T.C.
BEYKENT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
MATEMATİK BİLGİSAYAR ANABİLİM DALI
BİLGİ TEKNOLOJİLERİ BİLİM DALI

METİN MADENCİLİĞİ YÖNTEMLERİ İLE SOSYAL MEDYADAN TOPLANAN FOTOĞRAFLI PAYLAŞIMLARIN, METİN – FOTOĞRAF EŞLEŞMESİNİN İNCELENMESİ

Yüksek Lisans Tezi

Tezi Hazırlayan:
Aykut DEMİREL

İstanbul, 2015

T.C.
BEYKENT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ
MATEMATİK BİLGİSAYAR ANABİLİM DALI
BİLGİ TEKNOLOJİLERİ BİLİM DALI

METİN MADENCİLİĞİ YÖNTEMLERİ İLE SOSYAL MEDYADAN TOPLANAN FOTOĞRAFLI PAYLAŞIMLARIN, METİN – FOTOĞRAF EŞLEŞMESİNİN İNCELENMESİ

Yüksek Lisans Tezi

Tezi Hazırlayan:
Aykut DEMİREL

Öğrenci No:
130862009

Danışman:

Doç. Dr. Gökhan SİLAHTAROĞLU

İstanbul, 2015

YEMİN METNİ

Yüksek Lisans Tezi olarak sunduğum "**Metin Madenciliği Yöntemleri İle Sosyal Medyadan Toplanan Fotoğraflı Paylaşımların, Metin – Fotoğraf Eşleşmesinin İncelenmesi**" başlıklı bu çalışmanın, bilimsel ahlak ve geleneklere uygun şekilde tarafımdan yazıldığını, yararlandığım eserlerin tamamının kaynaklarda gösterildiğini ve çalışmamın içinde kullanıldıkları her yerde bunlara atıf yapıldığını belirtir ve bunu onurumla doğrularım 13/09/2015

Aykut DEMİREL

T.C.
BEYKENT ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZ/PROJE SAVUNMA SINAVI SONUÇ TUTANAĞI

Beykent Üniversitesi
Fen Bilimleri Enstitüsü Müdürlüğü'ne,

Aşağıda tez/proje adı belirtilen yüksek lisans öğrencisi 130862009 ..... no'lu ..Aşkın.....Damla.....'in 14/03/15 tarihinde yapılan tez/proje savunma sınavı[1] sonucunda ..52.. dakika süreyle sunduğu ve savunduğu tezi/projesi hakkında[2] oybirliğiyle, ..kabul..... kararı verilmiştir.

Bilgilerinize saygılarımızla arz ederiz.

Anabilim Dalı : Matematik Bilgisayar
Programı : Bilgi Teknolojileri
Tez/Proje Başlığı[3] : Metin Madenciliği Yöntemleri ile Sosyal Medyadan Toplanan Fotoğraflı Paylaşımların, Metin - Fotoğraf Eşleşmesinin İncelenmesi

---

Tez/Proje Sınav Jürisi		Öğretim Üyesi	İmza
Danışman	:	Doç. Dr. Gökhan SILAHTAROĞLU	
Üye	:	Yrd.Doç.Dr. Edite Saykol	
Üye	:	Doç.Dr. Kazım Sarı	

---

[1] Jüri üyeleri söz konusu tezin kendilerine teslim edildiği tarihten itibaren en geç bir ay içinde toplanarak öğrenciyi tez savunma sınavına alır. Belirlenen günde yapılamayan jüri toplantısı, katılanların hazırladığı bir tutanakla enstitü yönetimine bildirilir. Bu durumda jüri en geç onbeş gün içinde toplanarak adayı tez savunma sınavına alır. Tez savunma sınav süresi en az 45 dakikadır. Yüksek lisans tez savunma sınavı, tez çalışmasının sunulması ve bunu izleyen soru-yanıt bölümlerinden oluşur ve dinleyiciye açıktır. (Beykent Lisansüstü eğitim ve Öğretim Yönetmeliği-Madde30-3)

[2] Tez sınavının tamamlanmasından sonra jüri, tez hakkında "kabul", "düzeltme" veya "red" kararı verir. Jüri başkanı, jüri üyelerince imzalanmış sınav tutanağını, tez sınavını izleyen üç gün içinde ilgili enstitü yönetimine teslim eder. Tezi başarısız bulunan öğrencinin Enstitü ile ilişiği kesilir. Tezi hakkında düzeltme kararı verilen öğrenci en geç üç ay içinde gerekli düzeltmeleri yaparak ve yönetmelikte belirtilen usullere uygun olarak tezini aynı jüri önünde yeniden savunur. Bu savunma sınavında da tezi kabul edilmeyen öğrencinin enstitü ile ilişiği kesilir.(Beykent Lisansüstü eğitim ve Öğretim Yönetmeliği-Madde30-4)

[3] İleride doğabilecek aksaklıklarınengellenmesiiçintezinbaşlığınınyazılmasıgerekmektedir.

TEŞEKKÜR

Tezin en zor kısmı olan çalışma konusunun tespitinden itibaren değerli bilgilerini benimle paylaşan ve beni yönlendiren danışman hocam Gökhan Silahtaroğlu'na teşekkürlerimi sunarım.

Aykut DEMİREL

13.09.2015

Adı ve Soyadı : Aykut DEMİREL
Danışmanı : Doç. Dr. Gökhan SİLAHTAROĞLU
Türü ve Tarihi : Yüksek Lisans Tezi, 2015
Alanı : Bilgi Teknolojileri
Anahtar Kelimeler : Metin Madenciliği, Sosyal Medya, Metin-Fotoğraf Eşleşmesi

ÖZ

METİN MADENCİLİĞİ YÖNTEMLERİ İLE SOSYAL MEDYADAN TOPLANAN FOTOĞRAFLI PAYLAŞIMLARIN, METİN – FOTOĞRAF EŞLEŞMESİNİN İNCELENMESİ

İnternet kullanımının günden güne artması, insanların internet üzerinden etkileşime girebilecekleri sosyal mecraları doğurmuştur.

Bu mecralarda işitsel iletişim yerine, görsel ve yazılı iletişimi kullanarak yüzbinlerce kişi aynı anda görsel ve metin paylaşımları yapmaktadır. Yapılan bir paylaşım kişinin kayıtlı olduğu gruplar, takipçileri, arkadaşları gibi etkileşim alanlarına düşmekte ve bu sayede de dahil olduğu gruplar, takipçileri ya da arkadaşları ile tanımadığı bir grup, bir kişi ya da kişilerin önüne etkileşim olarak düşebilmektedir. Ve dahası bir paylaşım ile o paylaşımın milyonlara erişmesi mümkün olabilmektedir.

Bu tezde etkileşimin görsel ve yazılı olarak yapıldığı bu mecralarda paylaşılan metinler ile fotoğraflar arasında bir ilişki olup olmadığı araştırılmıştır.

İlk olarak sosyal mecralardan bu tez için geliştirilmiş yazılım ile veriler çekilmiştir.

Çekilen verilerin paylaşılmış fotoğrafları yine bu tez için geliştirilmiş olan renk analiz yazılımında analiz edilmiştir. Renk analiz programı her fotoğrafın her bir pixelini ele alarak renk ağırlıklandırılması yapmıştır.

Çekilen ve renk analizi tamamlanan veriler, veri analiz programına aktarılmıştır. Veri analiz programı olarak ise knime kullanılmıştır.

Aktarılan bu veriler karar ağacı ile sınıflandırılmıştır. Sınıflandırılmış olan veriler ışığında ise son olarak tahminleme yapılmıştır.

Name and Surname : Aykut DEMİREL
Supervizor : Assoc. Dr. Gökhan SİLAHTAROĞLU
Degree and Date : Master, 2015
Major : Information Technologies
Key Words : Text Mining, Social Media, Text- Photograph Match

ABSTRACT

TEXT FOTOGRAPH MATCH ANALYSIS WITH TEXT DATA MINING TECHNICS FOR COMPILED POSTS WHICH INCLUDE FOTOGRAPH THROUGH SOCIAL MEDIA

Increasing internet usage day by day created social platforms in which people can interact with each other through internet.

In these platforms, thousands of people can share visual and textual posts at the same time as using visual and textual communication instead of auditory communication. A shared post falls into interaction areas like groups which the sharing individual registered for, his followers and friends so that it can fall before of the groups he involved in, his followers and friends as well as a group, an individual or individuals who he does not know as an interaction. Therefore a post can reach millions of people by only one sharing.

In this thesis, it is conducted that whether or not there is a relationship between texts and photographs which are shared through these platforms in which interaction is made as visual and textual.

Initially, data is drawn from social platforms with specially developed software for this particular thesis.

The photographs of the drawn data are analyzed in color analysis software which is developed for this particular thesis. Color analysis program conducted color weighting as analysis each pixel of each photograph.

Data which is drawn and completed color analysis phase is transferred to data analysis program. Knime is utilized as data analysis program.

The transferred data is classified with decision tree. Lastly, estimations are made through classified data.

İÇİNDEKİLER

Sayfa No:

TABLO LİSTESİ

Sayfa No.

ŞEKİLLER LİSTESİ

KISALTMALAR

BIC : Bayesian Information Criterion
BKM : Bisecting K-Means
BPM : Bigram Proximity Matrix
CAP : Clustering Agreement Process
CBKM : Cooperative Bisecting K-Means
CC : Cooperative Clustering
CCS : Compressed Column Storage
CIE : Commission Internationale de L'Eclairage
CF : Concept Forest
CFWMS : Clustering based on Frequent Word Meaning Sequences
CFWS : Clustering based on Frequent Word Sequences
CLUTO : Clustering Toolkit
COMRAF : Combinatorial Markov Random Field
CST : Centroid Similarity Technique
DCF : Description Comes First
DCM : Dirichlet Compound Multinomial Model
DE : Differential Evolution
DHC : Dynamic Hierarchical Compact
DHS : Dynamic Hierarchical Star
EA : Evolutionary Algorithm
EDCM : Exponential DCM
EM : Expectation Maximization
EO : Entropy Overlap
ESA : Explicit Semantic Analysis
EWKM : Entropy Weighting K-Means
F2IHC : Fuzzy Frequent Itemset-Based Hierarchical Clustering
FCCM : Fuzzy Clustering for Categorical Multivariate Data
FCM : Fuzzy C-Means
FIHC : Frequent Itemset-based Hierarchical Clustering
FSC : Fuzzy Semantic Clustering

FTC : Frequent Itemset-based Clustering

FTSC : Frequent Term Set-based Clustering

FTSHC : Frequent Term Set-based Hierarchical Clustering

GAHC : Group-average Agglomerative Hierarchical Clustering

GBHS : Global Best Harmony Search

GST : Generalized Suffix Tree

H2-FCM : Hierarchical Hyperspherical Fuzzy C-Means

HAC : Hierarchical Agglomerative Clustering

H-FCM : Hyperspherical Fuzzy C-Means

HFTC : Hierarchical FTC

HKA : Harmony K-Means Algorithm

HS : Harmony Search

ICA : Independent Component Analysis

IGBHSK : Iterative Global Best Harmony Search K-Means

IR : Information Retrieval

ISOMAP : Isometric Figure Mapping

IST : Intra-Cluster Similarity Technique

KDD : Knowledge Discovery in Databases

LPI : Locality Preserving Index

LSI : Latent Semantic Indexing

MAC : Max Assignable Cluster

MC : Maximum Capturing

MCSKM : Multi-Cluster Spherical K-Means

MFI : Maximal Frequent Itemset

ML : Maximum Likelihood

MPE : Most Probable Explanation

MRF : Markov Random Field

MS : Multidimensional Scaling

MVS : Multi-ViewPoint Similarity

NLP : Natural Language Processing

NMI : Normalized Mutual Information

OSKM : Online Spherical K-Means

PC : Partition Coefficient

PCA : Principal Component Analysis

PFCC : Possibilistic Fuzzy Co-Clustering

PSO : Particle Swarm Optimization

SFCM : Spherical FCM

SHDC : Simple Hybrid Document Clustering

SKM : Spherical K-Means

SO : Standard Overlap

SRL : Similarity Ratio Limit

STC : Suffix Tree Clustering

SVD : Singular Value Decomposition

TFIDF : Terim Frekansı x Ters Doküman Frekansı

UPGMA : Unweighted Pair Group Method with Arithmetic Mean

VI : Variation of Information

VSM : Vector Space Model

VUM : Vektör Uzayı Modeli

WCM : Weighted Conceptual Model

WLC : WordNet Lexical Categories

WO : WordNet Ontologies

SEMBOL LİSTESİ

k : Küme sayısı

S : Benzerlik

n : Nesne/Doküman sayısı

m : Boyut sayısı

D : Doküman koleksiyonu

W : Terim ağırlığı

TF : Terim frekansı

IDF : Ters doküman frekansı

C : Kümeleme

c : Sınıf/kategori sayısı

Cj : j kümesi

Kh : h sınıfı

cj : j kümesinin ağırlık merkezi

nj : j kümesindeki nesne sayısı

nj(h) : j kümesindeki h sınıfından nesnelerin sayısı

J : Amaç fonksiyonu

NF : Normalizasyon katsayısı

MAC : Maksimum Atanabilir Küme

SRL : Benzerlik Oranı Sınırı

O : İşlemsel karmaşıklık

T : Zamana göre işlemsel karmaşıklık

F : Kosinüs benzerliği hesaplanması için gereken birim işlem zamanı

G : Küme ağırlık merkezi güncellenmesi için gereken birim işlem zamanı

H : Sıralama işleminde birim karşılaştırma için gereken işlem zamanı

L : K-Means döngüsündeki tekrar sayısı

1. METİN MADENCİLİĞİ

Veri madenciliği üzerine yapılan çalışmalar genelde veri ambarlarında ve ilişkisel veritabanı gibi yapısal veriler üzerine kurulmuştur. Erişilebilir ve kullanılabilir durumdaki verinin önemli bir bölümü metin veritabanlarında bulunmaktadır. Bu veritabanları çeşitli kaynaklardan oluşan (e-posta, araştırma bildirileri, haberler, sayısal kütüphaneler, kitaplar, makaleler, web sayfaları vs.) geniş döküman koleksiyonlarından oluşmaktadır. Bilgi miktarındaki artış nedeniyle metin veritabanlarının boyutları da günden güne hızla artmaktadır. Tahminlere göre iş dünyasıyla ilgili bilginin %85′i metin formatında saklanmaktadır [121].

Metin Madenciliği (Text Mining) ise, ilginç, yararlı ve henüz keşfedilmemiş bilginin, metin halindeki veriden, bilgi işlem metodları ile elde edilmesi olarak tanımlanabilir. "**bilgi patlaması**" (information explosion / information overload) adlı soruna çözüm bulmayı amaçlayan bir araştırma alanıdır. Bu sorunu çözmeye çalışırken; veri madenciliği, metin madenciliği, yapay zeka, istatistik, doğal dil işleme (NLP Natural Language Processing), bilgi yönetimi (Knowledge Management) ve bilgi erişim (IR Information Retrieval) tekniklerini kullanır. Metin Madenciliği, döküman koleksiyonlarının önişlemeden geçirilmesi, ara sonuçların saklanması, ara sonuçları analiz edebilmek için çeşitli tekniklerin kullanılması, ve elde edilen sonuçların görselleştirilmesi gibi aşamalardan oluşmaktadır [121].

Metin madenciliği çeşitli amaçlar için yapılabilir. Bunlardan bazıları, yazıyı özetleme, sınıflandırma, kategorize etme, kelime sıklığı ve kelimeler arası ilişki gibi istatistiki özelliklerini belirleme, duygu veya sentiment analizi yapma, bilgi çıkarımı gibi uygulamalardır. Bir örnek verilecek olursa, popüler bir araç olan Twitter yazı verileri üzerinde yapılan duygu (sentiment) analizi anlatılabilir. Mesela bir X markası reklam kampanyası yaptığında ve Twitter üzerinde belirli bir zaman diliminde o marka hakkında yapılan yazışmaların ne oranda olumlu ve olumsuz olduğunu duygu analizi ile öngörebilir. Bunu gerçeklemek için kullanılabilecek algoritma yaklaşımlarından bir tanesi en basit şekilde şöyle olabilir: Her twit de geçen olumlu kelimelere (iyi, güzel vb. gibi) +1 ve olumsuz kelimelere (kötü, çirkin vb. gibi) -1 puanı verilerek bir tweet için verilen tüm puanlar toplanır. Sonra o tweet için olumlu (sonuç pozitifse) olumsuz (sonuç negatifse) veya nötr (sonuç sıfırsa) kararına varılır

ve sınıflandırılır. Bu işlem eldeki tüm twit verilerine uygulanır. Daha sonra bunlar oranlanarak twit ifadelerinin mesela %67′si olumlu gibi bir sonuca varılabilir ki buda kampanyanın marka üzerine etkisini daha önce yapılan rutin analizlerle kıyaslayarak en hızlı bir şekilde ölçmek için önemli bir fikir verir. Elbette bu, konuyu en basit şekilde aktarabilmek için verilen yüzeysel bir yaklaşım ama daha sofistike algoritmalar benzer mantıkla uygulanabilmektedir.

Metin Madenciliği teknikleri temelde dört kategoriye ayrılır: **sınıflandırma** (classification), **kümeleme** (clustering), **bilgi çıkarım** (information extraction) ve **birliktelik analizi** (association analysis) . Sınıflandırma işlemi; nesnelerin hangi sınıfa ait bulmak için yakınlaştırma yapılarak o sınıfa veya ait olduğu kategoriye dahil edilmesidir. Birliktelik analizi; sıklıkla birlikte kullanılan, gelişen sözcük veya kavramların belirlenmesi ve döküman kümelerinin ya da döküman içeriğinin anlaşılmasını amaçlar. Bilgi çıkarım, bilgi çıkarım teknikleri ile dökümanların içerisinde yer alan yararlı veriyi ya da ifadeleri bulmayı çalışmak olarak özetlenebilir. Kümeleme analizi, döküman kümelerinin temelini oluşturan yapıları bulmak amacıyla uygulanmaktadır [121].

1.1. METİN KÜMELEMEDE KARŞILAŞILAN SORUNLAR

Metin dökümanları doğal dil ile yazıldıklarından yapısal veriden çok farklıdırlar. Bundan dolayı yapısal veriler için geliştirilmiş kümeleme algoritmaları metin dökümanları için yeterince başarılı olamamaktadır [1]. Metin kümeleme için yeni teknik ve algoritmaların geliştirilmesi gereklidir. Bunun için öncelikle metin kümelemenin kendine özgü gereksinim ve sorunları irdelenmelidir. Aşağıda bunlar ayrıntılı olarak verilmiştir:

- Metin dökümanlarını temsil etmek için uygun bir model bulmak kolay değildir. Dökümanların büyük çoğunluğu doğal dil kullanılarak yazılmaktadır ve içeriğe duyarlıdır (context sensitive). Bir cümlenin ifade ettiği anlam ile cümleyi oluşturan sözcüklerin sıralaması yakından ilişkilidir. Dolayısıyla kullanılacak olan modelin dökümandaki sözcüklerin sıralamasının etkisini dikkate alması önemlidir.

- Gerçek hayatta değişik konulardaki dökümanların sözcük dağarcığı birbirinden çok farklı olmasına rağmen Vektör Uzayı Modeli'nde tüm döküman vektörleri tüm koleksiyondaki sözcükleri içerecek şekilde normalize edilmek zorunda kalınmaktadır.

- Gerçek hayatta aynı anlamı ifade etmek için farklı sözcükler kullanılabileceği gibi aynı sözcük farklı anlamları ifade etmek için de kullanılabilmektedir. Metin kümelemede anlamların gözardı edilerek sadece sözcüklerin kullanılması hata oranını artırmaktadır [3].

- Dökümandaki tüm sözcükler dökümanın genel konusuyla her zaman ilgili değildir. Dökümanın geneliyle ilgisiz sözcükler çıkarılabilirse kalan sözcükler kümeleme için daha fazla bilgi içeriyor olur. Ancak konuyla ilgisiz sözcüklerin tespiti kolay değildir.

- Tespit edilen kümelerin anlamlı olarak isimlendirilmesi de önemli bir gereksinimdir. Sistemin kullanıcılarının tespit edilen kümelerin ilgili oldukları konuları görebilmeleri çok yararlı bir özelliktir. Ancak, kümeleme tamamlandıktan sonra kümelerin isimlendirilmesi oldukça zaman alıcı bir işlemdir.

- Bir dökümanın içerdiği bir ya da daha fazla konu nedeniyle aynı anda birden fazla kümeye dahil olabilmesine yani döküman kümelerinin belli ölçüde örtüşmesine izin verilmelidir.

- Genellikle dökümanlar 200-10000 arası eşsiz sözcük yani boyut içerirler. Geniş döküman koleksiyonlarını etkin ve verimli bir biçimde kümelemek için çok yüksek olan boyut sayısının indirgenmesi gerekmektedir [4].

- Kümeleme işleminden önce küme sayısı bilinmemektedir. Döküman koleksiyonunun içeriği hakkında bilgi sahibi olunmadan oluşacak küme sayısını doğru tahmin etmek zordur. Kümeleme algoritmasına oluşması beklenilen küme sayısının verilmesi yerine bu sayıyı kümeleme algoritmasının bulması daha anlamlıdır. Metin kümelemede karşılaşılan bu gereksinim ve sorunlar, metin dökümanlarının doğası da hesaba katılarak yeni teknik ve algoritmaların geliştirilmesini gerekli kılmaktadır.

1.2. BİLGİYE ERİŞİM

Bilgiye Erişim kavramı ilk kez Calvin Mooers tarafından 1948 yılında "Application of Random Codes to the Gathering of Statistical Information" başlığını taşıyan yüksek lisans tezinde Information Retrieval terimi altında kullanılmıştır. Vickery, Kullanıcı Yorumları (Yapılandırılmamış metinsel veri) BİLGİ METİN MADENCİLİĞİ YÖNTEMLERİ İşletmeler Kullanıcılar (Tüketici/Müşteri) 25 Mooers'in kavrama İngilizce olarak getirdiği ilk tanımı şu şekilde aktarır. Bilginin bir depodan özelliklerine göre konusal olarak aranarak erişilmesidir [12].

Bilgiye Erişim (BE), Metin Madenciliğinde ilk adım olarak nitelendirilmektedir Bilgiye Erişimin amacı kullanıcıların bilgi ihtiyaçlarını karşılayacak olan belgeleri bulmasına yardımcı olmaktır.

Bilgiye Erişim, birçok konu alanına sahipliği nedeniyle geniş bir alana yayılmaktadır ve kullanıcıların belirli konulardaki belgeleri bulabilmesi gibi büyük bir topluluktan oluşan metni sunması için modeller geliştirmiştir. Problem, kullanıcı şu an ne ile ilgilenmekte ve belirli bir konu kümesi hakkında belgeler nasıl sunulmalı ve tanımlanmalı gibidir [13].

Bilgiye Erişim, bilgi ihtiyacını karşılayan yapılandırılmamış materyalleri (genellikle dökümanlar) geniş bir koleksiyonun içerisinden bulmaktır. Eskiden bilgiye erişim sadece bazı meslek grupları tarafından özel amaçlar için kullanılmaktaydı. Fakat değişen günümüz dünyasında, milyonlarca insan mail ve web aramaları için kullanmaktadır. Böylelikle BE geleneksel veritabanı arama yöntemlerinin önüne geçmeye başlamıştır. BE bu tanımların dışında, yapılandırılmamış materyalleri içeren, temiz olmayan veriler ve anlamsız veriler ile ilgili problemleri de kapsar.

Birçok Bilgiye Erişim teknolojisinde kullanılan iki temel sunum şemaları şunlardır; vektör uzay modeli ve gizli anlambilimsel dizinleme (latent semantic indexing). Vektör uzayı modeli, belgeleri ve sorguları sunma maliyetini minimize edebilir. Belirli bir sorgunun kriterini, sırasıyla olası belgeleri ve özel sorguyu sunan iki vektör arasındaki öklit uzaklığını hesaplayarak karşılayan belgeleri etkin bir şekilde bulabilir. Gizli anlambilimsel dizinleme, özellikle eşanlamlılık ve çok

anlamlılık gibi vektör uzayı modeline ait bazı kısıtlamaları dengelemek için geliştirilmiştir [12,13,14].

Bilgiye Erişim sistemlerinde kullanılan standart iki ölçüt vardır [14].

a) Doğruluk (Recall): Araştırmacı tarama yaptığı konu da bütün kaynaklara erişmek istemektedir. Bilgi sistemlerinde araştırma yapanın bu isteğinin karşılanma derecesi Doğruluk ile ifade edilir. Doğruluk, bir bilgi sisteminin sorgu ile bulduğu sonuçların içindeki gerçekten sorgu ile ilgili olanların sayısının veritabanında bulunan ilgili sonuçların sayısına oranı ile ifade edilir [9, 13, 14].

$$\text{Doğruluk} = \frac{veritabanı\ içinde\ \ dönen\ ilgili\ belge\ sayısı}{ilgili\ toplam\ döküman}$$

b) Duyarlık (Precision): Araştırmacı edinmek istediği bilgileri fazla zaman harcamadan bulmayı istemektedir. Bundan dolayı bilgi sisteminin tarama hızıdır yüksek olmalıdır. Tarama sistemi hızlı olduğu gibi, doğru sonuçları getirecek kadar da akıllı bir tarama sistemi özelliğini barındırması gerekmektedir. Çünkü araştırmacının zamanın büyük kısmı gelen sonuçlar içerisinden doğru sonuçları bulması ile geçmektedir. Sonuç listesinin iyiliğini gösteren bu özellik ise Duyarlık olarak adlandırılır. Duyarlık bir bilgi sisteminin sorgu ile ilgili olarak bulduğu sonuçların içindeki araştırmacının istediği sonuçların sayısının bulunan sonuçların sayısına oranıdır [9, 13, 14].

$$\text{Duyarlık} = \frac{veritabanı\ içinde\ dönen\ ilgili\ belge\ sayısı}{geri\ dönen\ döküman}$$

Doğruluk ve Duyarlık ölçümlerinin her ikisini birden arttırmak bilgilerin tasnif edilmesi ile olur. Bu konudaki robotların Doğruluk(Recall) ve Precision(Duyarlık) oranları düşüktür. Kütüphanelerin ise yüksektir. Ağırlık verme Bilgiye Erişim sistemlerinde önemli bir rol oynar. Birçok ağırlık verme modeli geliştirilmiştir. En yaygın olarak kullanılan ağırlık verme modeli, genel(global) ve yerel(local) ağırlık verme şemalarının birlikte kullanılmasıdır. Terim frekansı (term frequency) tf yerel ağırlık vermede, ters döküman frekansı (inverse document frequency) idf ise genel ağırlık vermede, kullanılır [9, 13, 14].

Terim Frekansı (tf), bir döküman içerisinde bir terimin tekrar etme sıklığıdır Ters Döküman Frekansı(idf) ise bir terimin bütün döküman koleksiyonu içindeki önemidir ve aşağıda gösterildiği gibi Denklem 1 ile hesaplanır. Bunlara bağlı olarak terim ağırlığı formülü Denklem 2'de görülmektedir. Denklem 2'deki N değeri, veri seti içindeki toplam belge sayısını göstermektedir.

$$idf_i = \frac{\log N}{df_i} \quad (1)$$

$$w_i = tf_i * idf_i \quad (2)$$

Bilgiye Erişim terimini Türkçe'de ilk kez 1971'de Aydın Köksal kullanmıştır. Aydın Köksal daha sonra bu kavramı Bilişim Terimleri Sözlüğü'nde şöyle tanımlamıştır; "Bir bilgiye erişim dizgesini (sistemini) kullanarak içerik bakımından araştırılan konu ve kavramlarla ilgili olabilecek genellikle varlığı bile bilinmeyen belgelerin izini bulmayı amaçlayan araştırmadır". En eski Bilgiye Erişim çalışması yine aynı kişi tarafından 1981 yılında gerçekleştirilmiş ve 12 sorgu ile bilgisayar bilimindeki 570 belge kullanılmıştır [14].

Türkçe'de bu konudaki diğer çalışmalara bakılacak olursa, Solak ve Can (1994) 533 haber makalesine ait yığını ve 71 sorguyu kullanmışlardır. Solak ve Can'ın çalışması, kök bulma algoritması verilen bir kelimeyi sözlükte aramayı, kelimenin sonundan bir karakter silmeyi ve daha sonra yapısal analizi yapılandırmayı temel almaktadır [14].

Ekmekçioğlu ve Willett (2000), 6289 boyutunda bir Türkçe haber belge yığınını ve 50 sorguyu kullanmışlardır. Sadece sorgu kelimelerini köklerine ayırmışlar ve 28 köklerine ayrılmış ve ayrılmamış sorgu kelimelerini kullanarak kazanım etkililiğini karşılaştırmışlardır [14].

Sever ve Bitirim (2003), çalışmalarında 2468 kanun belgesi ve 15 sorguyu temel alan sisteme ait uygulamayı tanımlamışlardır. İlk önce, yeni bir kök bulucunun daha üstün performansa sahip olduğunu ispatlamışlardır. Daha sonra, çekimli ve türetmeli kök bulucuların, kök bulamama durumuyla karşılaştırıldığında erişim duyarlılığı açısından % 25 civarında ilerleme sağladığını göstermişlerdir [14].

Pembe ve Say (2004) Türkçe Bilgiye Erişim problemini Türkçenin morfolojik, lexico-semantik ve sözdizimsel seviyelerindeki bilgileri kullanarak çalışmışlardır. Bazı sorgu zenginleştirme teknikleri ile kök bulucunun etkilerini tartışmışlardır. Deneylerinde, webden elde edilen farklı konuları ele alan 615 Türkçe belgeyi ve 5 uzun doğal dil sorgularını kullanmışladır. Yedi farklı dizinleme ve kazanım kombinasyonunu kullanarak ve performans etkilerini ölçmüşlerdir [14].

1.3. BİLGİ ÇIKARIMI

Bilgi Çıkarımı konusu, doğal dil işleme teknikleri kullanılarak, genel olarak bir metin üzerinde bulunan belirli kriterdeki bilgileri elde etmeyi hedefler. Örneğin bu işlem sırasında, belirlenmiş kalıba uygun olan verilerin çıkarılması istenebilir. Amaç insan müdahalesini en aza indirmek, çok miktardaki veriyi otomatik olarak işleyen bir yazılım üretmektir. Genellikle bilginin çıkarılacağı ortam yazılı metinler olmaktadır, ancak bu metinlerin bulunacağı ortamlar değişebilir. Örneğin taranmış metinler, veri tabanları veya internet üzerindeki dökümanlar bu verinin kaynağını oluşturabilirler [15].

Bilgi Çıkarma yöntemleri metin içindeki unsurları varlıkları otomatik olarak çıkarır ve bunlar arasındaki ilişkileri ortaya koyar. Metin içindeki cümleler ve paragraflar içerdikleri önermelerle varlıklara ait bilgiler taşır. Bilgi çıkarma teknikleri bu önermelere bağlı olarak belgeyi oluşturan varlıkları ve bu varlıklar arasındaki ilişkileri çıkarırlar [15,16].

Bilgi çıkarımı başka bir ifade ile geniş ölçekli bilgilerden özet çıkarılması olarak adlandırılabilir. Başka bir ifadeyle büyük veri yığınları içerisinden özet bilgiler elde edilmesidir. Anahtar kelimeler veya örnek dökümanlar gibi kullanıcı girişleriyle bağlantılı olan bilgi ya da dökümanların bulunması bilgi çıkarımı örnekleridir. Bu çalışmalar sonucunda web sayfalarından bilgiler karşılaştırılarak bulunabilir, geniş ölçekli metinlerden özet bilgiler çıkarılabilir, sorgulara karşılık gelen ifadeler bulunabilir [17].

Bilgi çıkarım için veriyi işlerken belirli bir yapıya oturtmak en zor adımlardan biridir. Örneğin internet üzerinden yayınlanan verilerin herhangi bir standart yapısı

bulunmamakla beraber, dağınık halde yayınlanmaktadır. Bilgi erişim yöntemlerine nazaran daha etkin sonuçlar elde edilmesini sağlayan bilgi çıkarma tekniklerinin avantajı belge içindeki içeriğin anlamını ön plana çıkaran terimlerin ve terimler arası ilişkilerin bulunmasında yatar. Ancak bazen belgelerin incelenmesindeki amaç, daha önceden fark edilmemiş gerçeklerin ve ilişkilerin ortaya çıkarılmasıdır. Bu aşamada devreye bilgi keşfi teknikleri girer. Bilgi keşfi için kullanılan yöntemler metnin içeriklerini derler, birbiri ile entegre eder ve başka kaynaklardan elde edilen sonuçlarla birleştirilerek üst seviye bir anlam ve ilişki kümesi oluşturmaya çalışır. Özellikle konuya bağlı olarak terimler ve terimler arası ilişkilerin üzerine de çıkılır ve konuya özel yapılar ve fonksiyonlara bağlı bir ilişki kümesi oluşturulur. Bu amaçla geliştirilen sistemlerin sadece belgeleri değil veritabanlarındaki verileri de kullanması gerekir [18].

Bilgi çıkarım işlemi, temelde anahtar kelime ve/veya benzerlik tabanlı çıkarımlara dayanmaktadır [18]. Anahtar kelime tabanlı bilgi çıkarımında, herhangi bir döküman ya da metinden bilgi çıkarılırken anahtar kelimelerden oluşan bir küme oluşturulur. Benzerlik tabanlı çıkarım sistemleri ortak anahtar kelimeler kümesini temel alarak, benzer dökümanları bulmaktadır. Bu tür bir çıkarımın çıktısı, kelimelere yakınlığı ve birbirleriyle ilişki derecelerini temel almaktadır. Günümüzde internet ve bilgi teknolojilerinin hızla gelişmesi ve insanların hayatında önemli bir yer tutması sebebiyle, bu ortamlardan bilgi çıkarımı önem kazanmıştır. Herhangi bir ürünün satış sitelerinden aranması ve karşılaştırmalı olarak değerlendirilmesinden, elektronik posta içeriklerinin yorumlanmasına kadar çeşitli uygulamalar internetten bilgi çıkarımı işlemine örnek olarak düşünülebilir [15,19,20].

Bilgi Çıkarım sistemi sonuçlarının değerlendirilmesinde bilgi erişim sistemlerinde de olduğu gibi duyarlık ve doğruluk ölçütleri kullanılmaktadır. Fakat burada belgeler yerine, yapılan tahminler ölçüm değişkenleri olarak kullanılmaktadır. Duyarlık, sistemin doğru yaptığı tahminlerin tüm tahminlere bölümü ile hesaplanmaktadır. Doğruluk ise sistemin yaptığı doğru tahminlerin metinde bulunan bütün varlıkların sayısına bölünmesi ile elde edilmektedir [21,22].

Bilgi Çıkarımı konusunda literatürde birçok çalışma mevcuttur. Yapılan çalışmalar üzerinde çalışılan belgelerin metin özelliklerine göre farklılıklar

göstermektedir. Eğer bilgi çıkarımı yapılan belgeler içinde herhangi bir yapısal özellik taşımayan düz yazıdan oluşuyorsa dilbilimi ile ilgili özellikler önem taşır. Öte yandan belgeler bir yapısal düzen taşıyorsa örneğin web sayfaları gibi o zaman çıkarılacak alanlar arasındaki noktalama işaretleri vb. özellikler önem kazanır. Web sayfaları gibi yapısal metinlerden bilgi çıkarımı için kullanılan algoritmalara örnekler aşağıda verilmiştir [22].

WHISK, bilgi çıkarım algoritması hem düz yazılarda hem de yapısal özellikler içeren belgelerde çalışabilir. Bu algoritma öğrenme kümesini kullanarak düzenli ifadeler (regular expression) tarzında kurallar öğrenir. Öğrenilen bu kurallar ile bilgi çıkarımı işlemi gerçekleştirilir [23].

SRV (Stochastic Real Valued) Algoritması Freitag tarafından önerilmiş bir bilgi çıkarım yöntemidir. Bu algoritma makine öğrenme tekniğinin bilgi çıkarımı problemine bir uygulaması olarak görülebilir. Bu algoritma sonucunda da yine kurallar elde edilerek bilgi çıkarımı gerçekleştirilir. Web sayfaları üzerinde başarıyla ve makul sürede çalışabilen bir algoritmadır [23].

SRV algoritmasına ilave olarak literatürdeki iki farklı bilgi çıkarım yaklaşımı olarak RAPIER ve HMM tabanlı bilgi çıkarım yaklaşımı Sanner tarafından gerçeklenmiştir [23]. RAPIER bilgi çıkarım algoritması bir çeşit “detaydan-genele” (bottom-up) kural öğrenme algoritmasıdır. Algoritma önce bir ders sayfası için detaylı bir kural öğrenir. Daha sonra karşılaşılan diğer bir etiketlenmiş veriler için bu kuralı mümkün olduğunca genelleştirmeye çalışır ki yeni görülen örnek de bu kural tarafından kabul edilsin. Eğer başta öğrenilen kural bu örneği kapsayacak şekilde genelleştirilemiyorsa bu örnek için detaylı bir kural öğrenilir. Sonuç olarak öğrenme işlemi sonunda bir Bilgi Çıkarımı kümesi elde edilmiş olur [23].

RAPIER algoritmasının öğrendiği kurallar birçok çeşit bilgi üzerinde koşullar koyar. Bu bilgiler şöyle sıralanabilir; kelimeler, kelimelerin cümle içinde kullanım şekilleri (isim, fiil, zamir, sıfat vs.), kelimelerin anlamsal sınıfları ve çıkarılacak bilgiyi çevreleyen kelimeler. RAPIER algoritması kelimelerin cümle içindeki kullanım şekillerine gereksinim duyduğu için literatürde kullanılan bir program olan Brill’in konuşma kısmı etiketleme (Part of Speech Tagger) programı kullanılabilir [23].

Sanner'in HMM bilgi çıkarımı programı Saklı Markov Modeli (Hidden Markov Model) tabanlı Bilgi Çıkarımı algoritmasını çalıştırmaktadır. Saklı Markov Modeli etiketlenmiş öğrenme kümesini kullanarak bir durum geçiş (bağlantı) diyagramı öğrenir. Sanner HMM algoritmasının öğrendiği durum geçiş diyagramı 6 farklı tip durum bilgisi içerir. BAŞLANGIÇ, BİTİŞ, HEDEF, ÖN EK, SON EK, VE ARKA PLAN. Bu diyagram ilk başta BAŞLANGIÇ durumundadır. Durumlar arasındaki geçişler kelime ve ihtimal değerleri ile sağlanır. Bu kelime ve ihtimal değerleri öğrenme kümesinden öğrenilir. Bir test sayfası geldiğinde ise görülen kelimenin sistemin bulunduğu durumdan diğer bütün durumlara gitme ihtimallerine bakılarak ihtimal değeri en yüksek olan duruma sistem geçer bu kelime işlenmiş olur. Bilgi Çıkarımı işlemi ise HEDEF durum tipindeki durumlarda kullanılan kelimelerin belirlenmesiyle gerçekleştirilmiş olur [23].

Sanner HMM algoritmasında kullanılan model 3 ayrı HMM modelinin birlikte kullanılmasıyla oluşmuştur: tekdüze model (uniform model), bağlam modeli (context model) ve tam geçişli model (full transition model). Bu modellerin farklı alanlar için farklı derecede önemleri vardır. Sanner HMM programı her bir alan için bu modellerin önemini ifade eden katsayı değerlerinin belirtilmesine konfigürasyon dosyaları sayesinde izin verir [23]."

1.4. BİLGİYE ERİŞİM VE BİLGİ ÇIKARIMININ KARŞILAŞTIRILMASI

Bilgi Çıkarımı, bilgi parçalarını çıkarmak için doğal dil işlemeyi temel alan bir teknolojidir. Bu süreç girdi olarak metinleri ele alır ve çıktı olarak belirli bir formatta açık şekilde ifade edilebilecek veriler üretir. Bu veri kullanıcıların görüntü elde etmesi için doğrudan kullanılabilir veya daha sonra analiz etmek için veri tabanında veya elektronik tablolarda saklanabilir veya Google gibi internet arama motorlarında olduğu gibi Bilgiye Erişimi uygulamalarında dizinleme amaçlarını yerine getirmek için kullanılabilir [24].

Bilgi Çıkarımı, Bilgiye Erişimden oldukça farklıdır; -Bilgiye Erişim sistemi uygun metinleri bulur ve bunları kullanıcıya sunar. -Bilgi Çıkarımı uygulaması

metinleri analiz eder ve sadece kullanıcıların ilgilendikleri metinlerden özel bilgi elde eder [24]

Bir örnekle açıklanmak istenirse, tarım ürünleri pazarlarını ilgilendiren ticari grup yapılarından bilgi bekleyen bir Bilgiye Erişim sistemi kullanıcısı uygun kelime listesini girecektir ve karşılığında olası eşleşmeleri içeren belge kümesine (örneğin gazete makaleleri) ulaşacaktır. Daha sonra kullanıcı belgeleri okuyacaktır ve bilgilerin içerisinde bir ayıklama işlemi gerçekleştirecektir. İşlem uygulandıktan sonra elektronik tablo halinde bilgi girişi yapılabilir ve bunlardan rapor veya sunu çizelgeleri oluşturulabilir. Bunun tersine bir Bilgi Çıkarımı sistemi uygun şirket ve grup adlarını doğrudan ilgilendiren değerleri otomatik olarak elektronik tablo halinde sunacaktır [24].

Bilgiye Erişim sistemi ile karşılaştırıldığında Bilgi Çıkarımının avantajları ve dezavantajları vardır. Bilgi Çıkarım sistemleri işlem oluşturma için daha zor ve daha bilgi duyarlı bir sistemdir. Özel alanlara ve senaryolara bağlı olan değişken derecelerine göre çalışır. Bilgi Çıkarımı, Bilgi Erişimden hesaplama açısından daha duyarlıdır. Bununla birlikte, geniş çaptaki metin belgeleri ile ilgili uygulamalarda Bilgi Çıkarımı, Bilgiye Erişimden potansiyel olarak daha etkin çalışan bir sistemdir. Bunun nedeni insanların metin okuma sırasında kaybettiği zamanı önemli ölçüde azaltma potansiyeline sahip olmasından gelmektedir. Ayrıca, sonuçların farklı birkaç dilde sunulması gerektiği durumlarda, Bilgi Çıkarımının sabit biçimli, kesin sonuç veren yapısı Bilgiye Erişim tarafından bulunan birden çok dili içeren metinlerin yorumlanmasını gerektiren tam çeviri kolaylıklarını sağlama açısından karşılaştırıldığında nispeten anlaşılır durumlar sağlayacaktır [24].

Bu bilgiler ışığında Metin Madenciliği metotlarının girdi ve çıktıları Tablo 1.4'te görüldüğü gibi özetlenebilir.

Tablo I.4 Metin madenciliği metotlarının girdileri ve çıktıları [2]

Bilgiye Erişim	**Bilgi Çıkarımı**	**Web Madenciliği**	**Kümeleme**
Girdi: Metin Belgesi Kaynağı, Kullanıcı sorgusu	Girdi: Metinsel belgeler kaynağı İyi tanımlanmış	Webteki özel bilginin çıkarımı ve metinsel belgelerin	Benzer metin belgelerinin toplanması

(metin tabanlı) Çıktı: Sorgu ile ilişkili olan sıralanmış belgeler kümesi	sınırlandırılmış sorgu Çıktı: İlişkili bilgi cümleleri İlişkili bilginin çıkarımı ve ilişkili olmayan bilginin yok sayılması Önceden belirlenmiş formatta çıktı ve ilgili bilgi linki.	erişimi ve indekslenmesi	

2. ÖNİŞLEME TEKNİKLERİ

Geniş döküman kümeleri üzerinde veri madenciliği uygulayabilmek için öncelikle metin dökümanları bir önişlemden geçirilirler ve sonraki madencilik süreçlerinde kullanılmaya uygun veri yapılarında saklanırlar. Önişleme (preprocessing) teknikleri Metin Önişleme (Text Preprocessing) ve Dilbilimsel Önişleme (Linguistic Preprocessing) olarak ikiye ayrılabilir [4].

2.1. Metin Önişleme Teknikleri

Bir metin içerisindeki sözcükleri elde etmek için genellikle dizgeciklere (token) ayırma (tokenization) işlemi gereklidir. Bu işlem ile metin içerisindeki tüm noktalama işaretleri, tab ve satır sonu karakterleri ile diğer okunabilir olmayan (nontext ve non-readable) karakterler boşlukla (white space) değiştirilir ve metin bir sonraki süreç için daha uygun ve temiz bir hale getirilir. Koleksiyondaki tüm dökümanlarda dizgeciklere ayırma işlemi uygulandıktan sonra tüm dökümanlarda yer alan sözcüklerin tümü ilgili koleksiyonun "sözlüğü"nü (dictionary) oluşturur [4].

Sözlük boyutunun, dolayısıyla da koleksiyonlardaki dökümanları temsil eden veri yapılarının (örneğin Vektör Uzayı Modelindeki döküman vektörlerinin) boyutunun küçültülmesi için çeşitli önişleme yöntemleri kullanılabilir [4].

Filtreleme (filtering) yöntemleri ile sözlükteki ve dolayısıyla da dökümanlardaki sözcükler filtrelenebilir. En yaygın filtreleme yöntemi durak sözcükleri (stop words) filtreleme yöntemidir. Buradaki amaç bağlaç, edat, İngilizce'deki "the" artikeli, Türkçe'deki "böyle" ve "daha" sözcükleri gibi doğrudan herhangi bir bilgi taşımayan ve içeriğe bu anlamda etkisi olmayan sözcüklerin çıkartılmasıdır. Ek olarak, bütün dökümanlarda çok sık geçen ve böylelikle herhangi bir ayırdedici etkisi kalmayan ya da bütün koleksiyonda çok az sayıda bulunan ve istatistiksel olarak belirgin bir yararı olmayan sözcüklerin sözlükten çıkartılması da filtreleme tekniği olarak uygulanmaktadır [4].

Temel hale döndürme **(lemmatization)** yöntemleri genellikle fiil çekimlerini mastar duruma çevirmeye ya da çoğul haldeki isimleri tekil halde çevirmeye yarar. Bu işlem sözcüklerin cümle içerisindeki konumlarını ve görevlerini de bilmeyi gerektirdiğinden zor, pahalı ve hataya açık bir işlemdir. Bu nedenle pratikte daha çok "kökenine döndürme" (stemming) yöntemleri uygulanmaktadır [4].

Kökenine döndürme **(stemming)** yöntemleri sözcükleri basit hallerine çevirme işlemidir. Örneğin çoğul ekinin isimlerden atılması, fiil çekim eklerinin fiilden ayrılarak fiil kökünün ayrılması gibi işlemler stemming olarak adlandırılır. İngilizce lisanı için en yaygın kural tabanlı stemming algoritması Porter tarafından geliştirilmiştir ve halen en sık başvurulan yöntemdir [4, 15].

Türkçe için ise Doğal Dil İşleme kütüphanesi olan Zemberek geliştirilmiştir. Zemberek açık kaynak, platform bağımsız ve genel kullanım amaçlıdır. **Zemberek** kullanılarak Türkçe dökümanlarda stemming uygulanabilmektedir [7].

2.2. Dilbilimsel Önişleme Teknikleri

Çoğunlukla metin tabanlı yani mekanik önişleme teknikleri metin madenciliği operasyonları için yeterli olmaktadır. Ancak, çeşitli durumlarda dilbilimsel önişleme yöntemleri ile terimler hakkında daha fazla bilgi sahibi olmak ve bundan yararlanmak mümkün olabilmektedir. Aşağıda yaygın kullanımı olan dilbilimsel önişleme teknikleri açıklanmaktadır [4].

Cümlenin öğelerini etiketleme **(Part-of-speech Tagging)** işlemi ile cümlenin öğeleri tespit edilerek sözcüklere cümledeki görevine uygun ağırlık vermek, isim ya da fiil oluşuna göre lemmatization/stemming uygulamak vb. yaklaşımlar yararlı olabilmektedir.

Anlam belirsizliğini giderme **(Word Sense Disambiguation)** eşanlamlı sözcüklerin cümledeki ya da ifadedeki kullanım şekline göre hangi anlamda kullanıldığını tespit etmeye ve ona göre değerlendirmeye yardımcı olan işlemdir. Böylece örneğin vektör uzayı temsilinde terimin kendisi yerine cümledeki anlamını daha belirgin ifade eden başka bir sözcük kullanılır. Bu yöntem sözlüğün boyutunun büyümesine neden olmasına rağmen terimlerin anlamsal boyutlarıyla ele alınmasını sağlamaktadır [4].

Anlam belirsizliğini giderme amacıyla İngilizce dökümanlar için sıklıkla WordNet sözcüksel veritabanı kullanılmaktadır [8].

3. METİN KÜMELEME ALGORİTMALARI

Bu bölümde, Metin Kümeleme alanında yapılmış önemli çalışmalar, önerdikleri yaklaşım ve çözüm yöntemlerine göre çeşitli başlıklar altında incelenmiştir.

3.1. K-Means Algoritmasının Türev ve Çeşitlemeleri

K-Means, en fazla bilinen ve en yaygın kullanıma sahip bölünmeli kümeleme algoritmasıdır [1]. Basitliğinin yanı sıra işlemsel karmaşıklığının düşük olması ve metin halindeki veri setlerinin kümelenmesindeki başarımı nedeniyle tercih edilen bu algoritmanın döküman kümeleme için geliştirilmiş türevleri bulunmaktadır.

3.1.1. Küresel K-Means – Dhillon ve Ekibinin Çalışmaları

Dhillon ve Modha tarafından yüksek boyutlu ve seyrek vektörler olarak temsil edilen geniş döküman koleksiyonlarında veri madenciliği amacıyla K-Means algoritmasının özel bir durumu olan Küresel K-Means (Spherical K-Means) algoritması ve benzerlik ölçütü olarak da kosinüs benzerliği (cosine similarity) kullanılmıştır. Döküman vektörleriyle birlikte kümelerin ağırlık merkezlerini temsil eden vektörlerin de normalize edilmiş olması nedeniyle hem döküman vektörleri hem de küme ağırlık merkezleri daima birim kürenin yüzeyinde bulunmaktadır. Bundan dolayı bu algoritma Küresel K-Means olarak isimlendirilmiştir. İlgili çalışmada bu algoritma derinlemesine analiz edilmiş, algoritmanın ürettiği kümelerin belirli bir fraktal benzeri davranış sergilediği, bu kümelerin kavram vektörlerinin de seyrek olduğu gösterilmiş ve bu sayede verinin öz bir açıklamasının elde edilebildiği, küme etiketlerinin oluşturulması bakımından bunun çok yararlı olduğu öne sürülmüştür [22].

Dhillon, Fan ve Guan tarafından çok geniş döküman topluluklarının yüksek verimlilik ve başarım ile kümelenmesi konusuna odaklanılan çalışmada özellikle 100000'den fazla dökümanın tek işlemci üzerinde makul bir sürede kümelenmesi hedeflenmiştir. Yani, düşük bellek kullanımıyla yüksek hızda ve ölçeklenebilir bir kümeleme yapılmaya çalışılmıştır. Dökümanların diskten okunup önişlemeden geçirilmesi için çok-kanallı (multithreaded) ve bellek kullanımı bakımından da verimli olacak bir yaklaşım sergilenmiştir. Dökümanların vektör uzayı modelinin oluşturulması için yerel (local) ve küresel (global) karma tabloları (hash table) gibi verimli ve ölçeklenebilir veri yapıları kullanılmıştır. Kümelemede ise döküman temsili için kullanılan vektör uzayı modelindeki verinin seyreklik (sparsity) özelliğini avantaja dönüştüren, oldukça verimli ve etkili bir algoritma olan Küresel K-Means algoritması kullanılmıştır. Terim-döküman matrisi oluşturulurken "normalized term frequency-inverse document frequency" yani normalize edilmiş TFIDF yaklaşımı kullanılmaktadır. Normalize edilmiş olması her bir döküman vektörünün boyutunun 1 olması demektir. Bu sayede w boyutlu her bir döküman vektörü R^W uzayındaki birim kürenin yüzeyinde bulunmaktadır. Kümelemede döküman vektörünün uzunluğuyla değil sadece doğrultusuyla ilgilenilmektedir.

Döküman verisinin doğası gereği oluşan döküman vektörleri çok yüksek boyutlu olurken terim-döküman matrisi de son derece seyrek (sparse) olmaktadır. Bu seyreklikten yararlanılarak bu matris CCS (Compressed Column Storage) olarak isimlendirilen bir veri yapısında saklanmaktadır. Bu veri yapısı sayesinde terim-döküman matrisi için en az miktarda bellek kullanımı sağlanmaktadır. K-Means algoritmasındaki hesaplamalar ağırlıklı olarak döküman vektörleriyle kavram vektörlerinin skaler çarpımlarından oluşmaktadır. Bu çalışmada bu konuda bir iyileştirme öne sürülmüştür. K-means'in ilk birkaç döngüsünden sonra kümelerin kararlı (stable) duruma geçtikleri ve amaç fonksiyonunun değerinin çok fazla değişmediği görülmektedir. Dolayısıyla bu kararlı süreçte skaler çarpımların bazı durumlarda yapılmasına gerek kalmadan daha önce hesaplanmış değerlerin kullanıldığı bir yöntem geliştirilmiştir [23].

Dhillon, Guan ve Kogan tarafından yayınlanan diğer bir çalışmada Küresel K-Means algoritmasının özellikle az sayıda döküman içeren küçük kümeler söz konusu olduğunda optimal çözümden çok uzakta bir yerel maksimumda takılıp kalması sorununun aşılması için bir öneri getirilmektedir. Döküman koleksiyonu gibi çok boyutlu veri kümelerinde dökümanlar arasındaki kosinüs benzerliği değerleri 0,025 gibi çok küçük değerler olmaktadır. Bu nedenle de K-Means algoritması uygulandığında çoğu zaman sonuçta elde edilen kümelemenin başlangıçtaki rastgele ayarlanmış kümelemeden çok farklı olmadığı görülmektedir. Bu çalışmada, bu gibi sorunların aşılması için İlk Çeşitleme (First Variation) isimli teknik ile bu tekniğin iyileştirilmiş şekilleri ve deney sonuçları sunulmaktadır. İlk Çeşitleme tekniğinde, amaç fonksiyonunu maksimize etmek amacıyla bir kümeden bir vektörün çıkartılarak başka bir kümeye dahil edilmesi denenmektedir. İlk Çeşitleme tekniği her iterasyonda tek bir vektörü kümeler arasında hareket ettirmektedir. Tek bir vektörün hareketi amaç fonksiyonunda düşüşlere de neden olabilir. Bu nedenle bir dizi İlk Çeşitleme hareketinin arka arkaya uygulanmasıyla düşüşün ardından yükseliş de elde edilebileceği fikri üzerine graf kümelemede kullanılmakta olan Kernighan-Lin sezgisel yaklaşımına benzer bir yaklaşım olan Kernighan-Lin Chains yaklaşımı geliştirilmiştir. Amaç fonksiyonunda en yüksek artışı sağlayacak İlk Çeşitleme hareketi dizisinin tespitine çalışılmaktadır. Bu teknik de bir adım öteye götürülerek, Küresel K-Means'in içinde doğrudan uygulanmıştır. Burada, ping-pong stratejisi

olarak isimlendirilen iki adımlı teknikte, ilk adımda Küresel K-Means çalıştırılmaktadır. Eğer amaç fonksiyonunda bir artış olmazsa Kernighan-Lin Chains uygulanarak amaç fonksiyonunda artış denenmektedir. Eğer Küresel K-Means ile bir artış sağlandıysa Kernighan-Lin Chains tekniği uygulanmadan bir sonraki K-Means iterasyonuyla devam edilmektedir [24].

Küresel K-Means algoritması temel olarak yığın güncelleme (batch update) mantığıyla çalışmakta yani küme ağırlık merkezleri sadece ve sadece tüm dökümanlar en yakın oldukları kümelere atandıktan sonra hesaplanmakta ve güncellenmektedir. Zhong tarafından her bir döküman en yakın olduğu kümeye atanır atanmaz ilgili kümenin merkezinin Winner-Take-All olarak bilinen bir Rekabetçi Öğrenme (Competetive Learning) yaklaşımıyla güncellendiği online bir Küresel K-Means sürümünün yığın güncellemeye göre daha iyi bir kümeleme yaptığı ortaya konulmuştur [25].

3.1.2. İkiye Bölmeli K-Means – Karypis ve Ekibinin Çalışmaları

Karypis ve ekibi tarafından geliştirilen İkiye Bölmeli K-Means algoritması ile standart K-Means ve Birleştirici Hiyerarşik Kümeleme (Agglomerative Hierarchical Clustering) algoritmaları döküman kümeleme başarımı bakımından karşılaştırılmış ve karşılaştırma sonuçları döküman verilerinin doğası da dikkate alınarak ayrıntılı biçimde analiz edilmiştir. Çalışma kapsamında birleştirici hiyerarşik kümeleme için şu 3 teknik karşılaştırmalı olarak denenmiştir: Intra-Cluster Similarity Technique (IST), Centroid Similarity Technique (CST) ve Unweighted Pair Group Method with Arithmetic Mean (UPGMA). Bu üçü içerisinden en yüksek kümeleme kalitesine sahip olan tekniğin UPGMA olduğu görülmektedir. K-Means, İkiye Bölmeli K-Means ve UPGMA teknikleri de kümeleme kalitesi bakımından birbirleriyle karşılaştırıldığında İkiye Bölmeli K-Means algoritmasının hem K-Means algoritmasına hem de UPGMA algoritmasına üstünlük sağladığı deneylerle tespit edilmiştir. Ayrıca işlemsel karmaşıklığı bakımından da İkiye Bölmeli K-Means'in Birleştirici Hiyerarşik Kümeleme tekniklerine göre üstün olduğu görülmektedir – sırasıyla $O(n)$ ve $O(n^2)$ [26].

Zhao ve Karypis tarafından düşük hesaplama gereksinimine sahip kümeleme algoritmalarının kümeleme sorununa genellikle optimizasyon sorunu olarak yaklaşan ve belirli bir kümeleme ölçütü fonksiyonunu (clustering criterion function) maksimize ya da minimize etmeye çalışan algoritmalar olduğu vurgulanmaktadır ve döküman kümeleme için kullanılan çeşitli kümeleme ölçütü fonksiyonlarının başarımları değerlendirilmektedir [27]. Bu amaçla 7 farklı ölçüt fonksiyonunun başarımı 15 ayrı veri seti üzerinde yapılan deneylerle ölçülmeye çalışılmış, ek olarak bu ölçüt fonksiyonlarının karakteristiklerinin analizi yapılmıştır. Bu ölçüt fonksiyonları çeşitli kategoriler altında şu şekildedir:

- İçsel Ölçüt Fonksiyonları (Internal Criterion Function): Bu fonksiyonlar, dökümanların sadece dahil oldukları kümelere olan benzerliklerini dikkate almakta olup, diğer kümeleri hesaba katmamaktadır. Yani, kümeleme sürecini küme içi bir yaklaşımla ele aldıklarından bu ölçüt fonksiyonları İçsel Ölçüt Fonksiyonları olarak isimlendirilmektedir.
- Dışsal Ölçüt Fonksiyonları (External Criterion Functions): Genel olarak oluşan kümelerin birbirinden ne kadar farklı olduğu düşüncesi üzerine kurulu fonksiyonlardır. Kümeleme sürecini kümeler arası bir yaklaşımla ele almaktadırlar.
- Hibrit Ölçüt Fonksiyonları (Hybrid Criterion Functions): İçsel ve dışsal ölçüt fonksiyonlarının bir arada kullanılmasıyla elde edilen ölçüt fonksiyonlarıdır.
- Graf Tabanlı Ölçüt Fonksiyonları (Graph Based Criterion Functions): Her bir dökümanı çok boyutlu bir vektör olarak ele alan diğer ölçüt fonksiyonlarından farklı olarak dökümanlar arası ilişkilerin graflarla temsil edildiği yaklaşımlara uygun olarak kullanılan ölçüt fonksiyonlarıdır. Döküman kümelemede temel olarak 2 tür graf modeli vardır. Birinci tür dökümanlar arası benzerliklerden oluşturulan graftır. Diğer tür ise dökümanlar ve terimler arasındaki ilişkilerden oluşturulan iki parçalı (bipartite) graftır.

Karypis ve ekibi tarafından bu ve önceki çalışmada [26] bahsedilen ölçüt fonksiyonlarının ve çeşitli kümeleme algoritmalarının kullanılabildiği CLUTO (Clustering Toolkit) isimli bir yazılım geliştirilmiştir ve bu yazılım döküman kümeleme alanında karşılaştırma amacıyla sıkça kullanılmaktadır. Bu çalışmanın

geliştirildiği yeni bir çalışmayla esnek kümeleme (soft clustering) işlemi için oluşturulmuş yeni ölçüt fonksiyonları değerlendirilmiştir [28]. Esnek kümelemede bir dökümanın sadece belirli bir kümeye dahil edilmesi zorunluluğu yoktur; bir döküman belirli ölçüde bir veya daha fazla kümeye dahil olabilir. Bu çalışmada, bu amaçla esnek ölçüt fonksiyonlarının optimize edildiği keskin (hard) kümeleme tabanlı bir algoritma geliştirilmiş ve karşılaştırmalı deney sonuçları sunulmuştur. Deney sonuçlarına göre çoğu veri seti için esnek ölçüt fonksiyonları kümeleme kalitesini iyileştirmektedir.

Zhao ve Karypis'in döküman kümeleme alanındaki çalışmaları, döküman kümelerinin hiyerarşik bir biçimde oluşturulması amacıyla kullanılan kümeleme yaklaşımları ve çeşitli kümeleme ölçütü fonksiyonları değerlendirilerek devam etmiştir [29]. Temel olarak iki yaklaşım üzerinde durulmaktadır:

- Hiyerarşik Bölünmeli Kümeleme (Hierarchical Partitioning Clustering): Bu yaklaşımda veri kümesinin sürekli olarak ikiye bölünmesi söz konusudur. Veri kümesi önce ikiye bölünür, oluşan iki kümeden biri seçilerek o da ikiye bölünür ve bu şekilde yukarıdan aşağıya bölünme devam eder.
- Hiyerarşik Birleştirici Kümeleme (Hierarchical Agglomerative Clustering): Aşağıdan yukarıya bir yaklaşım olan bu yaklaşımda ilk olarak her bir dökümanın kendisi bir küme oluşturur. Benzerliklerine göre kümeler birleştirilerek daha büyük kümeleri oluşturur. Hiyerarşik olarak tek bir küme oluşuncaya kadar birleştirme işlemi devam eder.

Deney sonuçlarına göre bölünmeli kümeleme yaklaşımları birleştirici kümeleme yaklaşımlarına göre daha iyi sonuç vermektedir. Birleştirici kümeleme algoritmaları içerisinde ise en iyi sonucu UPGMA metodu vermektedir.

İşbirlikçi Kümeleme (Cooperative Clustering) yaklaşımı üzerine çalışmalar yapan Kashef ve Kamel tarafından bu yaklaşımın İkiye Bölmeli K-Means algoritmasına uygulandığı İşbirlikçi İkiye Bölmeli K-Means (Cooperative Bisecting K-Means – CBKM) adlı modelde, hiyerarşik küme ağacının oluşturulması sırasında tüm seviyelerde eş zamanlı olarak işletilen K-Means ve İkiye Bölmeli K-Means'ten elde edilen sonuçlar birleştirilerek tek başına K-Means ve İkiye Bölmeli KMeans'ten daha iyi kümeleme sonuçları elde etmeye çalışılmaktadır [30].

3.1.3. K-Means Tabanlı Diğer Algoritmalar ve İyileştirmeler

Zhuang ve Dai tarafından, Maximal Frequent Itemset (MFI) adı verilen yaklaşımda K-Means algoritmasının başlangıç koşullarını iyileştirmek ve böylece rastgele başlangıç koşullarına göre daha iyi kümeleme kalitesi elde etmek üzere, koleksiyondaki sık terimlerin yer aldığı dökümanların birbirine benzer olması gerektiği teziyle, Sık Nesne Kümeleri (Frequent Itemset) tespit yöntemi kullanılarak, yüksek yoğunluklu küme ağırlık merkezleri belirlenmektedir. Bu yöntemin az sayıda dökümanda çok sayıda küme elde edilmek istendiğinde daha başarılı olduğu öne sürülmektedir [31].

MFI K-Means algoritması döküman uzunluğuna karşı duyarlıdır ve uzun dökümanlar kısa olanlara göre daha fazla MFI içerdiğinden, başlangıç koşulu olarak uzun dökümanların baskın olduğu küme ağırlık merkezleri seçilmektedir. Bu sorunu gidermek için MFI K-Means algoritması dökümanlardaki sadece ilk 300 terimi kullanmaktadır [31]. Bu durumu ortadan kaldırmak ve başlangıç küme ağırlık merkezi elde etme sürecini basitleştirmek için Wang ve arkadaşları tarafından Sık Terim Kümelerine (Frequent Term Set) dayalı hibrit bir algoritma olan Simple Hybrid Document Clustering (SHDC) algoritması geliştirilmiştir. En sık k adet terim kümesinin kullanıldığı SHDC algoritmasında, bir sık terim kümesindeki bütün terimleri içeren dökümanlar küme adayı olarak işaretlenmekte ve bu dökümanlar kendilerine benzer dökümanlarla birleştirilerek başlangıç koşulu olarak kullanılacak küme ağırlık merkezleri bulunmaktadır [32].

Jing, Ng ve Huang tarafından geliştirilen Entropy Weighting K-Means (EWKM) algoritması, yüksek boyutlu ve seyrek veri setlerinde, alt vektör uzayında (subspace) yer alan kümelerin neden olduğu düşük kümeleme kalitesine çözüm olarak K-Means döngüsünde her bir küme için her bir boyutun özel olarak ağırlıklandırılması ve kümeleme ölçütü fonksiyonunda bu ağırlıkların entropisi kullanılarak kümelerdeki döküman sınıfları için önemli olan boyutların tespiti fikrine dayanmaktadır [33].

K-Means algoritmasının her bir iterasyonunda bir önceki iterasyonda hesaplanan küme ağırlık merkezleri, bu iterasyonun başlangıç koşulu konumundadır. Dolayısıyla, her bir iterasyonda uç değer (outlier) sayılabilecek dökümanlar

nedeniyle küme ağırlık merkezleri doğru sınıflardaki dökümanların yoğun olduğu bölgeden uç değerlere doğru sapabilmektedir. Xinwu tarafından K-Means algoritmasına önerilen iyileştirmede bu uç değerlerin tespiti ve küme ağırlık merkezi hesaplamasına katılmaması sayesinde kümeleme kalitesinin artırılmasına çalışılmaktadır [34].

K-Means algoritması genellikle yerel optimal çözümde takılıp kalmakta, bu da elde edilen kümelemenin başarımının düşük olmasına neden olmaktadır. Mahdavi ve Abolhassani tarafından bu soruna çözüm olarak Harmony Search (HS) optimizasyon metodu temel alınarak Harmony K-Means Algoritması (HKA) geliştirilmiştir [35]. HS müzisyenlerin doğaçlama yaparken mükemmel armoniyi oluşturacak sesleri arayışlarını taklit etme fikri üzerine geliştirilmiş meta-sezgisel (meta-heuristic) bir algoritmadır ve çok çeşitli optimizasyon problemlerinde başarıyla uygulanmaktadır. HS'nin ayrıca diğer optimizasyon tekniklerine göre çok az matematiksel hesaplama yükü getirme ve karar değişkenleri için başlangıç değeri atanmasını gerektirmeme gibi çeşitli üstünlükleri vardır. Stokastik bir yaklaşım kullanan HS algoritmasının yerel optimuma yakınsama sorunu yoktur. Ayrıca, genellikle stokastik yaklaşımların küresel optimuma yakınsaması uzun zaman alırken, bu çalışmada döküman kümeleme için HS'ye dayalı olarak geliştirilen Harmony K-Means Algoritması'nın küresel optimuma yüzde yüz olasılıkla yakınsadığı ispatlanmaktadır.

Harmony Search metodunun Parçacık Sürü Optimizasyonu (Particle Swarm Optimization – PSO) kullanılarak geliştirilen bir sürümü olan Global-Best Harmony Search (GBHS) algoritmasının uygulandığı diğer bir çalışmada, Cobos ve arkadaşları tarafından geliştirilen hibrit bir algoritma olan Iterative Global-Best Harmony Search K-Means Algorithm (IGBHSK) algoritmasında, çözüm uzayında küresel bir çözüm araştırmak için GBHS, çözümleri iyileştirmek için yerel bir strateji olarak K-Means ve küme sayısını otomatik olarak bulmak için de Bayesian Information Criterion (BIC) kullanılmaktadır [36].

Kalogeratos ve Likas tarafından yüksek boyutlu ve seyrek döküman verileri için K-Means algoritmasında küme prototipi olarak küme ağırlık merkezlerinin kullanılmasının özellikle az sayıda döküman içeren kümeler söz konusu olduğunda iyi bir seçim olmayabileceği, buna çözüm olarak da sentetik küme prototipleri

fikrinin uygulanabileceği öne sürülmektedir. Bu amaçla MedoidKNN adı verilen sentetik küme prototipi oluşturma yaklaşımında ilk olarak küme içerisinde baskın olan sınıftan dökümanlar belirlenir, bunlardan küme temsilcisi hesaplanır ve bu temsilciye göre hatalı olarak bu kümeye atanmış dökümanların daha uygun kümelere atanması sağlanır [37].

3.2. Model Tabanlı Kümeleme Yaklaşımları

Kümeleme yöntemleri genel olarak ayırıcı (discriminative) yani benzerlik tabanlı yaklaşımlar ve üretici (generative) yani model tabanlı (olasılıksal) yaklaşımlar olarak ikiye ayrılmaktadır. Benzerlik tabanlı yaklaşımlarda dökümanlar arası benzerlikler kullanılarak bir amaç fonksiyonu en iyileştirilmeye çalışılırken, model tabanlı yaklaşımlarda dökümanlar belirli bir istatistiksel dağılım açısından değerlendirilmekte ve her bir döküman kümesini temsil eden model parametrelerinin dökümanlardan öğrenilmesine çalışılmaktadır [38].

Metin sınıflandırmada yaygın olarak kullanılmakta olan model tabanlı yaklaşımlar Zhong ve Ghosh tarafından döküman kümeleme açısından ele alınmış ve şu üç olasılıksal (probabilistic) modelin karşılaştırması yapılmıştır: Çokdeğişkenli (Multivariate) Bernoulli Modeli, Çokterimli (Multinomial) Model ve von Mises-Fisher Modeli. Bu üç farklı model değişik veri setleri üzerinde denenerek Bernoulli modelinin döküman kümeleme için uygun olmadığı, von Mises-Fisher modelinin ise çok terimli modellere göre üstünlük sağladığı gösterilmiştir [38]. Ghosh ve ekibi tarafından Küresel K-Means algoritmasıyla von Mises-Fisher dağılımlarını içeren olasılıksal modellerin teorik bağlantısı ortaya konulmuş olup, normalize edilmiş döküman vektörleri gibi birim hiperkürenin yüzeyinde dağılmış yönlü (directional) veriler için özelleştirilmiş, von Mises-Fisher dağılımlarının karışımından oluşan bir model öne sürülmüştür [39] .

Çokterimli dağılımlara alternatif olarak Elkan ve grubu tarafından dökümanların modellenmesi amacıyla Dirichlet Birleşik Çokterimli Modeli (Dirichlet Compound Multinomial Model – DCM) önerilmiştir (ayrıca çokterimli Polya Dağılımı olarak da bilinir). DCM, diğer çokterimli dağılımlardan farklı olarak dökümanlardaki

"burstiness" olarak ifade edilen bir özelliği de hesaba katan bir dağılımdır. Bu nedenle de benzeri dağılımlara göre metin dökümanlarını daha iyi temsil ettiği öne sürülmektedir. "Burstiness" ile ifade edilen şudur: eğer bir sözcük bir dökümanda bir kez geçiyorsa, o sözcük çok yüksek olasılıkla aynı dökümanda bir kez daha geçecektir. Örneğin, otomotivle ilgili bir dökümanda Toyota ve Nissan sözcüklerinin yer alma olasılığı eşittir. Ancak, Toyota sözcüğü bir kez geçtikten sonra Toyota sözcüğünün bir kez daha geçme olasılığı Nissan sözcüğünün ilk kez geçme olasılığından daha yüksektir. İlk olarak döküman sınıflandırma için kullanılan DCM [40] daha sonra Elkan tarafından döküman kümeleme alanında uygulanmıştır [41]. İlgili çalışmada DCM derinlemesine incelenmiş ve DCM'ye çok yakın yeni bir dağılım türetilmiştir. EDCM olarak adlandırılan bu dağılım, DCM'den farklı olarak üstel dağılımlar ailesindendir (exponential family of distributions). Beklenti Ençoklama (Expectation Maximization – EM) tekniği EDCM ile birlikte kullanılarak DCM'ye daha hızlı ve daha başarılı bir kümeleme elde edilmiştir.

3.3. Dil Modeli Tabanlı Kümeleme Yaklaşımları

Erkan tarafından döküman kümelemede kullanılmak üzere geleneksel terim-döküman matrisi yerine n döküman sayısı olmak üzere n x n boyutlu bir matris önerilmektedir. Bu matrisin satırlarını oluşturan döküman vektörlerinin her bir boyutundaki değer, karşılık gelen dökümanın dil modeline (language model) bağlı olarak hesaplanan üretilebilirlik olasılıklarıyla (generation probability) ilişkili olarak hesaplanmaktadır. Bu matris bir kümeleme algoritmasına doğrudan girdi olarak verilebilmektedir. Bunun yerine Erkan tarafından bu matris kullanılarak bir üretilebilirlik grafı (generation graph) oluşturulmuştur. Bu yönlendirilmiş grafın köşelerini dökümanlar ve kenarlarını da vektörlerdeki ağırlıklar oluşturmaktadır. Bu graf üzerinde rastgele yürüyüş (random walk) tekniği kullanılarak döküman benzerliklerinin ölçümlenmesi ve dökümanların kümelenmesi sağlanmıştır [42].

Zhou, Zhang ve Hu tarafından dil modelinin yumuşatılması (smoothing) ihtiyacı ön plana çıkartılmış ve bunun için de anlambilimsel yumuşatma (semantic smoothing) kavramı üzerinde durulmuştur. Burada yumuşatılmadan kasıt, dil modelinin oluşturulması sırasında daha önce karşılaşılmamış sözcük ya da

kavramlara sıfır olasılık atanması hatasına düşülmemesinin sağlanmasıdır. Anlambilimsel yumuşatma ile de modelin içeriğe duyarlı (context sensitive) olması ile eşanlamlılık ve eşseslilik durumlarını dikkate alır hale getirilmesi hedeflenmektedir. İlgili çalışmada "konu imzası" (topic signature) dil modeli olarak adlandırılan bir dil modeli geliştirilmiştir. Konu imzası, birden çok dökümanda geçen ve konu bilgisi taşıyabilecek herhangi bir metin parçası olarak tanımlanmaktadır. Örneğin, tek başına sözcükler, çok sözcüklü ifadeler, ontolojik kavramlar ve kavram çiftleri birer konu imzası olabilirler. Dökümanlardaki konu imzaları tespit edildikten sonra her bir konu imzasının tek tek sözcüklere eşlenmesi yani anlambilimsel eşleme(semantic mapping) yapılır. Geliştirilen bu model ile deneysel olarak birleştirici kümeleme yaklaşımlarında anlambilimsel yumuşatma oldukça etkili olurken bölünmeli kümeleme yaklaşımlarında TFIDF kadar etkili olamadığı gösterilmiştir [43]. Aynı ekibin diğer bir çalışmasında anlambilimsel yumuşatmanın yanı sıra dökümanlar arası uzaklık ölçütü olarak Kullback-Leibler Iraksaması'nın (Kullback- Leibler Divergence) kullanıldığı birleştirici hiyerarşik kümeleme üzerinde başarılı sonuçlar alınmıştır [44].

Mevcut anlambilimsel yumuşatma uygulanan dil modeli tabanlı bölünmeli kümeleme yöntemlerinin döküman koleksiyonlarındaki "general" gibi eşseslilik (polysemy) sorununa yol açan sözcüklere doğru ağırlık verilememesi nedeniyle yeterince başarılı olamadığını öne süren Wen ve Li tarafından karışım olasılıksal modellerden (mixture probability model) ilham alınarak karışım dil modeli kullanan bir kümeleme yaklaşımı geliştirilmiştir. Bu yeni modelin TFIDF, karışım çokterimli model ve anlambilimsel modele göre üstün sonuçlar ürettiği gösterilmiştir [45].

3.4. Sonek Ağacı Kümeleme

Vektör Uzayı Modeli gibi geleneksel modeller genellikle dökümanları sözcük kümeleri olarak ele alırlar ve sözcüklerin döküman içerisindeki sıralanışını gözardı ederler. İlk olarak Zamir ve Etzioni tarafından önerilen Sonek Ağacı Kümelemede ise dökümanlar sözcük dizileri olarak ele alınırlar ve sözcükler arasındaki mesafe bilgisinden yararlanılır [10, 11]. STC algoritması, sonek ağacı kullanarak ortak sözcük ve ifadeleri içeren dökümanların verimli olarak tespiti fikrine dayanmaktadır.

Chim ve Deng tarafından STC'de bir iyileştirme olarak dökümanlar arası benzerliklerin tespiti için yeni bir benzerlik ölçütü önerilmiş, bu yeni ölçüt Gruportalaması Birleştirici Hiyerarşik Kümeleme (Group-average Agglomerative Hierarchical Clustering – GAHC) algoritmasına adapte edilerek NSTC olarak adlandırılan yeni bir STC algoritması geliştirilmiştir [46].

Li, Chung ve Holt tarafından geliştirilen Sık Sözcük Dizileri Tabanlı Kümeleme (Clustering based on Frequent Word Sequences – CFWS) ve Sık Sözcük Anlamı Tabanlı Kümeleme (Clustering based on Frequent Word Meaning Sequences – CFWMS) algoritmaları da sözcüklerin döküman içerisindeki sıralanışını önemsemekte ve bunun için Genelleştirilmiş Sonek Ağacı (Generalized Suffix Tree – GST) yapısını kullanmaktadırlar. Klasik STC'den farklı olarak CFWS algoritması sık sözcük dizilerini araştırmakta ve böylece bir çeşit boyut indirgeme gerçekleştirildiği için orijinal STC'nin çok boyutlu döküman topluluklarındaki performans sorunu ortadan kaldırılmış olmaktadır [47]. CFWMS algoritması ise sadece dökümandaki sözcükleri kullanmakla kalmayıp WordNet [8] tarafından sağlanan eşanlamlı olma ve kavramsal ilişki içinde olma (hypernym ya da hyponym olma) durumlarını kullanarak kümelemeye ontolojik bilgiler katmakta ve böylece CFWS'den daha iyi bir kümeleme yapmaktadır [47].

3.5. Sık Nesne Kümeleri Tabanlı Yaklaşımlar

Döküman kümelemenin kendine özgü çok boyutluluk sorununa çözüm getirmek, anlamlı küme etiketleri elde etmek ve büyük veri setlerini yüksek performansla kümeleyebilmek amacıyla ilk olarak Beil, Ester ve Xu tarafından Sık Nesne Kümeleri (Frequent Itemset) fikrinin döküman kümelemeye uyarlandığı yenilikçi bir algoritma olan Sık Nesne Kümeleri Tabanlı Kümeleme (Frequent Itemset-based Clustering – FTC) algoritması geliştirilmiştir [48]. Sık nesne kümeleri yani sık terim kümeleri, Apriori [49] benzeri bir ilişki kuralı madenciliği (Association Rule Mining) algoritması ile tespit edilmektedir. Bulunan sık terim kümeleri daha sonra Standart Overlap (SO) ve Entropy Overlap (EO) gibi örtüşme hesaplamaları yardımıyla gerçek döküman kümelerine dönüştürülmektedir. İlgili çalışmada düz kümelemenin yanı sıra (flat clustering) hiyerarşik kümeleme yapan bir algoritma (HFTC) da

geliştirilmiş ve bilinen yaygın kümeleme yaklaşımlarıyla deneysel karşılaştırmaları verilmiştir [48]. Shi ve Ester tarafından yayınlanan teknik raporda FTC'nin kümeleme kalitesini artırmaya yönelik iyileştirilmiş örtüşme hesaplamaları (Improved SO ve Improved EO) gibi çeşitli iyileştirmeler önerilmiştir [50].

Fung, Wang ve Ester tarafından geliştirilen Sık Nesne Kümesi Tabanlı Hiyerarşik Kümeleme (Frequent Itemset-based Hierarchical Clustering – FIHC) algoritması, HFTC'de olduğu gibi sık terim kümelerinin tespitine dayalı, döküman kümelemenin özel gereksinimlerini karşılamaya yönelik yenilikçi bir algoritma olup sık nesne kümelerinin seçilerek gerçek kümelerin oluşturulması bakımından HFTC'den ayrılmaktadır. HFTC, bir sık terim kümesinden başlayarak aralarındaki örtüşme değerini minimize edecek şekilde sırasıyla diğer sık terim kümelerini seçmekte, bu nedenle de sık nesne kümelerinin seçiliş sıralamasına bağımlı olmaktadır. FIHC'de dökümanlar olabilecek en iyi kümelere atanmakta, sıralamaya bağımlı olma dezavantajı ortadan kaldırılmaktadır [51, 52]. FIHC algoritması literatürde oldukça önemli bir çalışma olup, eksikliklerinin giderildiği ve çeşitli geliştirmelerin yapıldığı çok sayıda çalışmaya kaynaklık etmiştir.

Liu ve He tarafından geliştirilen Frequent Term Set-based Clustering (FTSC) ve bunun hiyerarşik çeşitlemesi olan Frequent Term Set-based Hierarchical Clustering (FTSHC) algoritmaları da Apriori [49] algoritması kullanarak sık terim kümelerini tespit etmekte, ardından bu sık terim kümelerindeki sık sözcüklere göre dökümanları kümelemektedir. Önceki yaklaşımlardan farklı olarak sık terim kümelerinin tespitinde tüm terimler değil, yalnızca dökümanlar için ayırıcı değere sahip öznitelik (feature) terimler kullanılmakta, böylece daha en başından bir boyut indirgeme yapılarak kümeleme verimliliği artırılmaya çalışılmaktadır [53].

Kryszkiewicz ve Skonieczny tarafından döküman koleksiyonundaki sık nesne kümeleri çok fazla olduğunda FIHC algoritmasının başarısız olduğu belirtilmekte ve çözüm olarak sık terim kümeleri yerine sık kapalı kümelerin (frequent closed set) kullanıldığı bir dizi iyileştirme önerilmektedir [54].

Malik ve Kender tarafından yüksek sıklığın her zaman yüksek kümeleme kalitesi anlamına gelmeyeceği öne sürülmekte, bu nedenle en ilginç ilişki kurallarının bulunmasına yönelik araştırmalardan ilham alınarak "kapalı ilginç" terim kümeleri

yaklaşımının kullanıldığı bir hiyerarşik kümeleme modeli önerilmektedir. Bu modelde 20 farklı "ilginçlik" ölçütü deneysel olarak karşılaştırılmış, Ortak Bilgi (Mutual Information), Katma Değer (Added Value), Yule's Q ve Ki-Kare (Chi-Square) ölçütlerinin en yüksek kümeleme başarımını ürettikleri gösterilmiştir [55].

Krishna ve Bhavani'nin sık terim kümeleri tabanlı döküman kümeleme çalışmasında ise öncelikle her bir dökümanda en sık geçen p adet terim belirlenmekte ve sadece bu terimler kullanılarak Apriori algoritmasıyla sık terim kümeleri oluşturulmaktadır. Daha sonra her bir döküman içerdiği sık terim kümelerine göre oluşturulan yalnızca bir başlangıç kümesine dahil edilmekte, ardından bu başlangıç kümeleri belirli bir benzerlik kriterine göre bölünerek gerçek döküman kümeleri elde edilmektedir [56].

Sık nesne kümeleri yaklaşımı kullanan başka bir çalışma da Zhang ve arkadaşları tarafından yapılmıştır. Bu çalışmada Maksimum Yakalama (Maximum Capturing – MC) olarak adlandırılan bir yöntem kullanılarak döküman kümeleri oluşturulmaktadır. MC yaklaşımında öncelikle dökümanlar içerdikleri sık terim kümeleriyle temsil edilmektedirler. Sık terim kümelerine bağlı olarak dökümanlar arası benzerlik hesaplamak için üç farklı yol kullanılmıştır. Bu benzerlik ölçütleri kullanılarak benzerlik matrisi oluşturulmakta, daha sonra birbirlerine en benzer iki dökümanın mutlaka aynı kümede olması koşuluyla benzerlik matrisinden yararlanılarak dökümanlar kümelere ayrılmaktadır. Eğer belirli sayıda küme oluşması isteniyorsa fazladan bir normalizasyon adımıyla fazla sayıda olan kümeler birleştirilmektedir [57].

Chen, Tseng ve Liang tarafından geliştirilen Fuzzy Frequent Itemset-Based Hierarchical Clustering (F2IHC) algoritması ile FIHC algoritmasının kümeleme kalitesini artırmak amacıyla bulanık ilişki kuralı madenciliği (fuzzy association rule mining) yöntemleri kullanılmaktadır [58]. Döküman temsilini zenginleştirmek üzere Wikipedia gibi harici bilgi kaynaklarının kullanımı oldukça yaygındır. Kiran, Shankar ve Pudi tarafından geliştirilen sık kapalı terim kümeleri tabanlı hiyerarşik kümeleme yaklaşımında da harici bilgi kaynağı olarak Wikipedia'dan yararlanılmıştır. Wikipedia'daki dış bağlantılar (outlinks) ve kategoriler kullanılarak başlangıç kümelerindeki dökümanların temsili zenginleştirilmekte, ardından

dökümanların dahil edileceği kümelerin tespitinde dökümanlar bu yeni halleriyle kullanılmakta, böylece kümeleme kalitesini artırıcı yönde daha doğru küme seçimleri gerçekleştirilmektedir [59].

3.6. Graf Tabanlı Kümeleme Yaklaşımları

Dökümanların kümelenmesinin sözcüklerin kümelenmesi, benzer şekilde de sözcüklerin kümelenmesinin dökümanların kümelenmesi sonucunu ortaya çıkartacağı şeklindeki, sözcük ve döküman kümelemenin ikiliği (duality of word and document clustering) yaklaşımını ele alan Dhillon tarafından dökümanları ve sözcükleri ayrı ayrı değil de eşzamanlı (simultaneously) kümeleyebilmek amacıyla döküman koleksiyonunun dökümanlar ve sözcükler arasında bir çift-taraflı graf (bipartite graph) olarak temsil edildiği bir model geliştirilmiştir. Bu modele göre döküman ve sözcüklerin eşzamanlı kümelenmesi problemi bir çift-taraflı graf bölmeleme (bipartite graph partitioning) problemi olarak ele alınmaktadır. Bu graf modelinde dökümanlar ve sözcükler grafın köşelerini oluşturmakta, eğer bir sözcük bir dökümanda yer alıyorsa ikisi arasında yönsüz bir kenar yer almaktadır. Dökümanları veya sözcükleri kendi aralarında birleştiren kenarlar bulunmamaktadır. Kenar ağırlıkları olarak döküman vektörlerindeki terim ağırlıkları kullanılmaktadır. Oluşan graf bölmeleme probleminin çözümü amacıyla yeni bir spektral kümeleme algoritması geliştirilmiştir [60].

Bekkerman, Sahami ve Learned-Miller tarafından Markov Random Field (MRF) olarak bilinen graf modeline dayalı olarak Combinatorial Markov Random Field (Comraf) olarak isimlendirilen yeni bir yönlendirilmemiş graf (undirected graph) modeli geliştirilmiştir. Graf tabanlı gözetimsiz öğrenme problemleri genellikle Maximum Likelihood (ML) çatısı kullanılarak çözülmesine rağmen Comraf için Most Probable Explanation (MPE) çatısının tercih edildiği belirtilmektedir. Comraf modelinin standart (gözetimsiz) kümeleme, yarı-gözetimli kümeleme, interaktif kümeleme ve tek-sınıf (one-class) kümeleme alanlarında, çok yönlü (multimodal) veri kümeleri üzerinde uygulamaları yapılmıştır. Burada çok yönlü veri kümesiyle kastedilen, kümelenecek verinin birden çok görünümünün (modality) olmasıdır. Örneğin, metin dökümanları için döküman içindeki sözcükler bir görünüm

olabilirken, dökümanların başlıkları, dökümanların yazarları ayrı birer görünümdür. Bunlardan sadece birini kullanarak çalışan bir sistem tek-yönlü (unimodal), ikisini bir arada kullanarak çalışan bir sistem ise iki-yönlü (bimodal) olarak isimlendirilmektedir. Bu çalışmada önerilen sistem, verileri çok yönlü olarak değerlendirebilmektedir [61].

Hossain ve Angryk tarafından yapılan çalışmada sık anlamlar (frequent senses) fikri üzerine kurulu bir kümeleme tekniği oluşturulmaktadır. GDClust algoritması yaygın kullanımı olan sık sözcükler (frequent keywords) yerine sık anlamlardan yararlanmaktadır. Bu algoritmada dökümanlar hiyerarşik döküman grafları olarak temsil edilmekte ve sık anlamları karşılayacak olan sık altgrafların (frequent subgraphs) bulunması için Apriori [49] tekniği kullanılmaktadır. Keşfedilen sık altgraflardan yararlanılarak da döküman kümeleri oluşturulmaktadır. Döküman grafları oluşturulurken WordNet [8] ontolojileri kullanılarak dökümandan özel olarak seçilmiş her bir anahtar sözcüğe karşılık gelen hiyerarşik sınıflandırma bilgisi elde edilir. Bu sınıflandırma bilgisindeki her bir seviye için grafa bir köşe eklenir. İki köşe arasında kavramsal bir ilişki (hypernym ya da hyponym olma durumu) var ise bu iki köşe bir kenar ile birleştirilir. Anlamsal olarak oluşturulmuş bu graflar, ortak anahtar sözcüklere sahip olmasalar bile ortak kavramları içerdiği için birbirine benzeyen dökümanları daha iyi temsil etmektedir. Geleneksel sepet analizinde birlikteliklerin bulunmasına benzer şekilde Apriori tekniği kullanılarak sık geçen altgrafların tespiti gerçekleştirilmektedir. Bulunan sık altgraflar daha sonra iki döküman arasındaki benzerliğin hesaplanmasında kullanılmaktadır. GDClust dökümanları gruplamak için HAC (Hierarchical Agglomerative Clustering) kullanmaktadır [62].

Tüm dökümanlardan tek bir sözcüksel grafın (lexical graph) oluşturulduğu, Sha, Zhang ve Jiang tarafından yapılan çalışmada, grafın köşeleri sözcüklerden, kenarları ise kenarların birleştirdiği sözcüklerin eş zamanlılık değerinden (concurrent) oluşmaktadır. Ayrıca, her bir köşe, o köşedeki sözcüğün geçtiği dökümanların bilgisini de nitelik olarak barındırmaktadır. Kenar değerleri, birleştirdikleri sözcüklerin korelasyon değeri yani dökümanlarda birlikte kaç kez geçtikleridir. Her bir sözcüğün yani köşenin derecesi ise o köşedeki sözcüğün toplam kaç farklı sözcükle birlikte geçtiğidir. Graftaki yüksek dereceli köşeler küme merkezi, o köşelerdeki sözcükler de küme adı/etiketi olarak alınmaktadır. Grafın kümelere

bölümlenmesinde şu kural uygulanmaktadır: Eğer iki sözcük derece, kenar değeri ve nitelik bakımından birbirine çok benziyor ise bu ikisinin kümelemeye katkısı da yakındır ve ilgili iki köşe birleştirilmelidir. Bu algoritma arama motoru sonuçları ve Reuters veri setleri üzerinde denenmiş, aynı veri setleri üzerinde işletilen K-Means ve STC kümeleme teknikleriyle karşılaştırılmıştır. Deney sonuçlarına göre Lexical Graph algoritması arama sonuçlarının kümelenmesinde diğer iki algoritmaya göre daha başarılı olmasına rağmen Reuters veri setinde başarısız olmuştur [63].

Vektör Uzayı Modeli gibi yapıların, dökümanların anlambilimsel (semantic) yapısını temsil edemeyişi sorununa çözüm olarak Wensheng ve Guohe tarafından yeni bir metin yapısı graf modeli (text structure graph model) ileri sürülmektedir. Bu modelde, dökümanı oluşturan karakteristik terimlerin sıklığı ve dökümandaki yerleşimleri önemsenmektedir. Karakteristik terimler arasındaki anlambilimsel (semantic) ilişkiyi tanımlayan iki anlambilimsel pencere (semantic window) tanımlanmaktadır: paragraf penceresi ve cümle penceresi. İki farklı terim aynı paragrafta ancak farklı cümlelerde yer alıyorsa paragraf penceresine dahildir. İki terim aynı cümlede yer alıyorsa cümle penceresine dahildir. Bu modelde her bir döküman için bir graf oluşturulur. Grafın köşelerini karakteristik terimler oluşturmaktadır. Eğer iki terim herhangi bir anlambilimsel pencereye dahil ise ilgili iki köşe arasında bir kenar oluşturulur ve bu kenara bir birliktelik derecesi (cooccurence degree) atanır. Bu birliktelik derecesinin hesaplanmasında terimlerin dahil oldukları pencerelerden yararlanılır. Böylelikle bu graf modelinde karakteristik terimler, sıklıkları ve döküman içerisindeki konumları arasındaki ilişkiler yansıtılmış olur, yani dökümanların anlambilimsel yapısı dikkate alınmış olur. Graf modelinin etkin olarak kümelenebilmesi için grafların benzerliği graf spektrumu kullanılarak tanımlanmakta ve graflar spektrumları ile serialize edilmektedir. Böylece grafların benzerliği problemi graf spektrumlarının benzerliği problemine dönüşmektedir. Bu teknik sayesinde graftaki köşe ve kenarlar arasındaki ilişkiler maksimum derecede korunmak suretiyle hesaplama karmaşıklığı azaltılmaktadır. K-Means ile yapılan karşılaştırmalı deneylerde, spektral kümelemenin K-Means'ten daha iyi kümeleme yaptığı görülmektedir. Ancak n kümelenen döküman sayısı olmak üzere bu algoritmanın işlemsel karmaşıklığı $O(n^2)$ düzeyindedir [64].

Yoshida tarafından kümeleme problemi için Ortak Bilgiye (Mutual Information) dayalı bir graf modelinin önerildiği çalışmada veri nesneleri yani dökümanlar grafın köşelerini, dökümanlar arası benzerlik ilişkisi için tanımlanan yeni bir fonksiyona göre hesaplanan değerler de kenarları oluşturmaktadır. Keskin Kümeleme (hard clustering) olarak ele alındığında, söz konusu kenar ağırlıklı graf modeli için ortak bilgi tabanlı kümeleme probleminin kombinatoryal (combinatorial) optimizasyon problemine yakınsadığı, böylece bu graf modeli üzerinde uygulanan spektral kümeleme yaklaşımının etkili sonuçlar verdiği gösterilmektedir [65].

Lee ve On tarafından gerçek kümeleme uygulamalarında küme sayısının 2-3'ten daha fazla olduğu ve döküman kümelerinin çoğunlukla çok farklı sayıda döküman içerecek şekilde son derece eğik (skewed) dağılımlara sahip olduğu, bunlardan dolayı da geleneksel kümeleme yaklaşımlarının yeterince etkin olamadığı belirtilmektedir. Bu duruma çözüm getirmek amacıyla, dökümanlardan oluşturulan benzerlik grafının (similarity graph) normalize edilmiş parçaları (cut) üzerinde ikiye bölme ve birleştirme (bisection and merge) adımları uygulayan bir algoritma geliştirilmiştir [66].

3.7. Bulanık Kümeleme Yaklaşımları

En çok bilinen ve en yaygın kullanılan bulanık bölünmeli kümeleme (fuzzy partitional clustering) algoritması Bulanık C-Means (Fuzzy C-Means) algoritmasıdır. İlk olarak Dunn tarafından ortaya atılmış [67] ve Bezdek tarafından geliştirilmiştir [68]. Bulanık kümelemede bulanık mantıkta (fuzzy logic) olduğu gibi her bir döküman, kümelerin her birine [0, 1] aralığında bir üyelik değeri ile dahildir. Dolayısıyla bir dökümanın tüm kümelere olan üyelik değerlerinin toplamı 1 olmaktadır. Döküman hangi küme merkezine yakın ise o kümeye ait olma üyeliği diğer kümelere ait olma üyeliklerinden daha büyük olacaktır.

İlk olarak Mendes ve Saks tarafından bulanık kümelemenin döküman kümeleme alanında uygulanabilirliği araştırılmış, bu amaçla Bulanık C-Means algoritmasındaki uzaklık (dissimilarity) fonksiyonunda ve amaç fonksiyonunda gerekli değişiklikler yapılarak, Hiperküresel Bulanık C-Means (Hyperspherical Fuzzy C-Means – H-

FCM) algoritması geliştirilmiş, geleneksel keskin kümeleme (hard clustering) yapan yöntemlerle karşılaştırılarak, H-FCM'nin çoğu durumda K-means algoritmasından daha iyi kümeleme yaptığı belirtilmiştir [69]. Mendes ve Sacks tarafından bu çalışma ilerletilerek Hiyerarşik Hiperküresel Bulanık C-Means (Hierarchical Hyperspherical Fuzzy C-Means – H2-FCM) adı verilen hiyerarşik kümeleme algoritması geliştirilmiştir. Anlamlı bir konu hiyerarşisi oluşturulması amacıyla bulanık kümeleri hiyerarşik olarak birbiriyle ilişkilendirmek için asimetrik benzerlik (asymmetric similarity) fikrini kullanan bu algoritmanın n döküman sayısı ve c bulanık küme sayısı olmak üzere işlemsel karmaşıklığının O(nc2) olduğu ve geniş döküman kümeleri için ölçeklenebilir olduğu gösterilmektedir [70].

Oh, Honda ve Ichihashi tarafından çok değişenli ve kategorik verilerin bulanık kümelenmesi için geliştirilen Fuzzy Clustering for Categorical Multivariate Data (FCCM) algoritmasının [71] döküman ve boyut sayısının yüksek olduğu geniş koleksiyonlarda uygulanabilirliğini artırmak üzere Kummamuru, Dhawale ve Krishnapuram tarafından FCCM algoritmasında değişiklik yapılarak Fuzzy CoDoK adı verilen algoritma geliştirilmiştir. Aynı çalışmada, Küresel K-Means algoritmasının bulanık bir sürümü olarak Küresel FCM (Spherical FCM – SFCM) algoritması da geliştirilmiş, yapılan deneylere göre az sayıda ayrık kümelerin bulunduğu verilerde Küresel K-Means algoritmasının Fuzzy CoDoK algoritmasından daha iyi sonuçlar verdiği, ancak genellikle örtüşen kümelerin bulunduğu verilerde SFCM ve Fuzzy CoDoK algoritmalarının daha başarılı olduğu ortaya konulmuştur[72].

Tjhi ve Chen tarafından geliştirilen Possibilistic Fuzzy Co-Clustering (PFCC) algoritması geniş ve yüksek boyutlu verilerin analizi için olasılıksal ve bulanık ortaklaşa kümeleme (co-clustering) yaklaşımlarını birleştirmektedir. Bu algoritmanın döküman ve sözcüklerde olabilecek uç değerlere karşı (outlier) dirençli olduğu, iyi derece açıklayıcı döküman kümeleri oluşturduğu, yüksek boyutlu verilerde iyi başarım gösterdiği ve olasılıksal kümelemedeki başlangıç koşullarına karşı duyarlılığının düşük olduğu öne sürülmektedir [73].

Deng ve arkadaşları tarafından klasik Bulanık C-Means algoritmasının zayıf yönlerinin iyileştirilerek döküman kümelemede uygulanması amacıyla yapılan

çalışmada öncelikle yüksek boyutlu döküman vektörlerinde sözcüklerin kalitesini belirleyen bir formülasyona göre öznitelik seçimi yapılmaktadır. Uzaklık ölçütü olarak Düzenleme Uzaklığı (Edit Distance) olarak da bilinen Levenshtein Uzaklığı (Levenshtein Distance) [74] kullanılan çalışmada ayrıca Bulanık C-Means algoritmasının başlangıç koşullarına yüksek derece bağımlı oluşuna çözüm olarak yüksek-güçlü örnek nokta kümesi (high-power sample point set) kavramı önerilmektedir [75].

Song, Guo ve Chen tarafından yapılan çalışmada döküman temsili için WordNet [8] referans sisteminden yararlanılarak Ağırlıklandırılmış Kavramsal Model (Weighted Conceptual Model – WCM) ortaya konulmakta ve bu anlambilimsel modele dayalı olarak Bulanık Anlambilimsel Kümeleme (Fuzzy Semantic Clustering – FSC) algoritması geliştirilmektedir [76]. WCM modelinde konu, ağırlık merkezi kavramları (centroid concepts) ve çevresel kavramlar (peripheral concepts) tanımlanmakta, bunlar arasındaki anlambilimsel ilişkilere dayalı olarak geliştirilen bir anlambilimsel benzerlik ölçütü kümeleme aşamasında kullanılmaktadır. Bulanık kümeleme amacıyla Mendes ve Sacks [69] tarafından oluşturulan amaç fonksiyonuna benzer bir yaklaşım kullanılan FSC algoritmasının klasik Bulanık C-Means ve K-Means algoritmalarına göre daha kararlı (stable) olduğu ve daha iyi kümeleme sonuçları ürettiği belirtilmektedir [76].

3.8. Harici Bilgi Tabanlarından Yararlanan Ontolojik ve Anlambilimsel Yaklaşımlar

Recupero tarafıdan yapılan çalışmada Vektör Uzayı Modeli'nin iki temel sorununun giderilmesi amacıyla WordNet [8] referans sisteminden yararlanılmıştır [3]. VUM'nin söz konusu sorunlarından ilki verinin çok boyutlu olması ve temsil eden vektörlerin seyrek olmasıdır. Diğeri ise döküman benzerliği hesaplanırken terimler arasındaki eşanlamlılık ve eşseslilik ilişkilerinden yararlanılmayışıdır. Bu çalışmada WordNet'in kullanımı için iki farklı strateji denenmiş ve karşılaştırması yapılmıştır. İlk olarak WordNet sözcüksel kategorileri (lexical categories) kullanılmıştır (WLC). WordNet'te toplam 41 sözcüksel kategori bulunmaktadır. Her bir dökümanın temsili için 41 boyutlu bir vektör kullanılmıştır. Vektörün her bir

elemanında ise ilgili dökümanda o boyuta karşılık gelen kategoriye kaç sözcüğün dahil olduğu yer almaktadır. Bir sözcük aynı anda birden çok kategoriye dahil olabilmektedir. Bu durumda da çeşitli anlam belirsizliği giderme (disambiguation) teknikleri geliştirilmiş ve kullanılmıştır. İkinci strateji ise WordNet Ontolojileri (WO) tekniğidir. Her bir sözcük için WordNet tarafından bu sözcükle ilişkili kavramları hiyerarşik olarak içeren listeler üretilir. Bu listelerden yararlanılarak ilk tekniğe benzer şekilde dökümanları temsil edecek vektörler oluşturulmaktadır. Çeşitli örnek veri setlerinden elde edilen vektörler FTC ve İkiye Bölmeli K-Means gibi algoritmalar ile kümelenerek karşılaştırmalı sonuçlar verilmiştir. Bu sonuçlara göre, WO tekniği ile WLC tekniğine göre çok daha iyi sonuçlar alınmakta ancak daha fazla hesaplama gücü gerekmektedir. Ayrıca, önişleme adımının kümeleme kalitesine etkisinin çok yüksek olduğu gösterilmiştir [3].

WordNet sözcüksel kategorilerinden yararlanan Gharib, Fouad ve Aref'e ait olan ve Bulanık C-Means algoritması temel alınan çalışmada yine dökümanlardaki terimler 41 WordNet sözcüksel kategorisinden birine eşlenmiş olup, elde edilen öznitelik matrisi K-Means, İkiye Bölmeli K-Means ve Bulanık C-Means algoritmalarıyla kümelenerek karşılaştırmalı sonuçlar analiz edilmektedir [77].

Geleneksel kümeleme yaklaşımlarının dökümanları sözcük torbası (bag-ofwords) olarak ele aldığını ve dökümanlar arasındaki kavramsal benzerliğin görmezden gelindiğini ileri süren Song ve Park, bu duruma çözüm olması amacıyla sözlük tabanlı ontoloji (thesaurus based ontology) kullanarak hibrit bir yaklaşım önermektedir [78]. Bu yaklaşımda dökümanlar arası benzerlik ölçütü olarak hem Li ve Bandar [79] tarafından geliştirilen anlambilimsel benzerlik ölçütü hem de VUM tabanlı kosinüs benzerlik ölçütü bir arada kullanılmaktadır. Söz konusu anlambilimsel benzerlik ölçütü, WordNet kavramları arasındaki hyponymy yani "ISA" bağlantılarını kullanmaktadır. Önerilen bu yeni benzerlik ölçütü, genetik algoritma tabanlı bir kümeleme algoritmasında kullanılarak ve kosinüs benzerlik ölçütü kullanımına göre daha başarılı sonuçlar elde edildiği gösterilmektedir.

Shehata tarafından geliştirilen modelde ilk olarak dökümanlardaki her bir cümlenin anlambilimsel yapısı analiz edilir ve cümledeki en önemli terimler belirlenir. Ardından, bu terimlerin WordNet'teki eşanlamlı karşılıkları ile karşılık

gelen hypernym'ler bulunur. Bulunan bu WordNet kavramları dökümanlar arası anlambilimsel benzerlik ölçütü hesaplanmasında kullanılmaktadır. Buradaki benzerlik ölçütü hem cümle seviyesinde hem de döküman seviyesinde terim ve kavramları dikkate alan kapsamlı bir ölçüttür. Bu benzerlik ölçütü kullanılarak dökümanlar arası benzerlikler hesaplanmakta ve oluşturulan benzerlik matrisi, Hiyerarşik Birleştirici Kümeleme (Hierarchical Agglomerative Clustering), Tek Geçiş Kümeleme (Single Pass Clustering) ve k-En Yakın Komşu (k-Nearest Neighbor) kümeleme algoritmaları tarafından kümelenerek karşılaştırmalı sonuçlar verilmektedir [80]. Shehata, Karray ve Kamel tarafından bu çalışma ilerletilerek yeni bir anlambilimsel benzerlik ölçütü geliştirilmiştir. Bu yeni çalışmada öncekinde yer alan cümle tabanlı kavramsal analiz ve döküman tabanlı kavramsal analize sözlük (corpus) tabanlı kavramsal analiz de eklenmekte, cümlenin anlamına katkısı olan bir terim hem cümle, hem döküman, hem de koleksiyon bazında analiz edilmekte, buna bağlı olarak yeni bir kavram tabanlı benzerlik ölçütü kullanılmaktadır [81].

Hu ve arkadaşları tarafından Wikipedia'nın harici bilgi ve başvuru kaynağı olarak kullanıldığı çalışmada, yaygın şekliyle döküman temsilinin harici bilgi kaynaklarıyla zenginleştirilmesinin doğurduğu iki temel sorun üzerinde durulmakta ve bunların çözümüne yönelik yaklaşımlar sergilenmektedir. İlk olarak, WordNet için bile ontolojinin kapsamı sınırlıdır. İkinci olarak, terimlerin kavramlarla değiştirilmesi ya da kavramların yeni öznitelikler olarak döküman temsiline eklenmesi bilgi kaybına ve ayrıca gürültü olarak nitelenebilecek fazlalıklara yol açmaktadır. Bu sorunların aşılması için dökümanların Wikipedia kavramlarına ve kategorilerine eşlendiği yaklaşımlar geliştirilmekte, bu eşlemelerin sonucunda elde edilen kavram ve kategori vektörleri de dökümanların sözcük vektörleriyle birlikte dökümanlar arası benzerlik hesaplamalarında belirli ağırlıklarla kullanılmaktadır. Bu vektörler şu kombinasyonlarda kullanılarak birleştirici ve bölünmeli kümeleme yöntemleriyle kümelenmektedir: sözcük, kavram, kategori, sözcük-kavram, sözcükkategori, sözcük-kavram-kategori. Deney sonuçlarına göre sözcük-kategori kombinasyonu en iyi sonucu vermekte, sözcük-kavram-kategori kombinasyonu da çoğunlukla en iyiye yakın sonuç vermektedir. Kavram bilgisinin katkısı kategori bilgisinin katkısı kadar fazla olmamaktadır [82].

Fodeh, Punch ve Tan tarafından yapılan çalışmada anlambilimsel ikili model (semantic binary model) ve ad frekans modeli (nouns frequency model) olarak adlandırılan iki modelin ayrı ayrı kümeleme sonuçlarının kombine edildiği bir kümeleme birliği çerçevesi (ensemble clustering framework) tasarlanmaktadır. Ad frekans modelinde, dökümanlardaki ad olan sözcüklerden oluşturulan yeni döküman vektörlerinin oluşturduğu matris klasik yöntemlerle kümelenir. Benzer şekilde, anlambilimsel ikili model için de dökümanlardaki ad olan sözcüklere karşılık gelen WordNet anlamlarından (sense) en uygun olanı seçilir ve her bir döküman vektörü için, bir ikili anlam vektörü oluşturulur. Bu yeni vektörlerin oluşturduğu matris de klasik yöntemlerle kümelenir ve her iki kümelemenin sonuçları kümeleme birliği yaklaşımına göre yeniden kümelenerek nihai kümeleme elde edilir [83].

Gabrilovich ve Markovitch tarafından doğal dilde oluşturulmuş metinlerin hassas biçimde anlambilimsel olarak yorumlanabilmesi amacıyla Açık Anlambilimsel Analiz (Explicit Semantic Analysis – ESA) olarak adlandırılan bir sistem geliştirilmiştir. Bu sistem bir metnin anlamını tüm Wikipedia kavramlarının ağırlıklandırılmış bir kombinasyonundan oluşan çok yüksek boyutlu kavram uzayı olarak temsil etmektedir. İhtiyaca yönelik olarak bu kavram uzayındaki tüm kavramlar kullanılabileceği gibi dökümanların sözcük torbası (bag-of-words) temsilini zenginleştirmek amacıyla seçilen en uygun kavramlar da kullanılabilmektedir. Bu çalışmada oluşturulan döküman vektörlerinin benzerlik ölçütü olarak kosinüs ölçütü kullanılmaktadır. ESA sisteminde döküman vektörlerinin oluşturulması yani öznitelik vektörlerinin üretilmesi için terimlerin karşılık gelen kavramlarla değiştirilmesi kesinlikle önerilmemekte, bunun yerine sözcük torbası şeklindeki vektörlerin kavramlarla zenginleştirilmesi ya da kavram vektörleriyle birleştirilmesi önerilmektedir. ESA sisteminde ikinci yöntem uygulanmaktadır. ESA sisteminin, doğal dil işlemenin (natural language processing) en önemli sorunlarından olan eşanlamlılık ve eşseslilik sorunları için oldukça başarılı bir çözüm ortaya koyduğu belirtilmektedir [84].

WordNet'ten harici bilgi kaynağı olarak yararlanan diğer bir çalışmada Zheng, Kang ve Kim tarafından dökümanlardaki isim tamlamaları (noun phrase) kullanılarak dökümanların anlambilimsel açıdan daha iyi temsiline çalışılmaktadır. İsim tamlamalarına ek olarak tek başına terimler de kullanılarak WordNet tarafından

sağlanan hypernymy, hyponymy, holonymy ve meronymy bilgilerinden yararlanılarak elde edilen döküman temsilleri K-Means ve İkiye Bölmeli K-Means algoritmalarıyla kümelenerek yaklaşımların etkinlikleri karşılaştırılmaktadır. Buna göre en iyiden en kötüye doğru hypernymy, hyponomy, meronymy ve holonymy olarak belirlenmiştir [85].

Jing, Ng ve Huang tarafından geliştirilen bilgi tabanlı (knowledge based) vektör uzayı modeli yaklaşımında döküman vektörlerindeki boyutlarda herhangi bir değişiklik yapılmayıp terim frekansları, terimler arasındaki anlambilimsel ilişkiye bağlı olarak yeniden ağırlıklandırılmaktadır. Terimler arasındaki ilişkinin belirlenmesinde kullanılan ilk yaklaşım WordNet gibi ontolojileri kullanmak şeklindedir. Diğer yaklaşım ise herhangi bir ontolojik bilgi kaynağının olmadığı durumlarda döküman koleksiyonunun bütününün kullanılması şeklindedir. Bu çalışmada, terimler arasındaki anlambilimsel ilişkinin değerlendirilebilmesi için WordNet ontoloji hiyerarşisindeki düğüm noktalarında bulunan kavramlar arasındaki mesafeden yararlanan bir terimler arası benzerlik ölçütü geliştirilmiş, bunun kullanıldığı ontolojik olarak ağırlıklandırılmış döküman-terim matrislerinin İkiye Bölmeli K-Means, K-Means ve hiyerarşik kümeleme algoritmalarıyla kümelenmesiyle elde edilen sonuçlara göre bu yeni modelin geleneksel terim tabanlı vektör uzayı modelinden daha iyi olduğu gösterilmiştir [86].

Duong ve arkadaşları tarafından yapılan çalışmada, dökümanlardaki sözcüklerin oluşturduğu döküman vektörlerine ek olarak dökümanlar içerisinde geçen adlandırılmış varlıklar (named entities) kullanılarak yeni döküman vektörleri oluşturulmakta ve böylelikle döküman içerisinde gizli kalmış ontolojik özniteliklerin kullanılmasıyla metin kümelemenin kalitesinin artırıldığı ileri sürülmektedir. Burada adlandırılmış varlıklar ifadesiyle genellikle kişiler, kurumlar, yerler gibi adlarıyla atıfta bulunulan varlıklar kastedilmektedir. Adlandırılmış varlıklar 3 temel öznitelik ile temsil edilmektedir: ad, tür ve belirteç (name, type, identifier). Örneğin, "UN team survey of public opinion in North Borneo and Sarawak on the question of joining the federation of Malaysia." cümlesindeki adlandırılmış varlıklar şunlardır: (UN/*/*), (North Borneo/Province/*), (Sarawak/Location/*) ve (Malaysia/Country/Country_T.MY). Burada, Malaysia adlı varlığın türünün Country olduğu ve bilgi tabanındaki belirtecinin ise Country_T.MY olduğu görülmektedir.

“*” karakteri ise tür ya da belirteç bilgisinin bilinmediğini ya da ontoloji bilgi tabanında yer almadığını göstermektedir. Çalışmada önerilen modele göre her bir döküman adlardan oluşan bir vektör, türlerden oluşan bir vektör, ad-tür çiftlerinden oluşan bir vektör ve belirteçlerden oluşan bir vektör olmak üzere toplam 4 vektör ile temsil edilmekte, benzerlik karşılaştırması için bu vektörler klasik TFIDF yöntemindeki gibi kullanılmaktadırlar. Bu 4 vektöre ek olarak sözcük ağırlıklarından oluşan bir vektör de döküman temsili için kullanılmaya devam etmektedir. Benzerlik ölçütü olarak toplam bu 5 vektörün karşılıklı olarak kosinüs benzerliklerinin değişen oranlarda ağırlıklı toplamı ile nihai bir benzerlik değeri elde edilmektedir. Bu teknik Reuters-21578 veri seti üzerinde K-Means algoritması kullanılarak denenmiştir. Ontoloji bilgi tabanı olarak Reuters dökümanlarının kendi içinde düzenli olarak yer alan PEOPLE, PLACES, ORGS gibi etiketli kısımlar kullanılmıştır. Sonuç olarak bu tekniğin kümeleme kalitesini büyük ölçüde artırdığı görülmektedir [87].

3.9. İşbirlikçi – Topluluk Tabanlı Kümeleme Yaklaşımları

Tek ve iyi ayarlanmış bir modelin sonuçlarına göre daha iyi sonuçlar elde etmek üzere birden çok sınıflandırma yönteminin sonucunun birleştirilmesi (kombine edilmesi) fikrini kümeleme alanında kullanmak üzere Strehl ve Ghosh tarafından Kümeleme Birliği (Cluster Ensemble) olarak adlandırılan bir bilginin yeniden kullanımı (knowledge reuse) çatısı tasarlanmış, bu çatı altında kümeleme birliği problemi bir optimizasyon problemi olarak tanımlanarak bu problemin çözümü için üç kombine etme yöntemi öne sürülmüştür. Buradaki temel fikir orijinal veriye ve özniteliklere erişmeye gerek kalmadan, daha önce çeşitli şekillerde elde edilmiş kümeleme sonuçlarının yeniden kullanılmasıdır. Bu yaklaşımın bir avantajı da aynı verinin farklı öznitelikleri kullanılarak elde edilmiş kümeleme sonuçlarının da değerlendirilebilmesidir. Ayrıca bu yaklaşım dağıtık kümelemeye (distributed clustering) de bir altyapı hazırlamaktadır [88].

Greene ve Cunningham tarafından hazırlanan raporda, standart kümeleme algoritmalarının kesinliğini ve kararlılığını artırmada etkili olduğu görülen kümeleme birliği tekniklerinde, verinin çoklu kümelemelerinin oluşturulması ve bunların birleştirilmesi sırasında kaçınılmaz olarak görülen yüksek işlemsel maliyetlerin, bu

yöntemleri döküman koleksiyonları gibi yüksek boyutlu ve geniş veri setlerinde uygulamaya engel oluşturduğu öne sürülmektedir. Bu duruma çözüm olarak birlik üyelerinin (ensemble members) oluşturulması için gereken zamanı azaltmak amacıyla orijinal veri için yeteri kadar iyi bir vekil (proxy) olabilecek küçük bir çekirdek matrisi (kernel matrix) oluşturmaya yönelik bir prototip indirgeme (prototype reduction) yöntemi kullanılmaktadır. Bu yöntemin önceki yöntemlere göre nihai kümeleme kalitesinden ödün vermeden işlem süresini azaltarak daha ölçeklenebilir bir çözüm olduğu belirtilmektedir [89].

Kümeleme Birliği alanındaki çalışmaların genellikle küme sayısının ya da başlangıç koşullarının kullanıcılar tarafından verildiği parametrik yöntemler üzerine olduğunu, parametresiz (non-parametric) yöntemlerin ihmal edildiğini öne süren Gonzales ve Turmo tarafından yapılan çalışmada parametresiz kümeleme yöntemlerinin kümeleme birliği içerisinde kullanımı ve bunun yanı sıra farklı kümeleme yöntemlerinin parametresiz kümeleme birliği bağlamındaki performansları araştırılmış ve uygulanan yöntemlerin diğer çeşitli kümeleme yöntemlerine göre daha iyi sonuçlar verdiği gösterilmiştir [90].

Bekkerman, Scholz ve Viswanathan tarafından gerçekleştirilen çalışmada kümeleme birliğine dahil olan algoritmaların tutarsızlıklarının ve uyuşmazlıklarının etkisini azalmak ve böylece kümelemenin kararlılığını artırmak amacıyla m adet kümelemeyi girdi olarak alan ve birbiriyle yüksek derecede uzlaşma (agreement) halinde olan ve yüksek kalitede sonuca sahip m adet kümeleme sonucu üreten bir yöntem geliştirilmiştir. Kümeleme Uzlaşma İşlemi (Clustering Agreement Process – CAP) olarak adlandırılan bu yöntem kümeleme kalitesini korumak için kümeleme algoritmasındaki optimizasyon prosedürünün aynısını uygulamaktadır. Bu yöntemde özellikle rastgele başlangıç koşulları nedeniyle her çalıştırıldığında farklı kümeleme sonucu üreten yöntemlerin kümeleme birliğinde kullanılması sırasında ortaya çıkan kararsızlık sorununa odaklanılmaktadır [91]. CAP sisteminin alt yapısında Bekkerman tarafından geliştirilen graf tabanlı bir sonuç çıkarma (inference) yöntemi olan Combinatorial Markov Random Field (Comraf) yer almaktadır [61].

Kümeleme Birliği yaklaşımlarından farklı olarak Kashef ve Kamel tarafından geliştirilen İşbirlikçi Kümeleme (Cooperative Clustering – CC) yaklaşımında aynı

verinin aynı anda farklı kümeleme teknikleriyle kümelenmesi sırasında bu teknikler arasındaki işbirliğine dayalı olarak kümeler içindeki nesnelerin homojenliğinin artırılması hedeflenmektedir. CC yöntemi farklı özelliklere sahip veriler üzerinde de uygulanabilmektedir. Bunun sağlanabilmesi için nesneler arası karşılıklı benzerlik değerlerini temsil eden bir histogram ve bir işbirlikçi olasılık grafı (cooperative contingency graph) şeklinde veri yapıları tutulmakta, bu veri yapıları farklı kümeleme tekniklerindeki eşleşen alt-kümelerin bulunmasında ve bir birleştirme aşaması ile nihai kümelerin oluşturulmasında kullanılmaktadır. İşbirlikçi Kümeleme yaklaşımının yüksek boyutlu döküman verilerinde tek başına çalışan kümeleme algoritmalarına göre çok daha iyi sonuçlar ürettiği gösterilmektedir [92].

3.10. Akan Metin Verilerinin Kümelenmesine Yönelik Yaklaşımlar

Veri madenciliği ve spesifik olarak kümeleme yaklaşımları genellikle durağan (static) haldeki veriler üzerinde uygulanmak üzere geliştirilmektedir. Ancak günümüzde analiz edilecek veriler sürekli ve kesintisiz bir şekilde üretilmekte, eski veriler önemini çabucak yitirmekte, yeni verilerin bir an önce analiz edilerek işe yarar bilgiye dönüştürülmesi gerekmektedir. Haber kaynaklarından yeni konu tespiti, sürekli ağ trafiğinden saldırı tespiti ve video görüntülerinden nesne tanımlama, Akan Veri Madenciliği (Stream Mining) alanındaki uygulamalara örnek olarak verilebilir [93]. Sürekli olarak koleksiyona eklenen yeni dökümanların ya da bloklar halinde gelen döküman topluluklarının yani akan dökümanların kümelenmesiyle ilgili literatürde yer alan çalışmalar bu bölümde verilecektir.

Zhong tarafından akan dökümanların kümelenmesi amacıyla, daha önceki bir çalışmada geliştirilen Online Spherical K-Means (OSKM) [25] algoritması ile çok büyük veri akışlarının kümelenebilmesi için ölçeklenebilir kümeleme tekniklerinin birleştirildiği bir çalışma gerçekleştirilmiştir [93]. Bu çalışmada, yeni gelen dökümanlara bellekte yer açılması amacıyla eski dökümanların bellekte tutulmadığı, ancak eski dökümanlara ait kısıtlı ve yeterli miktarda istatistiklerin saklandığı, böylece bu istatistiklerin yeni gelen dökümanların daha iyi kümelenmesinde kullanıldığı bir teknik ortaya konulmaktadır. Bu teknikte, insanların yeni bilgilere ve koşullara uyum sağlamak amacıyla eskiye ait bilgileri unutması gerektiği

düşüncesinden yola çıkılarak eski dökümanlara ait istatistiklerin etkisini ve katkısını üstel olarak azaltan bir azalma katsayısı (decay factor) kullanılmaktadır.

Aggarwal ve Yu tarafından, akan döküman verilerinin kümelenmesi amacıyla, düzenli aralıklarla eski verilere ilişkin çeşitli özet istatistiklerin saklandığı bir yapı önerilmektedir. Bu yapıda, gelen her yeni dökümana bir zayıflama fonksiyonu (fading function) tarafından zamana bağlı olarak bir ağırlık verilir. Yani her bir döküman için diğer bir deyişle yarı-ömür (half-life) verilmiş olur. Bu yarı-ömür, dökümanların sağladığı tarihçe bilgisinin zaman içerisindeki önemini belirlemektedir. Her gelen yeni dökümanın mevcut kümelere benzerliği hesaplanır ve eğer en yüksek benzerlik değeri bir eşik değerini aşıyor ise döküman o kümeye katılır. Eğer eşik değeri aşılmıyorsa, uzun süredir etkin olmayan bir küme varsa döküman o kümeye katılır, aksi halde en benzer olduğu kümeye katılır. Döküman uygun bir kümeye katılır katılmaz o kümeyle ilgili istatistikler, barındırdığı dökümanların yarı-ömürleri yani azalma katsayıları dikkate alınarak güncellenir [94].

Sahoo ve arkadaşları tarafından gerçekleştirilen çalışmada, döküman kümelemede pek yaygın olarak kullanılmayan bir artımlı hiyerarşik kümeleme algoritması (incremental hierarchical clustering algorithm) olan COBWEB ve bunun bir çeşitlemesi olan CLASSIT algoritması, akan dökümanların kümelenmesi problemi için ele alınmakta ve geliştirilen çözümler sunulmaktadır. CLASSIT algoritması kümelenecek verinin normal dağılıma uygun olması prensibine göre çalışmakta olup, metin dökümanları için daha uygun olduğu belirtilen Katz Dağılımı (Katz's Distribution) kullanacak şekilde uyarlanarak çeşitli veri setleri üzerinde etkinliği gösterilmektedir. Bu algoritma ile yeni gelen dökümanların küme hiyerarşisi içerisinde en uygun yere yerleştirilmesinin, artımlı kümeleme algoritmasının sağladığı en önemli yararlarından biri olduğuna değinilmektedir [5].

He ve arkadaşları tarafından klasik Vektör Uzayı Modeli temsil şeklinin durağan yapısının zaman içerisinde bir anlambilimsel bağlamdan diğerine geçişi anlamlı bir şekilde modelleyemediği, bir özniteliğin/terimin zaman içinde değişik anlarda değişik konulara karşılık gelebileceği öne sürülmekte, bu duruma çözüm olarak "Bursty" Feature Representation olarak adlandırılan yeni bir akan döküman temsil modeli ortaya konulmaktadır. Bu modelde "burst" (ya da patlama, sıçrama) ile, çok

kısa zaman dilimi içerisinde hakkında çok sayıda metin içerik üretilen önemli olay ya da durumlar ifade edilmektedir. "Bursty" özniteliklerin ağırlığı dökümanların yayınlanış zamanına son derece bağımlıdır. Bu model, akan dökümanların kümelenmesi bakımından terimlerin var/yok olarak ağırlıklandırıldığı İkili Vektör Uzayı Modeli (Binary Vector Space Model) ile deneysel olarak karşılaştırılmakta ve daha iyi sonuç verdiği gösterilmektedir [95].

Liu ve arkadaşları tarafından, TFIDF gibi ağırlıklandırma modellerine göre döküman kümeleme açısından daha uygun görülen anlambilimsel yumuşatma (semantic smoothing) modellerinin akan dökümanlar için yeterince iyi olmadığı öne sürülmekte ve geliştirilmiş bir anlambilimsel yumuşatma modeli ortaya konularak bu modele bağlı iki kümeleme algoritması sunulmaktadır [96]. Aggarwal ve Yu tarafından yapılan çalışmadakine [94] benzer şekilde bu algoritmalarda da küme profili (cluster profile) olarak adlandırılan ve kümelerle ilgili istatistikleri barındıran yapılar bulunmakta, dökümanlara yine bir zayıflama fonksiyonu vasıtasıyla yarıömür verilmektedir [96].

Gil-García ve Pons-Porrata tarafından geliştirilen Dynamic Hierarchical Compact (DHC) ve Dynamic Hierarchical Star (DHS) adlı kümeleme algoritmalarıyla dinamik veri setlerinin döküman kümelemenin temel gereksinimlerini karşılayacak şekilde hiyerarşik olarak kümelenmesi hedeflenmektedir. DHC algoritması ayrık küme hiyerarşileri oluştururken DHS algoritması örtüşen hiyerarşilere izin vermektedir. Bu algoritmalar dökümanların geliş sırasına bağımlı olmayıp, dökümanların dinamik olarak hiyearşiye eklenebilmesine olanak sağladıkları gibi hiyearşiden çıkarılmalarına da olanak sağlamaktadır [97].

3.11. Diğer Döküman Kümeleme Yaklaşımları

Yüksek boyutlu verilerin analiz edilmesinde yaşanan çok boyutluluk sorununa (curse of dimensionality) [6] çözüm bulmak amacıyla Strehl ve Ghosh tarafından verilerin bir benzerlik uzayına (similarity space) dönüştürüldüğü ilişki tabanlı (relationship-based) bir yaklaşım ortaya konulmuştur. Bu yaklaşımda dökümanlar arası benzerlik değerlerinden oluşturulan benzerlik matrisi, geliştirilen verimli ve

ölçeklenebilir graf bölmeleme tabanlı bir kümeleme algoritmasıyla kümelenmekte, böylelikle orijinal verinin bulunduğu çok boyutlu uzay yerine daha düşük boyutlu benzerlik uzayında çalışılmaktadır. Kümeleme sonucu daha sonra kümelerin bantlar halinde belirdiği iki boyutlu bir düzlem üzerinde verinin görselleştirmesinde kullanılmaktadır [98].

Kümeleme analizi bir gözetimsiz öğrenme (unsupervised learning) problemi olmasına karşın, Zhao ve Karypis tarafından, bazı bilgi yönetim sistemi uygulamalarında döküman koleksiyonu için eksiksiz bir taksonomi (tasnif sistemi) kullanılabilir durumda olmasa bile ilgili alanın uzmanları tarafından ya da kullanıcıları tarafından temel konu başlıklarının sağlanabileceği, bu konu başlıkları kullanılarak kullanıcıların beklentilerini daha iyi karşılayabilecek kümeleme çözümlerinin üretilebileceği öne sürülmüş, bu amaçla da konu güdümlü kümeleme (topic-driven clustering) yaklaşımı ortaya konulmuştur. İlk bakışta bir sınıflandırma (classification) problemi gibi görünse de bu yaklaşımda, sınıflandırma sisteminin eğitimi için kullanılmak üzere etiketlenmiş çok sayıda verinin ön bilgi olarak sisteme sağlanamayacağı, sadece konuların birkaç sözcükle tanımlanabileceği uygulama alanları hedeflenmektedir ve bu nedenle de sınıflandırmadan ziyade bir kümeleme problemidir. İlk olarak bu çalışma ile önerilen konu güdümlü kümeleme için yeni bir algoritma geliştirilmiş, konu başlıkları ile dökümanlar arasında ve dökümanların da kendi arasında benzerliklerinin ölçülmesi için çeşitli benzerlik fonksiyonları ile kümeleme sürecini yönlendirecek olan çeşitli optimizasyon ölçütü fonksiyonları önerilmiş, deneysel olarak karşılaştırmaları ve ayrıntılı analizleri yapılmıştır [99].

Luo, Li ve Chung tarafından yapılan çalışmada K-Means ailesinden kümeleme algoritmaları üzerine komşuluk matrisinin uygulanmasına yönelik algoritmalar önerilmektedir. Başlangıç küme ağırlık merkezlerinin belirlenmesi için yeni bir teknik önerilmiş ve döküman benzerliği için komşuluk matrisini de kullanacak şekilde kosinüs benzerliğinin geliştirilmiş bir hali kullanılmıştır. Burada komşu (neighbor) ve bağlantı (link) kavramları önerilmiştir. Eğer iki döküman birbirine yeterince benzer ise bu ikisi birbirinin komşusu olarak kabul edilir. Koleksiyondaki her bir dökümanın belli bir eşik değerinin üzerinde benzerliğe sahip olduğu bir dizi komşusu vardır. Bağlantı (link) ise iki nokta arasındaki ortak komşuların sayısını ifade etmektedir. Döküman koleksiyonundaki tüm dökümanlar için komşular ve

bağlantılarla ilgili bilgiler komşuluk matrisi (neighbor matrix) ile temsil edilmektedir. Kosinüs ve bağlantı fonksiyonlarının birlikte kullanıldığı yeni bir kümeleme kriteri fonksiyonu önerilmektedir. Bu kriter fonksiyonunda döküman ve küme vektörleri arasındaki bağlantı değerinin hesaplanabilmesi için komşu matrisine küme sayısı k kadar daha kolon eklenerek dökümanlar ve küme vektörleri arasındaki komşuluk durumu da dikkate alınmaktadır. Bu yaklaşımın standart K-Means ile yapılan kümelemeye göre daha iyi kümeleme sonucu verdiği deneylerle gösterilmektedir [100].

Aliguliyev tarafından döküman koleksiyonlarının bölünmeli kümelenmesi açısından çeşitli yoğunluk tabanlı kümeleme ölçütü fonksiyonlarının performanslarının karşılaştırmalı olarak ele alındığı çalışmada ölçüt fonksiyonlarının değerlendirilmesi için üçü dahili, yedisi harici olmak üzere toplam on farklı geçerlilik ölçütü (validity index) kullanılmıştır. Kümeleme algoritmalarının genellikle verinin şeklini önemsemeyişinin bu algoritmaların bir eksikliği olduğu öne sürülmekte ve buna alternatif olarak kümelenecek veri noktalarının ağırlıklandırılması yaklaşımı denenmektedir. Bu amaçla her bir döküman, koleksiyonun merkezine olan uzaklığına göre ağırlıklandırılmaktadır. Bu ağırlık, koleksiyon merkezi etrafındaki noktaların yoğunluk derecesi (concentration degree) olarak tanımlanmaktadır. Bu yaklaşımın kümeleme kalitesini artırdığı sonucu deneylerle desteklenmektedir. Söz konusu ağırlık değerini de dikkate alan ölçüt fonksiyonlarının optimizasyonunun denenebilmesi için bir çeşit Evrimsel Algoritma (Evolutionary Algorithm) olan Differential Evolution (DE) algoritması geliştirilmiştir. DE algoritmasının klasik EA'lardan farklı yanı, mutasyon operatörünün populasyon bireyleri üzerinde rastgele mutasyona neden olması yerine populasyondan rastgele seçilen iki birey arasındaki farkın mutasyon değeri olarak kullanılıyor olmasıdır. Algoritmanın adı da bu nedenle Differential Evolution'dır [101]. Aliguliyev tarafından yapılan diğer bir çalışmada dökümanlar yine benzer şekilde ağırlıklandırılmakta, kosinüs benzerlik fonksiyonuna alternatif olarak geliştirilen, terimlerin tüm koleksiyon çapında ortalama ağırlığı kullanılarak ayarlama yapılan Ayarlanmış Kosinüs (Adjusted Cosine) benzerlik ölçütünün katkısı deneysel olarak gösterilmektedir [2].

3.12. Benzerlik Ölçüm Yöntemleri İle İlgili Çalışmalar

Döküman kümeleme alanında yaygın olarak kullanılan benzerlik ölçütlerinin etkinliğini sistematik olarak karşılaştıran ilk çalışma Strehl, Ghosh ve Mooney tarafından yapılmış olup, çeşitli kümeleme algoritmaları üzerinde Öklid uzaklığı, Kosinüs benzerliği, Genişletilmiş Jaccard benzerliği ve Pearson korelasyonu uygulanarak sonuçları karşılaştırılmıştır. Sonuç olarak Kosinüs ve Genişletilmiş Jaccard benzerliklerinin insanların gruplama davranışına en benzer sonuçları ürettiği gösterilmiştir [16].

Wang ve Taylor tarafından WordNet ontolojileri [8] kullanılarak Kavram Ağaçlarının (Concept Forest – CF) oluşturulduğu bir algoritma ve bu CF'lerin dökümanlar arası anlambilimsel benzerliği ölçümlemek için kullanıldığı bir benzerlik ölçütü geliştirilmiştir [102]. Döküman içerisindeki terimlerin WordNet hypernym ilişkisi aracılığıyla anlamsal ilişkisi belirlendikten sonra bu ilişkilerin "ISA" mantığıyla kullanıldığı bir graf yapısı yani CF oluşturulmaktadır. İki döküman arasındaki benzerlik ise bu dökümanlar için elde edilen CF'lerdeki terim kümelerinin karşılaştırılması yoluyla hesaplanmaktadır. Geliştirilen bu yöntemin küçük dökümanlar için klasik vektör uzayı modeli kullanan benzerlik yöntemlerinden daha iyi sonuç verdiği gösterilmiştir.

Bisson ve Hussain tarafından, döküman-terim matrisinde hem dökümanları hem de terimleri bir arada kümelemeye dayalı Ortaklaşa Kümeleme (Co-Clustering) uygulamaları için χ-Sim olarak adlandırılan bir benzerlik ölçütü öne sürülmektedir. Bu benzerlik ölçütü, kosinüs gibi klasik yöntemlerle dökümanlar arası benzerlik ve terimler arası benzerlik hesaplanarak oluşturulan benzerlik matrisleri üzerinde tekrarlamalı bir biçimde işlem yaparak ortaklaşa kümeleme algoritmalarında kullanılmak üzere bir benzerlik sonucu üretmektedir [103]. Bu benzerlik yöntemi üzerinde gürültüye karşı duyarlılık sorunlarının giderilmesi gibi çeşitli iyileştirmeler yapılmış ve bu yaklaşım için genelleştirilmiş bir çerçeve sunulmuştur [104].

Türkçe dökümanların kümelenmesi amacıyla çeşitli benzerlik ölçütlerinin karşılaştırıldığı oldukça kapsamlı bir çalışma Madylova ve Öğüdücü tarafından yapılmıştır [15]. Bu çalışmada kosinüs benzerliği gibi terim tabanlı benzerlik

ölçütlerinin yanı sıra anlambilimsel benzerlik ölçütü yaklaşımları da değerlendirilmektedir. Dökümanlar arası anlambilimsel ilişkilerin belirlenebilmesi için WordNet ontolojsine benzer olarak, Türkçe'nin de içinde bulunduğu çeşitli Balkan dillerine ait sözcüksel ontolojilerin yer aldığı BalkaNet projesinden yararlanılmıştır [105]. Wu-Palmer anlambilimsel benzerlik ölçütüne [106] dayalı olarak geliştirilen çeşitli anlambilimsel benzerlik ölçütleri üzerinde deneysel çalışmalar yapılmış ve sonuç olarak bu benzerlik ölçütlerinden hiç biri oldukça basit ve işlemsel karmaşıklığı oldukça düşük olan kosinüs benzerliği kadar başarılı sonuç vermemiştir. Ayrıca, bu çalışmada kosinüs benzerliği kullanılarak elde edilen kümeleme sonuçlarının bir uzman tarafından elle yapılan gruplandırmayla daha fazla örtüştüğü vurgulanmaktadır [15].

D'hondt ve arkadaşları tarafından, Kosinüs Ayrıklığı (Cosine Dissimilarity) ölçütü ile özel bir öznitelik seçim yönteminin birleştirilmesiyle elde edilen yeni bir ayrıklık ölçütü öne sürülmüştür. Ayrıklık hesaplamasına katılacak özniteliklerin seçimi için iki farklı yaklaşım sunulmaktadır. İlk yaklaşımda çiftler halinde dökümanlar arası ayrıklık hesaplanırken tüm çiftlerde aynı öznitelikler seçilerek kullanılmakta, ikinci yaklaşımda ise her döküman çiftinde hangi özniteliklerin kullanılacağı devingen olarak belirlenmektedir. Bu çalışma kapsamında yapılan deneylere göre kümeleme sonucu bakımından bir iyileştirme sağlanırken aynı zamanda işlem süresi bakımından da bir kazanç elde edildiği belirtilmektedir [107].

Kosinüs gibi benzerlik ölçütlerinin dökümanların yapısal özelliklerinden faydalanmaması şeklindeki eksikliğini ortadan kaldırmak ve buna bağlı olarak daha iyi kümeleme sonucu elde etmek üzere Guan ve arkadaşları tarafından yeni bir benzerlik ölçütü önerilmiştir. Bu ölçüte göre döküman yapısına ve sözcüklerin dökümanda bulunduğu yerlerin önemine bağlı olarak (dökümanın başlığı ve içerik metni gibi) Cofeature Set, Uniliteral Feature Set ve Significant Cofeature Set olmak üzere üç farklı öznitelik kümesi tanımlanmakta, bu kümelerin birlikte değerlendirildiği bir formülasyon önerilmektedir [108].

Nguyen, Chen ve Chan tarafından kosinüs benzerliğinde tek bir referans noktası (yani 0 noktası) kullanarak iki döküman vektörü arasındaki açının kosinüsünün hesaplanması yerine birden çok referans noktasının bu ölçümlere katılmasına yönelik

olarak Multi-ViewPoint Similarity (MVS) olarak adlandırılan yeni bir benzerlik ölçütü önerilmektedir. Burada birden çok referans noktası olarak aralarındaki benzerliğin ölçümlendiği dökümanların birlikte yer aldıkları kümenin dışında kalan nesnelerden yararlanılmaktadır. MVS ölçütü ile birlikte bu ölçüte dayalı kümeleme ölçütü fonksiyonları (clustering criterion functions) da geliştirilerek deneysel olarak daha iyi döküman kümeleme sonucu ortaya konulmaktadır [109].

3.13. Kümelerin Adlandırılmasına (Etiketlenmesine) Yönelik Çalışmalar

Stefanowski ve Weiss tarafından, döküman kümeleme sonucunda elde edilen kümeler etiketlenirken bu etiketlerin anlaşılabilir olması, kısa ve öz olması, kümenin neden bu şekilde etiketlendiğinin açıkça anlaşılabilecek şekilde şeffaflık sağlanması şeklinde üç temel gereksinim ortaya konulmuş, bu gereksinimleri sağlamak üzere Önce-Açıklama-Gelir (Description Comes First – DCF) olarak adlandırılan bir kümeleme ve etiketleme yaklaşımı geliştirilmiştir. Bu yaklaşımda, geleneksel yaklaşımların önce kümeleme sonra etiketleme sıralaması tersine çevrilmekte, öncelikle dökümanlarda küme etiketi olmaya aday anlamlı sözcük grupları belirlenmekte, kümeleme işlemi ise ayrı bir adım olarak daha sonra gerçekleştirilmektedir. Son adımda, elde edilen aday küme etiketleriyle kümeler eşleştirilmektedir. Bu çalışma daha sonra büyük döküman koleksiyonlarında kullanılabilecek şekilde iyileştirilmiş, Descriptive K-Means adıyla oluşturulan yeni bir algoritmada uygulanmıştır [110, 111].

Carmel, Roitman ve Zwerdling tarafından önerilen etiketleme sisteminde Wikipedia ansiklopedisinden yararlanılmaya çalışılmıştır. Bu sistemde, oluşan kümelerin içeriğine en yakın Wikipedia konuları belirlenerek bunların başlıkları ve kategorileri küme etiketi adayı olarak çıkartılmaktadır. Çeşitli iyileştirme adımlarından sonra bu adaylar arasından gerçek küme etiketleri seçilmektedir. Deneylere göre Wikipedia tarafından kapsanan konulara yönelik döküman koleksiyonları kümelendiğinde küme etiketlerinin başarılı olarak oluşturulabildiği, ancak Wikipedia dışındaki konuları içeren döküman kümelerinde yeteri kadar iyi

sonuçlar elde edilemediği, böyle durumlarda başka harici kaynaklardan yararlanılması şeklinde zeki yaklaşımların kullanılabileceği belirtilmektedir [112].

Muhr, Kern ve Granitzer tarafından hiyerarşik kümeleme sonucu oluşan hiyerarşik yapının kümelerin daha iyi etiketlenmesini sağlayabileceği fikri üzerine hiyerarşide yer alan ebeveyn-çocuk ilişkileri, kardeş küme ilişkileri ve hiyearşinin derinliği gibi özelliklerin dikkate alındığı çeşitli etiketleme yaklaşımları geliştirilerek karşılaştırması yapılmıştır [113].

4. RENK UZAYI

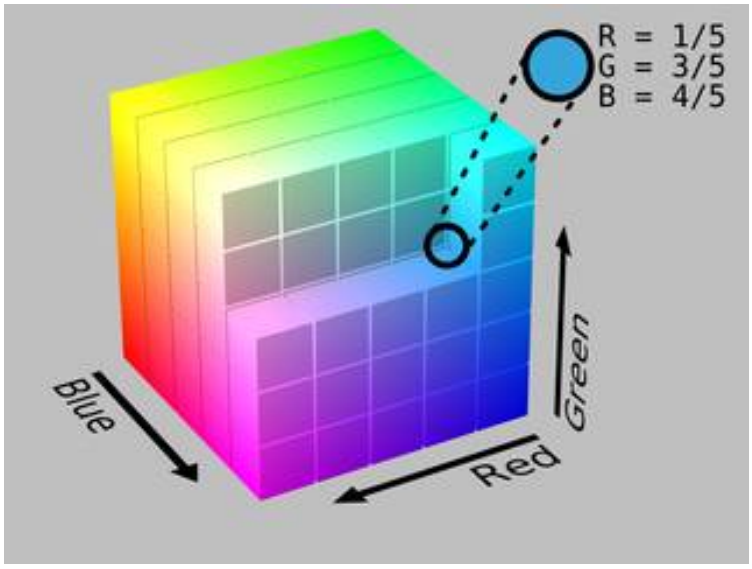
Renk uzayları renkleri tanımlamak için kullanılan matematiksel modellerdir, bütün renkleri temsil edecek şekilde 3D olarak tasarlanırlar. Renkmetri biliminin temelini oluşturan Grassmann'ın birinci kanununa göre; herhangi bir rengi tanımlamak için birbirinden bağımsız değişkenlere ihtiyaç vardır. Renklerin renk uzayındaki yerleri bu değişkenlere göre belirlenmektedir. Her renk uzayının kendine özgü renk oluşturmak için standartları vardır. Renk uzayları oluşturulurken başka bir renk uzayına doğrusal olan ya da doğrusal olmayan yöntemlerle dönüşüm yapılabilmelidir [122].

Renk uzayları genel olarak cihaz bağımlı ve cihaz bağımsız renk uzayları olarak iki gruba ayrılırlar. Cihaz bağımlı renk uzaylarında renkler cihazın teknik özelliklerine bağlı olarak üretilir. Cihaz bağımsız renk uzayları ise CIE (Uluslararası Aydınlatma Komisyonu) tarafından geliştirilen ve bütün renkler için renk ölçümünü sağlayan yani renkmetride kullanılan uzaylardır. CIE tarafından geliştirilen renk uzaylarında renk ile ilgili olarak ortaya konulan ve önerilen tanımlamalar (standart aydınlatıcı gibi ve standart gözlemci) kullanılmıştır [122].

4.1. RGB RENK UZAYI

RBG renk uzayı, bir birim küpün içinde toplamalı renk karışımı yöntemiyle renkleri tanımlayacak şekilde dizayn edilmiştir. (bilgisayar monitörleri, katodik televizyon tüpleri ve tarayıcılar gibi cihazlarda kullanılır) [122].

Bilgisayarda herhangi bir rengi görüntülemek için belirli yoğunluklarda karıştırılır. RGB renk uzayı koordinat eksenleri kırmızı, yeşil ve mavi olan 3D bir uzay olarak düşünülebilir. Oluşturulmak istenilen renkler bu ana rengin koordinatları cinsinden ifade edilebilir [122].



Şekil IV.1 RGB renk uzayı [117]

4.2. HSV RENK UZAYI

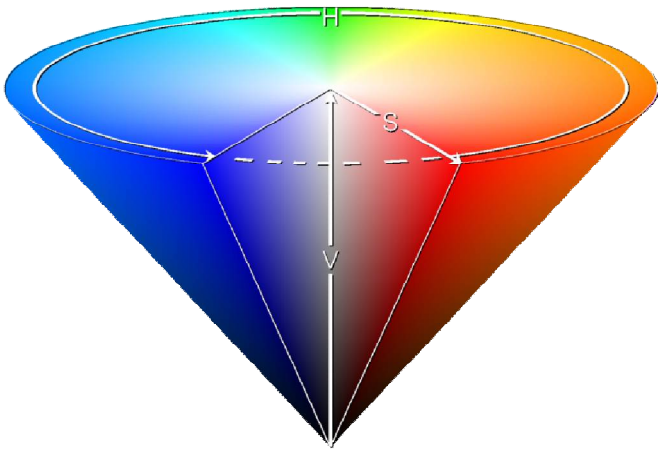
HSV (Hue, Saturation, Value) veya HSB (Hue, Saturation, Brightness) renk uzayı, renkleri sırasıyla renk özü, doygunluk ve parlaklık olarak tanımlar. **Alvy Ray Smith** tarafından 1978 yılında tanımlandı. Tanımlanmasının amacı, RGB uzayına nazaran insan görü düzlemine daha yakın bir yapı oluşturmaktı. HSV, doğrusal olmayan bir dönüşüm ile RGB renk uzayından elde edilir [118].

- Renk özü, rengin baskın dalga uzunluğunu belirler, örneğin sarı, mavi, yeşil, vb. Açısal değerdir 0° ile 360° arasındadır, uygulamaların bazılarında ise 0 ile 100 arası olağanlaştırılır [118].
- Doygunluk, rengin "canlılığını" belirler. Yüksek doygunluk ile canlı renkler elde edilirken, düşük doygunluk ise rengin gri tonlara yaklaşmasına neden olur. 0 ile 100 arasında değişmektedir [118].

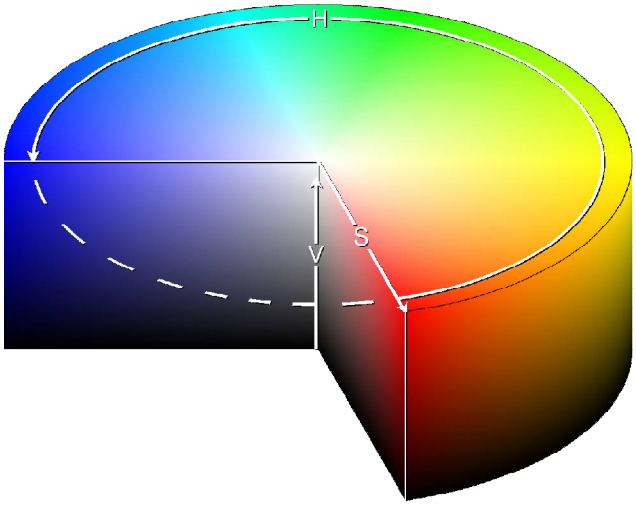
- Parlaklık ise rengin içindeki beyaz oranını belirler. 0 ile 100 arasından değişmektedir [118].

HSV'nin Gösterimi

HSV uzayı, ilk tanımlandığında konik biçimdeydi. Gerçek zamanlı geçerli koordinat denetimi için zamanın bilgisayarları yeterli olmadığı için silindir biçimine dönüştürüldü. Konik biçimde, aydınlık düzeyi azaldıkça koninin genişliği azalır, dolayısıyla insan örme yetisine uygun olarak, düşük aydınlıkta algılanabilen farklı doygunluk düzeyleri de azalmış olur. Fakat silindir biçimi ile sıfır aydınlık düzeyinde dahi yüksek doygunluk düzeyleri tanımlanabilir, böylece geçersiz renkler elde edilir. Bundan dolayı görüntü işleme uygulamalarında konik biçim tercih edilirken, renk seçimi görevlerinde silindir biçimi kullanılma eğilimi gösterilir [118].



Şekil IV.2 HSV renk uzayı konik [118].



Şekil IV.3 HSV renk uzayı silindir [118].

4.2.1. HSV Renk Uzayından RGB Renk Uzayına Dönüşüm

Aşağıda bulunan dönüşüm denklemleri RGB ile HSV'nin silindir biçimi arasında dönüşüm gerçekleştirir [118]:

$$H \in \{0, 360\}, S, V, R, G, B \in \{0, 1\}$$

RGB'den HSV'ye:

$$MAX = max\{R, G, B\}, \quad MIN = min\{R, G, B\}$$

$$H = \begin{cases} \text{tanimsiz}, & \text{eger } MAX = MIN \\ 60\frac{G-B}{MAX-MIN} - 0, & \text{eger } MAX = R \\ & \text{ve } G \geq B \\ 60\frac{G-B}{MAX-MIN} - 360, & \text{eger } MAX = R \\ & \text{ve } G < B \\ 60\frac{B-R}{MAX\ \ MIN} - 120, & \text{eger } MAX = G \\ 60\frac{R-G}{MAX-MIN} - 240, & \text{eger } MAX = B \end{cases}$$

$$S = \begin{cases} 0, & \text{eger } MAX = 0 \\ 1 - \frac{MIN}{MAX}, & \text{degilse} \end{cases}$$

$$V = MAX$$

HSV'den RGB'ye:

$$H_i = \left\lfloor \frac{H}{60} \right\rfloor \mod 6$$

$$f = \frac{H}{60} - H_i$$

$$p = V(1-S)$$

$$q = V(1-fS)$$

$$t = V(1-(1-f)S)$$

$$\text{eger } H_i = 0 \Rightarrow R = V, G = t, B = p$$

$$\text{eger } H_i = 1 \Rightarrow R = q, G = V, B = p$$

$$\text{eger } H_i = 2 \Rightarrow R = p, G = V, B = t$$

$$\text{eger } H_i = 3 \Rightarrow R = p, G = q, B = V$$

$$\text{eger } H_i = 4 \Rightarrow R = t, G = p, B = V$$

$$\text{eger } H_i = 5 \Rightarrow R = V, G = p, B = q$$

5. UYGULAMA

5.1. Tezin Amacı

Bu tez gündelik yaşamın bir parçası haline gelmiş olan sosyal medyadan alınan fotoğraflı paylaşımların beraberinde paylaşılmış metinler ile bir bağlantısı olup olmadığının araştırılmasını ele alır.

Oluşturulan sistemin çalışma adımları aşağıda belirtilmiştir.

1. Bu tez için geliştirilmiş yazılım ile sosyal medyadan, o sosyal medyaya ait API vasıtası ile veriler alınır. (ek1)
2. Toplanan veriler yine bu tez için geliştirilmiş olan renk analiz programına aktarılır. (ek2)
3. Renk analizinden geçirilmiş veriler, veri analizi yapılmak üzere ilgili programa aktarılır. Bu tezde veri analizi için knime isimli program kullanılmıştır. (ek3)
4. Veri analizinden çıkan sonuçlar değerlendirilir.

5.2. Çalışmanın Gerekçeleri

Basılı ve dijital görsel iletişimin altın çağında olduğumuz bir dünyada hedef kitle konusunda ciddi bütçelerin ayrıldığı ve ciddi araştırmaların yapıldığını biliyoruz.

Basılı veya dijital reklamcılık için hazırlanan görsel ve yazılı iletişimin daha isabetli olması gereklidir, bunun yolu ise insanların konular üzerinde nasıl düşündüğünü anlamaktan geçer. İnsanların düşüncelerine etken faktörleri;

-kültürel etkiler,

-bireysel etkiler,

-grup etkileri olmak üzere 3'e ayırabiliriz.

Kültürel etkiler, davranış normları ve sosyalleşmeyi içerir. Bireysel etkilerde güdünlenme, duygular, öğrenme ve hatırlama, tutum, algılama, rasyonel ve rasyonel olmayan düşünme, kişilik ve kişilik farklılıkları ve benlik özellikleri incelenir. Grup

etkileri faktörü ise taklit etme ve öneri alma, aile, sosyal etkiler, etnik ve dinsel etkiler, sosyal sınıf, rol ve önderlerin etkisi gibi alt faktörlerden oluşmaktadır [5].

Özetleyecek olursak;

- Basılı veya dijital renklamcılıkta nokta atışı yapabilmek için insanların paylaştıkları görsellerden konular ile ilgili olarak nasıl düşündüklerini anlayabiliriz.
- Sosyal medya üzerinden paylaşımların konumları alınabildiği için demografik analiz yapılabilir, harcanan bütçenin doğru kitle için harcanmış olması sağlanabilinir.
- İnsanların konular üzerindeki renk algılarını ölçerek nasıl düşündüğünü anlaşılabilir (insan aklında tatil kelimesinin hangi renkleri çağrıştırdığı ile örneklendirilebilir) ve bunun iletişimin gerekli yerlerinde kullanılması sağlanabilinir.

5.3. Tezin Kapsamı

Tez 5 bölümden oluşmaktadır;

Birinci bölümünde Metin Madenciliği hakkında ise detaylı bilgi verilmiştir. Zohar'a (2002) göre Metin Madenciliği metotları;

-Bilgiye Erişim (Information Retrieval),

-Bilgi Çıkarımı (Information Extraction),

-Web Madenciliği (Web Mining),

-Kümeleme (Clustering), olmak üzere dört grupta toplanmaktadır[2].

Tezin ikinci, üçüncü ve dördüncü bölümünde ise metin önişleme, metin gruplama algoritmaları ve renk uzayı hakkında bilgiler verilmiştir.

Beşinci bölümde ise uygulamadan bahsedilmiştir. Sosyal medya API'leri ile paylaşımları çekme, paylaşımların renk analizinin yapılması ve veri madenciliği ile renkler ile paylaşılmış fotoğraflar arasında bir bağlantı olup olmadığının araştırması yapılmıştır.

5.4. Sosyal Medya Ağlarından Verilerin Toplanması

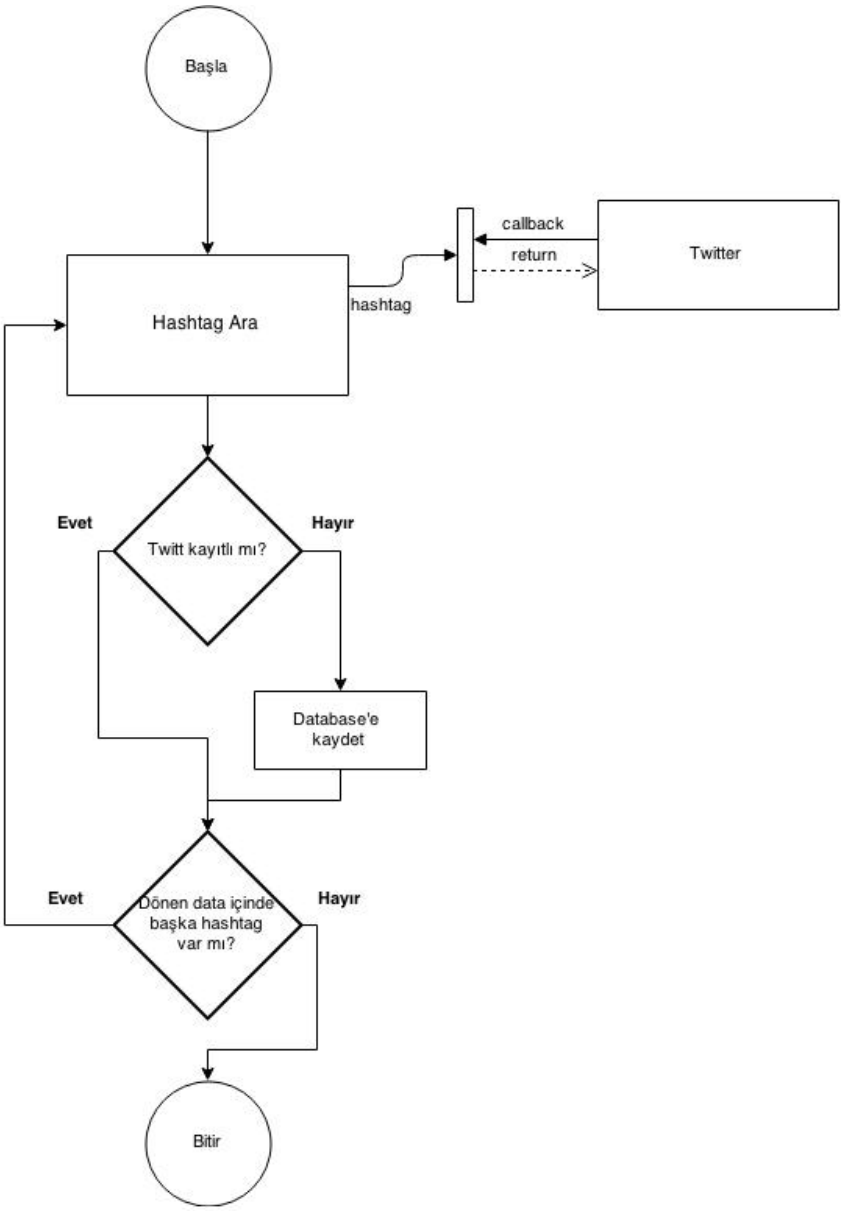
Sosyal medyadan data toplamaya başlamanın yolu önce datanın nasıl çekileceği ve sosyal medyada datanın nasıl biriktiğini anlamaktan geçer. Bir çok sosyal medya ağı bilginin vurgusu veya kategorizasyonu için hashtag kullanır.

Hashtag, genelikle sosyal ağlarda bir tümcenin ya da kelimenin başına hash(#) sembolü eklenerek oluşturulan bağlantıya verilen isimdir, anlık olarak bilgileri kategorize etmeye ve kitlelere ulaştırmaya yardımcı olur. Hashtag'ler sadece belirli bir ortama bağlıdır; video ve resim dosyalarının içerisinden bağlantı verilemez. Tüm bir cümle için kullanılabildiği gibi cümle içinde birkaç kelimeyi vurgulayabilir (örn., "Beşiktaş Barcelona'ı yenerek #tarihyazmak bir tarihe imza attı" ile ilgili sosyal medya platformu üzerinde herkesin #tarihyazmak hashtag'i adı altında paylaşımlarını görülmesi sağlanmış olur) [119].

Bu sosyal ağlarda hashtag'ler vasıtasıyla kategorize edilmiş veriye, erişmek için bir çok sosyal medya platformu erişim için API dediğimiz iki software arasındaki konuşmayı sağlayan yazılım geliştirmiştir.

API aracığılı ile aratılan hashtag'leri içeren bir metin listesine erişiriz. Metin listesinin her bir içeriği başka hashtag'ler içeriyor olabilir, dolayısıyla her bir metin listesi elemanı için tekrar bir aratma yaparak datanın devamlı olarak (bu tasarlanacak yazılım mimarisine göre değişir) toplanmasına olanak sağlanır.

Verinin toplanması amacıyla bu tez için geliştirilmiş olan uygulamanın (ek1) algoritması aşağıdaki gibidir:



Şekil V.1 Sosyal medyadan data toplama algoritması.

Toplanan verilerin json formatlı hali aşağıdaki gibidir (ek4) ve tez dökümanları arasında verilmiştir;


```
[
    {
    "_id": "55c7ca977647a34fb81296d8",
    "media": [
      {
          "id_str": "630059597840187393",
          "media_url": "http://pbs.twimg.com/media/CL5r3bSXAAECGM3.jpg",
```

```
            "type": "photo",
            "url": "http://t.co/1xUV9dsmEP"
         }
      ],
      "metadata": {
         "result_type": "popular"
      },
      "source": "<a href=\"http://twitter.com/download/iphone\"
rel=\"nofollow\">Twitter for iPhone</a>",
      "text": "Yunanistan yolculuğu başlayabilir. \n#beşiktaş #atina #komşu
#friendlymatch #gt7 #gökhantöre #türkiye http://t.co/1xUV9dsmEP",
      "user": {
         "name": "Gökhan Töre"
   }
]
```


5.5. Toplanan Verilerin Ön İşlemden Geçirilmesi

Toplanan verileri işlenecek hale getirmek ve bunun analizini yapmak için bir takım ön işlemlerden geçirmemiz gerekiyor. Verinin ham hali ile herhangi bir şekilde analiz yapmak mümkün değildir. Veri analizi için kullandığımız programda ek3'deki proje dosyasında sırasıyla yapılan işlemler aşağıdaki gibidir;

1) Dosya okuma işlemi ile json formatlı dosyamızı karakter dosyası olarak okuyoruz.
2) Satır satır okunan verinin '[,]' karakterlerini siliyoruz.
3) Karakter olarak okunan veriyi json formatına çeviriyoruz.
4) Json formatı içerisinde tek bir kolon iken içindeki metin alanlarını tabloya ikinci bir kolon olarak ekliyoruz. Örn;

Row ID	{JSON} Col0	S text
Row0	{ "_id":"55c7ca977647a34fb81296d8", "created_at":"", "id":"630059622947246080", "media":[{ "display_url":"pic.twitte... "expanded_url":"http://tw... "id_str":"630059597840187...	Yunanistan yolculuğu başlayabilir. #beşiktaş #atina #ko...

Şekil V.2 Metin kolonunun tabloya eklenmesi.

5) Media id kullanarak kaydettiğimiz fotoğrafların url'lerini tabloya kolon olarak ekliyoruz.
6) Metin alanını metin madenciliği yapmak üzere document formatına çeviriyoruz;

Row ID	S text	S	S	Document
Row0	Yunanistan yolculuğu başlayabilir. #beşikta...	...	...	"Yunanistan yolculuğu başlayabilir. #beşiktaş #atina #komşu #friendlymatch #gt7 #gökhan
Row1	RT @BirAsktir: Süleyman Seba : "Beşiktaş'ı ...	...	...	"RT @BirAsktir: Süleyman Seba : "Beşiktaş'ı üzmesinler." http://t.co/IcS5FdkIbJ"
Row2	"@Besiktas: Şenol Güneş: Top bizdeyken hı...	...	...	""@Besiktas: Şenol Güneş: Top bizdeyken hızlı ve etkili olmalıyız' http://t.co/6y0vGayvJV #B
Row3	RT @BirAsktir: Doğum günün kutlu olsun B...	...	...	"RT @BirAsktir: Doğum günün kutlu olsun Beşiktaş'ın çocuğu ! http://t.co/fv8iO0rQVN"
Row4	"@Besiktas: Olympiakos:2 Beşiktaş:1 http:...	...	...	""@Besiktas: Olympiakos:2 Beşiktaş:1 http://t.co/d1tzzIEmly #Beşiktaş #football http://t.cc
Row5	"@Besiktas: #MarioGomez #Beşiktaş #foo...	...	...	""@Besiktas: #MarioGomez #Beşiktaş #football http://t.co/d4RaskCbfK""
Row6	RT @SporSpikeri: Bir Beşiktaş taraftarının i...	...	...	"RT @SporSpikeri: Bir Beşiktaş taraftarının isyanı http://t.co/f6oRIYpIos"
Row7	"@Besiktas: .@07RQuaresma #Beşiktaş #...	...	...	""@Besiktas: .@07RQuaresma #Beşiktaş #football http://t.co/UEEru32ryH""
Row8	Beşiktaş son dakikada yıkıldı - Sabah http:/...	...	...	"Beşiktaş son dakikada yıkıldı - Sabah http://t.co/3iJuqH0FHu http://t.co/V8msmfdXzR"
Row9	Motta yeter be kardeşim ☐ ...	...	...	"Motta yeter be kardeşim ☐ #BEŞİKTAŞ #çarşı #iyigünde #kötügünde #süleymanseba #be
Row10	RT @neco_ates: Saldırı sonucu vefat eden...	...	...	"RT @neco_ates: Saldırı sonucu vefat eden Beşiktaş U16 kalecisi Servet Dündar'a Allah'tan r
Row11	"@Besiktas: .@Gokhan_Tore7 #Beşiktaş #...	...	...	""@Besiktas: .@Gokhan_Tore7 #Beşiktaş #football http://t.co/3z4vSUDkGE""
Row12	"@Besiktas: #TolgaZengin #Beşiktaş #foo...	...	...	""@Besiktas: #TolgaZengin #Beşiktaş #football http://t.co/jSK01dXMaU""
Row13	RT @apaci__ronaldo: "Derbi kazanmıyor di...	...	...	"RT @apaci__ronaldo: "Derbi kazanmıyor diye gönderdiğin hocaya son bir kez dön bak istedi
Row14	Oyunlara bakış açım #transfer #beşiktaş #...	...	...	"Oyunlara bakış açım #transfer #beşiktaş #lol http://t.co/zMUDcB3vh2"
Row15	"@Besiktas: #NecipUysal #Beşiktaş #foot...	...	...	""@Besiktas: #NecipUysal #Beşiktaş #football http://t.co/3LhC2Dyshq""
Row16	"@Besiktas: .@Ozyakup #Beşiktaş #footb...	...	...	""@Besiktas: .@Ozyakup #Beşiktaş #football http://t.co/aIethM7p7j""
Row17	"@Besiktas: Teknik Direktörümüz Şenol Gü...	...	...	""@Besiktas: Teknik Direktörümüz Şenol Güneş #Beşiktaş #football http://t.co/00txRINUiK""
Row18	"@Besiktas: .@AndreasBeck87 #Beşiktaş ...	...	...	""@Besiktas: .@AndreasBeck87 #Beşiktaş #football http://t.co/2m6ZKDqw3w""

Şekil V.3 Metin kolonunun document formatına çevirilecek yeni kolon olarak eklenmesi

7) Document içerisinde geçen terimleri kolon olarak ekleme işlemi yapıyoruz. Her terimin cümlesini karşısına yazıyoruz ki böylece renk

analizinde aynı terim aşağıda da geçiyor olsa farklı renk değerleri alabileceği için decision tree'de farklı noktalarda olacaktır. Terim çıkartma işleminin örneği aşağıdaki gibidir;

Row ID	Term	Document
Row1	Yunanistan[]	"Yunanistan yolculuğu başlayabilir. #beşiktaş #atina #komşu #friendlymatch #gt7 #gökhantöre #türkiye
Row2	yolculuğu[]	"Yunanistan yolculuğu başlayabilir. #beşiktaş #atina #komşu #friendlymatch #gt7 #gökhantöre #türkiye
Row3	başlayabilir[]	"Yunanistan yolculuğu başlayabilir. #beşiktaş #atina #komşu #friendlymatch #gt7 #gökhantöre #türkiye
Row4	.[]	"Yunanistan yolculuğu başlayabilir. #beşiktaş #atina #komşu #friendlymatch #gt7 #gökhantöre #türkiye
Row5	#beşiktaş[]	"Yunanistan yolculuğu başlayabilir. #beşiktaş #atina #komşu #friendlymatch #gt7 #gökhantöre #türkiye
Row6	#atina[]	"Yunanistan yolculuğu başlayabilir. #beşiktaş #atina #komşu #friendlymatch #gt7 #gökhantöre #türkiye
Row7	#komşu[]	"Yunanistan yolculuğu başlayabilir. #beşiktaş #atina #komşu #friendlymatch #gt7 #gökhantöre #türkiye
Row8	#friendlyma...	"Yunanistan yolculuğu başlayabilir. #beşiktaş #atina #komşu #friendlymatch #gt7 #gökhantöre #türkiye
Row9	#gt7[]	"Yunanistan yolculuğu başlayabilir. #beşiktaş #atina #komşu #friendlymatch #gt7 #gökhantöre #türkiye
Row10	#gökhantöre[]	"Yunanistan yolculuğu başlayabilir. #beşiktaş #atina #komşu #friendlymatch #gt7 #gökhantöre #türkiye
Row11	#türkiye[]	"Yunanistan yolculuğu başlayabilir. #beşiktaş #atina #komşu #friendlymatch #gt7 #gökhantöre #türkiye
Row12	http[]	"Yunanistan yolculuğu başlayabilir. #beşiktaş #atina #komşu #friendlymatch #gt7 #gökhantöre #türkiye
Row13	://t.co/1xUV...	"Yunanistan yolculuğu başlayabilir. #beşiktaş #atina #komşu #friendlymatch #gt7 #gökhantöre #türkiye
Row14	RT[]	"RT @BirAsktir: Süleyman Seba : "Beşiktaş'ı üzmesinler." http://t.co/IcS5FdkIbJ"
Row15	@BirAsktir[]	"RT @BirAsktir: Süleyman Seba : "Beşiktaş'ı üzmesinler." http://t.co/IcS5FdkIbJ"
Row16	:[]	"RT @BirAsktir: Süleyman Seba : "Beşiktaş'ı üzmesinler." http://t.co/IcS5FdkIbJ"
Row17	Süleyman[]	"RT @BirAsktir: Süleyman Seba : "Beşiktaş'ı üzmesinler." http://t.co/IcS5FdkIbJ"
Row18	Seba[]	"RT @BirAsktir: Süleyman Seba : "Beşiktaş'ı üzmesinler." http://t.co/IcS5FdkIbJ"
Row19	"[]	"RT @BirAsktir: Süleyman Seba : "Beşiktaş'ı üzmesinler." http://t.co/IcS5FdkIbJ"
Row20	Beşiktaş'ı[]	"RT @BirAsktir: Süleyman Seba : "Beşiktaş'ı üzmesinler." http://t.co/IcS5FdkIbJ"
Row21	üzmesinler[]	"RT @BirAsktir: Süleyman Seba : "Beşiktaş'ı üzmesinler." http://t.co/IcS5FdkIbJ"
Row22	.[]	"RT @BirAsktir: Süleyman Seba : "Beşiktaş'ı üzmesinler." http://t.co/IcS5FdkIbJ"
Row23	http[]	"RT @BirAsktir: Süleyman Seba : "Beşiktaş'ı üzmesinler." http://t.co/IcS5FdkIbJ"
Row24	://t.co/IcS5...	"RT @BirAsktir: Süleyman Seba : "Beşiktaş'ı üzmesinler." http://t.co/IcS5FdkIbJ"
Row25	"[]	""@Besiktas: Şenol Güneş: Top bizdeyken hızlı ve etkili olmalıyız' http://t.co/6y0vGayvJV #Beşiktaş #footb
Row26	@Besiktas[]	""@Besiktas: Şenol Güneş: Top bizdeyken hızlı ve etkili olmalıyız' http://t.co/6y0vGayvJV #Beşiktaş #footb
Row27	:[]	""@Besiktas: Şenol Güneş: Top bizdeyken hızlı ve etkili olmalıyız' http://t.co/6y0vGayvJV #Beşiktaş #footb
Row28	Şenol[]	""@Besiktas: Şenol Güneş: Top bizdeyken hızlı ve etkili olmalıyız' http://t.co/6y0vGayvJV #Beşiktaş #footb
Row29	Güneş[]	""@Besiktas: Şenol Güneş: Top bizdeyken hızlı ve etkili olmalıyız' http://t.co/6y0vGayvJV #Beşiktaş #footb
Row30	Top[]	""@Besiktas: Şenol Güneş: Top bizdeyken hızlı ve etkili olmalıyız' http://t.co/6y0vGayvJV #Beşiktaş #footb

Şekil V.4 Document kolonundan terimlerin elde edilip yeni kolon olarak eklenmesi.

8) Genel kabul görmüş renk paletlerinden birini kolonlar halinde tabloya ekliyoruz;

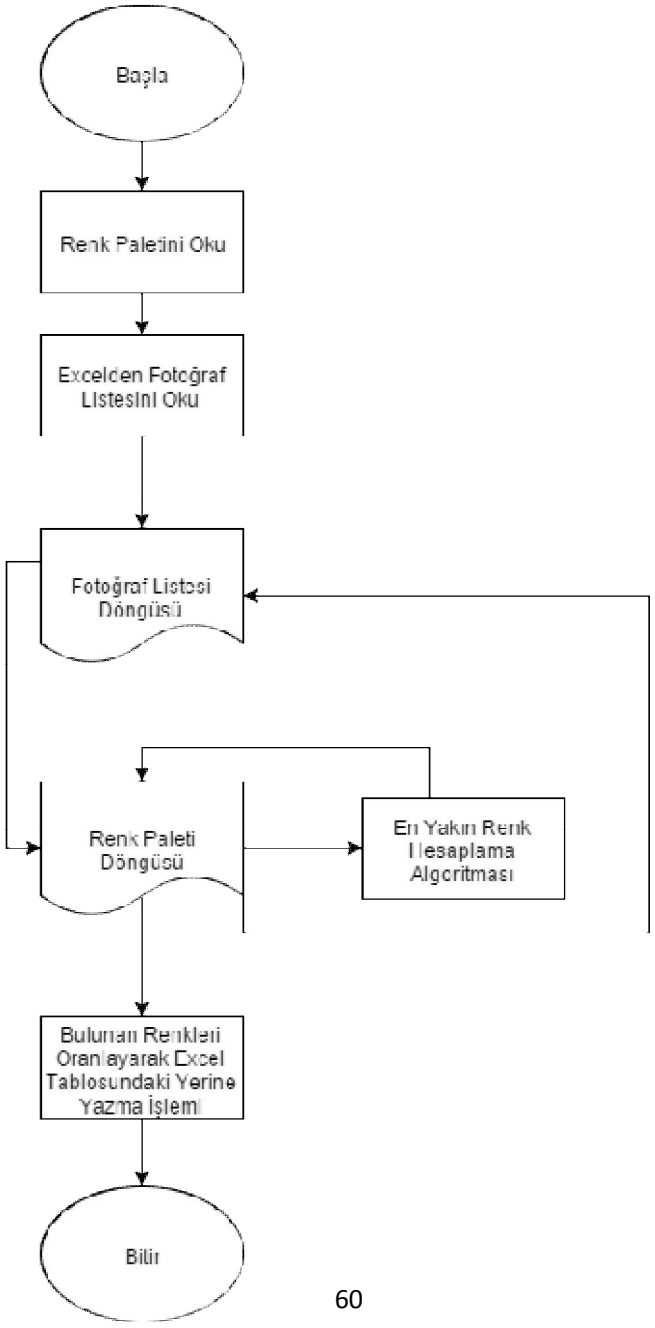
Row ID	Almond	Antique...	Apricot	Aquam...	Aspara...	Atomic ...	Banana...	Beaver	Bitters...	Black
Row0_Row1	?	?	?	?	?	?	?	?	?	?
Row0_Row2	?	?	?	?	?	?	?	?	?	?
Row0_Row3	?	?	?	?	?	?	?	?	?	?
Row0_Row4	?	?	?	?	?	?	?	?	?	?
Row0_Row5	?	?	?	?	?	?	?	?	?	?
Row0_Row6	?	?	?	?	?	?	?	?	?	?

Şekil V.5 Renk paletinin yeni kolonlar halinde tabloya eklenmesi

9) Verilerin renk analizlerinin yapılabilmesi için elde edilen tablonun excel formatına aktarılması. Elde edilen excel datası ekli tez dosyaları arasındadır [ek5].

5.6. Metin – Fotoğraf Eşlemesinin İncelenmesi

Ekli tez dosyaları içerisinde 5.5 nolu konuda anlatılan 9. adımda elde edilen excel dosyası içerisindeki yer alan fotoğrafların adresini alarak her bir noktası (pixel) için verilen renk paleti içerisinde en yakın rengi bulan, bu tez için geliştirilmiş bir yazılım bulunmaktadır. (ek2) Algoritması aşağıdaki gibidir;



Şekil V.6 Renk analizi algoritması

RGB uzayına göre hesaplanan yakınlık algoritması aşağıdaki gibidir;

kirmizi_uzaklik = $(paletteki\_rengin\_kirmizisi \; - \; noktanin\_kirmizi\_rengi)^2$

yesil_uzaklik = $(paletteki\_rengin\_yesili \; - \; noktanin\_yesil\_rengi)^2$

mavi_uzaklik = $(paletteki\_rengin\_mavisi \; - noktanin\_mavi\_rengi)^2$

uzaklik = $\sqrt{kirmizi\_uzaklik \; + yesil\_uzaklik \; + mavi\_uzaklik}$

Uzaklık sıfıra ne kadar yakın ise bu paletteki renge o kadar yakın demektir. Buradan elde edilen yakınlık bilgisi excele tekrar aktarıldıktan sonra aşağıdaki adımlarla metin fotoğraf eşleşmesi **incelenmiş ve test** edilmiştir;

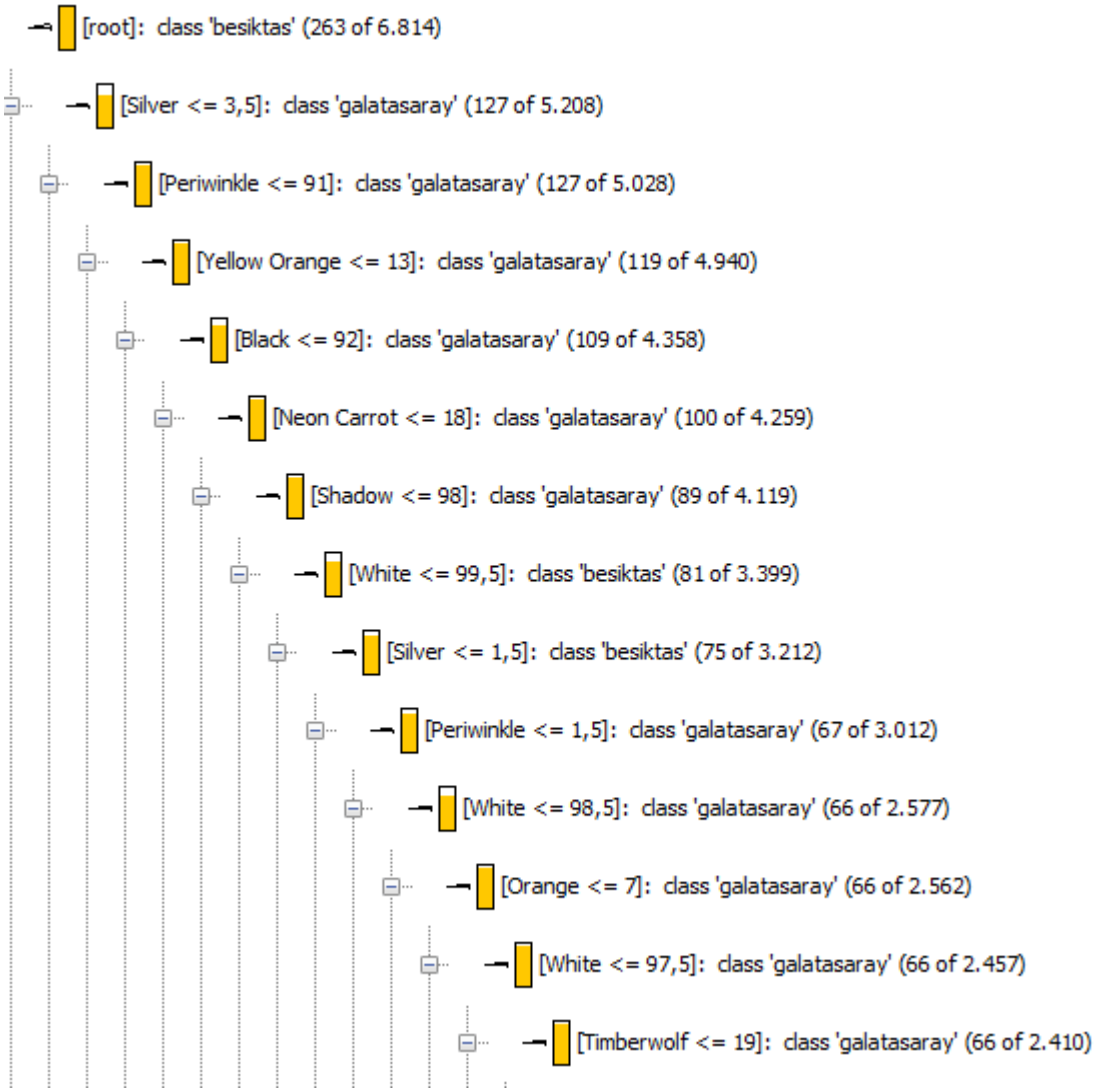
1) Renk analizinden gelen excel tablosu okuyor, tüm verileri küçük harfe çeviriyoruz, özel karakterleri siliyoruz.
2) Cümleye doğrudan etki etmeyen [ek06] ve iki harften küçük kelimeleri listeden siliyoruz;

Row ID	Term	Row ID	Term
Row228	http	Row228	
Row229	://t.co/2m6z...	Row229	tco2m6zkdq...
Row230	"	Row230	
Row231	@besiktas	Row231	besiktas
Row232	:	Row232	
Row233	.@glmersan	Row233	glmersan
Row234	#	Row234	
Row235	beşiktaş	Row235	besiktas
Row236	#football	Row236	football
Row237	http	Row237	
Row238	://t.co/xzwe...	Row238	tcoxzwexhrcyz
Row239	"	Row239	
Row240	@besiktas	Row240	besiktas
Row241	:	Row241	
Row242	futbol	Row242	futbol
Row243	takımımızın	Row243	takimimizin
Row244	@olympiaco...	Row244	olympiacos_...

Şekil V.7 Cümleye doğrudan etki etmeyen kelimelerin silinmesi

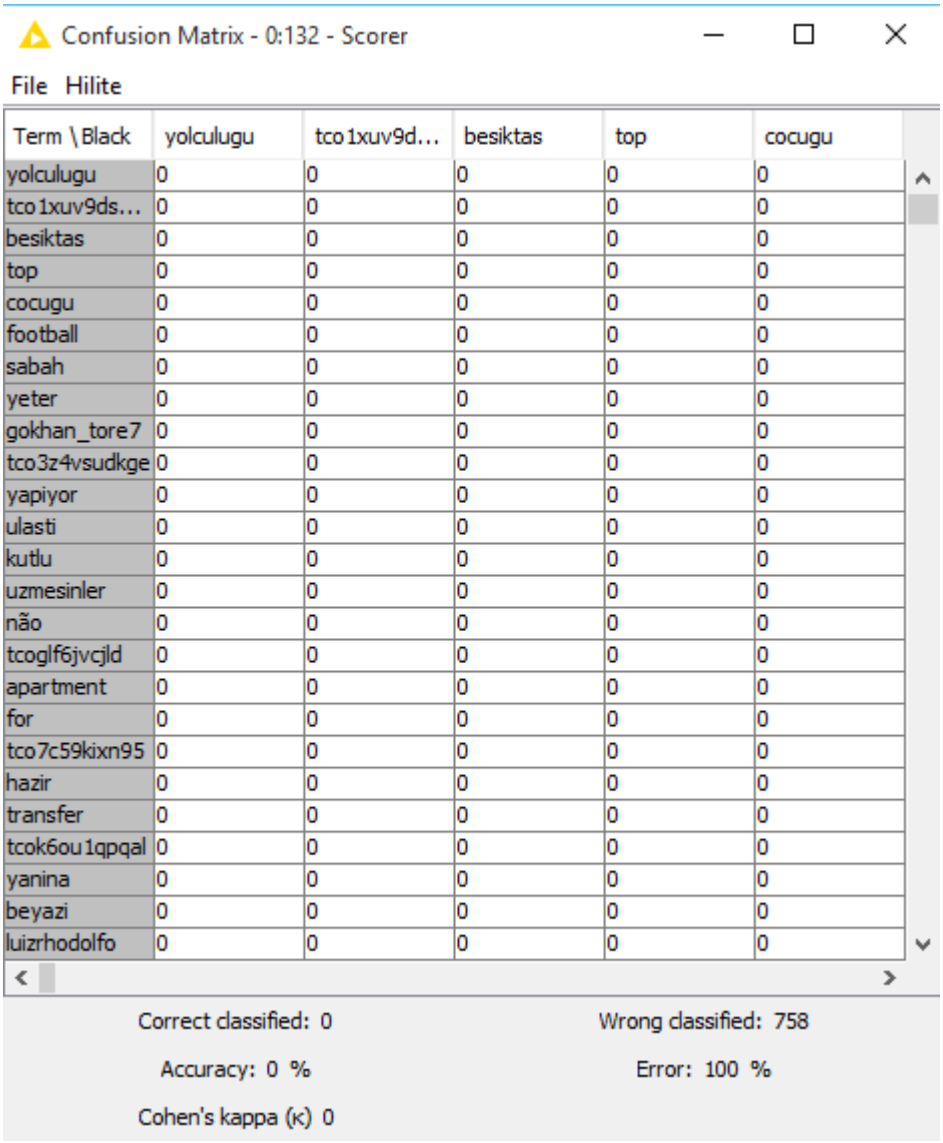
3) Boş kalan terim satırlarını siliyoruz.

4) Verinin 90'ını kelimeler ile renk oranları arasında bağlantı yakalayabilmek için sınıflandırma yapılmak üzere veri analiz programında **karar ağacı** kullanılmıştır**.** Karar ağacı çıktısının küçük bir bölümü aşağıdaki gibidir;



Şekil V.8 Karar ağacı çıktısı

5) Öğrenilmiş sınıflandırmayı sınamak için kalan 10%'luk kısmı tahminleme işlemine tabi tutuyoruz;



Şekil V.9 Karar ağacına göre test sonuçları

Yukarıda da görüldüğü üzere **758** tahminde toplam doğru sayısı **0** çıkmıştır. Tüm tahminleme işlemi yanlış tahminlenmiştir.

İşlemler özetlenecek olursa;

Fotoğraflar ile metinler arasında bir ilişki aramak amaçlı yazılmış olan bu tez için iki ayrı uygulama geliştirilmiş ve veri analiz programlarından biri ile elde edilen veriler işlenmiştir.

İlk uygulama ile Twitter'dan fotoğraflı paylaşımlar alınmıştır.

İkinci uygulama ile de fotoğrafların renk analizleri yapılmıştır.

İki ayrı uygulamadan geçen veriler, metin önişleme işleri için uygun hale getirilmiştir.

Noktalama işaretlerinin, ihtiyaç duyulmayan kelimelerin ve özel karakterlerin silinmesi gibi ayıklama işlemlerine tabi tutularak metin önişlemeden geçirilmiştir.

Metin önişlemeden geçirilen verilerin 90%'ı karar ağacına aktarılıp sınıflandırma yapılarak kelimeler ile fotoğraf arasında bir ilişki arayacak öğrenme yoluna gidilmiştir.

Son adım olan öğrenilmek üzere kullanılan verinin %90'ından kalan 758 veri ile tahminleme işlemi yapılmıştır.

Karar ağacından öğrenilen bilgiler ile tahmin için kullanılacak veriler tahminleme işlemine tabi tutulmuştur.

Bu işlem verilen renk oranlarına bakarak bize hangi kelime olacağını tahmin etmiştir.

Bu tezin konusu olan metin – fotoğraf ilişkisi saptanamamıştır, paylaşılan fotoğraf ile metinler arasında organik bir bağ olmadığı gözlemlenmiştir.

6. SONUÇ VE ÖNERİLER

İnternet kullanımının günden güne artması, insanların etkileşiminin bu yolla yapabileceği mecralar doğurmuştur. İnternet kullanımının bu denli artışı internet hızındaki teknolojiyi geliştirmiş iletişim, görsel yollar ile rahatlıkla yapılabilir olmuştur.

Bu etkileşimlerde görsel ve metin beraber kullanıldığında etkili olmaktadır. Kullanılan metin ile fotoğraf arasında bir bağ olduğu düşünülürse, bunu anlamak beraberinde basılı ve görsel iletişimde daha isabetli bir iletişim sağlayabilir.

Bu tezde bundan yola çıkılarak paylaşılan metin ile fotoğraf arasında bir bağ olup olmadığı incelenmiştir.

İlk olarak tez için geliştirilmiş bir yazılım ile belirlenen sosyal mecradan hastag'ler kullanılarak sosyal medyanın API'si ile veriler alınmıştır. Bu veriler döküman bazlı database denilen MongoDB veri ambarında saklanmıştır.

Alınan verilerin fotoğrafları yine bu tez için geliştirilmiş olan fotoğraf renk analiz programı ile analiz edilip tekrar saklanmıştır.

Fotoğraf analizlerinden sonra veri analiz programına aktarılan veriler üzerinde metin önişleme teknikleri ile metin verileri gerekli sadeliğe eriştirilmiştir. (Gereksiz kelimelerin atılması, noktalama işaretlerinin silinmesi vs)

Analize hazır olan temizlenmiş metin ve renk analizi yapılmış olan fotoğrafları veri madenciliği için açık kaynak kodlu olmasından ve kullanım alanının genişliğinden ötürü bu tezde veri analiz programı olarak seçtiğim Knime'a aktarılmıştır.

Verilerin yüzde 90'ını kelimeler ile fotoğraflar arasında nasıl bir ilişki olduğunu anlaması için sınıflandırma yapacak olan karar ağacına aktarılmıştır.

Öğrenilmiş bu veriler ile karar ağacından çıkan bilgilere göre kalan %10'luk verinin renk oranlarına bakarak doğru kelimeyi bulması amaçlanmıştır. Bu hedef doğrultusunda tahminleme işlemi yapılmıştır.

Tüm uygulamalar ve işlemler hatasız olmasına rağmen beklenilenin aksine yapılan tahminleme işleminde doğru tahmin oranı **%0** çıkmıştır. Karar ağacından elde edilen bilgiler ışığında tahminleme başarılı olamamıştır.

Bu durum gösteriyor ki bu araştırmada elde edilen veriler ile fotoğraf – metin ilişkisi kurulamamıştır. Bunun nedeni elde edilen verinin çok fazla sınıf (karar ağacından çıkan sonuç) içermiyor olması ya da metinler ile birlikte paylaşılan fotoğrafların, her defasında renk dağılımı olarak birbirinden çok farklı olmasından da kaynaklanıyor olabilir.

Ek olarak bir çok kişi farklı metinler ile aynı fotoğrafı paylaşıyor ve bu da verinin tutarlılığını azaltıyor olabilir.

Bu araştırmada beklenen sonuç elde edilmemesi, edilmeyeceği anlamına gelmemektedir. Daha büyük bir veri kümesi ile denenmesiyle farklı sonuç elde etmek mümkün olabilir.

Tez çalışması süresince kazanılan deneyimler ve elde edilen bilgiler ışığında büyük metin koleksiyonları üzerinde çalışacak araştırmacılara dağıtık uygulama mimarisi ile çalışmaları önerilir. Bu ciddi bir süre kazanımı olarak yansıyacaktır.

KAYNAKLAR

[1] Li, Y.: "High Performance Text Document Clustering", Doktora Tezi, Wright State University, **(2007)**.

[2] ZOHAR, E.Y., Introduction to Text Mining, Supercomputing, Automated Learning Group National Center for Supercomputing Applications, University of Illinois, **(2002)**.

[3] Recupero, D. R.: "A New Unsupervised Method for Document Clustering by Using Wordnet Lexical and Conceptual Relations", Journal of Information Retrieval, 10, **(2007)** 563-579.

[4] Shafei, M.; Wang, S.; Zhang, R.; Milios, E.; Tang, B.; Tougas, J.; Spiteri, R.: "A Systematic Study of Document Representation and Dimension Reduction for Text Clustering", Technical Report, CS-2006-05, Faculty of Computer Science, Dalhousie University, Halifax, Nova Scotia, **(2006)**.

[5] Sahoo, N.; Callan, J.; Krishnan, R.; Duncan, G.; Padman, R.: "Incremental Hierarchical Clustering of Text Documents", 15th ACM International Conference on Information and Knowledge Management, **(2006)** 357-366.

[6] Friedman, J. H.: "An Overview of Predictive Learning and Function Approximation", From Statistics to Neural Networks, Proc. NATO/ASI Workshop, **(1994)** 1-61.

[7] Akın, A. A.; Akın, M. D.: "Zemberek, an Open Source Nlp Framework for Turkic Languages", **(2007)**.

[8] http://wordnet.princeton.edu: "Wordnet Resmi Ağ Adresi", (Son erişim: Ağustos **2009**).

[9] Salton, G.; Wong, A.; Yang, C. S.: "A Vector Space Model for Automatic Indexing", Communications of the ACM, 18, (1975) 613-620.

[10] Zamir, O.; Etzioni, O.; Madani, O.; Karp, R. M.: "Fast and Intuitive Clustering of Web Documents", 3rd International Conference on Knowledge Discovery and Data Mining, **(1997)** 287-290.

[11] Zamir, O.; Etzioni, O.: "Web Document Clustering: A Feasibility Demonstration", 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Melbourne, Australia, **(1998)**.

[12] Tang, B.; Shepherd, M.; Milios, E.; Heywood, M. I.: "Comparing and Combining Dimension Reduction Techniques for Efficient Text Clustering", SIAM International Workshop on Feature Selection for Data Mining, Newport Beach, CA, **(2005)**.

[13] He, X.; Niyogi, P.: "Locality Preserving Projections", Advances in Neural Information Processing Systems 16, Vancouver, Canada, **(2003)**.

[14] Basili, R.; Marocco, P.; Milizia, D.: "Semantically Rich Spaces for Document Clustering", 19th International Conference on Database and Expert Systems Application, **(2008)** 43-47.

[15] Madylova, A.; Öğüdücü, Ş. G.: "Comparison of Similarity Measures for Clustering Turkish Documents", Intelligent Data Analysis, 13, **(2009)** 815832.

[16] Strehl, A.; Ghosh, J.; Mooney, R.: "Impact of Similarity Measures on Web-Page Clustering", Workshop on Artificial Intelligence for Web Search, **(2000)** 58-64.

[17] Strehl, A.: "Relationship-Based Clustering and Cluster Ensembles for HighDimensional Data Mining", Doktora Tezi, The University of Texas at Austin, **(2002)**.

[18] Bezdek, J. C.; Pal, N. R.: "Some New Indexes of Cluster Validity", IEEE Transactions on Systems, Man, and Cybernetics — Part B: Cybernetics, 28, **(1998)** 301-315.

[19] Mirkin, B.: "Mathematical Classification and Clustering", Kluwer Academic Press, Boston-Dordrecht, **(1996)**.

[20] Meila, M.: "Comparing Clusterings – an Information Based Distance ", Journal of Multivariate Analysis, 98, **(2007)** 873-895.

[21] Rosenberg, A.; Hirschberg, J.: "V-Measure: A Conditional Entropy-Based External Cluster Evaluation Measure", Joint Conference on Empirical

Methods in Natural Language Processing and Computational Natural Language Learning, Prague, **(2007)** 410–420.

[22] Dhillon, I. S.; Modha, D. S.: "Concept Decompositions for Large Sparse Text Data Using Clustering", Machine Learning, 42, **(2001)** 143-175.

[23] Dhillon, I. S.; Fan, J.; Guan, Y.: "Efficient Clustering of Very Large Document Collections", in Data Mining for Scientific and Engineering Applications, Kluwer Academic Publishers, **(2001)** 357-381.

[24] Dhillon, I. S.; Guan, Y.; Kogan, J.: "Iterative Clustering of High Dimensional Text Data Augmented by Local Search", 2nd IEEE International Conference on Data Mining, **(2002)** 131-138. [25] Zhong, S.: "Efficient Online Spherical K-Means Clustering", IEEE International Joint Conference on Neural Networks, **(2005)** 3180 - 3185.

[26] Steinbach, M.; Karypis, G.; Kumar, V.: "A Comparison of Document Clustering Techniques", 6th ACM SIGKDD International Conference on Data Mining, Workshop on Text Mining, Boston, MA, **(2000)**.

[27] Zhao, Y.; Karypis, G.: "Criterion Functions for Document Clustering: Experiments and Analysis", Technical Report, TR #04–22, Department of Computer Science, University of Minnesota, Minneapolis, MN, **(2001)**.

[28] Zhao, Y.; Karypis, G.: "Soft Clustering Criterion Functions for Partitional Document Clustering", Technical Report, TR #04–22, Department of Computer Science, University of Minnesota, Minneapolis, MN, **(2004)**.

[29] Zhao, Y.; Karypis, G.: "Hierarchical Clustering Algorithms for Document Datasets", Data Mining and Knowledge Discovery, 10, **(2005)** 141-168.

[30] Kashef, R.; Kamel, M. S.: "Enhanced Bisecting K-Means Clustering Using Intermediate Cooperation", Pattern Recognition, 42, **(2009)** 2557-2569.

[31] Zhuang, L.; Dai, H.: "A Maximal Frequent Itemset Approach for Web Document Clustering", 4th International Conference on Computer and Information Technology, **(2004)** 970 - 977.

[32] Wang, L.; Tian, L.; Jia, Y.; Han., W.: "A Hybrid Algorithm for Web Document Clustering Based on Frequent Term Sets and K-Means", APWeb/WAIM Workshops, **(2007)** 198-203.

[33] Jing, L.; Ng, M. K.; Huang, J. Z.: "An Entropy Weighting K-Means Algorithm for Subspace Clustering of High-Dimensional Sparse Data", IEEE Transactions on Knowledge and Data Engineering, 19, **(2007)** 1026-1041.

[34] Xinwu, L.: "Research on Text Clustering Algorithm Based on Improved KMeans", ETP International Conference on Future Computer and Communication, **(2009)**.

[35] Mahdavi, M.; Abolhassani, H.: "Harmony K-Means Algorithm for Document Clustering", Data Mining and Knowledge Discovery, 18, **(2009)** 370-391.

[36] Cobos, C.; Andrade, J.; Constain, W.; Mendoza, M.; León, E.: "Web Document Clustering Based on Global-Best Harmony Search, K-Means, Frequent Term Sets and Bayesian Information Criterion", IEEE Congress on Evolutionary Computation Barcelona, **(2010)** 1-8.

[37] Kalogeratos, A.; Likas, A.: "Document Clustering Using Synthetic Cluster Prototypes", Data & Knowledge Engineering, 70, **(2011)** 284–306.

[38] Zhong, S.; Ghosh, J.: "A Comparative Study of Generative Models for Document Clustering", SIAM International Conference on Data Mining - Workshop on Clustering High Dimensional Data and Its Applications, San Francisco, **(2003)**.

[39] Banerjee, A.; Dhillon, I. S.; Ghosh, J.; Sra, S.: "Clustering on the Unit Hypersphere Using Von Mises-Fisher Distributions", Journal of Machine Learning Research, 6, **(2005)** 1345–1382.

[40] Madsen, R. E.; Kauchak, D.; Elkan, C.: "Modeling Word Burstiness Using the Dirichlet Distribution", 22nd International Conference on Machine Learning, Bonn, Germany, **(2005)**.

[41] Elkan, C.: "Clustering Documents with an Exponential-Family Approximation of the Dirichlet Compound Multinomial Distribution", 23rd International Conference on Machine Learning, Pittsburgh, PA, **(2006)**.

[42] Erkan, G.: "Language Model-Based Document Clustering Using Random Walks", Human Language Technology Conference of the North American Chapter of the ACL, New York, **(2006)** 479-486.

[43] Zhang, X.; Zhou, X.; Hu, X.: "Semantic Smoothing for Model-Based Document Clustering", 6th International Conference on Data Mining, **(2006)** 1193-1198.

[44] Zhou, X.; Zhang, X.; Hu, X.: "Semantic Smoothing of Document Models for Agglomerative Clustering", 20th International Joint Conference on Artifical Intelligence, **(2007)** 2922-2927.

[45] Wen, J.; Li, Z.: "Research on Mixture Language Model-Based Document Clustering", IEEE International Conference on Granular Computing, **(2008)** 649-652.

[46] Chim, H.; Deng, X.: "A New Suffix Tree Similarity Measure for Document Clustering", 16th International Conference on World Wide Web, **(2007)**.

[47] Li, Y.; Chung, S. M.; Holt, J. D.: "Text Document Clustering Based on Frequent Word Meaning Sequences", Data & Knowledge Engineering, 64, **(2008)**.

[48] Beil, F.; Ester, M.; Xu, X.: "Frequent Term-Based Text Clustering", 8th International Conference on Knowledge Discovery and Data Mining, Edmonton, Alberta, Canada, **(2002)**.

[49] Agrawal, R.; Srikant, R.: "Fast Algorithms for Mining Association Rules in Large Databases", 20th International Conference on Very Large Data Bases, Santiago de Chile, Chile, **(1994)** 487-499.

[50] Shi, Z.; Ester, M.: "Performance Improvement for Frequent Term-Based Text Clustering Algorithm", Technique Report Computing Science, Simon Fraser University, **(2003)**.

[51] Fung, B. C. M.; Wang, K.; Ester, M.: "Hierarchical Document Clustering Using Frequent Itemsets", SIAM International Conference on Data Mining, **(2003)**.

[52] Fung, B. C. M.; Wang, K.; Ester, M.: "Hierarchical Document Clustering", in Encyclopedia of Data Warehousing and Mining, 2nd ed. vol. II, Wang, J., Ed., Information Science Reference, New York, **(2009)**.

[53] Liu, X.; He, P.: "A Study on Text Clustering Algorithms Based on Frequent Term Sets", in Advanced Data Mining and Applications, Lecture Notes in Computer Science. vol. 3584/2005, Springer, **(2005)** 347-354.

[54] Kryszkiewicz, M.; Skonieczny, L.: "Hierarchical Document Clustering Using Frequent Closed Sets", in Intelligent Information Processing and Web Mining, Advances in Soft Computing. vol. 35/2006, Springer-Verlag, **(2006)** 489-498.

[55] Malik, H. H.; Kender, J. R.: "High Quality, Efficient Hierarchical Document Clustering Using Closed Interesting Itemsets", IEEE International Conference on Data Mining, **(2006)**.

[56] Krishna, S. M.; Bhavani, S. D.: "An Efficient Approach for Text Clustering Based on Frequent Itemsets", European Journal of Scientific Research, 42, **(2010)** 385-396.

[57] Zhang, W.; Yoshida, T.; Tang, X.; Wanga, Q.: "Text Clustering Using Frequent Itemsets", Knowledge-Based Systems, 23, **(2010)** 379–388.

[58] Chen, C.-L.; Tseng, F. S. C.; Liang, T.: "Mining Fuzzy Frequent Itemsets for Hierarchical Document Clustering", Information Processing and Management, 46, **(2010)** 193–211.

[59] Kiran, G. V. R.; Shankar, K. R.; Pudi, V.: "Frequent Itemset Based Hierarchical Document Clustering Using Wikipedia as External Knowledge", International Conference on Knowledge-Based and Intelligent Information Engineering Systems, Wales, UK, **(2010)**.

[60] Dhillon, I. S.: "Co-Clustering Documents and Words Using Bipartite Spectral Graph Partitioning", 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, **(2001)** 269-274.

[61] Bekkerman, R.; Sahami, M.; Learned-Miller, E.: "Combinatorial Markov Random Fields", European Conference on Machine Learning, Berlin, Germany, **(2006)**.

[62] Hossain, M. S.; Angryk, R. A.: "Gdclust: A Graph-Based Document Clustering Technique", 7th IEEE International Conference on Data Mining Workshops, **(2007)** 417-422.

[63] Sha, Y.; Zhang, G.; Jiang, H.: "Text Clustering Algorithm Based on Lexical Graph", 4th International Conference on Fuzzy Systems and Knowledge Discovery, **(2007)** 277-281.

[64] Wensheng, G.; Guohe, L.: "Text Clustering Algorithm Based on Spectral Graph Seriation", Proceedings of Chinese Control and Decision Conference, **(2009)** 4255-4259.

[65] Yoshida, T.: "A Graph Model for Mutual Information Based Clustering", Journal of Intelligent Information Systems, **(2010)** 1-30.

[66] Lee, I.; On, B.-W.: "An Effective Web Document Clustering Algorithm Based on Bisection and Merge", Artificial Intelligence Review, 36, **(2011)** 6985.

[67] Dunn, J.: "A Fuzzy Relative of the Isodata Process and Its Use in Detecting Compact, Well-Separated Clusters", Journal of Cybernetics, 3, **(1973)** 32–57.

[68] Bezdek, J. C.: "Pattern Recognition with Fuzzy Objective Function Algorithms", Plenum Press, New York, **(1981)**.

[69] Mendes, M. E. S.; Sacks, L.: "Evaluating Fuzzy Clustering for RelevanceBased Information Access", 12th IEEE International Conference on Fuzzy Systems, **(2003)** 648-653.

[70] Rodrigues, M. E. S. M.; Sacks, L.: "A Scalable Hierarchical Fuzzy Clustering Algorithm for Text Mining", 4th International Conference on Recent Advances in Soft Computing, **(2004)** 269-274.

[71] Oh, C.-H.; Honda, K.; Ichihashi, H.: "Fuzzy Clustering for Categorical Multivariate Data", Joint 9th IFSA World Congress and 20th NAFIPS International Conference, **(2001)** 2154-2159.

[72] Kummamuru, K.; Dhawale, A.; Krishnapuram, R.: "Fuzzy Co-Clustering of Documents and Keywords", IEEE International Conference on Fuzzy Systems, **(2003)** 772-777.

[73] Tjhi, W.-C.; Chen, L.: "Possibilistic Fuzzy Co-Clustering of Large Document Collections", Pattern Recognition, 40, **(2007)** 3452-3466.

[74] Levenshtein, V. I.: "Binary Codes Capable of Correcting Deletions, Insertions and Reversals", Soviet Physics Doklady, 10, **(1966)** 707-710.

[75] Deng, J.; Hu, J.; Ch, H.; Wu, J.: "An Improved Fuzzy Clustering Method for Text Mining", 2nd International Conference on Networks Security, Wireless Communications and Trusted Computing, **(2010)** 65-69.

[76] Song, S.; Guo, Z.; Chen, P.: "Fuzzy Document Clustering Using Weighted Conceptual Model", Information Technology Journal, 10, **(2011)** 1178-1185.

[77] Gharib, T. F.; Fouad, M. M.; Aref, M. M.: "Web Document Clustering Approach Using Wordnet Lexical Categories and Fuzzy Clustering", International Workshop on Data Mining and Artificial Intelligence, Khulna, Bangladesh, **(2008)**.

[78] Song, W.; Park, S. C.: "An Improved Genetic Algorithm for Document Clustering with Semantic Similarity Measure", IEEE 4th International Conference on Natural Computation, **(2008)** 536-540.

[79] Li, Y.; Bandar, Z. A.; Mclean, D.: "An Approach for Measuring Semantic Similarity between Words Using Multiple Information Sources", IEEE Transactions on Knowledge and Data Engineering, 15, **(2003)**.

[80] Shehata, S.: "A Wordnet-Based Semantic Model for Enhancing Text Clustering", IEEE International Conference on Data Mining Workshops, **(2009)** 477-482.

[81] Shehata, S.; Karray, F.; Kamel, M. S.: "An Efficient Concept-Based Mining Model for Enhancing Text Clustering", IEEE Transactions on Knowledge and Data Engineering, 22, **(2010)** 1360-1371.

[82] Hu, X.; Zhang, X.; Lu, C.; Park, E. K.; Zhou, X.: "Exploiting Wikipedia as External Knowledge for Document Clustering", KDD2009 Conference on Knowledge Discovery and Data Mining, Paris, France, **(2009)** 389-396.

[83] Fodeh, S. J.; Punch, W. F.; Tan, P.-N.: "Combining Statistics and Semantics Via Ensemble Model for Document Clustering", Symposium on Applied Computing, Honolulu, Hawaii, USA, **(2009)**.

[84] Gabrilovich, E.; Markovitch, S.: "Wikipedia-Based Semantic Interpretation for Natural Language Processing", Journal of Artificial Intelligence Research, 34, **(2009)** 443-498.

[85] Zheng, H.-T.; Kang, B.-Y.; Kim, H.-G.: "Exploiting Noun Phrases and Semantic Relationships for Text Document Clustering", Information Sciences, 179, **(2009)** 2249–2262.

[86] Jing, L.; Ng, M. K.; Huang, J. Z.: "Knowledge-Based Vector Space Model for Text Clustering", Knowledge and Information Systems, 25, **(2010)** 35-55.

[87] Duong, V. T. T.; Cao, T. H.; Chau, C. K.; Quan, T. T.: "Latent Ontological Feature Discovery for Text Clustering", 7th IEEE International Conference on Research, Innovation and Vision for the Future in Computing and Communication Technologies, **(2009)**.

[88] Strehl, A.; Ghosh, J.: "Cluster Ensembles – a Knowledge Reuse Framework for Combining Partitionings", Journal of Machine Learning Research, 3, **(2002)** 583-617.

[89] Greene, D.; Cunningham, P.: "Efficient Ensemble Methods for Document Clustering", Tehcnical Report Department of Computer Science, Trinity College Dublin, **(2006)**.

[90] Gonzalez, E.; Turmo, J.: "Non-Parametric Document Clustering by Ensemble Methods", 13th International Conference on Natural Language and

Information Systems: Applications of Natural Language to Information Systems, **(2008)** 245-256.

[91] Bekkerman, R.; Scholz, M.; Viswanathan, K.: "Improving Clustering Stability with Combinatorial Mrfs", KDD2009 Conference on Knowledge Discovery and Data Mining, Paris, France, (2009) 99-107.

[92] Kashef, R.; Kamel, M. S.: "Cooperative Clustering", Pattern Recognition, 43, **(2010)**.

[93] Zhong, S.: "Efficient Streaming Text Clustering", Neural Networks, 18, **(2005)** 790-798.

[94] Aggarwal, C. C.; Yu, P. S.: "A Framework for Clustering Massive Text and Categorical Data Streams", SIAM Conference on Data Mining, Bethesda, MD, **(2006)** 477-481.

[95] He, Q.; Chang, K.; Lim, E.-P.; Zhang, J.: "Bursty Feature Representation for Clustering Text Streams", SIAM International Conference on Data Mining, **(2007)** 491-496.

[96] Liu, Y.-B.; Cai, J.-R.; Yin, J.; Fu, A. W.-C.: "Clustering Text Data Streams", Journal of Computer Science and Technology, 23, **(2008)** 112-128.

[97] Gil-García, R.; Pons-Porrata, A.: "Dynamic Hierarchical Algorithms for Document Clustering", Pattern Recognition Letters, 31, **(2010)** 469–477.

[98] Strehl, A.; Ghosh, J.: "Relationship-Based Clustering and Visualization for High-Dimensional Data Mining", INFORMS Journal on Computing, **(2003)** 208-230.

[99] Zhao, Y.; Karypis, G.: "Topic-Driven Clustering for Documents", SIAM International Conference on Data Mining, **(2005)** 358-369.

[100] Congnan Luo; Li, Y.; Chung, S. M.: "Text Document Clustering Based on Neighbors", Data & Knowledge Engineering, 68, **(2009)** 1271-1288.

[101] Aliguliyev, R. M.: "Performance Evaluation of Density-Based Clustering Methods", Information Sciences, 179, **(2009)** 3583-3602.

[102] Wang, J. Z.; Taylor, W.: "Concept Forest: A New Ontology-Assisted Text Document Similarity Measurement Method", IEEE/WIC/ACM International Conference on Web Intelligence, **(2007)** 395-401.

[103] Bisson, G.; Hussain, F.: "X-Sim: A New Similarity Measure for the CoClustering Task", 7th International Conference on Machine Learning and Applications, **(2008)** 211-217.

[104] Hussain, F.; Bisson, G.; Grimal, C.: "An Improved Co-Similarity Measure for Document Clustering", 9th International Conference on Machine Learning and Applications, **(2010)** 190-197.

[105] http://people.sabanciuniv.edu/~oflazer/balkanet/index.htm: "Balkanet - Turkish Wordnet Resmi Ağ Adresi", (Son erişim: Mayıs **2015**).

[106] Wu, Z.; Palmer, M.: "Verbs Semantics and Lexical Selection", 32nd Annual Meeting of the Association for Computational Linguistics, **(1994)** 133-138.

[107] D'hondt, J.; Vertommen, J.; Verhaegen, P.-A.; Cattrysse, D.; Duflou, J. R.: "Pairwise-Adaptive Dissimilarity Measure for Document Clustering", Information Sciences, 180, **(2010)** 2341–2358.

[108] Guan, R.; Shi, X.; Marchese, M.; Yang, C.; Liang, Y.: "Text Clustering with Seeds Affinity Propagation", IEEE Transactions on Knowledge and Data Engineering, 23, **(2011)** 627-637.

[109] Nguyen, D. T.; Chen, L.; Chan, C. K.: "Clustering with Multi-Viewpoint Based Similarity Measure", IEEE Transactions on Knowledge and Data Engineering, 23, **(2011)**.

[110] Osinski, S.; Weiss, D.: "A Concept-Driven Algorithm for Clustering Search Results", IEEE Intelligent Systems, 20, **(2005)** 48–54.

[111] Stefanowski, J.; Weiss, D.: "Comprehensible and Accurate Cluster Labels in Text Clustering", 8th Conference on Information Retrieval, Pittsburgh, PA, **(2007)** 198-209.

[112] Carmel, D.; Roitman, H.; Zwerdling, N.: "Enhancing Cluster Labeling Using Wikipedia", ACM Special Interest Group Confernce on Information Retrieval, Boston, Massachusetts, **(2009)** 139-146.

[113] Muhr, M.; Kern, R.; Granitzer, M.: "Analysis of Structural Relationships for Hierarchical Cluster Labeling", ACM Special Interest Group Confernce on Information Retrieval, Geneva, Switzerland, **(2010)**.

[114] Işık, M.: "Bölünmeli Kümeleme Yöntemleri İle Veri Madenciliği Uygulamaları", Yüksek Lisans Tezi, Marmara Üniversitesi Fen Bilimleri Enstitüsü, İstanbul, **(2006)**.

[115] Kadriye E.:" Metin madenciliği yöntemleri ile ürün yorumlarının otomatik değerlendirilmesi", Sakarya Üniversitesi / Fen Bilimleri Enstitüsü / Endüstri Mühendisliği Anabilim Dalı **(2012)**

[116] Volkan T.: "Metin madenciliği için iyileştirilmiş bir kümeleme yapısının tasarımı ve uygulaması", Marmara Üniversitesi / Fen Bilimleri Enstitüsü / Elektronik-Bilgisayar Eğitimi Anabilim Dal **(2011)**

[117] Selçuk Üniversitesi Jeodezi ve Fotogrametri Mühendisliği Öğretiminde 30. Yıl Sempozyumu, **16-18 Ekim 2002**, Konya

[118] http://tr.wikipedia.org/wiki/HSV_renk_uzay%C4%B1 (Son erişim: **Mayıs 2015**).

[119] B.W. Kernighan & d.Ritchie (1978). The C Programming Language. Prentice Hall. s. 86, 207

[120] http://en.wikipedia.org/wiki/List_of_colors:_A%E2%80%93F (Son erişim: **Mayıs 2015**).

[121] http://www.ustatlar.net/c-denetim/bt-denetimi/bt-it-acl/254-acl-bt.html (Son erişim: **Mayıs 2015)**.

[122] http://ammaraslan.blogspot.com.tr/2014/05/renk-uzaylar.html (Son erişim: **Mayıs 2015)**.

EKLER

Ek1- Sosyal ağlardan veri toplama yazılımı

Ek2- Renk analiz yazılımı

Ek3- Veri analiz yazılımı projesi

Ek4- Toplanılan veriler

Ek5- Elde edilen renk verileri

ÖZGEÇMİŞ

2 Temmuz 1986 tarihi, İstanbul İli Gaziosmanpaşa ilçesi doğumluyum. İlk ve Ortaokulu aynı ilçede tamamladıktan sonra liseyi Maçka Akif Tunçel Anadolu Meslek Lisesi Bilgisayar Bölümü'nde tamamladım. Ardından meslek yüksek okulu olarak İstanbul Üniversitesinde Bilgisayar Programcılığı bölümünü bitirdim. Mezun olduktan sonra çeşitli yazılım evlerinde çalışırken bir yandan da Eskişehir Anadolu Üniversitesi İşletme Bölümü'nü bitirdim. Çalıştığım yerlerde aktif olarak yazılım projeleri yürütürken 2013 yılında da, Beykent Üniversitesi, Matematik Bilgisayar Anabilim Dalında Bilgi Teknolojileri yüksek lisans eğitimine başladım.

Özel ilgi alanlarım mikrochip programlama, haberleşme sistemleri, teknik proje yönetimi ve "internet of things" projeleridir.

Yabancı dilim İngilizce olmakla beraber, biraz japonca da bilirim. Evliyim ve bir çocuk babasıyım.

Aykut DEMİREL