır. Örneğin akademik yazılarda yazıların referanslarına bakılarak ve yazının içeriği yardımıyla yazılardaki kümeler ortaya çıkarılabilir.

4.6.3. 5. Alt-graf Tespiti

Ağdaki belirli karakteristikteki alt grafları bulmayı hedefler. Örneğin protein yapıları, kimyasal alt yapılar bu şekilde bulunabilir.

4.6.3. 6. Nesne Teyidi

Birbirinden farklı olan iki nesnenin gerçekte de farklı olup olmadığını nesnenin özellik ve bağlarına bakarak bulmaya çalışır. Örneğin farklı web adresine sahip iki web sayfasının bağlarına bakarak ve içeriğine bakarak bu iki web sayfasının gerçekte aynı sayfalar olup olmadığını araştırır.

4.6.3. 7. Bağ Varlığının Tahmini

İki nesne arasındaki bağın türünü tahmin etmekten farklı olarak iki nesne arasında bağ olup olamayacağını tahmin etmeye çalışır.

4.6.3. 8. Bağ Önemi Tahmini

Bir web sayfasına pek çok web sayfası bağ vermiş olabilir. Bu durumda bu web sayfası diğer web sayfalarına göre ilgili konuda daha otoritedir. SNA bu şekilde belirli bir konuda otoriteleri bulmaya çalışabilir.

4.6.4. Bağ Madenciliğinde Karşılaşılan Zorluklar

Bağ madenciliği yaparak yukarıda sıraladığımız hedefleri gerçekleştirebiliriz ancak bunları yaparken birtakım zorluklarla karşılaşmaktayız. Bunların ilki özellik oluşturmadır. Bağ madenciliğini sınıflandırma amacıyla kullandığımızı düşünelim. Bu durumda gerek düğümlerin özellikleri gerekse bağların özellikleri bulunabilir. Bu durumda ya sınıflandırma için özelliklerin bir alt kümesini (düğümleri ve düğümlerin bağlarını en iyi temsil eden) seçmek gerekir ya da bu özellikleri birleştirerek daha az özellik (mümkünse 1 özellik) ile ağı temsil etmek gerekir. Ancak uygulamada bu pek mümkün olmamaktadır.

Etiketli ve etiketsiz verinin birlikte kullanılması da SNA için faydalı sonuçlar vermektedir. Etiketsiz verinin bağları etiketli verinin özelliklerinin kullanılmasını sağlamaktadır.

4.7. Merkezilik

Bir düğümün ağ içerisindeki merkeziliğini ölçmek için çeşitli metotlar vardır. Merkezilik bir düğüm için ele alındığında diğer düğümlerle ne kadar bağlı olduğuyla ilgilidir. Yıldız graf yapısını ele alacak olursa yıldızın merkezindeki düğüm en

merkezidir diyebiliriz. Çünkü diğer bütün düğümlere doğrudan bağı vardır. Merkezilik çeşitleri sırasıyla aşağıda açıklanmıştır.

4.7.1. Derece Merkezilik

Derece merkezilik bir düğüme gelen bağ sayısıyla ilişkildir. Daha önce de hatırlanacağı gibi iç-derece ve dış-derece bağın yönünü ifade etmekteydi. Benzer şekilde iç-derece merkezilik düğüme gelen bağlar için yapılır. Dış-derece merkezilik ise düğümden çıkan bağlar için söz konusudur.

G: = (V,E) şeklindeki n düğümlü bir graftaki *v* düğümü için derece merkezilik;

$$C_D(v) = \frac{\deg(v)}{n-1} \qquad (6.2.)$$

Düğümlerdeki derece merkezilik ağın tamamına uygulanabilir. *v** G grafındaki en derece merkeziliği en yüksek düğüm ve G' := (V',E') 6.3.'teki ifadeyi maksimize eden n düğümlü bağlı bir graf olsun.

$$H = \sum_{j=1}^{|V'|} C_D(v'*) - C_D(v_j') \qquad (6.3)$$

Buradan G grafı için derece merkezilik 6.4'te verilmiştir.

$$C_D(G) = \frac{\sum_{i=1}^{|V|}[C_D(v^*) - C_D(v_i)]}{H} \qquad (6.4)$$

H yıldız (star) graf olduğu zaman maksimum olduğundan *H* 6.5'deki gibi yazılabilir.

$$H = (n-1)\left(1 - \frac{1}{n-1}\right) = n-2 \qquad (6.5.)$$

Dolayısıyla G'nin derece merkeziliği 6.6'daki gibi olacaktır.

$$C_D(G) = \frac{\sum_{i=1}^{|V|}[C_D(v^*) - C_D(v_i)]}{n-2} \quad (6.6)$$

[en.wikipedia.org/wiki/Centrality]

4.7.2. Arada Olma Merkeziliği

Arada olma Merkeziliği (AOM) bir düğüm eğer diğer pek çok düğümün birbirleriyle en kısa yoldan bağlantısında bulunuyorsa bu düğüm için arada olma merkeziliği fazladır diyebiliriz.

4.7.3. Yakınlık Merkeziliği

Yakınlık merkezliği de graf teorisinde kullanılmaktadır. Mesafesi diğer düğümlere göre daha kısa olanlar yakınlık merkeziliği daha fazla olan düğümlerdir. Burada yakından kasıt ilgili düğümler arası mesafesi (ör. bağ sayısı) az olanlardır.

Bu üç merkezilik için özet tablosu çizelge 4.7'de verilmiştir.

Çizelge 4.7. Derece, arada-olma ve yakınlık merkeziliği kıyas tablosu

	Tanım	Etki
Derece Merkezilik	Bir düğüm için bağ sayısı	Daha fazla imkan ve alternatif sunuyor
Yakınlık Merkeziliği	Diğer düğümlere olan mesafe	Diğer düğümlerle doğrudan iletişim imkanı
Arada-olma merkeziliği	Diğer düğümler arasında bulunma	Diğer düğümler için bir düğüm/kırım noktası oluşturma

[Mohsen Jamali et al. IEEWICIACM International Conference on Web Intelligence, 2006

5. METİN MADENCİLİĞİ

5.1. Metin Madenciliği Tanımı ve Genel Kavramlar

Metin madenciliği veri madenciliği görevlerinin metin verisi üzerinde özelleşerek uygulanmasına verilen isimdir. Metin madenciliği için veri kaynağı olarak dokümanlar, makaleler, e-postalar ve aslında metin içeren bütün alanlar dahildir. Çalışmamızda e-postalar metin kaynakları olarak değerlendirilmiş ve bu alanda metin madenciliği yapılmıştır.

Metin madenciliği için kullanılacak metinler yarı-düzgündür. Yani ne tam olarak belirli bir düzensizlik içerisindedir ne de tamamen düzenlidir. Örneğin web sayfalarının içeriği gerek HTML kodlarını gerekse metni barındırmaktadır. Bu sebeple metin madenciliği yapılırken veri madenciliğindeki veri ön işleme adımları metin madenciliği için biraz daha özelleşerek yapılmaktadır.

5.1.1. Bilgi Getirim Sistemi

Bilgi Getirim Sistemi (Information Retrieval - IR) metin havuzundan kullanıcının istediği kriterlere en uygun olan doküman yada metin varlıklarını getiren sistemdir. Kullanıcı bu metin koleksiyonundan ilgisi olan metni çekmektedir. Bu yapı klasik veri tabanı sistemlerinden farklıdır. Çünkü metin koleksiyonu yarı düzenli bir yapıya sahiptir, ancak veri tabanı sistemleri tamamen düzenli yapı üzerinde çalışmaktadırlar.

5.1.2. Metin Getiriminde Temel Ölçütler

Kullanıcının metin koleksiyonundan belirli kriterlerle bazı dokümanları getirdiğini farz edelim. Peki getirilen dokümanların doğruluk oranını nasıl ölçeceğiz. Bu durumda ***duyarlılık*** ve ***doğruluk*** kavramları öne çıkmaktadır.

Kullanıcının belirli bir sorguyla doküman getirmek istediğini düşünelim. Getirilen doküman kümesi {Getirilen}, ilişkili doküman kümesi ise {İlişkili} olsun. Bu durumda;

$$duyarlılık = \frac{|\{İlişkili\} \cap \{Getirilen\}|}{|\{Getirilen\}|} \quad (5.1)$$

Ve doğruluk;

$$doğruluk = \frac{|\{İlişkili\} \cap \{Getirilen\}|}{|\{İlişkili\}|} \quad (5.2)$$

Şeklindedir.

5.2. E-Posta Listelerinde Metin Madenciliği

Çalışmamız e-postaları veri olarak kullanmaktadır. E-postalar maalesef oldukça keyfi, rastgele ve hatalı ifadeler içerebilmektedir. Bu ise oldukça gürültülü bir veri ile çalıştığımızı göstermektedir. Ayrıca veri düzgün yazılmamış kelimeler içerebildiği gibi reklam, web sayfası, formül gibi ifadeler de barındırabilir. Bu durumda metin madenciliği yapmak daha zor hale gelmektedir.

E-posta listeleri farklı sayıdaki kişi ve e-posta sayısına sahip olabilir. E-posta listeleri kök dizini için ***alt.politics*** seçilmiştir. Ancak bu dizin içinden en uygun e-posta listesi seçilmelidir. Bunun için bu grubun altındaki bütün eposta listeleri geliştirilen bir uygulamayla gezilmiş (uygulama bir web crawler gibi çalışmaktadır) ve bu listelerde mesaj sayıları en fazla olanlar (***alt.politics.democrats, alt.politics.bush*** vb.) seçilmiştir.

Dil olarak İngilizce bir e-posta listesin seçilmesindeki temel sebep Türkçe metinlerin kullanıcılar tarafından İngilizce harflerle yazılmasıdır. Örneğin

“seçiyorum” kelimesini “seciyorum” olarak yazan pek çok kullanıcı bulunmaktadır. Bu durum veri ön işleme aşamasında problemlere sebep olacaktır.

alt.politics dizini içerisinden bizim seçimimiz ***alt.politics.bush*** olmuştur. E-posta listesini seçmemizde birden fazla etken rol oynamıştır. E-posta sayısının fazla olması, ilgili eposta grubunu çalışma için en etkin yapmamaktadır. Çalışmamız için posta sayısının çokluğu kadar farklı kişilerin aynı konuya mesaj atmaları da önemlidir. Sosyal ağ analizi yapmamıza olanak veren bilgi kişilerin birbirlerinin mesajlarına cevap vermeleridir.

Bu işlemin ardından e-postalar ilgili gruptan ***mesajın id’si*** (message-id), mesajı ***gönderen*** (from), mesajın ***konusu*** (subject) ve ***mesaj içeriği*** grubun ilk gününden itibaren gönderilmiş bütün mesajlar için elde edilmiştir.

Veri boyutu çok büyük olmadığından örnekler üzerinde değil verinin tamamı üzerinde çalışılmıştır. Verinin tamamında çalışılmasının diğer sebepleri ise [Yablonsky, L. et al. 1962] ve [Kerr et al. 1957]’da belirtildiği örneklerin ağı mükemmel şekilde temsil etmekten yoksun olmasıdır [John Scott, 2000 “Social Network Analysis”, Second Edition, SAGE Publications].

İnternet beklendiği gibi pek çok çöp verinin de bulunduğu bir ortamdır. Aynı durum e-posta listeleri için de geçerlidir. E-posta listesinden 4210 tane e-posta sisteme alınmış ve bunlardan 121 tanesi bozuk e-posta yapısında olduğundan listeye eklenememiştir. Bozuk olarak nitelenen epostalarda dil kodlaması problemi, gönderen bilgisinin bulunmaması gibi problemler bulunmaktadır. %2’lik bir kayıpla epostalar sisteme alınabilmiştir.

5.3. Bilinen Problemler

E-postalar kullanıcılar tarafından gönderilen ve içerisinde her şeyin olabileceği metinler olduğunu daha önce de belirtmiştik. Ancak tek problem bu değildir. Bunun yanında;

- Aynı kişiye ait birden fazla e-posta adresinin bulunması
- Hatalı yazılan kelimeler
- Reklam amaçlı gönderiler
- www, http gibi bağlantı bildiren ifadeler
- C, C++, python gibi dillere ait kaynak kodları içeren ifadeler
- Farklı e-posta istemcilerinin farklı eklenti yazılarını e-postalara eklemeleri

Gibi problemli konular da bulunmaktadır. Bazı çalışmalarda [Christian Bird et al. 2006] birden fazla e-postaya sahip kullanıcılar tespit edilmeye çalışılmıştır. Ancak çalışmamızda kaynak kod içeren e-posta bulunmamaktadır. Diğer problemler ise geliştirilen uygulama ile minimize edilmiştir.

E-posta listelerindeki her bir gönderi birbirinden bağımsız dokümanlar olarak değerlendirilebilir. Bu şekilde yaklaşıldığında her bir doküman için kelime frekans vektörü ortaya konulabilir. Ancak her bir doküman için oluşturulacak vektör çok büyük boyutlarda olabilir. Bu nedenle vektör boyutunu belirli bir sayının altında tutmak için;

- Dur kelimeleri (stop-words)
- Minimum terim frekansı
- TF/IDF değeri

Yöntemleri kullanılmıştır.

5.3.1. Dur Kelimeleri

Dur kelimeleri bir metinde sıklıkla görmeyi umduğumuz kelimeleridir. Bu kelimeler Türkçe için örnek verecek olursak: ***ile, ve, veya*** gibi kelimelerdir. Dur kelimeleri metin madenciliği için gürültü gibi düşünülecek olsa da bir dilde bulunabilecek dur kelimelerinin neler olduğu bellidir. Bu şekilde hazırlanmış olan bir listeden bu kelimeler tespit edilerek metin içerisinde görüldüğünde bu kelimeyi işleme almamak mümkündür. Böylelikle metindeki gürültü belirli bir miktarda ortadan kaldırılabilir.

Her metin için ayrıca dur kelimeleri de olabilir. Örneğin programlama ile ilgili bir metinde programlama kelimesinin sıklıkla geçmesi beklenen bir durumdur. Dolayısıyla bu kelime dur kelimesi olarak değerlendirilmelidir. Benzer şekilde ***alt.politics.bush*** eposta listesinde de ***bush*** kelimesinin dur kelime listesine eklenmesi gerekmektedir.

Bunun yanında dur kelimeleri için ek bir koruma mekanizmasıda TF-IDF olarak düşünülebilir. Çünkü dur kelimeleri hem ilgili e-posta içerisinde hem de bütün e-postalar içerisinde sıklıkla karşılaşılacak ifadeler olduğundan düşük TF-IDF değerine sahip olacaklardır.

5.3.2. Kök Bulma

Her e-posta için kelime frekans vektörü hazırlanırken çizelge 5.1.'deki gibi bir listeyle karşılaşmak olasıdır. Bu listede aslında aynı kelime olan ancak çeşitli ekler nedeniyle yanlış olarak yeni bir kelimeymiş gibi algılanıp frekans değeri hesaplanabilecek kelimeler görünmektedir (Ör. Signal ve Signals). Kök bulma işlemi işte bu yanlışlıkların belirli ölçüde kaldırılması için gerekli bir işlemdir. Elde edilen kelime listesinde kök bulma algoritmalarından snowball kullanılarak belirli ölçüde bu hatanın önüne geçilmiştir. Bu algoritma İngilizce kelimeler için kök bulma işlemini yapmaktadır. Böylelikle "Democrat" ile "Democrats" aynı kelime olarak tespit edilip frekansı ona göre hesaplanmış olacaktır. (Bkz. Çizelge 5.1)

Çizelge 5.1. Örnek E-postalar

Message-ID	**Gönderen**	**Konu**	**İçerik**
1	MailAddress1	Bush visits iraq	Signal signals visits…
2	MailAddress2	Re: Bush visits iraq	Came across democrat
3	MailAddress3	Heath payments	Decrease democrats…
4	MailAddress4	Re: Heath payments	İncreased last years…
5	MailAddress5	Re: Bush visits iraq	Tax modify signal…

5.4. TF-IDF Değeri

Adım 5.3.1 ve 5.3.2. yapılsa bile vektörün boyutu çok fazla olabilir. Bu durumda vektör belirli bir boyutun altında tutulmalıdır. Bunun için TF-IDF değeri her terim için hesaplanarak en anlamlı olanlar vektöre dahil edilerek vektör boyutu indirgenir. TF-IDF basit ama oldukça etkin bir boyut indirgeme yöntemidir [Ramos, 2003]. TF her e-posta için ayrı ayrı yapılan bir hesaplamadır.

$$tf(i,j) = \frac{n(i,j)}{\sum_k n(k,j)} \qquad (5.3)$$

TF terim frekansını ifade etmektedir. n(i,j) terimin eposta içerisinde ne kadar geçtiğini, $\sum_k n(k,j)$ ise bütün terimlerin ilgili eposta içindeki frekanslarının toplamını ifade etmektedir. Böyle bir normalizasyona gidilmesinin sebebi uzun epostalardaki yüksek terim frekanslarının oluşma ihtimalinin önene geçmektir.

IDF ise ters terim frekansıdır. Kelimenin kaç tane doküman içerisinde geçtiğiyle ilgili olan bu hesaplamada da normalizasyon yapılmaktadır.

$$idf(i) = log \frac{|D|}{|\{d : t(i) \in d\}|} \quad (5.4)$$

|D| doküman sayısını, payda ise terimin kaç tane doküman içerisinde geçtiğinin sayısıdır.

Her bir terimin toplam ağırlığı ise TF x IDF hesaplaması ile bulunacaktır. Bu şekilde hesaplanan ağırlıklar içerisinden en büyük "n" tane terim alınarak vektör uzayının boyutu n'e indirgenir.

Çalışmamızda n değeri 100 olarak kullanılmıştır. Ancak elde edilen vektörde ***com, wrote*** gibi kelimeler bulunmakta bunun yanında sayılarda görülebilmektedir. Bu kelimeler vektörden kullanıcı tarafından çıkartılmıştır.

6. E-POSTA LİSTELERİNDE SOSYAL AĞ ANALİZİ

E-posta göndereni ve alıcısı olan ve bu haliyle basit bir ikili bir ilişkinin söz edilebildiği bir sosyal yapıdır. Sosyal ağ analizi varlıkların birbirleriyle olan ilişkilerinin yapısı üzerinden çıkarımlar yapmamızı sağlayan bir veri madenciliği yaklaşımıdır. Sosyal ağ analizi graf ve link madenciliği ile de yakından ilişkilidir.

Metin madenciliği ile sosyal ağ analizini kıyaslamamıza olanak veren öngörümüz kişilerin benzer konular etrafında mesaj yazacağı bilgisidir. Yani programlama ile ilgili kişiler genellikle programlama konusu hakkında yazı yazacaklardır. Dolayısıyla metin madenciliği yapıldığında programlama konusunun ait olduğu kümede genellikle bu konu ile ilgili kişilerin bulunmasını bekleriz. Benzer şekilde programlama ile ilgili kişiler benzer sosyal ağ yapısında olacaklardır. Çünkü atılan bir epostaya yine cevap olarak bu konuyla ilgili olan diğer kişiler cevap verecektir. Böylelikle ilgili kişiler kendi aralarında yoğun bir bağ (link) yapısına sahip bir sosyal ağ oluşturacaktır. Beklenen her iki durumda oluşan kümelemelerin birbiriyle ilişkili olmalarıdır.

Pek çok çalışmada ya sadece sosyal ağın yapısı değerlendirilmiş [Gattrel 1984b], ya da konu içeriği metin madenciliğine salt olarak tabi tutulmuştur. Çalışmamızda her iki açıdan ayrı ayrı kümelemeye tabi tutulmuş olan e-postaların birbiriyle olan tutarlılığı değerlendirilmiştir.

6.1. Sosyal Ağ Analizi için Veri Ön İşleme

Sosyal ağ analizi için de veri ön işleme yapılmalıdır. Çünkü bazı konular sadece bir kişinin (yani konuyu açan kişi) gönderisini barındırmaktadır. Bu durumdaki veri sosyal ağ için outlier durumdadır ve ağı yorumlamada hiçbir katkısı olmayacaktır. Bu sebeple en az iki posta bulunan konular dışındaki konular (yani tek eposta bulunduran konular) listeden çıkarılmıştır. Bu şekilde olan e-posta sayısı oldukça fazladır. Muhtemelen bu e-postalar reklam amaçlı ya da belirli bir mesajı gruba

aktarmak için atılan e-postalardır. Sosyal ağ analizi için en az bir e-posta kısıtı elimizdeki veriye uygulanınca 4210 postadan 948 posta elenmiştir. Yani verinin %22'si sosyal ağ analizi için uygun değildir. Sonuç olarak elimizde sosyal ağ analizi ve metin madenciliği için 3262 e-posta bulunmaktadır.

6.2. Veri Gürültüsü ile Uğraşmak

E-posta listelerindeki gönderilerin tamamı anlamlı e-postalar olmayabilmektedir. Daha öncede ifade edildiği gibi sıklıkla reklam amaçlı ya da sadece kızgınlık, sinir gibi hallerle belirli gönderilere cevap olarak gönderilen iletilere rastlamak mümkündür. Bu gibi iletiler sosyal ağ analizi için veriyi daha anlamlı kılarken, metin madenciliği açısından problemlidir.

Bir kişinin başka birisinin iletisine sinir haliyle cevap yazması cevap yazan kişiyi cevap yazdığı kişiye kuvvetli bir şekilde bağlar. Aslında bu durum aralarındaki link ikili bir link değilse (link var yada yok) daha kuvvetli bir ağırlıklandırmayla değerlendirilebilir. Metin madenciliği bakımından çalışmamızda veri içerisindeki bu gibi konu dışı e-postalar gürültü olarak kabul edilse bile bu durumda metin madenciliği ve sosyal ağ analizi kümeleme yöntemlerinin gürültü durumundaki davranışını gözlemleyebileceğiz. Zaten bir başka e-postaya tepki olarak gönderilen e-postalar sosyal ağ analizine katkı sağladığını daha önce de belirtmiştik. Ancak reklam amaçlı gönderilerin tamamen gürültü olduğunu ve veriyi bozduğunu kolaylıkla söyleyebiliriz.

6.3. Metin Madenciliği Alan Adı Değişikliği

Elimizdeki vektör uzay modeline çevrilmiş e-postalarda kümeleme yapılması için k-means algoritması kullanılmıştır. k = 2 ve k = 4 için kümeler bulunmuştur. Dolayısıyla her e-posta bir kümeye atanmıştır. Ancak fark edileceği gibi kişilerin grubu henüz hesaplanamamıştır. Bu işlem için e-posta varlık alanından kişi alanına bir çevrim yapılacaktır. Bu çevrim için greedy bir yaklaşım kullanılmaktadır.

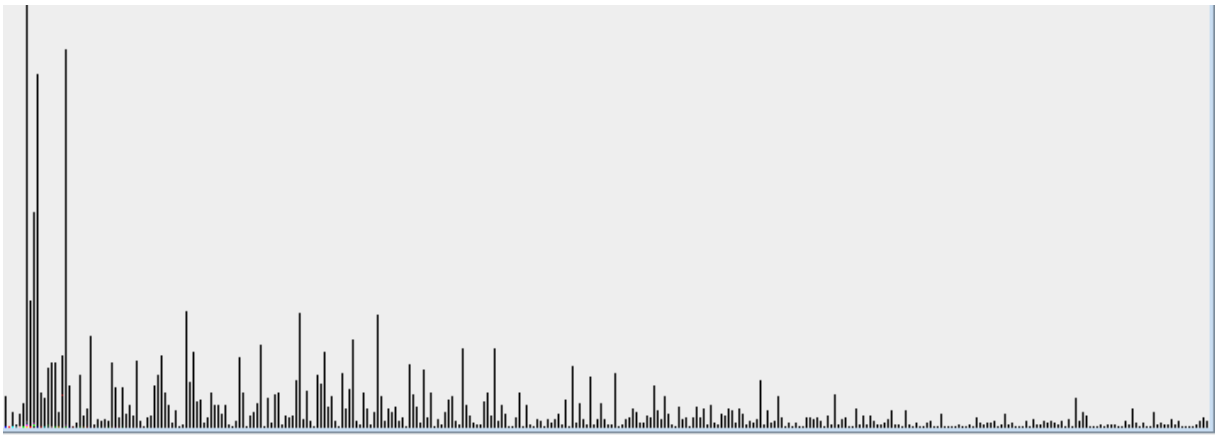
Greedy Algoritma:

Her bir kişi için gönderdiği epostaların konularının frekanslarını say. Bu frekanslardan en fazla olanın kümesi ile kişiyi etiketle.

Buna göre her bir kişi için kişinin gönderdiği epostaların sonuç kümeleri arasından en fazla hangi kümede eposta yazmış ise kişi o kümeyle etiketlenir.

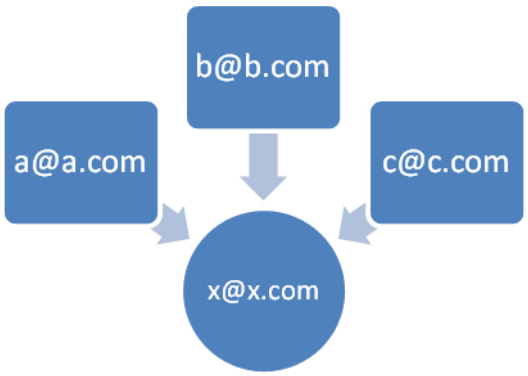
6.4. Sosyal Ağ Algoritmalarının Uygulanması

E-posta listeleri metin içeriği dışında belirli bir bağlantı (link) verisi de içermektedir. Bağlantı bilgisi bir kişiden diğerine gönderilen e-posta olarak göze çarpmaktadır. Burada konuyu açan kişi ve bu kişiye verilen cevaplar düşünüldüğünde ortaya sosyal ağ analizi yapılabilecek bir yapı çıkmaktadır. Ağdaki kişi ile e-posta sayılarını gösteren grafik göz önüne alındığında scale-free ağların özelliklerinden power law gözlemlenebilir [Albert Laszlo et al. 2003] (Bkz. Şekil 6.1).



Şekil 6.1: Kişi E-posta Sayısı Grafiği. X: kişi Y: E-posta Sayısı

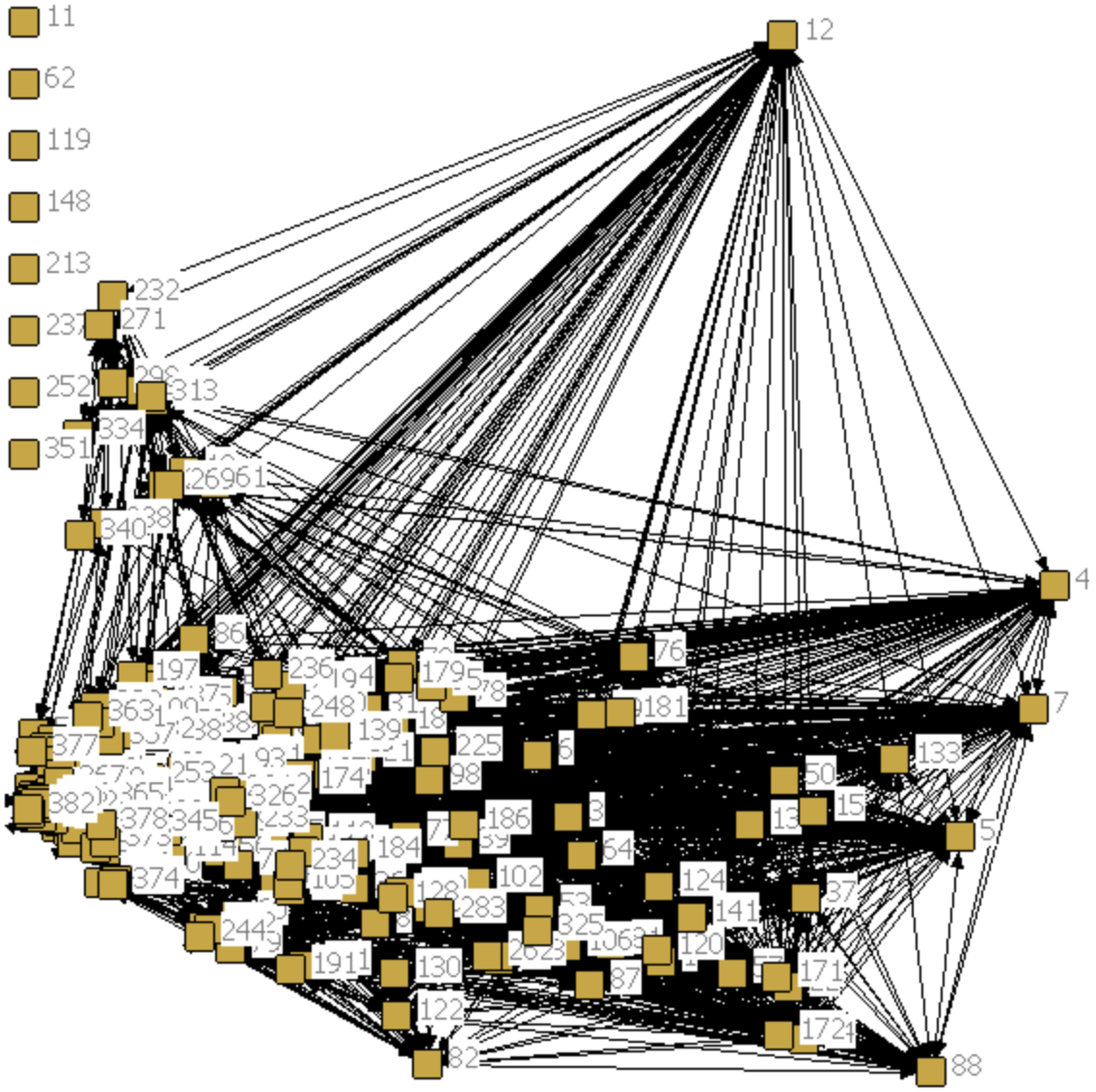
Dikkat edileceği gibi az sayıda kişi çok e-posta göndermiş ve çok sayıda kişi belirli bir eşiğin altında e-posta göndermiştir. Genel bir e-posta listesindeki tek bir e-posta için durum şekil 6.2'deki gibidir.



Şekil 6.2: Bir E-posta'daki İlişki[1]

Görüldüğü gibi 4 kişini oluşturduğu bu grup bir sosyal ağ teşkil etmektedir. Bu veri bir matriste tutulmuştur. Bizim çalışmamızda 382 farklı kişi için bu ağ yapısının görünümü aşağıdaki gibidir. (bkz. şekil 6.3)

[1] x@x.com, a@a.com, b@b.com, c@c.com eposta adresleri örnek amaçlıdır.



Şekil 6.3: alt.politics.bush ağ çizimi

Ağ oluşturulurken bağların yönlü ya da yönsüz olmasının yanında ikili yada ağırlıklı olması da ortaya çıkacak sonucu etkilemektedir. Örneğin yukarıdaki ekran çıktısında ikili ve yönlü bağlar üzerinde bir gösterim yapılmıştır.

Çalışmalarımızı yapacağımız matris 382x382 boyutlarında ve ağırlıklandırılmış yönsüz bir matristir. Matrisin hücreleri bir kişiden diğerine e-posta gittiği takdirde ilgili hücre bir arttırılacak şekilde doldurulmuştur.

$$M_{382x382} = \begin{bmatrix} 0 & 12 \cdots 24 & 4 \\ \vdots & \ddots & \vdots \\ 4 & 24 \cdots 13 & 0 \end{bmatrix}$$

6.5. Ağın Özellikleri

"alt.politics.bush" e-posta listesinde sosyal ağ analizi yapmak için hazırlanan matris üzerinde gözlemlenen sonuçlara göre:

- Ağı en fazla olan kişinin ağ büyüklüğü 114 kişi
- Ağdaki iki kişi arasındaki ortalama mesafe 2.765 bağ
- Derece merkezlilik (degree centrality) ortalama %26.41
- Arada olma merkezliliği (betweenness centrality) 3 kişi için yüzde 10 üzerindedir. Bu üç kişi otoritedir de denilebilir.

7. SONUÇLAR

Metin kümelemede sadece k-means algoritması kullanılmıştır. K-means için mesafe fonksiyonu olarak euclidean mesafe kullanılmıştır. Çalışmamızda sonuçları daha detaylı görebilmek için metin kümelemede kullanılan fonksiyonları sabit tutup sosyal ağ'daki algoritmaları değiştirerek sosyal ağ algoritmalarının metin kümelemelerine yakınsamasını test ettik. Sonuçlar ve algoritmalar aşağıda verilmiştir.

Çizelge 7.1. Metin Kümeleme ve Sosyal Ağ Analizi Uyum Sonuçları

Algoritma	k-means (k = 2)	k-means (k = 4)
CONCUR	%58	%36
FACTIONS	%55	%26

CONCUR algoritması en uyumlu algoritma olarak ortaya çıkmıştır. Bunda FACTIONS algoritmasının veri üzerinde eşit boyutta alt gruplar yaratma meyili bu farka sebep olmuştur. FACTIONS algoritması için mesafe ölçütü olarak hamming uzaklık kullanılmıştır. Görüldüğü gibi her iki algoritmadan 2 küme için alınan sonuçlar birbirine yakındır. Ancak 4 küme için %10'luk bir farkla CONCUR daha "uyumlu"dur denilebilir. Uyumlu olması doğru olmasını ile aynı anlama gelmemektedir.. Uyumlu olması k-means ile daha tutarlı sonuçlar ortaya koyduğunu ifade etmektedir.

Çalışmamız algoritmaların seçiminin önemli olduğunu ancak bunun yanında küme sayısının da seçimde önemli olduğunu ortaya koymaktadır. Burada dikkat edilmesi gereken husus bir algoritmanın diğerine göre daha başarılı olduğu değil metin madenciliğinde kullanılan algoritma ile uyumluluğudur.

Yukarıdaki sonuçlara göre 2 kümeli gruplamada kullanıcılar en fazla e-posta gönderdikler konu ile ilgili olan diğer kişilerle yaklaşık %60 oranında aynı sosyal

grupta bulunmaktadır. Bu ise bizim çalışmamızın başında savunduğumuz tezimizi destekler niteliktedir. Ancak 4 kümeli gruplarda ise başarımın düşmesi sosyal ağın ortaya koyduğu yapının metin içeriğinden farklılaştığını göstermektedir. Ancak yine unutulmamalıdır ki bizim eposta isim alanından kişi isim alanına geçiş için uyguladığımız greedy algoritmanın da sonuca etkisi olduğu göz önünde bulundurulmalıdır.

8. GELECEK ÇALIŞMALAR

Metin madenciliği için farklı algoritmalar ve farklı benzerlik ölçütleri kullanılarak bu algoritmaların CONCUR ve FACTIONS ile uyumu araştırılacaktır. Ayrıca uygulanan greedy algoritma daha da geliştirilebilir. Veri ön işlemedeki farklı e-posta istemcilerinin eklediği mesajlar daha başarılı bir şekilde filtrelenebilir. Bunun yanında Türkçe e-posta listelerine de mevcut çalışmalar uygulanmalıdır. Türkçe listelerde genellikle “ı” harfi yerine “i”, “ç” harfi yerine “c” harfi gibi Türkçe ifadeler yerine İngilizce ifadelerle Türkçe yazma gibi bir problem mevcuttur. Bu durum metin madenciliğinde kelime köklerini bulurken zemberek gibi kütüphanelerin başarısını düşürmektedir. Bunun için dil sözlüğü ve ilgili kelimeye benzerlik değerleri ile bu problemin önüne geçilebilir.

9. ÖNERİLER

Gerek sosyal ağ analizi yapılarak bulunan kümelerde gerekse metin madenciliği ile ortaya çıkarılan kümelerde benzerlikler ortaya konmuş olsa da, her iki kümelemenin birlikte ele alınması daha sağlıklı bir kümeleme sonucu verecektir. Bunun için her iki metodu birleştiren algoritmalar veriye uygulanmalıdır. Yani sosyal ağın yapısıyla metnin içeriğini birlikte ele alan yöntemler uygulanabilir.

KAYNAKLAR

[1] U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy. Advances in Knowledge Discovery and Data Mining. AAAI/MIT Press, 1996.

[2] Morzy, Mikolaj, "Advanced database structures for effective association rule mining", phD, Pozna´n University of Technology, 2004

[3] J. Han and M. Kamber. Data Mining: Concepts and Techniques. Morgan Kaufmann, 2000.

[4] S.K Madria, W.K Bhowmick,.ve E.P NG, "Research Issues in Web Data Mining". In Proceedings of Data Warehousing and Knowledge Discovery", First International Conference. DaWak1999. ss 303-312.

[5] Tug E. Genetik Algoritmalar ile Tıbbi Veri Madenciligi, Yüksek Lisans Tezi, 2005, Konya.

[6] Sinan Ataseven, Üniversitelerin Adaylar Tarafından Tercih Edilme Desenlerini Veri Madenciliği Yöntemleri ile Belirleyen Bir Model Önerisi, Yüksek Lisans Tezi, 2008

[7] en.wikipedia.org (Supervised Learning)

[8] Vapnik, V. (1995) The Nature of Statistical Learning Theory, Springer-Verlag Publishing, New York, s.187

[9] Vapnik, V. (1998a) Statistical Learning Theory, John Wiley Publishing, New York, s.768.

[10] Vapnik, V. (1998b) The Support Vector Method of Function Estimation, In Nonlinear Modelling Advanced Black Box Techniques, Kluwer Academic Publishers, Boston, s55-85

[11] R.O. Duda, P.E. Hart, and D.G.Stork. Pattern Classification (2nd Edition). Wiley-Interscience, 2000

[12] Johan Galtung, Theory and Methods of Social Research, 1967, 543 pp.70

[13] John Scott, 2000 "Social Network Analysis a Handbook", Second Edition, SAGE Publications

[14] [Lise Getoor et al., Link Mining A Survey, SIGKDD Explorations, Vol.7 Issue 2]

[15] [en.wikipedia.org/wiki/Centrality]

[16] [Mohsen Jamali et al. IEEWICIACM International Conference on Web Intelligence, 2006]

[17] Yablonsky, L. (1962) “The Violent Gang. Harmondsworth: Penguin”

[18] Kerr, C. And Fisher, L.H. (1957) “Plant Sociology: The Elite and the Aborigines” in Komarovsky (ed), 1957

[19] Christian Bird, Alex Gourley,Prem Devanbu, Michael Gertz, Anand Swaminathan (2006) “Email Social Networks” MSR’06, May 22–23, 2006, ACM

[20] Juan A. Ramos, 2003 "Using TF-IDF to Determine Word Relevance in Document Queries", iCML-03

[21] Gattrel, A.C. (1984b) “The Growth of a Research Speciality” Annals of the Association of American Geographers, 74

[22] Albert Laszlo, Barabasi, Eric Bonabea(2003) “Scale Free Networks” Scientific American May 2003

[23] Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C. and Wirth, R., 1996. CRISP-DM 1.0: Step-by-Step Data Mining Guide, http://www.crisp-dm.org/CRISPWP-0800.pdf

[24] Freeman, L. C. (1979). “Centrality in social networks: Conceptual clarification” Social Networks, 1, 215-239.

[25] Mark S. Granovetter (1973) “The Strength of Weak Ties” American Journal of Sociology, Volume 78, Issue 6, 1360-1380

ÖZGEÇMİŞ

1981 yılında Şanlıurfa'da doğdu. İlköğretimi Vatan İlkokulu'nda, orta ve lise eğitimini Şanlıurfa Anadolu Lisesi'nde tamamlayarak 1999 yılında İstanbul Üniversitesi bilgisayar mühendisliği bölümüne girdi. 2004 yılında mezun oldu. 2005 yılından beri TÜBİTAK UEKAE'de çalışmaktadır.