AĞIRLIKLI KAPSAM YOĞUNLUĞU ALGORİTMASI YAKLAŞIMI İLE RSS TABANLI HABER TAVSİYE SİSTEMİ

ÇAĞLAR DUMAN

YÜKSEK LİSANS TEZİ
BİLGİSAYAR MÜHENDİSLİĞİ

TOBB EKONOMİ VE TEKNOLOJİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

TEMMUZ 2011
ANKARA

Fen Bilimleri Enstitü onayı

_____

Prof. Dr. Ünver Kaynak
Müdür

Bu tezin Yüksek Lisans derecesinin tüm gereksinimlerini sağladığını onaylarım.

_____

Doç. Dr. Erdoğan Doğdu
Anabilim Dalı Başkanı

Çağlar DUMAN tarafından hazırlanan AĞIRLIKLI KAPSAM YOĞUNLUĞU ALGORİTMASI YAKLAŞIMI İLE RSS TABANLI HABER TAVSİYE SİSTEMİ adlı bu tezin Yüksek Lisans tezi olarak uygun olduğunu onaylarım.

_____

Yrd. Doç. Dr. Tansel ÖZYER
Tez Danışmanı

Tez Jüri Üyeleri

Başkan : Doç. Dr. Kemal BIÇAKÇI _____

Üye : Doç. Dr. Bülent TAVLI _____

Üye : Yrd. Doç. Dr. Tansel ÖZYER _____

TEZ BİLDİRİMİ

Tez içindeki bütün bilgilerin etik davranış ve akademik kurallar çerçevesinde elde edilerek sunulduğunu, ayrıca tez yazım kurallarına uygun olarak hazırlanan bu çalışmada orijinal olmayan her türlü kaynağa eksiksiz atıf yapıldığını bildiririm.

Çağlar DUMAN

Üniversitesi: TOBB Ekonomi ve Teknoloji Üniversitesi

Enstitüsü: Fen Bilimleri

Anabilim Dalı: Bilgisayar Mühendisliği

Tez Danışmanı : Yrd. Doç. Dr. Tansel ÖZYER

Tez Türü ve Tarihi: Yüksek Lisans – Temmuz 2011

ÇAĞLAR DUMAN

AĞIRLIKLI KAPSAM YOĞUNLUĞU ALGORİTMASI YAKLAŞIMI İLE RSS TABANLI HABER TAVSİYE SİSTEMİ

ÖZET

Günümüzde internetin yaygınlaşması ile birlikte yapısal olmayan verilerin özellikle de metinsel verilerin miktarı çok fazla artmıştır. Haber siteleri metinsel verilerin en önemli örneğidir. Veritabanlarında, internet ve intranetlerde çok büyük miktarda haber bilgileri depolanır. Bu bilgiler dokümanlarda veya metin dokümanlarında tutulmaktadır. Bu bilgilerden önemli bilgiler çıkartmak, işimize yarayan haber metinlerini çekerek kişinin haber alışkanlığını öğrenmek esas problemimizdir.

Bu tezde metin halindeki veriye erişimi kolaylaştıran, zaman ve hız kazandıran metin kategorizasyon tekniği olan ağırlıklı kapsam yoğunluğu ağırlıklandırma algoritması incelenmiş olup, bu teknik kullanılarak rss tabanlı dinamik içerikli, içeriği farklı haber sitelerinin farklı kategorilerinde yer alan haberleri tarayarak, kullanıcının okuduğu benzer haberleri dinamik olarak kümeleyen, kullanıcının haber alışkanlığını öğrenen, gerekli gruplara göre okuyabileceği haberleri tavsiye eden akıllı bir sistem gerçekleştirilmiştir.

Anahtar Kelimeler: rss, metin benzerliği, metin ayıklama, kümeleme, anahtar kelime çıkarımı,

University: TOBB Economics and Technology University

Institute: Institute of Natural and Applied Sciences

Science Programme: Computer Engineering

Supervisor: Assistant Prof. Dr. Tansel ÖZYER

Degree Awarded and Date: Master of Science – July 2011

ÇAĞLAR DUMAN

RSS BASED NEWS RECOMMENDATION SYSTEM WITH A WEIGHTED COVERAGE DENSITY ALGORITHM APPROACH

ABSTRACT

Today the amount of non-structural data, especially textual data, significantly increased due to become widespread of internet. News portals are the most important examples of textual data. Huge amount of news documents are stored in databases, internet and intranet. This information is kept in a document, or text documents. Our primary problem is to extract significant information and obtain the documents that we need. The main aim is to learn the people's preferences and habits of reading news documents.

In this thesis, weighted coverage density, which is a timesaving text categorization technique, enabling access to data in text form, algorithm was analyzed. Accordingly an rss-based smart system with dynamic content, a system that dynamically aggregates the similar news read by the users by searching the news included in different categories of the news sites with different contents that determines news habits of the user and recommends the news that might be read in accordance with the related groups, was developed.

Keywords: rss, data similarity, word stemming, clustering, title generation,

TEŞEKKÜR

Bu tezin hazırlanmasında yardım ve katkılarıyla beni yönlendiren değerli Hocam Yrd. Doç. Dr. Tansel Özyer'e, yüksek lisans eğitimim boyunca bana değerli katkılarda bulunan TOBB Ekonomi ve Teknoloji Bilgisayar Mühendisliği bölümü Hocalarıma, ofiste bana yardımlarını esirgemeyen mesai arkadaşlarıma, beni her zaman destekleyen ve yanımda olan değerli aileme ve kıymetlim Zerrin'e çok teşekkür ederim.

İÇİNDEKİLER

ÇİZELGELERİN LİSTESİ

ŞEKİLLERİN LİSTESİ

TABLOLARIN LİSTESİ

GRAFİKLERİN LİSTESİ

KISALTMALAR

Kısaltma	Açıklama
RSS	Really Simple Syndication
XML	Extensible Markup Language
XSL	Extensible Stylesheet Language Family
AKY	Ağırlıklı Kapsam Yoğunluğu
VTBK	Veritabanı Bilgi Keşfi
KY	Kapsam Yoğunluğu
BAKY	Beklenen Ağırlıklı Kapsam Yoğunluğu
BFI	Birleştirme Farklılık İndeksi (Birleştirme Maliyeti)
TBFI	Türevsel Birleştirme Farklılık İndeksi (Maliyetin Tepe Noktası)
BKPLOT	En iyi K Görselleştirme Methodu
KK	Kolerasyon Katsayısı
DP	Doğru Pozitif
DN	Doğru Negatif
YP	Yanlış Pozitif
YN	Yanlış Negatif
OH	Okunmayan Haber
THK	Taranan Haberin Kümesi
TFS	Toplam Frekans Sayısı
DPO	Doğru Pozitif Oranı
DNO	Doğru Negatif Oranı
YPO	Yanlış Pozitif Oranı
YNO	Yanlış Negatif Oranı

SEMBOL LİSTESİ

Bu çalışmada kullanılmış olan simgeler açıklamaları ile birlikte aşağıda sunulmuştur.

Simge	Açıklama
$\mathbf{W_j}$	Kelime ağırlığı (j kelimesinin ağırlığı)
$\mathbf{M_k}$	Öğe sayısı (k kümesindeki ayrık kelime sayısı)
$\mathbf{N_k}$	İşlemsel veri sayısı (k. kümesindeki haber sayısı)
$\mathbf{S_k}$	Frekans toplamı (k kümesindeki kelimelerin frekans toplamı)
$\mathbf{T_k}$	İşlemsel veriler (k kümesindeki haber verileri)
N	Toplam işlemsel veri sayısı (toplam haber sayısı)
KK(w,c)	Kelime w'nin korelasyon katsaysı
$\mathbf{Frekans(I_{kj})}$	Kelime frekansı (j. kelimenin k. kümesinde geçme sıklığı)
$\mathbf{C_i}$	Küme (i. haber kümesi)
θ	En iyi küme sayısı için destek değeri
$\mathbf{CM_k}$	k kümesi için küme modu farklılığı

BÖLÜM 1

1. GİRİŞ

Günümüzde, dinamik verilerin fazla olduğu, enformatik akış yoğunluğu içerisinde yaşamaktayız. Dünya çapında ağ[1], ya da yaygın bilinen adıyla internet, dinamik yapısı ve inanılmaz sayıdaki kaynakları nedeniyle hayli karmaşık ve sofistikedir. İnternet, her ne kadar anlamsal ağ[2] olarak tanımlanan, otomatik çevrilebilir web olmaya doğru yöneliyorsa da, hali hazırda büyük ölçekte, denetimsiz ve dağınık bir haldedir. Hemen her gün binlerce satır haber, ajanslara akmakta, on binlerce internet sitesinin içeriği güncellenmekte ve bu bilgiler her gün dünya üzerinde iki milyar civarında olan internet kullanıcısı arasında gidip gelmektedir.[1]

Uzmanlar, webi en azından kısmen yapılandırabilmek amacıyla çeşitli standartlar geliştirmektedirler. Bu tür kısmi yapılandırmanın en popüler örneği XML[3]'dir. XML, birbiriyle etkileşim içinde olan web uygulamaları arasında bir veri değiş-tokuş standardı getirmek için kullanılmıştır.[2] Şema tanımlarıyla birlikte kullanılan bir metin tabanını ve metne bağlı etiketler arasındaki verileri içerdiğinden, söz konusu etiketler içinden gerekli verilerin çıkarılması, ayrıştırma[4] yoluyla mümkün olmaktadır. RSS[5] ise XML'in bir formu olup, dinamik kullanım amacıyla kaynağa ulaşması gereken bilgi ve servislerin yayımlanması amacını taşır. Günümüzde servis sağlayıcılar, forumlar ve bloglar yaygın olarak RSS kullanmaktadır.[3]

Çeşitli RSS besleme kaynaklarından haber toplamayı sağlayan birçok RSS okuyucusu vardır. Ancak, bunların çoğu alıcı tarafında çalışan uygulamalar olduğundan, haberlerin içeriğiyle bağlantılı kümeleme yapma kapasiteleri yetersizdir. Bunun çözümü hayli güç bir sorundur. Birçok servis RSS'den haberleri topladıktan

[1] ing: *world wide web*
[2] ing: *semantic web*
[3] ing: *extensible markup language (XML)*
[4] ing: *parsing*
[5] ing: *really simple syndication (RSS)*

sonra, verileri okuyucunun ilgisine kullanıcı-dostu[6] bir usulde sunmaktadır. Çoğunlukla bir haber seli söz konusudur ve insanlar bu haber seli içerisinde boğulup aradıklarına ulaşamamaktadır. Bir haber sitesinin, kullanıcının tercihlerine göre kişiselleştirilmesi, geçmişte okunan haberlere dayalı bir sıralama oluşturulması daha uygun bir yaklaşım olacaktır.

Veri işleme[7], yeni ve önemli kalıpların tanımlanması ve tahmin amaçlı olarak ortaya çıkarılması süreci olarak ifade edilebilir. Tavsiye sistemleri bu teknikleri dağınık veriler üzerinde uygulamaktadır.[1]

Tez kapsamında geliştirilen haber tavsiye sistemi, kullanıcılara haber önerisi sunmaktadır. Bu çalışmanın katkıları aşağıdaki gibi ifade edilebilir:

1. Sistem tamamen web ortamında çalıştığından, herhangi bir yükleme yapmaya gerek duymadan kullanıcıyla bağlantı kurabilmektedir.
2. Sistem kullanıcı profillerini hiyerarşik bir biçimde oluşturur ve bu hiyerarşi ve profil bilgisi okunan haberlere göre dinamik olarak değişir.
3. Profiller ayırt edici kelime bulma yöntemi kullanılarak tanımlanır.
4. Optimal küme sayısı analiz yoluyla hesaplanır.

Kullanıcılar her bir profil için önceden okudukları ve sistem tarafından kümelenmiş haberlere göre birden fazla profile sahip olabilirler ve her bir profil için okunmamış haberler tavsiye edilir.

Bunlara ilaveten, bu tez çalışmasında haberleri içeriklerine göre toplayarak sınıflandıran akıllı çevrimiçi bir rss okuyucu sunulmaktadır. Ayrıca, çoklu profil oluşturmakta kullanılacak bir kümeleme algoritması da önerilmektedir. Söz konusu algoritma ağırlıklı kapsam yoğunluğu[8] [4] olarak adlandırılan etkin bir kümeleme algoritmasına dayanmaktadır. Temel olarak kapsam ve yoğunluk konuları üzerinde çalışmakta olan bu algoritma yoluyla okuyucuyu profil sayılarına bağımlı kılmadan,

[6] ing: *user friendly*
[7] ing: *data processing*
[8] ing: *weighted coverage density (WCD)*

ulaşılabilecek sayıda bir küme kullanılarak haberlerin kümelenmesi ve tavsiye edilmesi hedeflenir. Söz konusu profillerin her birinin bir etiket tanımı, başlığı vardır. Bu başlık, kümedeki haberler içinden otomatik olarak oluşturulur. Basitçe, küme içindeki ayırt edici kelimeler tespit edilerek kullanıcı profili belirlenmeye çalışılır. Oluşturulan kullanıcı profilleri ya da diğer değişle kullanıcı kümeleri sayesinde sistem ilgili kullanıcı için eğitilmiş olur. Böylece kullanıcıya, okunmayan yeni haberler arasından kullanıcının ilgilendiği haberler tavsiye edilir hale gelmektedir.

Bu tez çalışması şu şekilde düzenlenmiştir. Bölüm 1'de, tez çalışmasının fikir temelleri ve yapılan çalışmanın kısaca ifadesini içeren giriş bölümü bulunmaktadır. Bölüm 2'de, ağırlıklı kapsam yoğunluğu algoritması yaklaşımı anlatılmıştır. Bölüm 3'de, kullanılan yöntemler hakkında genel bilgi verilerek, ilgili çalışmalardan bahsedilmiştir. Bölüm 4'te, önerilen haber tavsiye sisteminin yapısı ve aşamaları ortaya konulmuştur. Bunu takiben Bölüm 5'te, sistemin kullanıcılar üzerinde test edilmesi ve deney sonuçları gözlemlenmiştir. Bölüm 6, sonuç bölümünde ise, son yorumlar açıklanarak, gelecek çalışmalara değinilip tez çalışması sonuçlandırılmıştır.

BÖLÜM 2

2. AĞIRLIKLI KAPSAM YOĞUNLUĞU ALGORİTMASI YAKLAŞIMI

Bu bölümde, tez çalışmasında önerilen haber tavsiye sisteminin temelini oluşturan ağırlıklı kapsam yoğunluğu algoritması anlatılmaktadır. İlk olarak, Bölüm 2.1'de işlemsel veri setlerine[9] değinilmiştir. Daha sonra, Bölüm 2.2'de, küme-içi benzerlik ölçütlerinden[10] bahsedilmiştir. Bölüm 2.3'de, kümedeki aynı öğelerin yaklaşık dağılımını ölçen ve temel olarak öğe dağılımları arasındaki farkı kullanan, kapsam yoğunluğu[11] (KY) algoritmasına değinilmiştir. Bunu takiben Bölüm 2.4'de, kapsam yoğunluğu algoritmasının problemleri ortaya konmuştur. Bölüm 2.5'de, tercih edilen öğe dağılımı olarak öğe-setini tanımlayan ve kümelemede en sık tekrarlanan öğe setlerine daha fazla yer vermek için kullanılan ağırlıklı kapsam yoğunluğu (AKY) algoritması tanımlanmıştır. Son olarak, Bölüm 2.6'da, kümeleme kriterimizi oluşturan beklenen ağırlıklı kapsam yoğunluğu[12] (BAKY) algoritmasından bahsedilmiştir.

2.1. İşlemsel Veri Setleri

Tez kapsamında kullanılan haber verileri birer işlemsel veridir. İşlemsel veriler, tabloya dönüştürülebilen özel bir tür veri çeşididir.[4] İşlemsel verilerin hacmi genellikle büyüktür. Bu nedenle, yüksek ölçekli işlemsel veri setlerini kümelemek için, hızlı ve yüksek kalitede algoritmalara dönük talep sürekli artmaktadır.

Bir işlemsel veri seti her biri çok sayıda öğeden oluşan N sayıda işlemsel veriden oluşur.[4] Örneğin, t1=(gazete, dergi, broşür) üç öğeli bir işlemsel veridir. t2= (dergi, broşür) ise iki öğeli bir işlemsel veridir. Bu iki işlemsel veri için; her öğe bir işlem ve her veri bir satır gibi değerlendirilerek, işlemsel veri setine dönüştürülebilir. İki verinin birleşmesi N=2 sayılı işlemsel veri setini oluşturur. İşlemsel veri setleri çeşitli

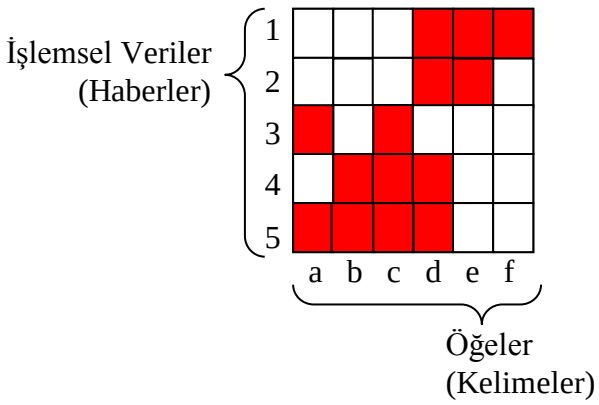
[9] ing: *transactional datasets*
[10] ing: *intra-cluster similarity measures*
[11] ing: *coverage density*
[12] ing: *expected weighted coverage density*

algoritmalar vasıtasıyla kümelendirilebilir. Ancak, genel kümeleme algoritmaları işlemsel veri setlerinin işlenmesinde yetersiz kalmaktadır. Çünkü işlemsel veri setleri yüksek hacimli ve yüksek boyutludur.[4]

Ağırlıklı kapsam yoğunluğu algoritması (AKY), bu işlemsel veri setleri için tasarlanmış bir yaklaşımdır. Bu algoritmanın temel fikri, küme öğeleri arasındaki kesişmeyi kontrol etmeye ve kümelerdeki öğe setini mümkün olduğunca korumaya çalışan bir kümeleme kriterine dayanmaktadır.[5]

2.1.1. İşlemsel Veri Setlerinin Kümelenmesi

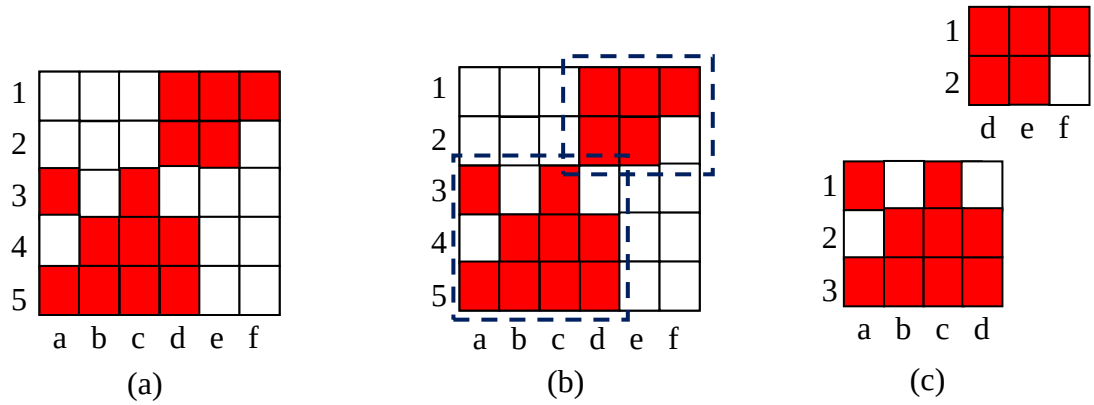
İşlemsel veri setine örnek olarak bir veri, veri seti grafiğinde eşleştirildiğinde; yatay eksen, öğeleri, dikey eksen de işlemsel verileri göstermektedir. Her bir hücre (i, j) j. işlemsel verisindeki i. öğeyi gösterdiği düşünülmektedir. Örneğin, basit bir işlemsel veri seti {abcd, bcd, ac, de, def} Şekil 2.1'de gösterilmiştir.



Şekil 2.1. İşlemsel veri seti grafik örneği [4]

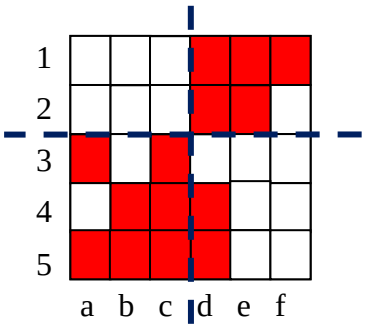
Şekil 2.1'de birinci işlemsel veri {def}, ikinci veri {d,e}, üçüncü veri {a,c}, dördüncü veri {b,c,d} ve beşinci veri {a,b,c,d}'yi göstermektedir. Hücrelere bakıldığında ise örneğin (d,1); 1.işlemsel veri "d" öğesini göstermektedir. (a,5) ise beşinci işlemsel verideki "a" öğesini simgelemektedir.

Tez çalışması için önerilen haber tavsiye sisteminde işlemsel veriler haberleri, öğeler ise kelimeleri temsil etmektedir.



Şekil 2.2. İşlemsel veri seti kümeleme örneği [5]

Şekil 2.2-a grafiğinde *{*abcd, bcd, ac, de, def*}* veri setinin tümünü tek bir küme olarak kabul etmiştir. Grafikteki dolu alana bakıldığında, doğal olarak oluşmuş iki küme, *{*abcd, bcd, ac*}* ve *{*de, def*}* Şekil 2.1-b'deki iki dikdörtgen ile belirtilmektedir. Şekil 2.2-a orjinal grafikte on altı hücre boşken, Şekil 2.2-b ayrıştırılmış grafikte ise yalnızca dört hücre boştur. Boş hücreler ne kadar azsa, kümeler o kadar sıkışık olacaktır. Bu yüzden işlemsel veri setlerinin kümelenmesi problemi, uygun sayıda ayrıştırma ile minimum doldurulmamış hücre sayısını elde etme problemine dönüştürülebilir.



Şekil 2.3. Örnek işlemsel veri setini eş-kümeleme ile ayrıştırılması [5]

İşlemsel verileri ve öğeleri ayrıştırmak için eş-kümeleme yöntemi[13] kullanılacak olursa, Şekil 2.3'deki iki düz çizgi tarafından ifade edilen sonuca ulaşılabilir. Açıkça, eş-kümeleme c öğesi ile d öğesi arasında bir eşleştirme kaybına neden olacaktır. Dolayısıyla işlemsel veri setinin nasıl uygun biçimde ayrıştırılacağı AKY algoritmasının temel meselelerinden biridir.

2.2. Kapsam Yoğunluğu (KY) Yaklaşımı

Kapsam yoğunluğu (KY) bölümlerin sıkışıklığının yalnızca doldurulmamış hücrelerle ölçülmesi olarak tanımlanmaktadır.[4] Kısacası, KY, dolu hücrelerin bir kümedeki ayrı ayrı öğeler ve işlemsel verilerin sayılarınca belirlenen toplam dikdörtgen alanı içindeki yüzdesidir.[4] Kümedeki bir öğenin ağırlığı o öğenin o kümeye katkısını göstermektedir. (i, j) hücresi için i=işlemsel veriyi (haberleri), j ise öğeleri (kelimeleri) temsil ettiği varsayılırsa $W_j = T = 1 / M_k$ ile j öğesinin ağırlığı hesaplanır, i işlemsel verinin ağırlığı ise $T_i = T = 1 / N_k$ ile hesaplanabilir. Bu verilere göre bir işlemsel veri kümesi için bu kümenin kapsam yoğunluğunu hesaplamak oldukça kolay ve basittir. Kapsam yoğunluğu o kümedeki işlemsel verilerin ağırlığı ile kümedeki tüm öğelerin gelme sıklığı çarpımı ve öğelerin ağırlık çarpımı ile bulunur. Formül 2.1'de formülün çıkarımı anlatılmıştır.[4]

$$KY(Ck) = Tk \times Sk \times Wk = Tk \times \sum_{j=1}^{M_k} frekans(I_{kj}) \times Wk$$
$$= \frac{1}{N_k} \times \frac{1}{M_k} \times \sum_{j=1}^{M_k} frekans(I_{kj}) \qquad (2.1)$$

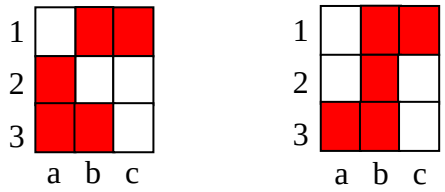
Küme içindeki öğelerin sayısı Mk, küme'nin öğe seti $Ik = \{Ik_1, Ik_2, \ldots, I_kM_k\}$, kümedeki işlemsel verilerin sayısı Nk, ve küme içindeki öğelerin frekans sayısı toplamı Sk olduğunda, Ck kümesinin kapsam yoğunluğu 2.2'deki formül gibi olacaktır.[4]

[13] ing: *co-clustering method*

$$KY(C_k) = \frac{S_k}{N_k \times M_k} = \frac{\sum_{j=1}^{Mk} frekans(I_{kj})}{N_k \times M_k} \tag{2.2}$$

2.3. Kapsam Yoğunluğu Yaklaşımındaki Problemler

Kapsam yoğunluğu, bir kümenin birlikteliğini yansıtır. Genel olarak, kapsam yoğunluğu ne kadar büyükse, bir kümenin işlemsel verileri arasındaki küme-içi benzerlikler o kadar fazladır. Ancak, KY ölçütünde kümedeki her bir öğe eşit derecede önemlidir. Örnek olarak grafik üzerinde göstermek gerekirse Şekil 2.4'de iki farklı kümede iki farklı işlemsel veri seti mevcuttur. Birinci küme {ab, a, bc} ve ikinci küme {ab, b, bc} veri setlerine sahiptir. Görüldüğü gibi iki küme'deki dağılım normal şartlarda birbirinden farklıdır.



Şekil 2.4. Aynı kapsam yoğunluğuna sahip iki küme [5]

Kümedeki öğelerin dağılımı Tablo 2.1'de incelendiğinde birinci kümede a ve b öğesinin yoğunluğunun fazla ikinci kümede ise b öğesinin tek başına yoğunluğunun fazla olduğu görülmektedir.

Öğe	Öğe	Frekans
Küme 1	a	2
	b	2
	c	1
Küme 2	a	1
	b	3
	c	1

Tablo 2.1. Birinci ve ikinci küme için öğelerin dağılımı

Tablo 2.1'de gösterilen dağılıma göre 2.1'deki formül kullanılarak kümelerin kapsam yoğunlukları hesaplandığında 2.3'deki kümelerdeki dağılımın farklı olmasına rağmen sonuçların aynı çıktığı gözlemlenmiştir. Bu nedenle, kapsam yoğunluğu bir kümenin frekans yoğunluğunu ölçmekte yetersizdir. "Farklı dolu hücre dağılımına sahip, ancak, aynı KY değerine sahip iki küme arasında fark var mıdır?" sorusu KY'nin en büyük problemini ortaya koymaktadır. Bu problem daha iyi bir dağılım belirlemek ve seçebilmek için, deneme yanılmaya dayalı bir kural geliştirmemizi gerektirir.

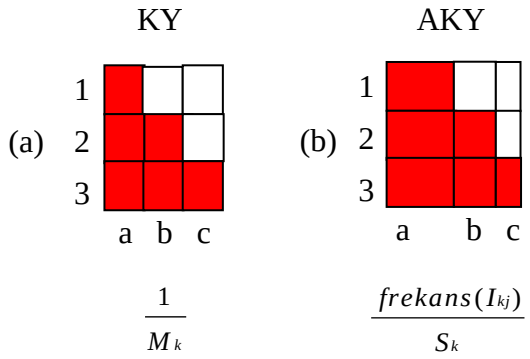
$$KY(Ck) = \frac{\sum_{j=1}^{Mk} frekans(I_{kj})}{N_k \times M_k} \qquad \begin{array}{l} \text{Nk: öğe sayısı}\{a,b,c\} = 3 \\ \text{Mk: işlemsel verilerin sayısı}\{1,2,3\} = 3 \end{array} \tag{2.3}$$

$$KY(Küme1) = \frac{2+2+1}{3\times 3} = \frac{5}{9} \qquad KY(Küme2) = \frac{1+3+1}{3\times 3} = \frac{5}{9}$$

Kapsam yoğunluğu, yüksek frekanslı öğelerden oluşan kümeler için sıkışıklık açısından aynı KY değerine sahip, ancak dolu-hücre dağılımı daha dağınık olan kümeye göre daha iyidir. Bu nedenle ağırlıklı kapsam yoğunluğu olarak adlandırılan yeni bir kavram tanımlanmaktadır.[5]

2.4. Ağırlıklı Kapsam Yoğunluğu (AKY)

Ağırlıklı kapsam yoğunluğu yaklaşımının kapsam yoğunluğu yaklaşımından en büyük farkı, bir kümede, frekansı (öğe geçme sıklığı) fazla olan öğelerin daha fazla ağırlık ile temsil edilmesidir. Bu tanım her bir öğenin ağırlığının kümeleme prosedürü sırasında sabit olmaması anlamını taşır ve ağırlık küme elemanlarının dağılımınca belirlenir. Bu yüzden öğe ağırlığı Wj Şekil 2.5(b) de görüldüğü üzere aynı olmaktan çıkar.



Şekil 2.5. KY ile AKY arasındaki öğe ağırlık farkı [5]

Öğe ağırlığı; W_j; her bir öğenin frekansının, o kümedeki tüm öğelerin frekansına oranı olarak, formül 2.4'de tanımlanmıştır. [5]

$$W_j = \frac{frekans(I_{kj})}{S_k} \qquad \sum_{j=1}^{Mk} W_j = 1 \tag{2.4}$$

Bölüm 2.2'de aktarılan 2.2 numaralı denkleme göre, işlemsel verilerin ağırlığı T ile sembolize edilmektedir. Bu değer değiştirilmeden, birinci kümenin ağırlıklı kapsam yoğunluğu formülü 2.5'de gösterilen şekilde çıkartılabilir. İkinci küme'nin ağırlıklı kapsam yoğunluğu değeri birinci kümeye göre daha iyidir. Bu sonuç da belirtilen kural ile tutarlıdır.

$$AKY(Ck) = \frac{\sum_{j=1}^{Mk} frekans(I_{kj})^2}{S_k \times N_k}$$

Sk: öğelerin frekans (geçme sıklığı) toplamı
Nk: işlemsel verilerin sayısı$\{1,2,3\}$ = 3

(2.5)

$$AKY(K1) = \frac{2^2 + 2^2 + 1^2}{(2+2+1)\times 3} = \frac{9}{15} \qquad AKY(K2) = \frac{1^2 + 3^2 + 1^2}{(1+3+1)\times 3} = \frac{11}{15}$$

2.5. Beklenen Ağırlıklı Kapsam Yoğunluğu (BAKY)

AKY-temelli kümeleme kriterini tanımlayabilmek için her bir kümedeki işlem sayısı dikkate alınmalıdır. K<N iken CK = C1, C2, ...CK, kümeleme sonuçları için aşağıdaki Beklenen Ağırlıklı Kapsam Yoğunluğu[14](BAKY) fonksiyonunu kümeleme kriteri fonksiyonu olarak tanımlanabilir. [5]

$$\begin{aligned} BAKY(C^K) &= \sum_{k=1}^{K} \frac{Nk}{N} \times AKY(C_k) \\ &= \sum_{k=1}^{K} \frac{\cancel{Nk}}{N} \times \frac{\sum_{j=1}^{Mk} frekans(I_{kj})^2}{S_k \times \cancel{N_k}} \\ &= \sum_{k=1}^{K} \frac{1}{N} \times \frac{\sum_{j=1}^{Mk} frekans(I_{kj})^2}{S_k} = \frac{1}{N} \sum_{k=1}^{K} \frac{\sum_{j=1}^{Mk} frekans(Ikj)^2}{S_k} \end{aligned} \qquad (2.6)$$

BAKY kümeleme algoritması BAKY kriterini maksimize etmeye çalışır. Kümelerin toplam beklenen ağırlıklı kapsam yoğunluğunun maksimum olması temeline dayanmaktadır. Ancak küme sayısının sınırlandırılmadığı durumlarda bir istisna ortaya çıkar; her bir bireysel işlemin bir küme olarak değerlendirildiği durum, tüm kümeleme sonuçlarının arasında en yüksek BAKY değerini verecektir. Bu nedenle küme sayısı ya açıkça belirtilmeli ya da parametreler tarafından belirlenmelidir.

[14] ing: *expected weighted coverage density*

BÖLÜM 3

3. İLGİLİ ÇALIŞMALAR

Bu bölümde öncelikle tez çalışması için gerekli olan temeller aktarılacaktır. Bölüm 3.1'de genel olarak XML ve RSS gibi belirli veriyi saklamak ve servisler arasında haberleşmek için kullanılan protokollerden bahsedilmiştir. Bölüm 3.2'de veri madenciliğinin genel kuralları anlatılmıştır. Bölüm 3.3'de doğal dil işleme ve kök bulma algoritmalarından bahsedilmiştir. Bölüm 3.4'de kümeleme algoritmaları ve Bölüm 3.5'de tavsiye sistemleri incelenmiştir.

3.1. Protokoller

3.1.1. Xml

Xml, diğer adı ile genişletilebilir işaretleme dili, hem insanlar hem bilgi işlem sistemleri tarafından kolayca okunabilecek dokümanlar oluşturmaya yarayan, W3C tarafından tanımlanmış bir standarttır.[2] Bu özelliği ile veri saklamanın yanında farklı sistemler arasında veri alışverişi yapmaya yarayan bir ara format görevi de görür. Xml, farklı türdeki ve yapıdaki dokümanların tanımlanması için kullanılan uluslararası bir standart olan sgml[15]'nin basitleştirilmiş bir alt kümesidir.[2]

Xml dosyalarının en önemli özelliği, geliştiricilerin özgürce ve ihtiyaçları doğrultusunda etiketler (imler) ve etiket özellikleri (nitelikler) kullanabilmesidir. Unicode karakter kodlamasını desteklemesi sayesinde Xml hemen hemen dünyadaki bütün dillerin desteklediği bir platform haline gelmiştir.[2] Bu nedenle günümüzde birçok yazılım, diğer yazılımlarla veri alışverişini Xml formatı üzerinden yapmaktadır. Ayrıca xml'i esas format olarak kullanan uygulamalara, veri tabanlarına rastlamak mümkündür. Microsoft'un geliştirdiği .NET teknolojisinde kullanılan veri nesneleri xml formatındadır.[2]

[15] ing: *standard generalized markup language*

Xml dokümanları ağaç veri yapısındadır. Bağımsız imler yapıyı oluştururken, içerik yerinin özelliği olarak ya da iki im arasında gösterilir. Yapıyla ilgili ayrıntılar belge tip tanımlayıcısı[16] ya da Xml şeması[17] adı verilen harici dokümanlar ile tanımlanır.[2] Çizelge 3.1'de örnek bir Xml doküman verisinin nasıl tanımlandığı gösterilmektedir.

Çizelge 3.1. Örnek XML dokümanı

```
<haberler>
  <haber id="1">
    <baslik> Benzine %20 Zam </baslik>
    <tanim> Benzin fiyatlarına %20 zam yapıldı. </tanim>
  </haber>
  <haber id="2">
    <baslik> ... </baslik>
    <tanim> ... </tanim>
  </haber>
</haberler>
```

3.1.2. Rss

Rss, genellikle haber sağlayıcıları, bloglar ve podcastler tarafından kullanılan, yeni eklenen içeriğin kolaylıkla takip edilmesini sağlayan özel bir Xml dosya formatıdır. Kullandığı dosya biçimleri .rss ve .xml'dir.[3] Rss kısaltması zaman içinde Rich Site Summary (Rss0.91), RDF Site Summary (Rss0.9 and 1.0), Really Simple Syndication (Rss2.0.0) olarak değişiklik göstermiştir.

İnternet kullanıcısı, Rss teknolojisi ile düzenli olarak içerik sunan sitelere abone olabilir ve çeşitli Rss istemcileri sayesinde içeriği takip edebilir. Site yöneticisi veya sahibi bu hizmeti sunmak için bir takım teknik düzenlemeler yapmalı ve uygun formatta xml'i RSS istemcisi talep ettiğinde göndermelidir.[3] Çizelge 3.2'de bir RSS haber doküman formatı örneği gösterilmiştir.

[16] ing: *document type definition*
[17] ing: *schema*

Çizelge 3.2. RSS 2.0 örnek haber dokümanı

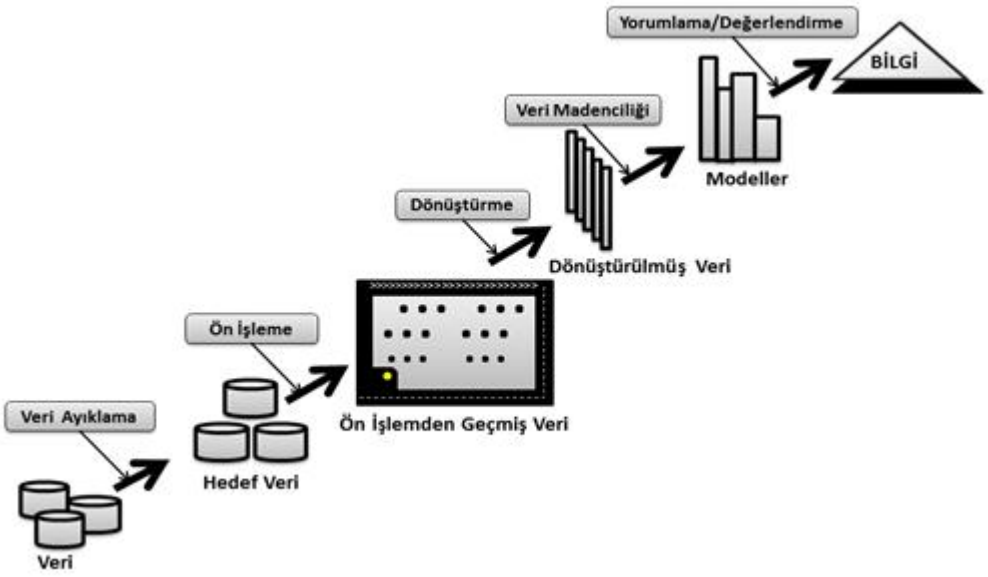
```
<?xml version=”1.0” ?>
<rss version=”2.0”>

<channel>
  <title>Yahoo News!</title>
  <link>http://rss.yahoo.com</link>
  <description>Yahoo News Rss News!</description>
  <language>en</language>
  <item>
    <title> Obama, Medvedev set to seal nuclear arms pact </title>
    <link> http://rss.yahoo.com/news/politics/344285 </link>
    <description> President Barack Obama will call Russian
      President Dmitry Medvedev on Friday to discuss a landmark
      nuclear arms reduction treaty, the White House said,
      signaling that formal announcement of an agreement was
      imminent.
    </description>
    <pubDate> Tue, 14 Jun 2011 03:33 GMT </pubDate>
  </item>
  <item>
    <title> ... </title>
    <link> ... </link>
    <description> ... </description>
    <pubDate> ... </pubDate>
  </item>
</channel>
```

Rss yöntemini destekleyen sitelerin hazırladıkları Xml biçimli dosyalara birçok programla erişmek mümkündür. Xml okuyucusu olan bu programlar, web gezgini veya e-posta istemcisi olabileceği gibi sadece rss içeriği izlemek için hazırlanan masaüstü programları da olabilir. Bu tip programlara rss okuyucu programlar denir. [3] Rss okuyucu programlar, listesine eklenmiş sitelerdeki değişen içerikten, siteye gitmeden haberdar olmamıza yararlar. Tez çalışmasında, internet tabanlı Rss verilerini çekmek için bir Rss okuyucu tasarlanmıştır. Kullanıcı Rss haber kategorilerini ekleyebilir ve yönetebilir. Sistem, eklenen dinamik kategoriler üzerinden haber verilerini çekerek kullanıcıya kategorili bir şekilde göstermektedir.

3.2. Veri Madenciliği

Geleneksel olarak kurumlar kendi verilerini depolamak ve bunlarla analiz ve inceleme yapmak amacıyla birçok farklı bilgi kaynağını kullanırlar. Veri toplamak ve depolamak gelişen teknolojiyle birlikte daha kolay, daha ulaşılabilir ve daha sık başvurulan bir olgu haline gelmiştir. Bunun sonucunda inanılmaz bir hızla toplanan devasa veri bankaları ortaya çıkmıştır. Mevcut istatistiki yöntemler ve veritabanı inceleme araçları söz konusu verileri incelemekte yetersiz hale gelmeye başlamıştır. Diğer yandan bilgi keşfi süreci[18] olarak adlandırılan yöntem devasa veri tabanlarını incelemekte önemli ölçüde başarılı olmuştur. Veritabanında bilgi keşfi[19] (VTBK) [10], veri tabanlarındaki bilginin elde edilebilmesi için gerekli süreci ifade etmek üzere oluşturulmuştur. Şekil 3.1 VTBK sürecinin temel aşamalarını özetlemektedir. Veri ayıklama, ön-işleme ve dönüştürme süreçleri ham verilerin işlenmek üzere hazırlanması için gerekli aşamalardır. [10]



Şekil 3.1. Veri tabanlarında bilgi keşfi süreci (VTBK) [10]

[18] ing: *knowledge discovery process*
[19] ing: *knowledge discovery in databases*

Veri işleme, VTBK sürecinin temel adımı olup, kalıplar, kurallar, kısıtlar ve düzenlilikler gibi karmaşık, zımni, önceden bilinmeyen ve potansiyel olarak yararlı bilgilerin keşfi olarak ifade edilebilir.[11] Veri işleme, istatistik, beritabanı, görüntüleme ve yapay zeka[20] (YZ) ile özellikle makina öğrenmesi (MÖ) ve doğal dil işleme (DDİ)'nin alt alanları gibi farklı araştırma topluluklarının buluşma noktası olarak değerlendirilmektedir.[11] Veri işleme bu bağlamda bir ayrıntılar öbeği içerisinden daha önceden bilinmeyen yapılar ve ilişkileri ortaya çıkarabilmek amacıyla büyük veri setlerinin analizin yapılabilmesini sağlayacak bütün yöntem ve teknikleri içeren bir kavramdır.[11] Bu tür bilgilerin filtrelenmesi, ön-işlemeye tabi tutulması/hazırlanması ve sınıflandırılması karar almada ve strateji oluşturmada değerli bir yardımcı olmaktadır. Veri işlemede kullanılan birçok farklı yöntemden bazıları ilişkilendirme kuralına göre işleme, ardışıklık kuralına göre işleme, sınıflandırma, kümeleme, ayırıcı çözümleme ve tahmin, bağımlılık modelleri ve sınıflayıcılar için kural yaratma olarak sayılabilir.[11]

Tez çalışmasındaki temel amaç, küme sayısını otomatik olarak belirleyecek ve gerçeğe en yakın kümelemeyi sağlayacak yeni kümeleme teknikleri geliştirmek olarak ifade edilebilir. Dolayısıyla bu bölümün kalan kısmında önce bu teknikler gözden geçirilecek, ardından da var olan kümeleme yaklaşımları anlatılarak bölüm sonlandırılacaktır.

3.3. Metin Madenciliği

Metin madenciliği[21] yeni ve daha önce bilinmeyen bilginin farklı yazılı kaynaklardan otomatik olarak bilgi çıkarımı yolu ile keşfidir.[9] Metin madenciliğinin ana fikri, bilginin daha konvansiyonel deneyler vasıtası ile daha ileri biçimde araştırılacak yeni olgular ya da yeni hipotezler oluşturmak üzere bir araya getirilmesidir.[9] Metin madenciliği, web aramalarında alışkın olduğumuz sonuç çıkarımından farklıdır. Web aramalarında, kullanıcı tipik olarak halihazırda bilinen ve bir başkası tarafından

[20] ing: *artificial intelligence*
[21] ing: *text mining*

yazılmış olan bir şeyi aramaktadır. Sorun amacına uygun bilgiyi elde etmek için halihazırda sizin ihtiyacınızla ilgisi olmayan tüm materyali bir kenara itmektir.

Metin madenciliğindeki amaç ise şimdiye kadar bilinmeyen bilgiyi, henüz kimsenin bilmediği ve yazılı hale getiremediği bir şeyi keşfetmektir. Metin madenciliği, geniş veri tabanlarından ilginç desenler[22]bulmaya çalışan ve veri madenciliği olarak adlandırılan alanın bir çeşididir. [9] Bu konu için tipik bir örnek; müşteri satın alma modellerinin raflarda hangi ürünlerin birbirine yakın biçimde yerleştirileceği, ya da hangi ürünler için kupon verileceği vb. konuları tahmin etmek üzere kullanılmasıdır. Örneğin, eğer bir el feneri alıyorsanız muhtemelen bu fenerle birlikte pil de satın alacaksınızdır. Bu örneğe istinaden, tez kapsamında geliştirilen uygulama okuyucunun okuduğu haberlerin yakınlık derecelerine göre aynı kümelere konması ve okuyacağı haberleri bu kümelere göre tahmin edip, tavsiye etmesine dayanmaktadır.

Metin madenciliği teknikleri dört temel kategoriye ayrılır:

1) Sınıflandırma[23]
2) Birliktelik analizi[24]
3) Bilgi çıkarımı[25]
4) Kümeleme analizi[26]

Sınıflandırma işlemi nesnelerin daha önceden bilinen sınıflara ya da kategorilere dahil edilmesidir.[9] Birliktelik analizi ise sıklıkla birlikte yer alan ya da gelişen sözcük ya da kavramların belirlenmesi ve böylece doküman içeriğinin ya da doküman kümelerinin anlaşılmasını amaçlamaktadır.[9] Bilgi çıkarım teknikleri ile dokümanların içerisindeki yararlı veri ya da ifadeler bulunmaya çalışılmaktadır.[9] Kümeleme analizi, doküman kümelerinin temelini oluşturan yapıların keşfedilmesi amacıyla uygulanmaktadır.[9]

[22] ing: *pattern*
[23] ing: *classification*
[24] ing: *association analysis*
[25] ing: *information extraction*
[26] ing: *clustering*

3.4. Doğal Dil İşleme

Doğal dil işleme[27], (DDİ) Türkçe, İngilizce, Almanca, vb. gibi doğal dillerin işlenmesi ve kullanılması amacı ile araştırma yapan bilim dalıdır. Bilgisayar ortamında doğal dil işleme ise dilin kurallı yapısının çözümlenerek anlaşılması veya yeniden üretilmesidir. [13] Bu çözümlemenin insana getireceği kolaylıklar, yazılı dokümanların otomatik çevrilmesi, soru-cevap makineleri, otomatik konuşma ve komut anlama, konuşma sentezi, konuşma üretme, otomatik metin özetleme, bilgi sağlama gibi birçok başlıkla özetlenebilir. Bilgisayar teknolojisinin yaygın kullanımı, bu başlıklardan üretilen uzman yazılımların gündelik hayatımızın her alanına girmesini sağlamıştır.[13] Örneğin, tüm kelime işlem yazılımları birer imla düzeltme aracı taşır. Bu araçlar aslında yazılan metni çözümleyerek dil kurallarını denetleyen doğal dil işleme yazılımlarıdır.[13]

Batı dillerinde SAPI (Microsoft şirketinin konuşma sentezleyici üretmek amacı ile satışa sunduğu geliştirici program) tabanlı konuşma sentezleyici bileşenleri, yazılımcıların çoklu ortam[28] sunuları hazırlamaları için hizmete sunulmuştur.

Konuşma ve komut anlama yazılımları ise gelecekte insan ve bilgisayar arasındaki klavye, fare gibi veri girişi aygıtlarını ortadan kaldıracak yazılımlardır. Bu gelişmeler makine-insan iletişiminde yeni ve devrimci değişimlere neden olacak ve bilgisayarların daha çok insan tarafından kabul görmesine yol açacaktır.

3.5. Kök Bulma Algoritmaları

Türemiş ya da değişmiş bir kelimenin orijinal haline (köküne) indirgeme işlemlerine kök bulma[29]denilir. Kök bulma işleminin sonucunda elde edilen kelime, dilin morfolojik yapısıyla aynı olmak zorunda değildir; genellikle aynı köke eşleştirilen alakalı kelimeler yeterli olmaktadır. [16]

[27] ing: *natural language processing*
[28] ing: *mutimedia*
[29] ing: *stemming*

Dilin morfolojik yapısına bağlı kalan takılar ile kök bulmada kullanılan çeşitli algoritmalar bulunmaktadır. İngilizce'de kullanılan en popüler kök bulma algoritması Porter Stemming [12] algoritmasıdır. Bu algoritmada kök bulma işlemi takıların atılması prensibine dayanmaktadır. Özellikle herhangi bir kelimenin kökünü bulmakta yararlı olan bir işlem gerçekleştirir. Bir tür tarayıcı gibi çalışarak, kelimeyi tarar ve kelimenin takılarını kendi kaydettiği takılarla eşleştirir, örneğin "CONNECTED", "CONNECTING", "CONNECTION", "CONNECTIONS" kelimelerinin hepsi "CONNECT" fiilinden türemiştir. Takılara bakarsak; -ED (geçmiş zaman fiil takısı) -ING (şimdi zaman fiili ya da isim), -ION(isimleştirme takısı), -IONS(isimleştirme çoğul takı) gözlemlenmektedir. Porter Stemming algoritması [12] kendisinde barındırdığı bu takıları atarak kelimeyi kök haline getirir. Ancak bu algoritmada bazı sorunlar mevcuttur. Bu sorunlardan en önemlileri İngilizcedeki bütün takıların sistemde yer alamaması ve özel kelimeleri ayıramamasıdır. Örneğin "airliner" özel kelimesi porter stemming algoritmasına sokulduğunda, algoritma kelimenin ne olduğunu bilmediği için ilk olarak kelimenin ekine odaklanacaktır. Kelimenin sonundaki "-er" ekini takı olarak kabul edip atarak "airlin" köküne ulaşacaktır.

Dilin morfolojik yapısına bağlı kalan kök bulma algoritmalarının yanı sıra kelimelerin anlamlarını, içeriğini inceleyen ve buna göre kök bulma işlemlerini gerçekleştiren algoritmalar da kullanılmaktadır. Bu tip algoritmalara İçeriğe Duyarlı Kök Bulma[30]algoritması denir. [17] İngilizce dili için yapılmış birçok içeriğe duyarlı kök bulma algoritması bulunmaktadır. Bu algoritmaların hepsi temelinde açık kaynak doğal dil işleme kütüphanelerini kullanmaktadırlar.

3.6. Açık Kaynak Doğal Dil İşleme Kütüphanesi

İngilizce adı ile açık kaynak doğal dil işleme kütüphanesi olarak bilinen OpenNLP, doğal dil işleme ile proje geliştirenler için bir açık kaynak kod grubunun adıdır. [14] Open NLP Kütüphanesi cümle dallandırıcısı, simgeleyici, sözcük türleri etiketçisi, ad

[30]ing: *context sensitive stemming*

öbekleri yığınları gibi özyinelemesiz, sözdizimsel ek açıklamaları bulmak için kullanılan yığın ayırıcısı, ayrıştırmacı ve ad bulucu gibi çeşitli bileşenler içerir.[14] Open NLP kütüphanesi kullanılarak WordNet::Similarity [17] gibi birçok uygulama geliştirilmişidir. Open NLP kelimelerin anlamlarını cümle bazında analiz ederek çıkartmaktadır. Bu bölümün ilerleyen kısımlarında, açık kaynak doğal dil işleme kütüphanesinin bazı araçlarından bahsedilecektir. Bölüm 3.6.1'de cümle ayrıştırma, 3.6.2'de cümleleri simgeleme, 3.6.3'de cümleden isim çıkarımı anlatılmıştır.

3.6.1. Cümle Ayrıştırma

Bir metin paragrafımız varsa, onu cümlelere bölmenin basit ve sınırlı yolu, bir dizgiler dizisi elde etmek için input.Split('.') kullanmak olacaktır. Bu komutu, input.Split('.', '!', '?') şeklinde genişletmek, daha çok vakanın doğru bir şekilde ele alınmasını sağlayacaktır. Fakat bu yöntem ile bu karakterler sadece cümle sonlarında alınıp alınmayacağı ayrımı yapılamaz. Örneğin bir cümle bitmediği halde ortalarında nokta (.), ünlem (!) yada soru işareti (?) gibi cümle sonu noktalama işaretleri olabilir.

Çizelge 3.1'deki metin düşünülürse; bu girdide Split metodunu kullanmakla, aslında iki elementli bir dizi olan metin, beş elementli bir diziyle sonuçlanacaktır. Bunu '.', '!', '?' karakterlerinin her birini kesin cümle sonu işaretleri olarak değil potansiyel işaretler olarak ele alarak yapabiliriz. Verilen metin taranıp, her defasında bu karakterlerden birine gelindiğinde, bunun cümle sonunu işaret edip etmediğine karar verme yoluna ihtiyaç duyulmaktadır. Bu durum maksimum entropi modelinin yararlı olduğu yerdir. Olası cümle sonu konumlarıyla ilgili bir dizi öngörü üretilir. Olası cümle sonu işaretlerinin önündeki veya arkasındaki karakterlerle ilgili çeşitli özellikler, bu öngörüler dizisini meydana getirmek için kullanılır. Bu öngörüler dizisi daha sonra MaxEnt modelinin karşısında değerlendirilir. Eğer, en iyi sonuç bir cümle arasını işaret ediyorsa, o zaman cümle sonu işaretinin konumunu içeren ve bu konuma kadar olan karakterler yeni bir cümle olarak ayrılırlar.

Örnek Metin	Mr.Bachelet says the tsunami killed 700 people. Damages are nearly $30.1 billion.
Ayrıştırma Sonrası	
Cümle 1	Mr.Bachelet says the tsunami killed 700 people.
Cümle 2	Damages are nearly $30.1 billion.

Tablo 3.1. İki cümleli örnek bir haber metni

3.6.2. Cümleleri Simgelemek

Bir cümleyi yalnız bıraktıktan sonra, ona bazı doğal dil işleme teknikleri olan sözcük türleri etiketçisi ya da tam ayrıştırma uygulanmak istenebilir. Bu işlemdeki ilk adım, cümleyi "simgelere", yani kelimeler ve noktalama işaretlerine bölmektir. Yine, ayrıştırma metodu[31]bunu tam olarak başarabilmek için tek başına yeterli değildir. Bunun yerine "EnglishMaximumEntropyTokenizer" nesnesinin "Tokenize" metodunu kullanabiliriz. Bu sınıf ve OpenNLP.Tools.Tokenize ad alanındaki ilgili sınıflar, cümleleri simgelemek için kullanılmaktadır. [14]

Simgeleyici, kısaltmalardan oluşan kelimeleri bölmektedir. Örneğin, "don't" kelimesini "do" ve "n't" olarak böler, çünkü "do" kelimesinin bir fiil ve "n't" = "not" ın kısaltması olarak, kendisinden önceki "do" fiilini tanımlayan bir zarf olarak algılandığı diğer doğal kaynak araçlarına geçirmek üzere tasarlanmıştır. Verilen bir örnek cümle, metod'dan geçirildiğinde Tablo 3.2'deki gibi kelime dizisi oluşacaktır.

Cümleler	
Cümle 1	Mr.Bachelet says the tsunami killed 700 people.
Cümle 2	Damages are nearly $30.1 billion.
Simgeleme Sonrası	
Kelimeler 1	Mr. | Bachelet | says | the | tsunami | killed | 700 | people
Kelimeler 2	Damages | are | nearly | $30.1 | billion

Tablo 3.2. Örnek bir metin cümleleri üzerinden kelime simgeleme

[31] ing: *split method*

3.6.3. Cümleden İsim Çıkarımı

Bir cümle içerisinden isim çıkarımı cümle içindeki varlık sınıflarının tanılamasını -örneğin kişi adlarını, yerleri, tarihleri vb.- ima etmek için OpenNLP kütüphanesi tarafından kullanılan terimdir. [14] Ad bulucu, yedi maksimum entropi modeli – tarih, yer, para, organizasyon, yüzde, kişi ve zaman- tarafından temsil edilen varlık çeşitleri ile gösterilir. [14]Diğer varlık sınıflarını bulmak için SharpEntropy kütüphanesini kullanan yeni modeller geliştirmek olasıdır. Bu durum, algoritma geliştirilen verinin kullanımına bağlı olduğu için ve "kişi" ya da "yer" gibi bir kategoriye gelebilecek çok fazla simge olduğu için kesin olmaktan çok uzaktır. Sonuç, varlıkların nerede bulunduğunu gösteren Xml benzeri etiketleri olan biçimlenmiş bir cümledir. Böylece istenilen isimler çıkartılabilir. Tablo 3.3'de gösterilen haber örneğinde, haber içindeki isimlerin çıkarımı gösterilmiştir.

<table>
<tr><td>Örnek Haber</td><td colspan="3">The earthquake and tsunami that struck Chile last month killed 700 people and caused damages of nearly $30.1 billion, according to the government. The ground hasn't stopped shaking.</td></tr>
<tr><td>Ayrıştırılmış Cümle 1</td><td>The <noun> earthquake</noun> and <noun>tsunami<noun> that struck <location>Chile</location><date>last month</date>killed 700 <pnoun>people</pnoun> and caused damages of nearly <money>$30.1 billion</money>, according to the <noun>government</noun></td><td>earthquake
Tsunami
Chile
last month
People
$30.1 billion
governm nt</td><td>Tekil isim[32]
Tekil isim
Yer[33]
Tarih[34]
Çoğul isim[35]
Para[36]
Tekil isim</td></tr>
<tr><td>Ayrıştırılmış Cümle 2</td><td>The <noun>ground</noun> hasn't stopped shaking.</td><td>Ground</td><td>Tekil İsim</td></tr>
</table>

Tablo 3.3. Örnek bir metin üzerinden cümlelerden isim çıkarımı

[32] ing: *singular noun*
[33] ing: *yer*
[34] ing: *date*
[35] ing: *plural noun*
[36] ing: *money*

3.7. Kümeleme

Kümeleme literatürde yaygın olarak incelenmiş veri işleme tekniklerinden biridir. Veri setini oluşturan gözlemlerin, küme adı verilen anlamlı alt gruplara ayrılmasına dayanır. Kullanıcılara veri setinin yapısını anlamada yardımcı olan bir yöntemdir. Kümeleme analizi özellikle açıklayıcı desen-analizi, gruplama, karar alma ve makine öğrenmesi gibi alanlarda, belge çağırma, imaj segmentasyonu ve desen sınıflandırması gibi işlemlerde yararlı bir araçtır [20]. Kümeleme bir tür güdümsüz sınıflandırma[37] olarak nitelenebilir. Yani kaç sınıf olduğu ve her bir sınıfın özelliklerinin neler olduğu bilinmemektedir ve herhangi bir örneğe dayanmaz. Söz konusu kümeler istatistiki olarak ya da nöral ve sembolik güdümsüz tümevarım yöntemleriyle oluşturulurlar. Değişik nöral ve sembolik yöntemler, kabul edilebilir özellik değeri tiplerine (nümerik, nominal ve yapısal), küme tasarımına ve küme organizasyonuna (hiyerarşik ya da düzlemsel) göre birbirinden ayrılırlar [21].

Tipik kümeleme bileşenleri:

1. Örüntü tasarımı[38]
2. Örüntü yakınlığının tanımı
3. Kümeleme ya da gruplama
4. Veri soyutlaması
5. Kümeleme değerlendirmesi

olarak özetlenebilir. [20] Bu bileşenler aşağıdaki şekilde tanımlanabilirler: örüntü tasarımı, sınıf sayısı, mevcut örüntüler, kümeleme algoritması için geçerli özelliklerin tip ve ölçekleri ile ilişkilidir. Özellik çıkarma[39] kümeleme işiyle ilgili özellik setinin bulunmasıdır. Örüntü yakınlığı kümelemede kullanılan uzaklık fonksiyonunu ifade eder. Uzaklık fonksiyonlarının bir kaç farklı anlamı olabilir. Kümelemede kullanılan yakınlık fonksiyonlarına ilişkin iyi bir inceleme [20] de bulunabilir. Gruplama aşaması da bir kaç şekilde ele alınabilir. Verilerin kümelenmesi katı (her bir örnek en fazla bir küme için üyelik değeri olarak 1

[37] ing: *unsupervised classification*
[38] ing:*pattern representation*
[39] ing: *feature extraction*

değerini alır (yani o kümeye aittir) ve diğer kümeler için sıfır değerini alır), ya da belirsiz (her bir örnek küme içinde 0 ile 1 arası bir üyelik değeri alır) olabilir.[20]
Genel olarak, kümeleme sürecinin tamamı belli bazı temel aşamaları içerir.[10] Özellik seçimi, belirli bir kümeleme algoritmasının çalıştırılması, sonuçların doğruluğunun sınanarak yorumlanmasıdır. Ancak, literatürde tanımlanan kümeleme yaklaşımları çoğunlukla kümeleme algoritması ve doğrulama süreci üzerine yoğunlaşır.[20] Örneğin katı kümeleme için N adet örnek ve k adet küme varsa, olası toplam kümeleme sayısı 107 olur.

$$\frac{1}{k!}\sum_{l=1}^{k}\binom{k}{k}(-1)^{k-1_{l}m} \quad \approx k^{n}/k! \tag{3.1}$$

Bölümlemeli algoritmalar n sayıda nesneyi önceden belirlenmiş k kadar kümeye dağıtır. Bunu yapabilmek için bölümlemeli algoritmalar genellikle rassal olarak seçilmiş bir başlangıç dağılımını alıp, nesneleri bir kümeden diğerine tekrar tekrar kaydırarak sonucun kalitesini optimize etmeye çalışır. Ancak, bölümlemeli algoritmalar, başlangıç seçimlerine hayli duyarlı olduklarından yerel optimumlara yatkındırlar. Genelde bilgiye ulaşılamadığından gözlenen veri setinden yola çıkarak küme sayısının tahmin edilmesi bir zorunluluk haline gelmektedir. Bu problem, küme geçerliliği ile ilintilidir ve esas itibarıyla çözümsüzdür. Bölümlemeli algoritmalara örnek olarak k-ortalama[40][22] verilebilir. K-ortalama kümeleme bilinen bir bölümlemeli algoritma tekniğidir.[22] Küme sayısı, k, girdi olarak verilir ve algoritma veri setini k sayıda ayrık alt kümeye böler. Daha sonra her kümenin merkezindeki elemanın diğer merkez elemanlarla arasındaki uzaklığın karelerinin toplamını minimize ederek kümelerin optimizasyonunu gerçekleştirmeye çalışır.[22] Bu durumda yerel minimumlar dışında sonuç alınamaması durumu ortaya çıkabilir. Literatürde k-ortalama yönteminin farklı varyasyonları tanımlanmıştır. Bunlar üç kategoride gruplandırılabilir: Başlangıç bölünmesinin saptanması, süreç içindeki ayrışma ve birleşme operasyonlarına izin verilmesi ve farklı şekillerdeki kümeleme sonuçları için farklı kriterlerin varlığı k-ortalama yöntemine ilişkin yeni bir

[40] ing: *k-means*

çalışmada karma bir veri seti için özellik ağırlıklandırması yer almaktadır, başka bir deyişle veriler bölümlenirken aynı zamanda özelliklere göre ağırlıklandırılmaktadır.[22] Örneğin, Modha ve Spangler, k-ortalama için bir başka değişken-ağırlıklandırma yöntemi geliştirmişlerdir.[23] Bu yöntem değişken ağırlıklarını en iyi kümelemeyi bulmak üzere optimize etmeye dayanır: bunun için kümeler arası ortalama çarpıklıkların kümeler içi ortalama çarpıklıklara oranı minimize edilir.[23] Temel olarak kullandıkları yöntem, bilgi teorisinde var olan gürültü azaltma yöntemi ile özellik ağırlıklandırma algoritması için Fisher'in doğrusal ayrılabilirlik kriterini bir arada kullanmaya dayanır.[23] Bir başka çalışma W-k-ortalama yöntemi olarak ifade edilebilecek ve değişkenlerin kümeleme içindeki önemlerine göre otomatik olarak ağırlıklandırılmalarını sağlayan bir yöntemi içermektedir.[24] Bu yöntem özellik ağırlıklarındaki artış/azalışlardan kaynaklanan hatanın minimize edilmesi için Lagrange çarpanını kullanmaktadır.[24] Bu çalışma daha sonra yalnızca sınıf sayılarının gerçek sınır etiketlerinin sayısı olarak kullanılması yoluyla veri setinin doğruluğunu sınayacak şekilde genişletilmiştir.[24]

Hiyerarşik kümeleme algoritması içiçe geçmiş kümeler arasında bir hiyerarşik seri oluşturur.[25] Çıktı grafiksel olarak bir dendrogram ile gösterilebilir. Hiyerarşik kümelemede ayrıştırma-birleştirme kriteri vardır.[25] Veri seti tekrarlanan biçimde alt gruplara ayrılır. Hiyerarşik kümeleme algoritmalarında iki yaklaşım söz konusudur: yığılımcı ve bölücü yığılımcı kümeleme[41] birbirine en yakın küme çiftlerinin birleştirilmesiyle alttan üste doğru işlerken bölücü kümeleme[42], bütün veri setini tek bir küme olarak varsayıp her bir aşamada verileri ayrıştırarak her bir kümede tek bir gözlem kalana dek kümeleme işlemine devam eder.[25] Hiyerarşik kümeleme veri setinin grafiksel gösteriminde doğal bir yol sağlıyor olsa da sağlamlığı tartışılır ve hesaplaması oldukça karmaşıktır. Ayrıca kümeleme sürecinin açgözlü yapısı kötü kararların daha sonraki aşamalarda düzeltilebilmesini engeller.[25]

[41] ing: *agglomerative clustering*
[42] ing: *divisive clutering*

CURE, hiyerarşik kümelemenin iyi örneklerinden ikisidir.[25] Grafiksel teorik kümeleme G = (V, E) olarak tanımlanan ve her bir gözlemin bir tepe noktasını gösterdiği yakınlık grafiği yoluyla yapılabilir. Köşeler ya bütün tepe noktaları arasında yer alırlar ve yakınlıklarına göre ağırlıklandırılırlar ya da belli bir eşiğe göre yerleştirilirler. Kümeleme en az sayıda dilimin veya en fazla sayıda gruplaşmanın bulunması şeklinde ifade edilebilecek bir grafik problemidir.[25] Grafiksel-teorik kümelemenin bir örneği CAST [26]'dır. CAST uyum derecesini temel alarak, her biri bir kümeye tekabül eden gruplaşmaları bulmaktadır.[26] Kullanıcı tarafından tanımlanmış küme sayısına bağımlı değildir ve bu nedenle uç değerleri konusunda başarılıdır.[26]

Yoğunluğa dayalı kümeleme yoğun bölgeleri genişletmeyi ve ayrıştırmayı yoğunluğa göre yapmayı amaçlar.[27] Etkin olarak yüksek yoğunluklu, düşük yoğunluklu ve gürültülü verileri ayırt eder. Yüksek bağıntılı özellikler içeren verileri analiz etmekte ve kümeler ile küme elemanları arasındaki ilişkileri göstermekte hayli başarılıdır. Ancak ciddi hesaplama güçlükleri vardır. Yoğun bölgelerin ayrışmasını kontrol edecek kullanıcı tanımlı parametreler gerektirir. Kümeleri, veri uzayında birbiriyle bağlantılı alanlar olarak konumlar.[27]

Modele dayalı kümeleme, küme yapısını modellerken istatistiki bir çerçeve kullanır.[27] Beklenti maksimizasyonu kullanarak kümelerin yoğunluk fonksiyonlarına ait parametreleri tahmin eder. Bu yolla her bir kümeye ait veri değerinin tahmini olasılığını verir. Diğer yandan, belli bir dağılıma uygun verilere gereksinim duyar. Gen ifadesi verileri gibi az örnekli ve yüksek boyutluluğa sahip veriler söz konusu olduğunda bu önerme daha fazla geçerli hale gelir.[27]

Ağ temelli kümeleme[43], alt-uzaylardaki değişkenlerin değerlerini sınırlayarak, kümeleme işlemini kuantal uzayda gerçekleştirir.

Öz organize harita[44] (ÖOH) [28] tek tabakalı nöral ağları girdi olarak kullanıp çıktı olarak nöronları sunan bir yapıya sahiptir. İki boyutlu bir ağ üzerindeki nöronların

[43] ing: *grid-based clustering*

her biri ağırlıklandırılmış bir referans vektörü ile ilişkilendirilir ve her gözlem kendisine en yakın referans vektörü yardımıyla bir nöron ile eşleştirilir. Referans vektörler başlangıçta rassal olarak ağırlıklandırılabilir.[28] Her gözlem için kendisine en yakın ağırlığa sahip bir vektör seçilir. Seçilen vektör daha sonra dağılıma en uyumlu hale gelecek şekilde güncellenir.[28] Tüm gözlemler için bu egzersizler tamamlandıktan sonra her bir gözlem çıktı nöronları ya da kümelerle eşleştirilirler. Sezgisel olarak çok boyutlu verileri eşleştirir ve ayrıca benzer kümeleri birbirine yakın yerleştirir. Gürültüye k-ortalama yöntemine göre daha az duyarlıdır. Ancak, kullanıcının küme sayısını ve nöron haritasındaki ağ[45] yapısını girmesini gerektirir. Ayrıca fazla sayıda ilgisiz gözlem içeren veri setleri için etkin bir yöntem değildir.[28]

3.7.1. Kümelemedeki Bazı Anahtar Konular

Kümeleme, literatürde yoğun olarak incelenmiştir ve geniş bir alanda uygulama ve geliştirme imkanı olduğundan halen aktif olarak incelenmeye devam edilen bir araştırma konusudur. Fakat literatürde açıklanan ve yukarıda özetlenen yaklaşımlar beklentileri tam olarak karşılamamaktadır. Küme sayısının otomatik tahmini, verilen data setinden en doğal kümelemenin üretilmesi, büyük ve yüksek boyutlu veri setlerinin ölçeklendirilmesi gibi konular kümelendirme algoritmalarının geliştirmesinde hala dikkat çeken anahtar kavramlardır. Küme sayısının tahmini, önceden belirlenmiş bir değere zorlanmasından daha gereklidir. Bu kümelemeye daha doğal bir yaklaşım olmalıdır, çünkü kümeleme kontrol edilmeyen bir öğrenme sürecidir ve genellikle küme sayısının önceden bilinmesi mümkün değildir. Bu kümeleme ve sınıflandırma arasındaki en önemli farktır. İkinci fark ise önceden tanımlanmış sınıflar ile kontrol edilen öğrenme süreçleridir. Daha ötesi, k-ortalama yöntemini farklı küme sayıları için çalıştırmak ve elde edilen alternatif sonuçlar arasından en uygun çözümü seçmek mümkündür. Fakat bu tür bir yaklaşım başlangıçtaki ağırlık merkezinden etkilenebilir, araştırmacılar hala k-ortalama

[44] ing: *self organized map(SOM)*
[45] ing: *grid*

yönteminin başlangıç ağırlık merkezini tahmin etmek için teknikler geliştirmek üzere çalışmaktadır.[23]

Yukarıda vurgulandığı üzere, diğer birçok araştırmacı kümelendirmede geleneksel algoritmalardan yararlanmaktadır. Sonuç olarak, iyi bir kümelendirme algoritması farklı küme sayılarında nasıl çalışır, nasıl ölçeklenebilir ve küme sayısı nasıl tahmin edilir gibi sorular kümelendirme algoritmalarına ilişkin devam eden çalışmaların temel soruları olarak ilgi çekmeye devam etmektedir.

3.7.2. Küme Doğrulanması

Literatürde birçok kümelendirme algoritması tanımlanmıştır ve her biri verileri değişik yöntemlerle ayrıştırmaktadır. Ancak, kümelendirmenin sonunda, süreçten elde edilen optimal küme sayısını ve kümelendirmenin yeterince iyi olup olmadığını kendimize sormalıyız. Küme doğrulama endeksi kullanarak çeşitli girdi parametrelerine göre, kümeleme sonuçları sıralaması oluşturulur.[29] Küme doğrulaması, temeldeki verileri en uygun ayrıştırma yöntemini sorgulamak için kullanılır. Burada amaç optimal küme sayısını bulmaktır. Eğer veri seti biliniyorsa, sınıflar içerisinde hangi adayın belirleyici olduğunu anlamak üzere, her bir alternatif adayı farklı parametrelerle değerlendirerek optimal küme sayısını kontrol etmek için kullanılır.[29] Bu içsel, dışsal ve göreceli değerleme endeksleri çalıştırılarak yapılır.[29] İlk iki kategori, uzun sayısal işlemlerle dayalı istatistiksel testleri gerektirmektedir. İçsel değerleme endeksleri[46] verilerin miktarını ve özelliklerini kullanır.

Her değerleme endeksi alternatif sonuçları kontrol etmek, değerlemek ve sıralamak için, kümeleme parametlerini çalıştıran özel formüllerden yararlanır.[29] Ancak, endeksleri sınıflandıran ve eldeki veri setlerine uygun özel bir endeks öneren bir çalışma henüz bulunmamaktadır. Bu tezde sunulan çalışma değerleme sonuçlarına

[46]ing:*internal validity indices*

güveni arttırmak ve veri setlerine uygun bir endekslemeye yardımcı olabilecek niteliktedir.

3.8. Tavsiye Sistemleri

Tavsiye sistemlerinin temelleri yakınsama teorisi, bilişsel sistemler, bilgi toplama, tahmin yöntemleri ve yönetim bilimine dayanmaktadır. 1990'ların ortasından itibaren öneri sistemleri kendi başına bir araştırma konusu olmuştur.[18] Resnick ve Varian tavsiye sistemleri, bireylere farklı tercihler arasından ihtiyaçlarına uygun seçimler yapmada yardımcı olmak amacıyla topluluktaki kullanıcıların fikir sağlaması olarak tanımlamaktadır.[19]

Tavsiye sistemleri internette özellikle e-ticaret uygulamalarında iş dünyasını şekillendirmek amacıyla kullanılmaktadır.[19] Amazon, ve CDNOW bu tavsiye sistemlerine örnek olarak verilebilir. Son zamanlarda gelişen teknoloji ile birlikte tavsiye sistemleri ve karar destek sistemleri sadece e-ticaret uygulamalarında değil çeşitli metinsel içerikli olan sitelerde de kullanılmaya başlamıştır. Ancak henüz haber tavsiye sistemleri ile ilgili çalışmalar akademik ortamda devam ettirilmektedir. [18]'de tavsiye yöntemlerinin bir sınıflandırması yer almaktadır. Bu yöntemler genellikle kullanıcıya ihtiyaçlarını öneren tavsiye sistemlerinde kullanılır. [18]'de geçmiş ve gelecek tavsiye sistemleri özetlenmektedir.

Internet sayesinde mevcut bilgi kümesinin hızla büyümesi nedeniyle kullanıcılar gerçekten ihtiyaçları olan ya da aradıkları bilgilere ulaşmakta zorluk yaşamaya başlamışlardır. Araştırmacılar da bu sorunu çözmek amacıyla bilgi filtreleme teknikleri ve veri çıkarma tekniklerine önem vermişlerdir.

Tavsiye sistemlerinin incelenmesinde farklı yaklaşımlar vardır. Temel olarak, içerik tabanlı filtreleme[47], ortak filtreleme [48], ve hibrit yöntemler kullanılmaktadır. İçerik tabanlı filtrelemede kişiye geçmişte yaptığı tercihler doğrultusunda önerilerde

[47] ing: *content based filtering*
[48] ing: *collaborative filtering*

bulunulur. Geçmişte yapılan tercihler, mevcut durum ve geçmiş arasında bağlantı kurmakta kullanılır. Ortak filtreleme belli bir içerik üzerindeki kişisel görüşleri bir araya getirerek, ayni ilgi alanları olan ve benzer tür bilgiye ihtiyaç duyan bireyleri eşleştirme yöntemidir. Bu ilişkiye dayanarak birbiriyle benzer içeriğe sahip olan kişiler için tavsiyeler sunulurken dikkate alınır. Hibrid metodlar[49] ise her iki yöntemin birlikte kullanılmasıdır [18,19].

[49] ing: *hybrid methods*

BÖLÜM 4

4. ÖNERİLEN HABER TAVSİYE SİSTEMİ

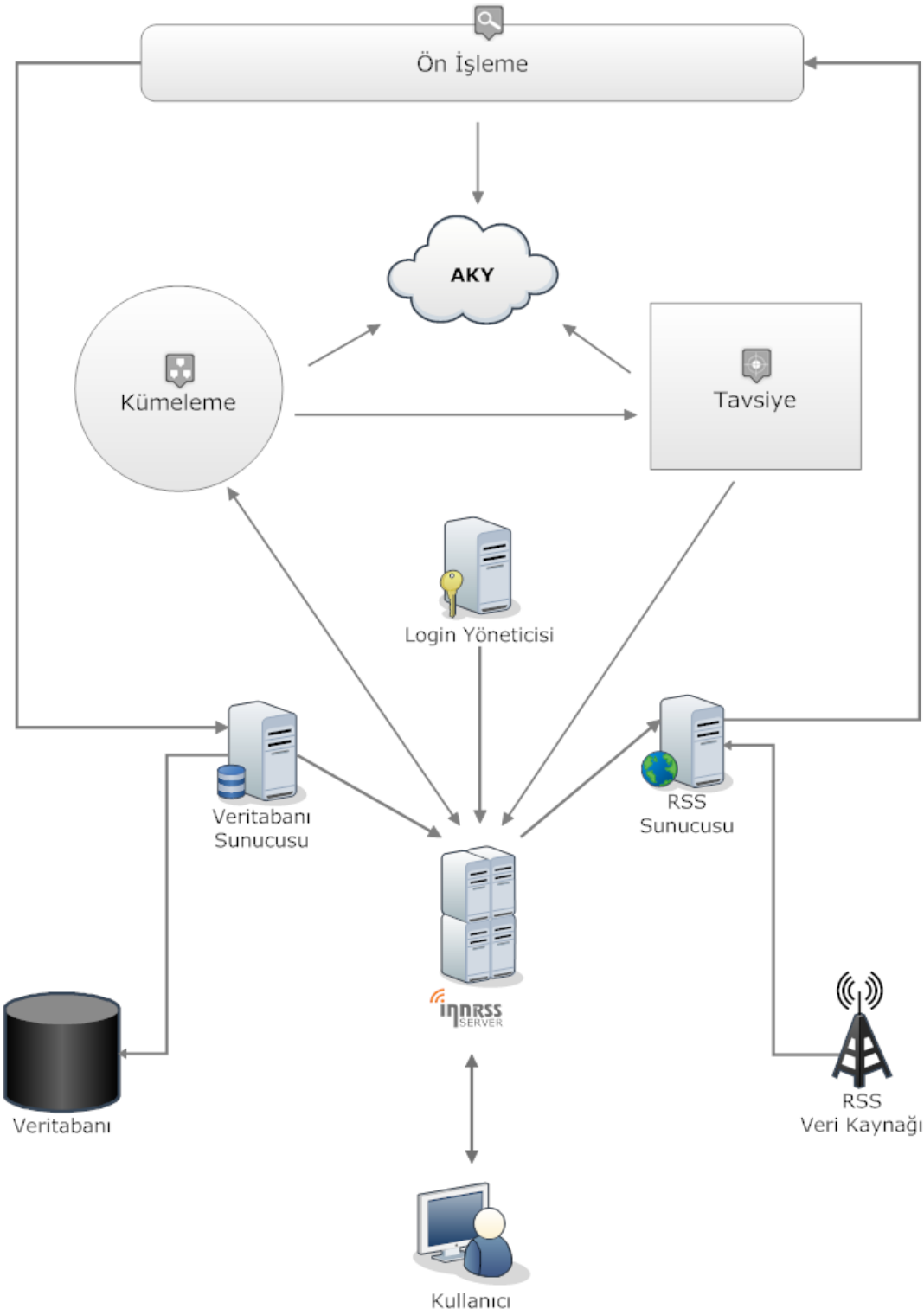
Tez kapsamında geliştirilen haber tavsiye sisteminin amacı kullanıcıların profilini oluşturmak ve oluşturulan profile göre haber tavsiyesi sunmaktır. Profil oluşturulması ikinci bölümde anlatılan ağırlıklı kapsam yoğunluğu algoritması temeline dayanmaktadır. Bu bölüm şu şekilde organize edilmiştir. Bölüm 4.1'de sistem mimarisinden bahsedilecek. Bölüm 4.2'de ön işleme aşaması anlatılacak. Bölüm 4.3'de haber okuma modülü gösterilecek. Bölüm 4.4'de haber kümeleme konusuna değinilecek. Bölüm 4.5'de okunan haberler için kümeler arası haber optimizasyonu tartışılacak. Bölüm 4.6'de haber kümeleri için en iyi küme sayısını bulmamızı sağlayan kümeler arası haber optimizasyonu uygulanması anlatılacak. Bölüm 4.7'de kümelerin özet bilgisini niteleyen anahtar kelime çıkarma metodundan bahsedilecek. Bölüm 4.8'de haber tavsiye kısmı anlatılarak bölüm sonlandırılacaktır.

4.1. Sistem Mimarisi

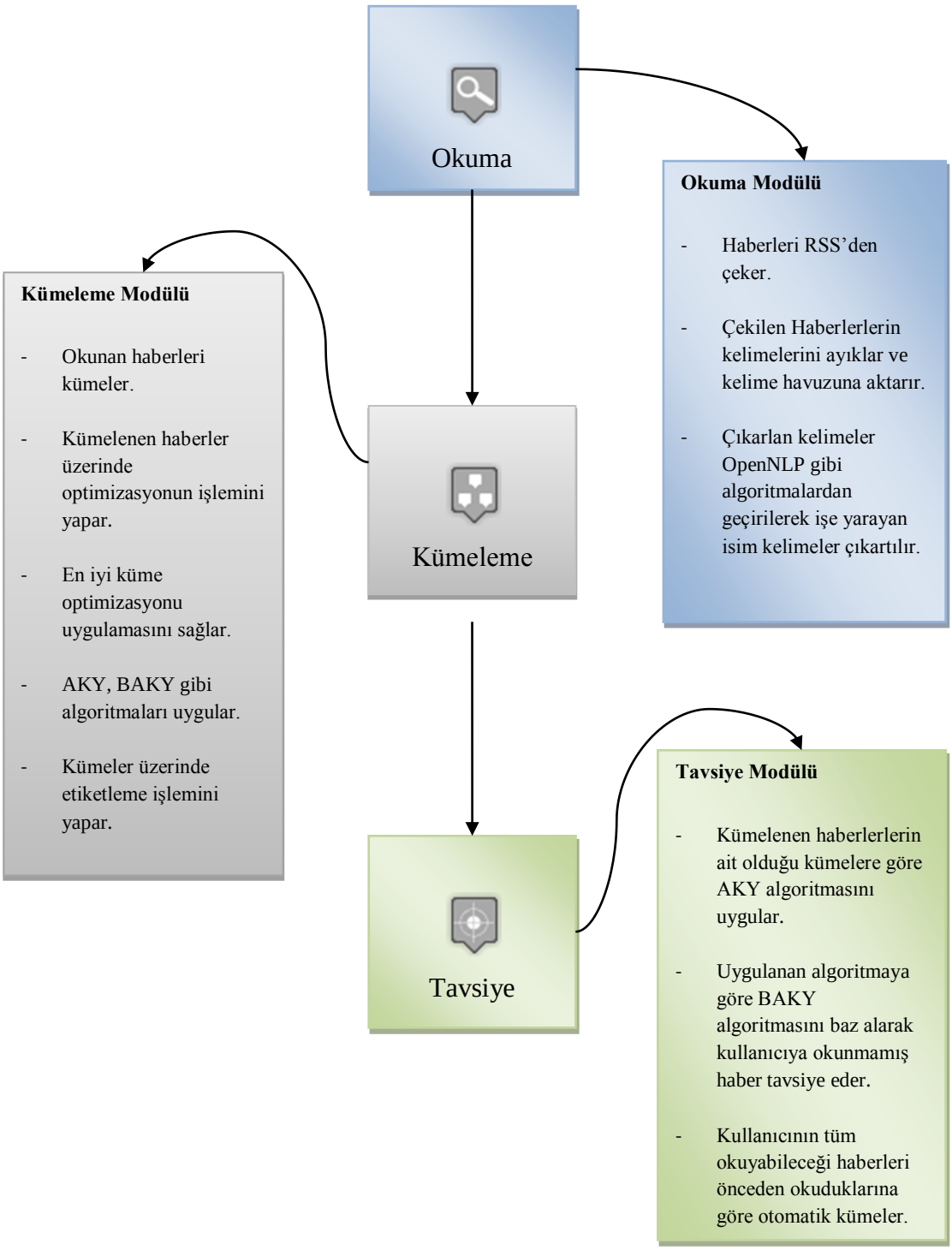
Sistem üç ana modül ve çok katmanlı mimari modeline göre tasarlanmıştır. Şekil 4.1 sistem diyagramında sistemin genel görünümü gösterilmektedir. Modül bazında sistem diyagramı Şekil 4.2'de sunulmaktadır. Sistem güçlü bir veritabanı işletme sisteminin yüksek kaliteli bir örneğidir. Depolanmış yordamlar[50] yardımıyla oturumlar, aktiviteler, kullanıcılar, haberler ve gruplar hakkında bilgi toplar ve saklar.

Veritabanı sistemimiz ilgili verilerin daha etkin ve daha güvenilir biçimde çağrılabilmesini sağlamak üzere, ilişkisel olarak tasarlanmıştır. Ayrıca, veritabanı tasarımı gelecekteki modül eklemelerine uyum sağlayabilmek için modifikasyona da açıktır. Çeşitli tablolar, görünümleri ve kayıtlı süreçleri içermektedir. Ayrıntılı veritabanı diyagramı EK-A 'da sunulmuştur.

[50] ing: *stored procedure*



Şekil 4.1. Sistem diyagramı



Şekil 4.2. Modül bazında sistem diyagramı

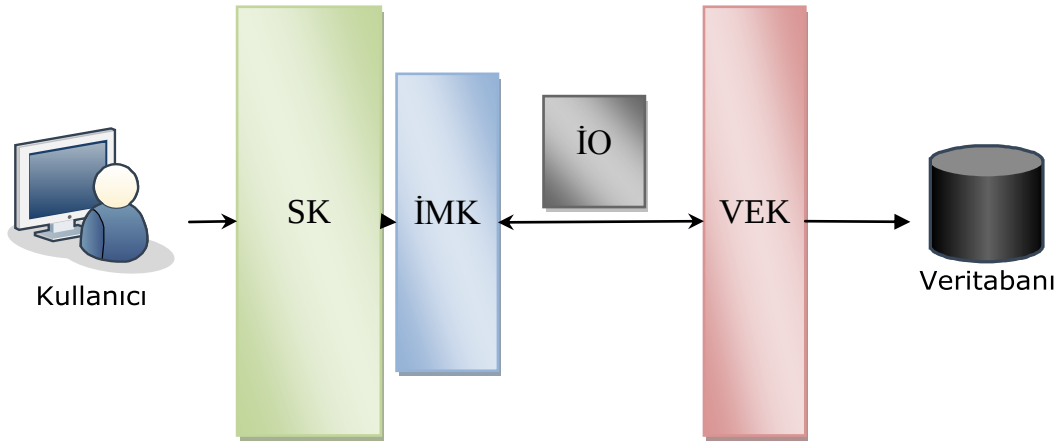
4.1.1. Sistem Aşamaları

Sistem, beş ana aşamadan oluşmaktadır. Haberler online olarak RSS kaynağından çekildikten sonra bir ön işleme aşamasından geçirilir. Bu aşamadan geçirilmesinin nedeni haberlerdeki gereksiz kelimelerden kurtularak veritabanı boyutunu düşürmektir. Haber kümeleme aşaması okunan benzer haberleri aynı grup içine alma temeline dayanır. Küme optimizasyonu kümelenen haberlerin iyileştirme çalışmasıdır. Anahtar kelime çıkarımı ise kümelenen haberler için özet bilgi vermesi için oluşturulur. Son aşamada ise kümeler arası optimizasyon tekniği olan en iyi K görselleştirme metodu ile kümeler arası iyileştirme hedeflenmiştir. Tüm bu işlemler sonrası oluşan kümeler arasından kullanıcıya okunmayan haberler tavsiye edilir.

Şekil 4.3. Sistem aşamaları gösterimi

4.1.2. Çok Katmanlı Mimari Modeli

Sistem çok katmanlı mimarinin bir örneğidir. Temel olarak üç ana katmandan oluşmaktadır. Katman mimari modeli Şekil 4.4'de gösterilmiştir. Veritabanı ile iletişimi sağlayan VEK (Veri Erişim Katmanı)[51] en alt seviyede bulunmaktadır. Veri erişim katmanının görevi yordamlar[52] yardımıyla veritabanı ile haberleşerek gerçek veriye ulaşımı ve güncellemeyi sağlamaktadır. İMK (İş Mantık Katmanı)[53] Veri erişim katmanı ile kullanıcı arayüzünün bulunduğu Sunum katmanını bağlayan katmandır. Tez kapsamında geliştirilen sistemde karmaşık işlemler iş mantık katmanında yapılmaktadır. Veri erişim katmanı ise sadece veritabanı güncellemeleri için kullanılmıştır. Kullanıcı için geliştirilen arayüzler SK (Sunum Katmanı)[54] olarak tasarlanmıştır. Katmanlar birbirleri arasında geliştirilen sistem tarafından tekil olarak üretilen İO (İş Objelerini)[55] ile haberleşmektedir.



Şekil 4.4. Çok katmanlı mimari modeli

[51] ing: *DAL (fata access layer)*
[52] ing: *procedure*
[53] ing: *BLL (business logic layer)*
[54] ing: *presentation layer*
[55] ing: *BE (business entity)*

Veri katmanı ve iş katmanındaki tüm sınıflar çoklu mimariyi korumak için üst sınıflardan türetilmiştir. Sınıf metotlarını standartlaştırma amacıyla üst sınıflar için arayüz[56] tanımlamaları yapılmıştır. Ayrıntılı sınıf diyagramı EK-D'de, detaylı olarak gösterilmiştir.

4.2. Ön İşleme Aşaması

Veri madenciliğinde[57] analiz edilecek giriş verilerinin belirli bir formata sahip olması ayrıca bozuk veya gereksiz verilerden temizlenmiş olması gerekmektedir. Bu noktadaki en büyük sorun, işleyeceği veri kümesinin yapısal olmamasıdır. Genellikle doğal dil kullanılarak yazılmış dokümanlar üzerinde çalışılan metin madenciliği alanında ön işleme aşaması[58], veri temizlemenin yanında veriyi uygun formata getirme işlemini de gerçekleştirmektedir.[9] Tez kapsamında gerçekleştirilen ön işleme adımı üç aşamadan oluşmaktadır. İlk aşamada gereksiz karakterler atılmaktadır. ikinci aşamada etkisiz kelimeler temizlenir. Son aşamada ise sözcük türleri etiketlenip, sözcükler arasından isim-kök çıkarımı yapılmaktadır. Bu bölüm şu şekilde organize edilmiştir; Bölüm 4.2.1'de ilk aşama olan gereksiz karakterlerin atılması anlatılmıştır. Bölüm 4.2.2'de etkisiz kelimelerin temizlenmesi aşaması aktarılmıştır. Bölüm 4.2.4'de sözcük türlerinin etiketlenmesi ve isim-kök çıkarımı anlatılmıştır.

4.2.1. Gereksiz Karakterlerin Atılması

Etkili bir ön işleme yapabilmek için haber metinlerinden noktalama işaretlerini çıkarmak önemli bir ihtiyaçtır. Ayrıca metinlerde bulunan karakterlerden tamamı veya bir kısmı büyük veya küçük harfle yazılmış olabilir. Böylece birbirinden farklı olarak yazılmış bu iki metin, programlama dillerindeki eşit operatörü tarafından aynı metinler olarak değerlendirilmezler.

[56] ing : *interface*
[57] ing : *data mining*
[58] ing : *pre-processing state*

Bir metini büyük harfe veya küçük harfe çevirirken de çeviren fonksiyonun dil ayarları olması gerekmektedir. Örneğin "I" karakteri küçük harfe çevrildiğinde "ı" olacaktır. Bu yanlışlık yüzünden haber metinlerindeki karşılaştırılan haberlerin kelimeleri aynı olmaz. Bu durum bizim doğru sonuç elde etmemizi engeller. Bu nedenle fonksiyona dil desteği uygulamak gerekmektedir. C# için yaratılan globalleşme sınıfı[59] kullanılmıştır.

Örnek: Chile says**;**- earthquake and Tsunami - Left 700 dead **!**
Sonuç: chile says earthquake and tsunami left 700 dead

4.2.2. Etkisiz Kelimelerin Temizlenmesi

Etkisiz kelimeler (durak kelimeler)[60]; Türkçe'de "bir", "bu", "şu" .., İngilizce'de "and", "or".. gibi bir dilde çok sık kullanılan kelimelerdir. Bu tip kelimeler arama motorları tarafından gözardı edilir. Arama motorlarının bu kelimeleri görmezden gelmesinin sebebi hemen her yazıda geçtiklerinden ötürü arama sonuçlarına pozitif bir katkı sağlamamaları, hatta bu sonuçları negatif yönde etkileyip daha isabetsiz sonuçların dönmesine sebep olmalarıdır. Etkisiz kelimeler, her uygulamaya göre farklılık göstermektedir. Bu nedenle seçilen uygulamaya göre kullanılan etkisiz kelime listesi dikkatli oluşturulmalıdır.

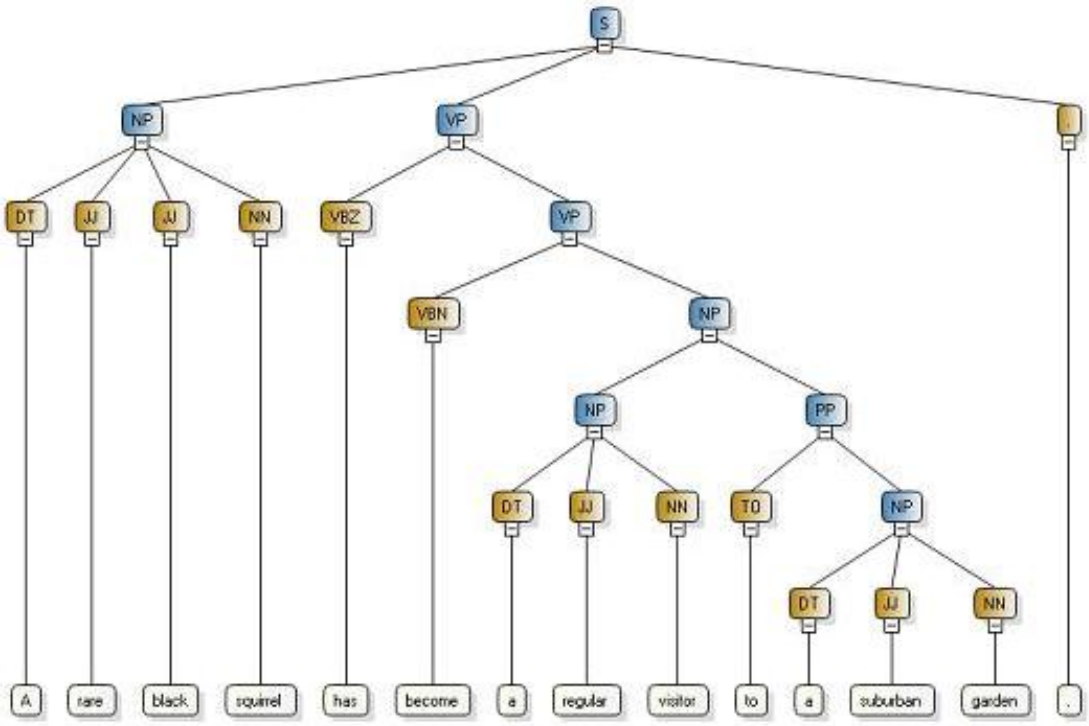
Tez çalışmasında önerilen haber tavsiye sistemi için çeşitli haberler incelenmiş ve doğal dilde sık kullanılan kelimelerin bir listesi oluşturulmuştur. Bu liste bir text dosyasında saklanmaktadır. Kelime birinci aşamadan geçirildikten sonra bu aşamada sık kullanılan kelimelerden arındırılır.

Örnek: **the** euro rebounded **from a** 10-month low **on** friday
Sonuç: euro rebounded 10-month low friday

[59] ing: *globalization class*
[60] ing: *stop words*

4.2.3. Sözcük Türlerinin Etiketlenmesi ve İsim-Kök Çıkarımı

Sözcük türlerini etiketleme işlemi ön işleme aşamasının en önemli safhalarındandır. Sözcük türleri etiketlemek, bir sözcük türünü bazen bir sözcük türünün kısaltmasını bir cümledeki her kelimeye atama işlemidir. Simgeleme işleminden bir dizi simgeler elde ettikten sonra, bu dizi sözcük türü etiketçisine verilir. Bu işlem için Bölüm 2'de bahsedilen açık kaynak doğal dil işleme kütüphanesi (OpenNLP)[14] geliştirilerek kullanılmıştır.



Şekil 4.5. Etiketleme ağaç örneği [15]

"A rare black squirrel has become a regular visitor to a suburban garden." örnek cümlesi Penn TreeBank etiketlerine göre etiketlendiğinde Şekil 4.5'deki sonuç elde edilmiştir.

Çizelge 4.1. Penn Treebank etiket şeması [15]

CC	Bağlaç	RP	Particle
CD	Sayma sayısı	SYM	Sembol
DT	Belirtme sıfatı	TO	mek, mak- mastar
FW	Yabancı kelime	VB	Fiil, temel biçimi
IN	Edat	VBD	Fiil, geçmiş zaman
JJ	Sıfat	VBG	Fiil, -ing eki almış biçimi
JJR	Sıfat, karşılaştırma	VBN	Fiil, -ed eki almış biçimi
JJS	Sıfat, üstünlük	VBP	Fiil, düzensiz 3.hal
UH	Ünlem	VBZ	Yardımcı fiil
MD	Kip belirteci	WDT	wh-belirteç
NP	İsim tamlaması	WP	wh-zamir
NN	İsim, tekil veya topluluk	WP$	İyelik wh-zamiri
NNP	Özel isim, tekil	WRB	wh-zarf
NNPS	Özel isim, çoğul	``	Sol açık çift tırnak işareti
NNS	İsim, çoğul	"	Sağ kapalı çift tırnak işareti
POS	İyelik eki	.	Cümle sonu noktalama işareti
PRP	Kişi zamiri	:	İki nokta üst üste, noktalı virgül
PRP$	İyelik zamiri	$	Dolar işareti
RB	Zarf	#	Pound işareti
RBR	Zarf, karşılaştırma	LRB	Sol parentez
RBS	Zarf, üstünlük	RRB	Sağ parentez

Sözcük türleri etiketleri, simgeler dizisiyle aynı uzunlukta bir dizide - dizinin her bir indeksindeki etiketi, simgeler dizisindeki aynı indekste bulunan simgeyle eşleştiği yerde döndürülür.[15] Sözcük türleri etiketleri[61], Çizelge 4.1 Penn Treebank[15] şemasına, Pensilvanya Üniversitesi tarafından geliştirilen dilbilimsel araca, uyan kodlanmış kısaltmalardan oluşmaktadır.

Sözcük türleri etiketçisine bir sözcük türleri arama listesi sağlayarak, daha iyi kontrol etmek mümkündür. Standart sözcük türleri etiketçisi bir arama listesi kullanmaz, fakat tam ayrıştırmacı liste kullanır. Arama listesi, bir kelimeli metin dosyası ve onun her satırdaki olası sözcük türleri etiketlerinden oluşur. Bunun anlamı şudur:

[61] ing: *word tag types*

etiketlediğiniz cümledeki bir kelime arama listesinde bulunuyorsa, sözcük türleri etiketçisi olası sözcük türleri etiketleri listesini arama listesinde belirlenmiş olanlarla sınırlar, böylece doğru etiketi seçmeyi daha olası kılar.

Tablo 4.1'de örnek haber olarak; yahoo haber kaynağından çekilen örnek bir haberin cümle etiketleyicisinden geçirildikten sonraki durumu yer almaktadır.

The earthquake and tsunami that struck Chile last month killed 700 people and caused damages of nearly $30.1 billion, according to the government. The ground hasn't stopped shaking.	
NP The/DT earthquake/NN and/CC tsunami/NN	Belirtme Sıfatı, İsim, Bağlaç, İsim
NP that/WDT	Belirteç
VP struck/VBD	Fiil (geçmiş zaman)
NP Chile/NNP	Özel isim (tekil)
NP last/JJ month/NN	Sıfat, İsim
killed/VBD	Fiil (geçmiş zaman)
NP 700/CD people/NNS	Sayma sayısı, İsim (çoğul)
and/CC	Bağlaç
VP caused/VBD	Fiil (geçmiş zaman)
NP damages/NNS	Fiil (geniş zaman)
PP of/IN]	Bağlaç
NP nearly/RB $/$ 30.1/CD billion/CD	Zarf
,/,	Virgül
PP according/VBG	Fiil (-ing eki almış fiil)
PP to/TO]	To
NP the/DT government/NN	Belirtme sıfatı, İsim
./.	Nokta
NP the/DT ground/NN	İsim
VP has/VBZ n't/RB stopped/VBN shaking/VBG	Fiil (geçmiş zaman), -ing ekli fiil
./.	Nokta

Tablo 4.1. Örnek haberin cümle etiketleyicisinden gerçime sonrası etiket tablosu

Etiket tablosunda; “the earthquake and tsunami” cümleciği (belirtme sıfatı, isim, bağlaç, isim)‘den oluştuğu görülmektedir. Haberin tüm cümleleri ayrı ayrı analiz edilerek her kelimenin türü simgelenmiştir. Kümeleme aşamasındaki amaç düşük kelime, maksimum kazanç olmalıdır. Bu nedenle isim kök bulma bizi veritabanı şişmesinden kurtaracak ve kümelerin doğru ayrılmasını sağlayacaktır.

Çizelge 4.2. İsim kökleri için seçilen etiketler

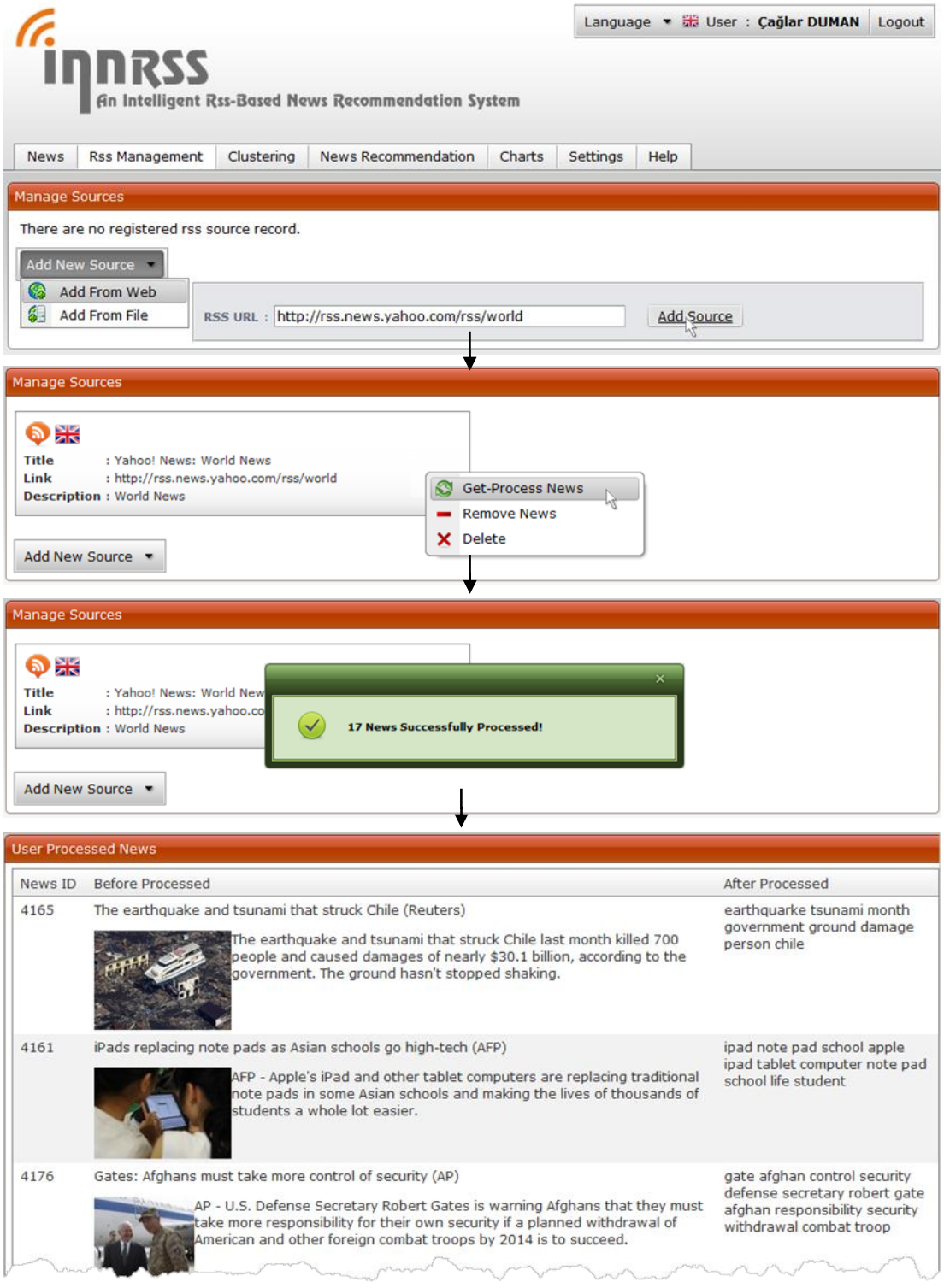
NN	İsim - tekil veya topluluk
NNP	Özel isim - tekil
NNPS	Özel isim - çoğul
NNS	İsim - çoğul

Önerilen haber tavsiye sistemi için Penn Treebank[15] şemasından sadece isim etiketleri seçilmiştir. Etiket listesi Çizelge 4.2’de verilmiştir. İşlenecek haber için ilgili etiketler kullanıldığında isim ve köklerin çıkarımı yapılması sağlanır. Üst bölümde verilen örnek haber için isim ve kök çıkarımı yapıldığında sonuç;

- NN (İsim - tekil) | earthquarke, tsunami, month, government, ground
- NNP (Özel isim - tekil) | chile
- NNS (İsim - çoğul) | damages, people

Bu kısımdaki en büyük sorun çıkan köklerdeki çoğul tekil isim farklılıklarıdır. Bu sorunun giderilmesi için ingilizcedeki tüm eklerin istatistiği çıkarılıp özel bir kütüphane geliştirilerek kullanılmıştır.[30] Kütüphane’deki en büyük sorun düzensiz çoğul isimler yaratmaktadır. Bu sorunu gidermek amacıyla düzensiz çoğul isimler için bir liste oluşturulmuştur. Ön işleme, algoritmanın her çalışmasında düzensiz çoğul isim kontrolü yapılmaktadır. Kontrol sonrası haberin kelimelerinin son durum şu şekilde gerçekleşir;

- NN (İsim, tekil) | earthquarke, tsunami, month, government, ground, damage, person
- NNP (Özel isim, tekil) | chile



Şekil 4.6. Rss ekleme ve ön işleme ekran görüntüsü

4.3. Haber Okuma

Geliştirilen sistemde kullanıcı RSS kaynaklarına göre teker teker yada toplu olarak haberleri okumaktadır. Şekil 4.7'de haber okuma ekran görüntüsü sunulmuştur.

Şekil 4.7. Kullanıcı haber okuma ekran görüntüsü

Kullanıcı Şekil 4.6'da gösterilen RSS kaynak yönetimi ekranından, Bölüm 4.2'de bahsedilen ön işleme aşaması ile haberleri kaynaktan çektiğinde, haberler işlenerek genel haber havuzuna HABER[62] tablosuna kaydedilir. Okunan her haber için KULLANICIHABER[63] tablosuna yeni bir kayıt oluşturulmaktadır. Bu yöntem ile bir RSS kaynağından eklenen haber sadece bir kere ön işlem algoritmasından geçirilmiş olur. Bu sayede çok kullanıcılı bir ortam sağlandığında hız ve performans elde edilmektedir. Haber havuzu arttıkça kullanıcıların haber kaynaklarından haber çekme ve okuma süresi kısalmaktadır.

4.4. Haber Kümeleme

Numerik verileri kümelemek için geometrik ve istatistiksel özelliklerini temel alan bazı genel ölçümler ya da interaktif gözlem yöntemleri geliştirilmiştir.[23] Anlamlı bir ikili uzaklık fonksiyonunun bulunmayışı nedeniyle, kategorik kümelendirmede bir ölçü olarak dağılım temelli ölçümler geniş bir alanda kullanılmaktadır.[23] Fakat bu gibi genel ölçümler, işlemsel veriler gibi özel tip veri setleri için etkin olmaktan uzaktır. Bu alana özel, anlamlı kalite ölçümleri ise daha ilginç olarak kabul edilmektedir.[23]

Önerilen sistemde haber içerikli işlemsel veri setlerini[64] etkin olarak işleyebilen, hafızada az yer kaplayan ve ölçeklendirilebilir bir kümelendirme algoritması önerilmektedir. Yaklaşım işlemsel veri setlerinin kümelendirme ölçümü için ağırlıklı kapsam yoğunluğu[65] (AKY)[66] kavramına dayanmaktadır. Tanım kümesi özelinde kümelendirme kriteri olarak ağırlıklı kapsam yoğunluğunu kullanmaktaki amaç, işlemsel veri birleşimindeki kuralların gözlemlenmesinin doğal olarak yoğunluk temelli veri kümelemesi ile ilişkili olmasıdır. Böylece, ağırlıklı kapsam yoğunluğunu öğe setinin frekansı kavramı temelinde sunulmaktadır.

[62] ing: *NEWS*
[63] ing: *USERNEWS*
[64] ing: *transactional dataset*
[65] ing: *weighted coverage density*
[66] ing: *WCD*

Kümeleme aşaması, başlangıçtaki küme tahsisini ve tekrarlanan kümeleme iyileştirmelerini kümeleme sonucunun beklenen ağırlıklı kapsam yoğunluğu[67] değerini maksimize edene dek ağırlıklı kapsam yoğunluğu algoritmasını kullanır.

Kullanıcının okuduğu haberler değişik kategorilerden oluşabilir. Bu haberler için ayrı kümelerin oluşması hem bilgisayarlar hem de insanlar için zor bir işlemdir. Tez çalışmasında önerilen tavsiye sistemi bu yaklaşımı iki ana faza ayırmıştır.

- Birinci Faz: Haberler okunmaya başladığında bir kısıt küme sayısı belirlenmesi ve o sayıya ulaşılana kadar okunan her haber için yeni bir küme oluşturulmasıdır.
- İkinci Faz: Kümeler arası haber optimizasyonu, diğer değişle küme içerisindeki haberleri iyileştirme çalışmasıdır.

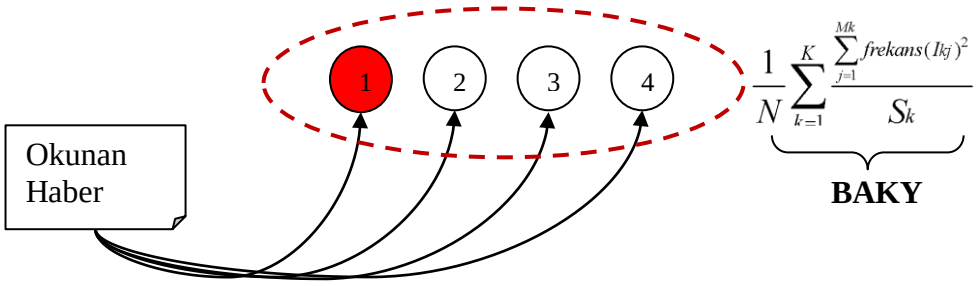
Birinci fazda kümelemeye başlanmadan önce bir kısıt değeri[68] tespit edilmektedir. Kümeler, kısıt ayar panelinde belirtilen kısıt sayısına göre oluşmaktadır. Kısıt sayısı kadar kümeye yerleştirilen haberlerden sonra gelen haberler önerilen kümeleme algoritmasına göre yapılmaktadır.

Çizelge 4.3. Küme yerleştirme sözde kod [4]

Girdi	T: İşlenecek veri,
Varsayılanlar	D: Veri seti, K: Küme sayısı,
Algoritma	Başlangıçtaki K küme sayısını sabit kabul et. Tüm veri setini tara (i...k) **{** D veri setine T işlemsel verisini eşleştir. T yi Ci kümesinin BAKY değerine göre maksimize eden kümeyi bul. <T,i> ikilisi için gelen T işlemini ilgili kümeye yerleştir. **}**
Çıktı	Ci..k : Tüm kümeler

[67] ing: *expected weighted coverage density*
[68] ing: *constraint value*

Çizelge 4.3'de kısıt küme yerleştirme algoritmasının sözde kodu verilmiştir. T: İşlenecek veriyi yani yeni okunan haber olarak kabul edilmektedir. D: veri seti[69] ise kullanıcının önceden okuduğu haberlerin profilini temsil etmektedir. K: küme sayısı başlangıçtaki boş kümeleri temsil etmektedir. Önerilen sistemde değişiklik kullanıcı tarafından manuel olarak yapılabilmektedir.



Şekil 4.8. BAKY değerinin maksimize edilmesi

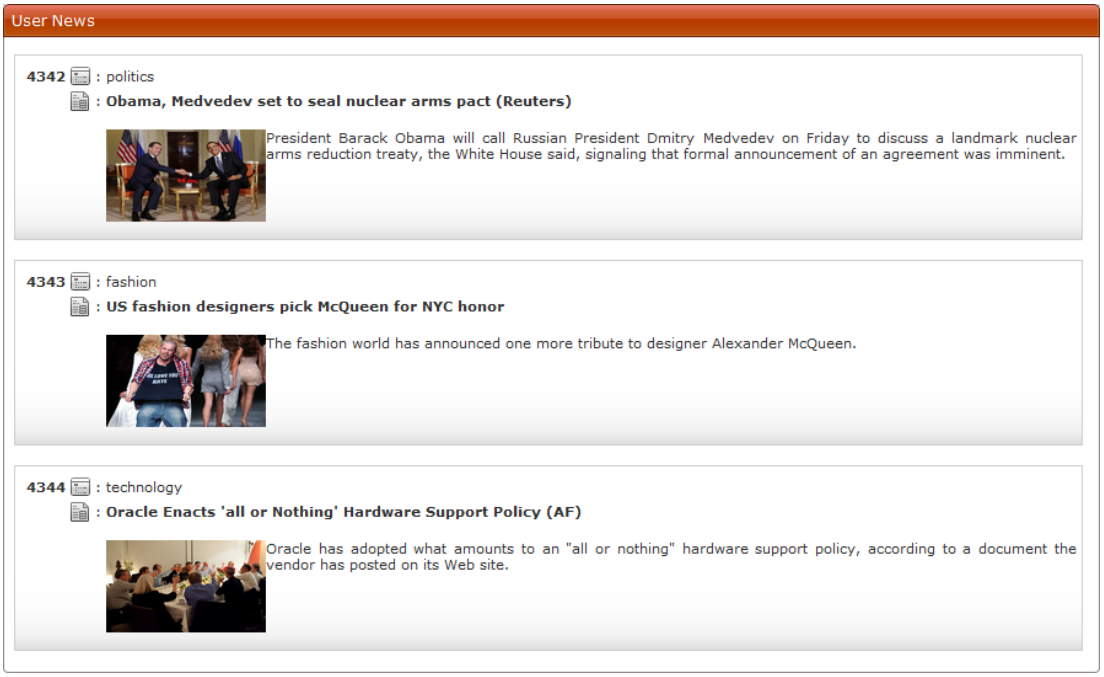
Beklenen ağırlıklı kapsam yoğunluğu algoritması formülü [4] Şekil 4.8'de gösterilmektedir. N: Tüm haberlerin sayısı, K: Küme sayısı, Mk: K kümesindeki kelime sayısı, Sk: K kümesindeki kelimelerin frekans[70] toplamları, $frekans(Ikj)^2$: K kümesindeki j kelimesinin frekans karesini temsil etmektedir.

Okunan haberin, varolan uygun kümelerden hangisine gireceğine, beklenen ağırlıklı kapsam yoğunluğu değerinin maksimize edilmesi karar verir. Okunan haber sırasıyla kullanıcının profilindeki kümelere eklenir ve çıkarılır. Her eklemede beklenen ağırlıklı kapsam yoğunluğu hesaplanır. Haberin hangi profile ait olduğu analiz ile belirlenir. Analiz kısmında BAKY değerleri küçükten büyüğe doğru sıralanır ve maksimum olan değer haber için asil küme olarak değerlendirilir. Önerilen algoritma Bölüm 4.4.1'de örnek veriler ile doğrulanmıştır.

[69] ing: *dataset*
[70] ing: *ocurry*

4.4.1. Haber Kümelemenin Örnek Veriler İle Doğrulanması

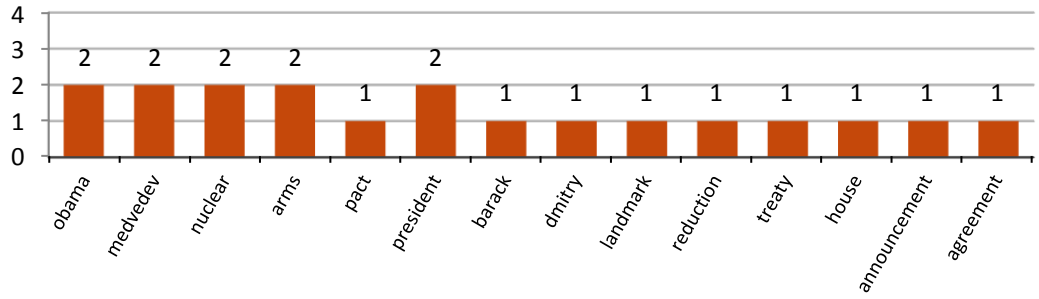
Kümeleme işlemini örnek veri seti üzerinde gösterilmesi için en popüler RSS haber sağlayıcılarından olan Yahoo haber kaynağı [32] kullanılmıştır. Bu noktada üç spesifik kategori seçilmiştir.



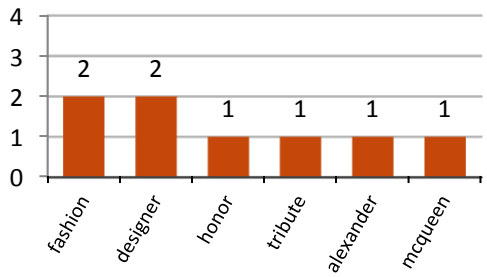
Şekil 4.9. Küme doğrulama için örnek haberler

Şekil 4.9'da görüldüğü gibi 4342, 4343 ve 4344 numaralı politika, moda ve teknoloji kategorilerine ait haberler bulunmaktadır. Haber verilerini test etmek için küme kısıt sayısının üç olarak ayarlandığı varsayılmıştır. İyi ve hızlı bir kümeleme yapılması için saklanan haberlerin histogramsal verileri veritabanında saklanmalı ve işlenmelidir. Örnek olarak seçilen üç haber ayrı kümelere yerleştiğinde oluşan histogram verileri sırasıyla Grafik 4.1, 4.2 ve 4.3'de verilmiştir. Veriler incelendiğinde politika kategorisine ait 4342 nolu haberin histogramı on dört kelime, moda kategorisine ait 4343 nolu haberin altı kelime, teknoloji kategorisine ait 4344 nolu haberin ise sekiz kelimeden oluştuğu görülmektedir. 4342 nolu haberde "obama, medvedev, nuclear, arms ve president" kelimeleri diğer kelimeleri, 4343

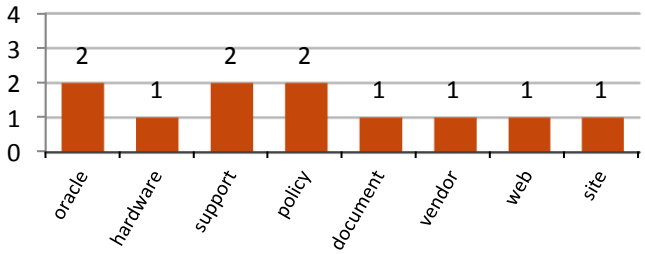
nolu haberde “fashion, designer” kelimeleri ve 4344 nolu haberde ise “oracle, support, policy” kelimeleri ağır basmaktadır.



Grafik 4.1. Birinci küme kelime histogram verisi



Grafik 4.2. İkinci küme kelime histogram verisi



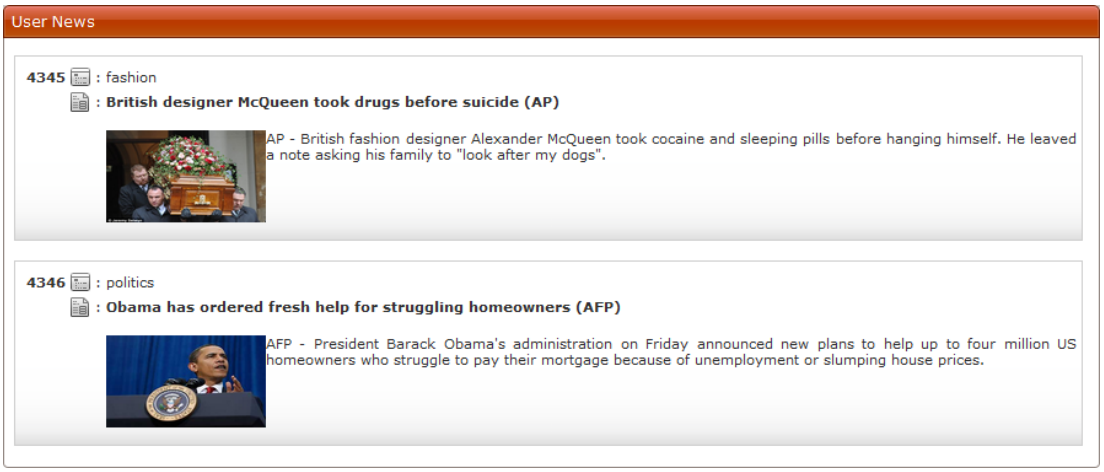
Grafik 4.3. Üçüncü küme kelime histgoram verisi

Algoritmada belirtilen kısıt sayısına kadar olan haberler ayrı kümeye yerleşme eğilimi gösterecektir. Bu nedenle üç yeni kümenin oluşacağı beklenmektedir. Kullanıcı okuma işleminden sonra oluşan kümeler Şekil 4.10’da gösterilmiştir.

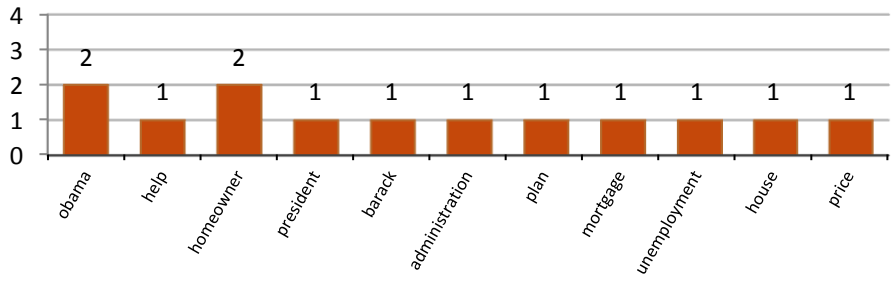
Şekil 4.10'ds gösterilen; N: haber sayısını, W: kelime sayısı, WCD: ağırlıklı kapsam yoğunlu algoritmasından çıkan sonucu temsil etmektedir. Kümeler içerisindeki ilk sayı haber numarasını, ikinci yazı haberin kategorisini, diğer kelimeler ise haberin ön işleme aşamasından geçirildikten sonraki işlenmiş kelimelerini temsil etmektedir.

Şekil 4.10. Küme doğrulama için örnek haberlerin kümeleri

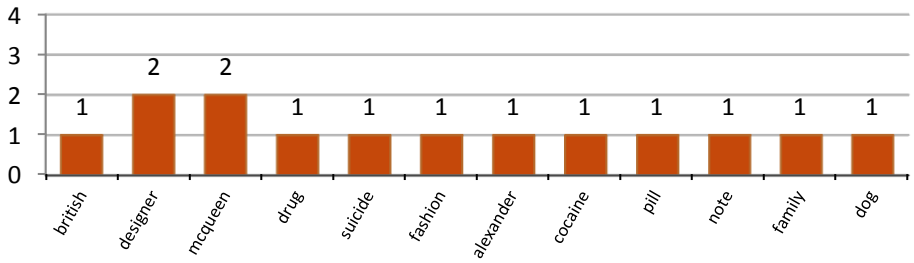
Tez çalışmasında önerilen haber tavsiye sisteminin kümeleme bölümünü oluşturan en önemli sorulardan bir tanesi var olan küme durumunda yeni bir haber okunursa hangi kümeye gireceğidir. Bu durum için benzer kategorilerde iki farklı haber sisteme okutulmuştur. Şekil 4.11'de kullanıcıya okutulacak örnek haber listesi sunulmuştur.



Şekil 4.11. Küme doğrulama için okutulacak örnek haberler



Grafik 4.4. Haber 4346 histogram verisi



Grafik 4.5. Haber 4345 histogram verisi

Okuma işlemi başlatıldığında algoritmadaki amacımız beklenen ağırlıklı kapsam yoğunluğu değerini maksimize etmektir. Geliştirilen sistemde bu değerin maksimize edilebilmesi için okunan haberin kümelere etkisi hesaplanmıştır. Bu hesaplamada BAKY değerinin maksimum olduğu küme haberin gireceği küme olarak tespit edilmektedir. Formül 4.1'de beklenen ağırlıklı kapsam yoğunluğu algoritması formülü verilmiştir.

$$BAKY(C^K) = \frac{1}{N}\sum_{k=1}^{K}\frac{\sum_{j=1}^{Mk} frekans(Ikj)^2}{Sk} \qquad (4.1)$$

C^K: Küme adını, S_k: Küme içindeki haber kelimelerinin frekans (gelme sıklığı) toplamını, $frekans(I_{kj})^2$: Küme içindeki kelimelerin frekanslarının kareleri toplamını, M_k: Küme içindeki kelime sayısını, N:Kullanıcının okuduğu tüm haber sayısını temsil etmektedir. Grafik 4.1, 4.2 ve 4.3'deki histogram verileri her biri bir küme

olacak şekilde 4.1'deki formüle göre hesaplandığında Tablo 4.2'deki sonuçlar elde edilmiştir.

$C1 = \frac{2^2+2^2+2^2+2^2+1^2+2^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2+1^2}{2+2+2+2+1+2+1+1+1+1+1+1+1+1+1}$	1,52631578
$C2 = \frac{2^2+2^2+1^2+1^2+1^2+1^2}{2+2+1+1+1+1}$	1,5
$C3 = \frac{2^2+1^2+2^2+2^2+1^2+1^2+1^2+1^2}{2+1+2+2+1+1+1+1}$	1,54545454
$BAKY = \frac{1}{3} C1 + C2 + C3$	$BAKY = 1,54244195$

Tablo 4.2. Küme doğrulama için varolan BAKY değerlerinin hesaplanması

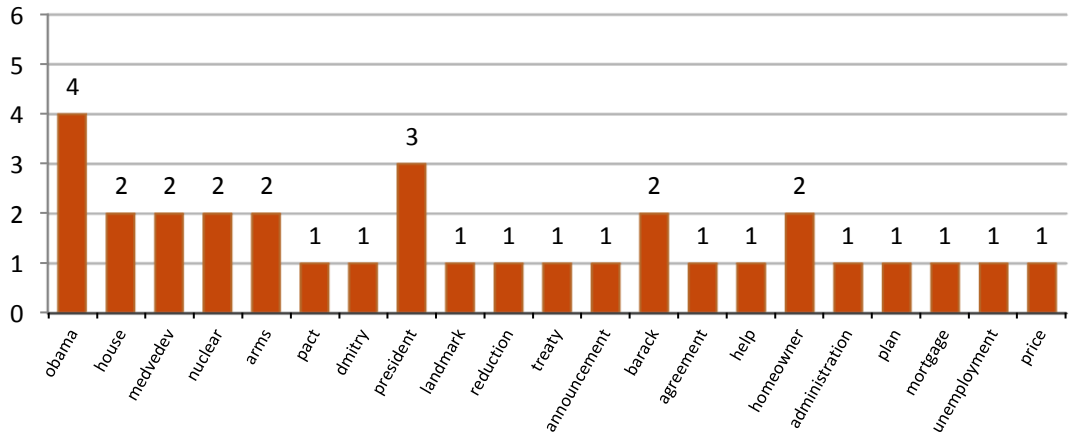
Uygun kümenin seçilmesi için öncelikle okunan haberin histogramının tüm kümelerin histogramlarıyla ayrı ayrı birleştirilmesi ve denenmesi gerekmektedir. Histogramların her birleşmesinden sonra yoğunluk değeri değişir ve bu değişim beklenen ağırlıklı kapsam yoğunluğu değerini arttırır ya da azaltır. Amaç beklenen değeri maksimize etmek olduğu için değişimin maksimize edildiği en büyük değer alınmaktadır. Tablo 4.3'de 4346 nolu haber için hesaplama sonuçları verilmiştir.

Varolan BAKY = 1,54244195 / N = 4					
Haber	Küme	Eski AKY'	Yeni AKY'	BAKY	AKY Değişim
4346	1	1,5263157812	1.9375000000	1,245738500	0.4111842105263
4346	2	1,5000000000	1.3809523823	1,113180345	-0.119047619047
4346	3	1,5454545454	1.4166666667	1,110745620	-0.128787878787

Tablo 4.3. Küme doğrulama için 4346 nolu haberin BAKY değerleri hesaplaması

Küme: okunan haberin test edildiği kümeyi, eski AKY: haber kümeye girmeden önceki AKY değerini, yeni AKY: haber kümeye girdikten sonraki AKY değerini, BAKY; toplam beklenen ağırlıklı kapsam yoğunluk değerini, değişim ise haber kümeye girmeden önceki ağırlıklı kapsam yoğunluk değeri ile girdikten sonraki

ağırlıklı kapsam yoğunluk değerinin farkını temsil etmektedir. Tablo 4.3 detaylı incelendiğinde değişimin ve beklenen ağırlıklı kapsam yoğunluk değerinin maksimum olduğu nokta birinci kümedir.

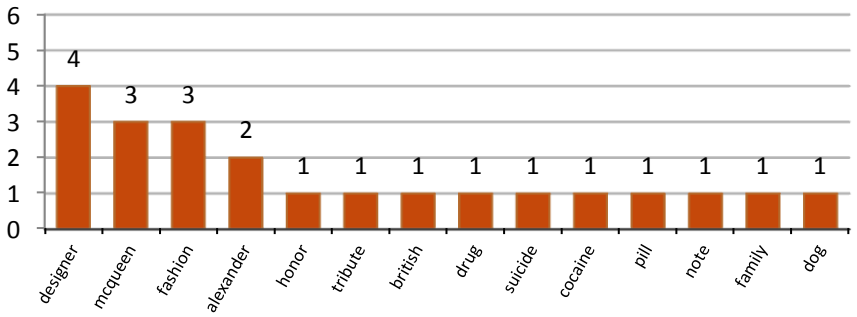


Grafik 4.6. Haber 4346'nın birinci kümeye girdikten sonra oluşan histogramı

Gösterilen hesaplamalar doğrultusunda küme doğrulanması için haber 4346 birinci kümeye daha yakındır ve birinci kümeye alınır tezi savunulmaktadır. Grafik 4.6'da 4346 nolu haberin birinci kümeye girdikten sonraki oluşan histogramı verilmiştir. Kümedeki toplam haber sayısı dörde yükselmiştir. Tablo 4.2'deki hesaplama tekrar yapıldığında varolan BAKY değerinin değiştiği görülmektedir. 4345 numaralı haber için hesaplama sonuçları Tablo 4.4'de gösterilmiştir.

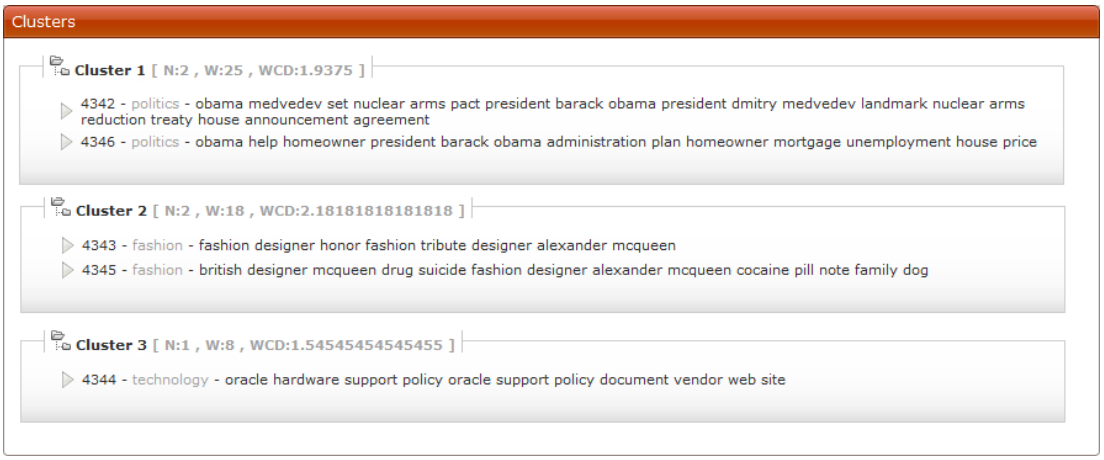
Varolan BAKY = 1,245738500 / N = 5					
Haber	Küme	Eski AKY'	Yeni AKY'	BAKY	AKY Değişim
4345	1	1.9375000000	1.739130434	0,956916886	-0.1983695652
4345	2	1.5000000000	2.181818188	1,132954444	0.68181818181
4345	3	1.5454545454	1.400000000	0,967500000	-0.1454545454

Tablo 4.4. Küme doğrulama için 4345 nolu haberin BAKY değerleri hesaplaması



Grafik 4.7. Haber 4345'in ikinci kümeye girdikten sonra oluşan histogramı

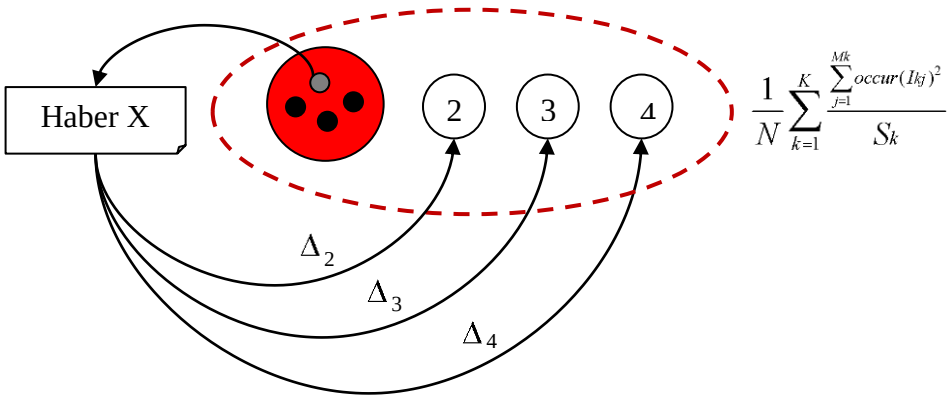
Haber 4345'in sonuçları incelenirse; birinci küme ile test edildiğinde değişim -0.198, yoğunluk 1.1329, üçüncü küme ile test edildiğinde değişimin -0.1454, yoğunluğun 1.1329 olduğu görülmektedir. İkinci küme ile denendiğinde değişimin arttığı (0.6818) yoğunluğunda değişime parlel olarak yükseldiği 1.329 gözlemlenmektedir. Bu durumda 4345 numaralı haber için en uygun yer ikinci kümedir tezi savunulmaktadır. Grafik 4.7'de 4345 numaralı haber ikinci kümeye aktarıldıktan sonra oluşan histogramsal veri sunulmuştur. Okuma işlemleri bittiğinde 4342, 4346 nolu haberler birinci kümede, 4343, 4345 numaralı haberler ikinci kümede, 4344 numaralı haber ise üçüncü kümede kalmaktadır. Oluşan kümelerin ekran görüntüsü Şekil 4.12'de gösterilmiştir.



Şekil 4.12. Küme doğrulama için test edilen haberlerin küme ekran görüntüsü

4.5. Kümeler Arası Haber Optimizasyonu

Haber kümeleme aşamasındaki bir diğer önemli işlem oluşturulan kümelerin optimizasyonudur. Küme optimizasyonu kümelerin iyileştirmesini hedeflemektir. Küme optimizasyonu var olan kümeler arasında yapılmaktadır. Bu nedenle bu aşamada geliştirilen algoritmada optimizasyon uygulanacak kümelerin en az iki haber içermesi gerekmektedir. Şekil 4.13'de kümeler arası haber optimizasyonunun görsel hali verilmiştir.



Şekil 4.13. Kümeler arası haber optimizasyonu

Şekil 4.13'de kırmızı küme, işlem yapılan kümeyi temsil etmektedir, kırmızı küme içerisindeki siyah noktalar ise kümeye ait haberleri temsil etmektedir.

Küme optimizasyonunda küme içerisindeki tüm haberler ele alınmaktadır. Bu nedenle optimizasyonu uzun ve maliyetli bir işlemdir. Tez kapsamında geliştirilen kümeleme sistemindeki temel amaç olan yoğunluğun maksimize edilmesi hedefi küme optimizasyonunda da korunmuştur, ancak bu kısımda haberin kümeler arası yer değiştirmelerine BAKY değişimleri karar vermektedir. Bu nedenle optimizasyon iki aşamaya ayrılmaktadır. Birinci aşamada bir eşik değeri kontrolü üretilmiştir. Eşik değerinin asıl amacı; "kontrol edilen haber kendi kümesinde kalmalı mı?" sorusuna

cevap aramaktır. Bu aşamada kullanıcının okuduğu tüm haberler sırası ile kontrol edilir. Çizelge 4.7'de küme optimizasyonun genel sözde kodu gösterilmektedir.

Çizelge 4.4. Küme optimizasyonu sözde kod [4]

Girdi	D: Veri seti, K: Küme sayısı,
Algoritma	Başlangıçtaki K kümesi kadar K sınıflanır. While(Taşı İşaretcisi == Doğru) **{** D veri setinden T işlemini oku. T yi Ci kümesinin BAKY değerine göre maksimize eden kümeyi bul. <T,i> ikilisi için gelen T işlemini ilgili kümeye yerleştir. **}**
Çıktı	Ci..k : Tüm kümeler

Formül 4.2'de varolan BAKY değeri kontrol edilen haber kendi kümesindeyken hesaplanan beklenen ağırlıklı kapsam yoğunluğu değerini, yeni BAKY değeri ise kontrol edilen haber diğer kümelere girdiğinde oluşan BAKY değerini vermektedir.

$$\theta_{c_1} = YeniBAKY > VarolanBAKY \tag{4.2}$$

İkinci aşamada ise kontrol edilen haberin diğer kümelere uygunluğu test edilmelidir. Bu aşamadaki en önemli nokta kontrol edilen haberin kendi kümesi hariç diğer tüm kümelerle ilişkisini bulup saklamaktır. Hesaplamaların düzgün yapılabilmesi için kümeler arasındaki değişimler bulunmalıdır. Bu nedenle Şekil 4.14'de gösterilen delta (Δ) değerleri hesaplanmaktadır.

$$\Delta_{c_1} = YeniBAKY - VarolanBAKY \tag{4.3}$$

Değişim hesaplamaları Formül 4.3'e göre yapılmaktadır. Kendi kümesinden çıkarılan haber beklenen kapsam yoğunluğu toplam değerinde değişiklik yapar. Delta (Δ) değeri haberin kendi kümesi hariç diğer kümelere girmeden önceki değerinden,

girdikten sonraki değerinin farkı ile elde edilmiş olur. Bu değişim değeri bize kontrol edilen haberin diğer kümelere ne kadar yapışık olduğunu gösterir.

$$\Delta_{k1} = T_{k1} \times \sum_{j=1}^{M_{k1}} frekans(I_{k1}) \times W_{k1} - \frac{\sum_{i=1}^{Mk2} frekans(I_{kj})^2}{S_{k2} \times N_{k2}} \quad (4.4)$$

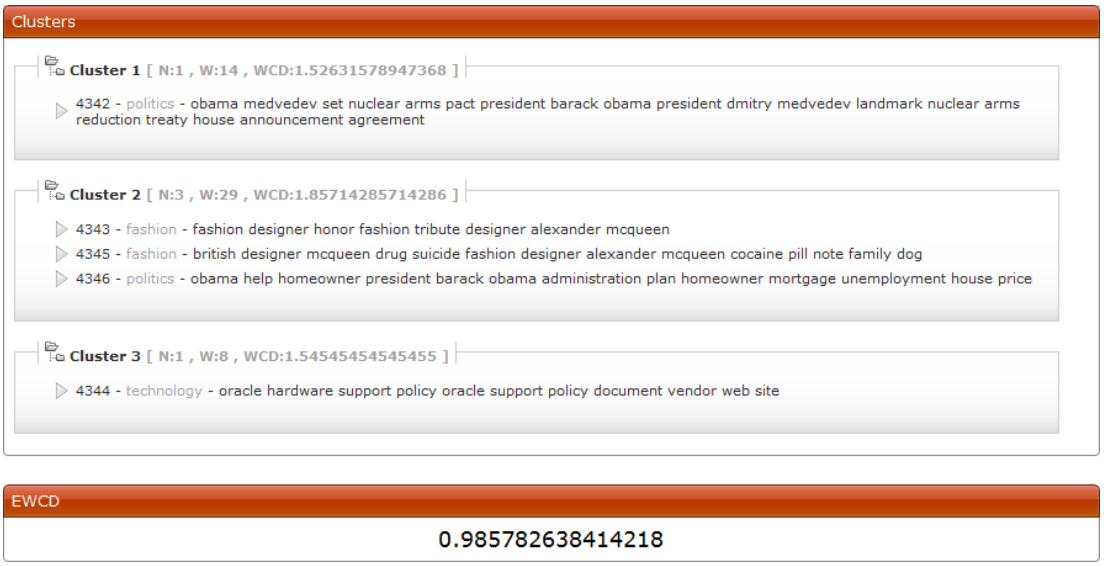
Formül 4.3 açıldığında Formül 4.4'e ulaşılmaktadır. Formüldeki senaryo şu şekildedir; Bir X haberi diğer kümeler ile denenmektedir ve bir değişim değeri elde edilmek istenmektedir. Bu değişim değeri; haberin yeni kümesine konulduğundaki BAKY değeriyle haber kendi kümesindeyken oluşan BAKY değerinin farkı ile bulunur. Formül 4.4'deki T_k1 = haberin yeni kümesine girdiğinde olan haberin ağırlığını, M_k1 = haberin yeni kümesine girdiğinde oluşan kelime sayısını, W_k1 ise haberin yeni kümesine girdiğinde oluşan kelime ağırlığını vermektedir. M_k2 = haber kendi kümesindeyken kelime sayısını vermektedir. S_k2 = haber kendi kümesindeyken frekans toplamlarını, N_k2 ise haber kendi kümesindeyken kelime sayısı toplamını vermektedir.

Şekil 4.14'de gösterildiği gibi kontrol edilen her haber diğer kümeler ile denenir ve bunun neticesinde üst kısımda anlatılan delta değerleri hesaplanır. Bulunan delta değerleri arasındaki en büyük değer kontrol edilen haberin o kümeye girmesi gerektiğini gösterir. Ancak haberi gerçekten delta değeri en yüksek kümeye sokmak için delta değerinin birinci aşamada hesaplanan eşik değeri eşitsizliğini sağlaması gerekir. Kısacası haber kendi kümesinde mi kalmalı yoksa hesaplanan en büyük delta değerine mi girmeli konusunda bir yaklaşım gerçekleştirilmiştir.

4.5.1. Haber Optimizasyonunun Örnek Veriler İle Doğrulanması

Haber optimizasyonu doğrulamak için Bölüm 4.4.1'deki örnek haberler kullanılmıştır. Haberlerin kategorileri incelendiğinde iki haberin "politika", iki haberin "moda" ve bir haberin "teknoloji" kategorisine ait olduğu görülmektedir.

Küme kısıt[71] değerimiz üç olarak ayarlanmış ve tüm haberler sırası ile kullanıcıya okutulmuştur. Okutulma sırası; "Politika → Politika → Teknoloji → Moda → Moda" olduğu varsayılmıştır. Bu durumda küme kısıt değeri üç olduğu için ilk üç haber kümelere yerleşir ve daha sonra gelecek olan iki haber bunu takiben bu üç kümeden birine yerleşecektir. Önerilen beş haber kullanıcıya okutulduktan sonra oluşan kümeler Şekil 4.14'de gösterilmiştir. Kümelere detaylı bakıldığında moda haberlerinin arasında bir adet politika haberi görülmektedir. Bu yanlış kümelemeyi düzeltmek tez kapsamında savunulan ve geliştirilen optimizasyon yönteminin ana amacıdır.



Şekil 4.14. Haber optimizasyon öncesi kümelerin görünümü

Tez çalışmasının ikinci bölümünde bahsedilen beklenen ağırlıklı kapsam yoğunluğu yaklaşımı dağıtık bir yaklaşımdır.[4] Bir haber kendisine çok yakın X kümesine girmesi yerine yeni bir küme oluşturulması BAKY değerinde daha yüksek bir sonuç verebilir.[4] Bu durumu engellemek için kısıt değeri konulduğu bölüm başında tartışılmıştır. Bu durum optimizasyonda ele alındığında en büyük sorun; "eğer bir

[71] ing: *cluster constraint*

kümede tek haber varsa optimizasyon ne olacak?" sorusudur. Eğer bir kümede tek haber varsa o haber aynı haberlerden oluşan bir kümeye konulsa bile sonuç tek başına ayrı bir küme olmasının daha iyi olacağı yönündedir. Tez kapsamında yapılan çalışmada bu durumu engellemek için kümede tek bir haber kalmışsa o haber için optimizasyon dışı olması önerilmiştir.

Önerilen optimizasyon yöntemi örnek haberlere uygulandığında tüm iterasyonlar[72] Tablo 4.5, 4.6, 4.7, 4.8 ve 4.9'da gösterilmiştir.

Varolan BAKY = 0.985782638414218 / N = 5						
Haber	Küme	AKY	AKY`	BAKY	BAKY (Δ)	
4342	1	1.5263157894	1.526315789412	0.985782638414	0	✓
4342	2	1.8571428571	2.037037037031	0.716498316498	-0.2692843384	✗
4342	3	1.5454545454	1.533333333333	0.678095238095	-0.3076874004	✗

Tablo 4.5. Haber optimizasyon doğrulama iterasyon I

Tablo 4.5 incelendiğinde 4342 numaralı haberin tüm kümeler için denendiği görülmektedir. 4342 numaralı haber ikinci ve üçüncü kümede beklenen ağırlıklı kapsam yoğunluğu düşmüştür ve değişimin negatif olduğu izlenmektedir. Bu sebeple 4342 numaralı haberin kendi kümesinde yani birinci kümede kalması daha uygun görülecektir ve algoritma bu yönde bir seçim yapmaktadır.

Varolan BAKY = 0.985782638414218 / N = 5						
Haber	Küme	AKY	AKY`	BAKY	BAKY (Δ)	
4343	1	1.5263157894	1.518518518521	0.872053872053	-0,1137287684	✗
4343	2	1.8571428571	1.857142857171	0.985782638414	0	✓
4343	3	1.5454545454	1.526315789473	0.869785575048	-0,1159970684	✗

Tablo 4.6. Haber optimizasyon doğrulama iterasyon II

Tablo 4.6'da 4343 numaralı haberin denendiği kümeler gösterilmektedir. Birinci ve üçüncü kümedeki yoğunluk değerleri varolan yoğunluk değerinden düşük çıkmıştır.

[72] ing: *iteration*

Sistem haberin ikinci kümede kalmasını sağlamaktadır. Varolan beklenen ağırlıklı kapsam yoğunluğu değeri değişmemiştir.

Varolan BAKY = 0.985782638414218 / N = 5						
Haber	Küme	AKY	AKY`	BAKY	BAKY (Δ)	
4345	1	1.5263157894	1.424242424242	0.870129870129	-0.1156527612	✗
4345	2	1.8571428571	1.857142857171	0.985782638414	0	✓
4345	3	1.5454545454	1.533333333333	0.840000000000	-0.1457826315	✗

Tablo 4.7. Haber optimizasyon doğrulama iterasyon III

Üçüncü iterasyon'da yine moda kategorisine ait 4345 numaralı haber işlenmiştir. haber önce birinci küme, daha sonra ikinci küme ve en son üçüncü kümeye sokularak yoğunlukları ve beklenen ağırlıklı kapsam yoğunlukları hesaplanmıştır. Sonuçlar Tablo 4.8'de verilmiştir. Çizelgedeki BAKY delta değerleri dikkatli incelendiğinde birinci ve üçüncü kümede değerlerin negatif'e düştüğü gözlemlenmiştir. Bu nedenle 4345 numaralı haberin ikinci kümede kalması uygun görülmüştür.

Varolan BAKY = 0.985782638414218 / N = 5						
Haber	Küme	AKY	AKY`	BAKY	BAKY (Δ)	
4346	1	1.5263157894	1.937560000000	1.132954545454	0,1471719154	✓
4346	2	1.8571428571	1.857142857171	0.985782638414	0	✓
4346	3	1.5454545454	1.533333333333	0.840000000000	-0.1457826315	✗

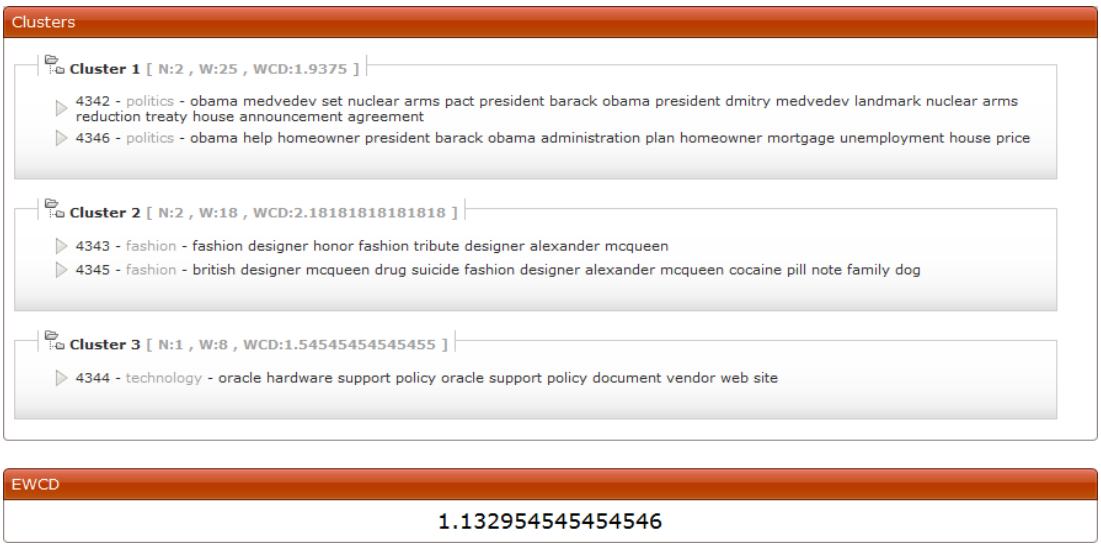
Tablo 4.8. Haber optimizasyon doğrulama iterasyon VI

Tablo 4.8'de gösterilen beşinci iterasyon incelendiğinde 4345 politka kategorisine sahip haberin birinci küme ile etkileşiminden doğan yoğunluk değişimi pozitiftir. Bu nedenle 4345 numaralı haber hem birinci küme hemde ikinci küme için adaydır. Bu durumda altogritma bu iki kümeden değişimin en büyük olduğu yeri seçecektir. 4546 numaralı haber birinci kümeye girmeye hak kazanır. Haberler arası küme değişimi olduğundan varolan beklenen ağırlıklı kapsam yoğunluk değeri yeniden hesaplanacak ve değişime uğrayacaktır. Bu değişimin, tez çalışmasında savunulan algoritma gereği yükselmesi beklenmektedir.

Varolan BAKY = 1.132954545454546/ N = 5						
Haber	Küme	AKY	AKY`	BAKY	BAKY (Δ)	
4344	1	1.9375000000	1.837209302325	0.803805496828	-0,32914904400	✖
4344	2	2.1818181818	1.969696969699	0.781439393939	-0,35151515151	✖
4344	3	1.5454545454	1.545454545454	1.132954545454	0	✔

Tablo 4.9. Haber optimizasyon doğrulama iterasyon V

Örnek olarak seçilen beş haber verisi üzerinde yapılan optimizasyonun son aşaması olan beşinci iterasyonun hesaplama sonuçları Tablo 4.9'da verilmiştir. Tablo detaylı incelendiğinde 4344 numaralı haberin sırasıyla birinci, ikinci ve üçüncü (kendi kümesi) üzerindeki değerlerine bakıldığında ikinci ve üçüncü küme yoğunluk değişimleri negatif oludğundan 4344 numaralı haberin kendi kümesinde kalması uygun görülmüştür.



Şekil 4.15. Haber optimizasyon sonrası kümelerin görünümü

Örnek veriler üzerinde gerçekleştirilen beş iterasyonlu optimizasyon sonrası kümelerin son durumu Şekil 4.15'de ekran görüntüsü olarak gösterilmiştir. Görüntü detaylı incelendiğinde birinci kümede politika kategorisine ait haberler, ikinci kümede moda kategorisine ait haberler ve son kümede ise teknoloji kategorisine ait haber bulunmaktadır. Optimizasyon öncesi ve sonrası beklenen ağırlıklı kapsam yoğunluğu değerleri incelendiğinde ise optimizasyon öncesi 0.985782638414218 olan değer optimizasyon sonrası 1.132954545454546 değerine ulaşmıştır. Bu sonuç bize kümeler üzerinde bir iyileştirmenin yapıldığını kanıtlamaktadır.

4.6. En İyi Küme Optimizasyonu

Kategorik veriler için gerekli olan uzaklık değerlendirme eksikliğinden dolayı sayısal veri kümelemede kullanılan teknikler, kategorik veri kümelemede kullanmak için uygun değildir.[9] Bu sebeple kümeleme[73] sonuçlarını veren görselleştirme aracıyla birlikte en iyi küme sayısı bulan bir uygulama olarak BKPlot yöntemi öne sürülmüştür.[7]

En iyi K görselleştirme metodu[74] (BKPlot) Georgia Tech'de geliştirilmiştir.[6] Belirli alandaki en iyi aday K'ları[75] bulmadaki arama alanını büyük ölçüde düşüren BKPlot metodu, değişken K'lı kümelenme yapıları arasındaki entropi farklılıklarını inceler ve sadece kümelenme yapısının önemli derecede değiştiği yerdeki K'ları en iyi K'lara aday olarak gösterir.[6]

Bu bölüm şu şekilde organize edilmiştir; Bölüm 4.6.1'de en iyi küme sayısını saptamak için gerekli olan işlemsel küme modu farklılığından bahsedilecek. Bölüm 4.6.2'de en iyi küme sayısını bulmak için BFI ve TBFI yöntemleri ele alınacaktır.

[73] ing: *clustering*
[74] ing: *visualization method*
[75] ing: *candidate K*

4.6.1. İşlemsel Küme Modu Farklılığı

Küme içi benzerliğinin yanısıra küme içi farklılığı da kümelerin kalitesini ölçmek için kullanılmaktadır. İşlemsel küme modu[76] farklılığı Bölüm 2.2'de anlattığımız kapsam yoğunluğu algoritması temeline dayanmaktadır. Deneyler, işlemsel küme modu farklılığı ile en iyi küme sayısını bulmada kullanılacak en verimli ölçümdür.[5]

İşlemsel küme modu haber kümesindeki her kelimenin geçme sıklığı kullanıcının belirlediği işlemsel oranın alt kümesidir.[5] Ck: K kümesi için işlemsel küme modu bileşenleri şöyledir; Nk: işlemler (haber sayısı), Mk: ayrık nesneler (kelimeler), θ : kullanıcının belirlediği en düşük destek değeridir. $I_{kj} \mid frekans(Ikj)$: I_{kj} nesnesi için j kelimesinin gelme sıklığıdır. Ck için İşlemsel küme modu CMk 4.5'de formülize edilmiştir.[5]

$$CM_k = \{ I_{kj} \mid frekans(I_{kj}) \geq (N_k \times \theta), 1 \leq j \leq M_k \} \tag{4.5}$$

Ayrık kelimeler Formül 4.5'de şu kurala göre alınmaktadır; Kullanıcının belirleyeceği bir θ destek değeri N_k haber sayısı ile çarpılarak bir değer elde edilir. Bu değer, kümedeki her ayrık kelime için o kelimenin geçme sıklığı hesaplanan sonuçtan büyükse kabul edilir. θ tekrar sayısı eşik değerinden büyük olan kelimelerin kesişim kümesi bu iki kümenin birleşek bilirlik olası artırıcı bir etken olarak düşünülmüştür.

İşlemsel küme modu farklılığı[77] iki küme arasındaki modların ayrıklığının ölçülmesiyle bulunur. Verilen iki küme C_i ve C_j için düşünecek olursak, bu iki küme arasındaki küme modu CM_i and CM_j olacaktır. C_i ve C_j kümeleri arasındaki farklılık değeri 4.6'daki formül ile bulunur.[5]

$$dm(C_i, C_j) = 1 - CD(CM_i \cup CM_j) \tag{4.6}$$

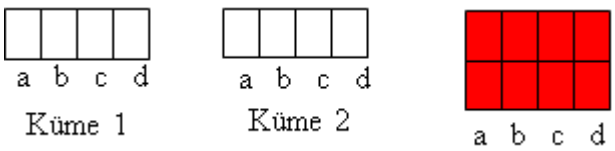
[76] ing: *transactional cluster modes*
[77] ing: *cluster mode dissimilarity*

CD: birinci bölümde açıklanan kapsam yoğunluğu[78] algoritmasına dayanmaktadır. Formül 4.6 detaylandırıldığında Formül 4.7 elde edilmektedir.[5]

$$dm(C_i, C_j) = 1 - \frac{|CM_i| + |CM_j|}{2 \times |CM_{ij}|} \qquad (4.7)$$

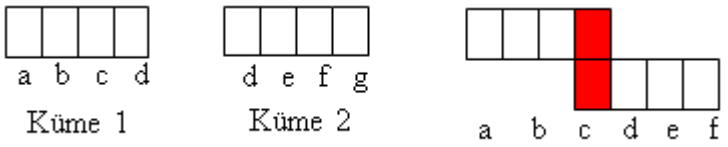
$$|CM_{ij}| \geq \max\{|CM_i|, |CM_j|\}$$

$|CM_{ij}|$ i ve j kümeleri birleştiğinde oluşan ayrık kelime sayısı formül 4.5 denkleminde anlatılmıştır.[5] Ayrık kelime sayısı seçilirken iki kümenin en büyük alt değere[79] sahip olanı seçilmelidir. Çıkarılacak sonuç: $dm(C_i, C_j) = 0$ ile $1/2$ arasında bir gerçel sayıdır.[5] İki küme tamamen aynı kelimeleri içeriyorsa CMi = CMj = CMij ve $dm(C_i, C_j) = 0$. İki kümedeki kelimelerin birbirleri üzerinde hiç bir etkisi bulunmuyorsa CMij = CMi + CMj $dm(C_i, C_j) = 1/2$ olur ve en büyük ayrıklığı oluşturur.[5] Eğer iki küme içerisindeki kelimelerin birbirleri üzerinde kısmen etkisi bulunuyorsa ayrıklık $dm(C_i, C_j) = 0$ ile $1/2$ arasında gerçel bir değer alır. Bu durumu özetlemek için üç örnek Şekil 4.16, 4.17, 4.18'de verilmiştir. İlgili şekillerde gösterilen harfler küme içindeki ayrık kelimeleri temsil etmektedir.

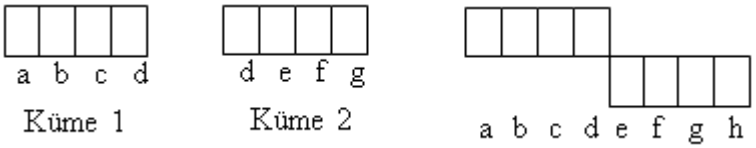


Şekil 4.16. İki küme arası sıfır ayrıklık [5]

[78] ing: *coverage density*
[79] ing: *treshold*



Şekil 4.17. İki küme arası pozitif ayrılık [5]



Şekil 4.18. İki küme arası maksimum ayrılık [5]

İşlemsel küme modu farklılığı, kümeler arası farklılığı yansıtmaktadır. Yapısal olarak iki küme benzer olduğu zaman, kümeleri birleştirmek büyük bir yapısal değişime neden olmaz.[5] Kümeler birbirinden çok farklı olduğu zaman, kümeleri birleştirmek büyük yapısal değişime neden olur ve bu yüzden kümeler arasında farklılık da büyük olur. Üst bölümde bahsedilen ölçüm, kümeler arası yapısal farklılıkları değerlendirir ve işlemsel küme modu farklılığı, işlemsel veriler için ideal bir kümeler arası farklılık ölçütüdür.[6]

4.6.2. En İyi Küme Sayısını Bulmak

Genel olarak istatistiksel indeks değerleri geometri ve yoğunluk dağılımına uygulanır. Tipik bir indeks eğrisi farklı K sayıda küme için istatistiksel indeks değerlerini içerir. Zirvedeki K'lar en uygun küme sayısı olarak adlandırılır.[7] Tez çalışmasında gerçekleştirilen algoritma yardımıyla en iyi küme sayısını bulmak için BFI (Birleştirme Farklılık İndeks)[80] ağacı oluşturulmuş ve TBFI (Türevsel Birleştirme Farklılık İndeks)[81] eğrisi kullanılarak en iyi küme sayısı tespit edilmiştir.

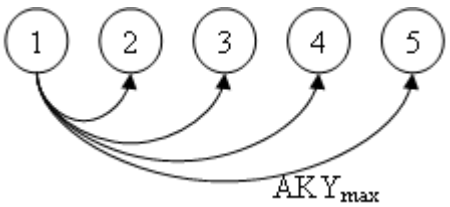
[80] ing: *merging dissimilarity index*
[81] ing: *differential merging dissimilarity index*

4.6.2.1. Birleştirme Farklılık İndeksi (BFI) Ağacının Oluşturulması

Birleştirme farklılık indeksi tüm birleştirme işlemlerinin en düşük işlemsel küme modları farklılığını tutar.[5] Diğer bir değişle BFI'da birleştirme masrafı[82] tutulmaktadır. İki küme arasındaki birleştirme masrafı kümelerin büyüklüğüne ve kelimelerin çokluğuna göre farklılık göstermektedir. 4.7 numaralı denklem ile Bölüm 4.6.1 de tartıştığımız işlemsel küme modu farklılığı bulunabilir.

$$BFI(C_i, C_j) = 1 - \frac{|CM_i| + |CM_j|}{2 \times |CM_{ij}|} \qquad |CM_{ij}| \geq \max\{|CM_i|, |CM_j|\} \tag{4.8}$$

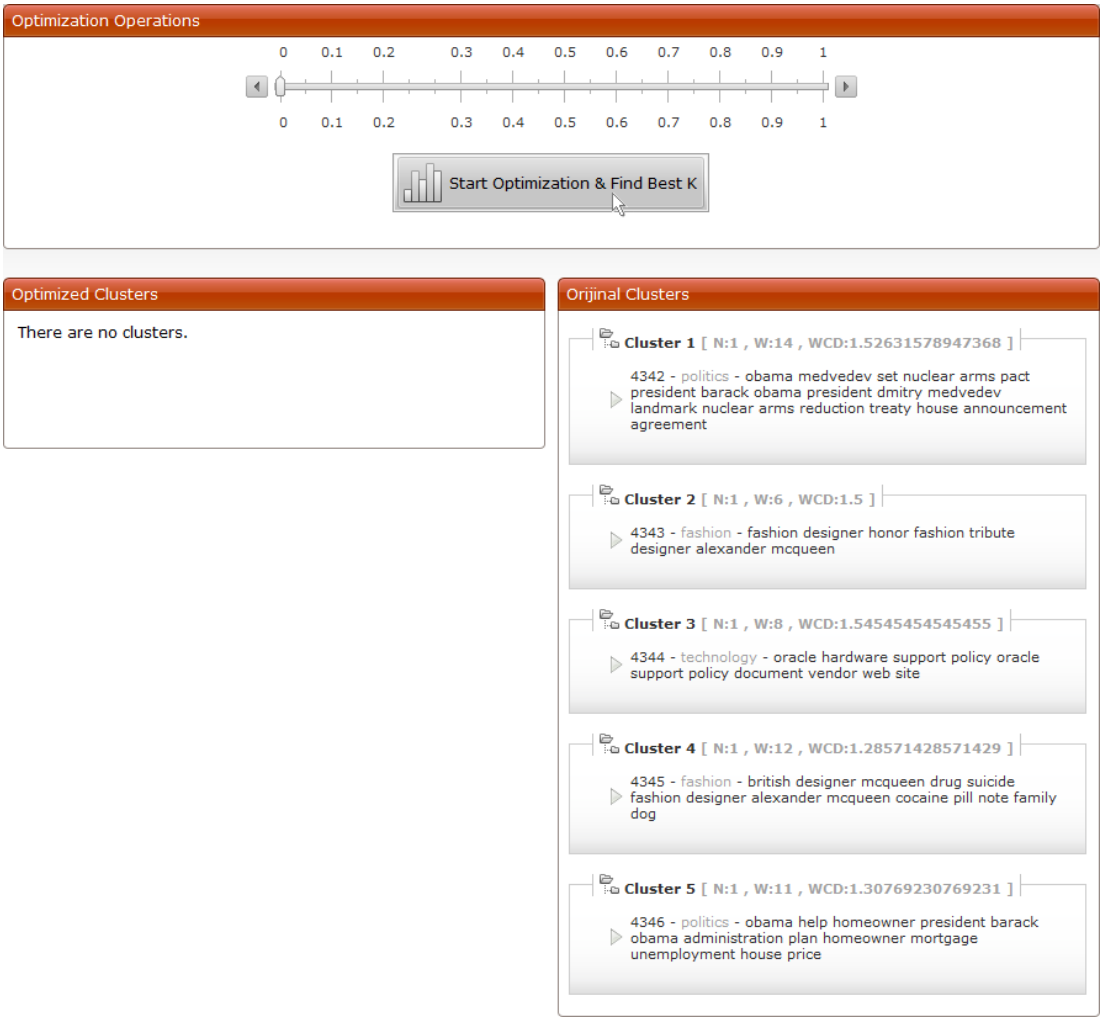
Tez kapsamında gerçekleştirilen çalışmada birleştirme masraflarını tutabilmek için bir ağaç yapısı tasarlanmıştır. Kümeler arası benzerlik ölçütleri ise ağırlıklı kapsam yoğunluğu algoritması yaklaşımı ile sağlanmaktadır. Şekil 4.19'da örnek kümeler için benzerlik ölçüt kontrolü gösterilmiştir.



Şekil 4.19. Örnek kümeler için benzerlik ölçüt kontrolü

Benzerlik ölçüt kontrolünün ana amacı iki küme arasındaki benzerliği tespit etmektir. Benzerlik ölçütleri ağırlıklı kapsam yoğunluğu algoritması kullanarak, AKYmax değeri ile birleştirilmeye en uygun yani benzerliğin en fazla olduğu kümeler olduğu savunulmaktadır. Birleştirme işlemi tek bir küme kalana kadar devam eder. Bu aşamadan sonraki adımda ise birleştirme fazı BFI ağacını oluşturur. Ağaç oluşumunda hesaplanan birleştirme masrafları veritabanı üzerinde saklanmaktadır. Bu yapı her kullanıcı için dinamik olarak tutulmaktadır. Tablo 4.10'da örnek bir işlemsel veri tablosu üzerinden birleştirme yapıldığı varsayılmıştır.

[82] ing: *merge cost*



Şekil 4.20. Küme optimizasyonu öncesi kümelerin durumu

Bölüm 4.5'de doğruladığımız haberlerden oluşan bir grup haber kullanıcıya herhangi bir kısıt değeri olmaksızın okutulmuştur. Sistem'e herhangi bir kısıt değeri belirtilmediği için okunan her haber ayrı bir öbek olarak değerlendirip Şekil 4.20'de oluşan kümelere yerleşmiştir. BFI ağacını oluşturmak için ilk çıkarılması gereken önemli tablo işlemsel veri tablosudur. İşlemsel veri tablosu kümelerdeki haberlerin ayrık kelime sayılarını gösteren ve 4.7 denklemindeki metrikleri bulmamızı sağlayacak olan en önemli verilerdir. Şekil 4.20'de gösterilen kümeler için işlemsel veri tabloları Tablo 4.10'da sunulmuştur.

Küme	Kategori	Ayrık Kelime	AKY
1	Politika	14	1.52631578947368
2	Moda	6	1.50000000000000
3	Teknoloji	8	1.54545454545455
4	Moda	12	1.28571428571429
5	Politika	11	1.30769230769231

Tablo 4.10. Kümelerin işlemsel veri tablosu

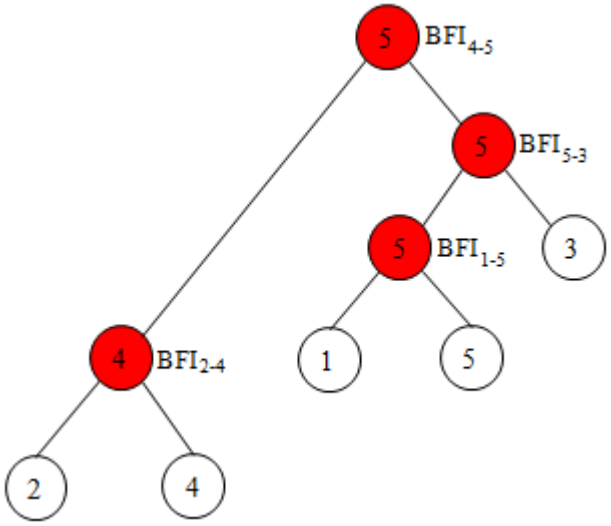
Kümeler arası benzerlik ölçütü yapılabilmesi için her küme birbirleri arasında çapraz eşleştirme işlemine[83] tabi tutulur.

Küme A	Küme B	Küme AUB AKY	BFI	AKY^{max}
1	2	1.5185185185185185	0.28571428571428571	-
1	3	1.5333333333333333	0.21428571428571428	-
1	4	1.4242424242424242	0.07142857142857123	-
1	5	1.9375000000000000	0.10714285714285712	-
2	3	1.5263157894736842	0.12500000000000000	-
2	4	2.1818181818181181	0.25000000000000000	✓
2	5	1.3809523809523812	0.22727272727272727	-
3	4	1.4000000000000000	0.16666666666666666	-
3	5	1.4166666666666666	0.13636363636363636	-
4	5	1.2962962962962962	0.04166666666666666	-

Tablo 4.11. Çapraz eşleştirme işlemi (KS^{84}=5)

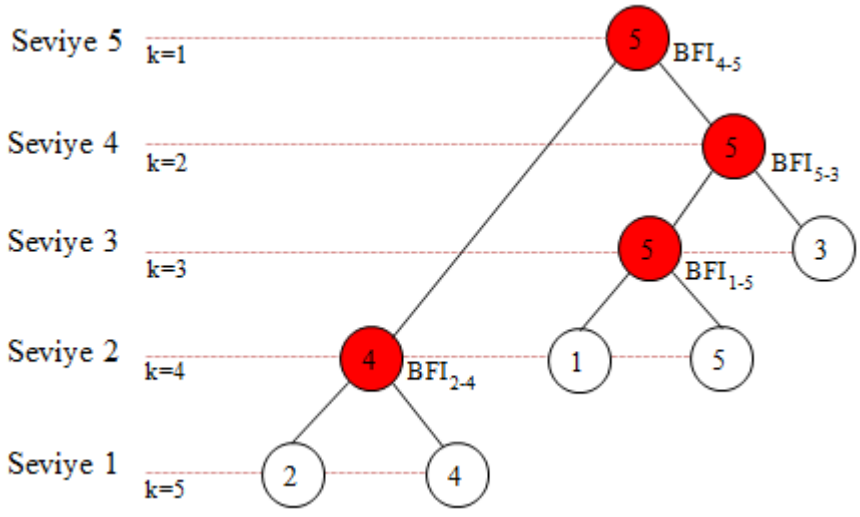
Çapraz eşitleme işleminin asıl amacı hangi kümelerin birleşerek BFI ağacını oluşturacağını bulmaktır. Tablo 4.11'de gösterildiği gibi ilk eşleştirmeden sonra ikinci ve dördüncü kümeler birleşmeye adaydır ve bu iki kümenin tüm kelimeleri ve histogramsal verileri birleştirilir. Birleştirme işlemi tamamlandığında geriye dört küme kalmaktadır. Geriye kalan dört küme birinci iterasyonda olduğu gibi tekrar birbirleri arasında çapraz eşleştirme yapılır. Bu işlemler tekrarlı bir şekilde gerçekleştirildiğinde Şekil 4.21'deki BFI ağacı oluşmaktadır.

[83] ing: *cross matching process*
[84] KS: *kalan küme sayısı*



Şekil 4.21. Örnek kümeler için oluşan BFI ağacı

Ağaç yapısı dikkatli incelendiğinde ikinci ve üçüncü küme dördüncü kümede, birinci ve beşinci küme beşinci kümede, beşinci ve üçüncü küme beşinci kümede ve son olarak dördüncü ve beşinci küme beşinci kümede birleşmiştir.



Şekil 4.22. Örnek kümeler için oluşan BFI ağacı (seviyeli gösterim)

Her birleşmede sistem üzerindeki birleştirme maliyeti[85] yani birleştirme farklılık indeksi BFI değerleri hesaplanmıştır. Bu değerler Tablo 4.12'de gösterilmiştir. Şekil 4.22'de gösterilen $BFI_{2\text{-}4}$ ikinci ve dördüncü kümenin birleşme maliyetlerini temsil etmektedir.

4.6.2.2.Türevsel Birleştirme Farklılık Eğrisinin[86] Bulunması

Türevsel birleştirme farklılık indeksi kümeler için BFI değerleri arasındaki değişimi bulmak için kullanılmıştır.[5] 4.9'daki küme için türevsel birleştirme farklılık indeks formülü önerilmiştir.[5] Bu formüle göre k kalan küme sayısını temsil etmektedir. Ağacımızın her seviyesi için BFI değeri hesaplanıp bir önceki kümenin BFI değerinden çıkarılarak değişim bulunmaktadır.

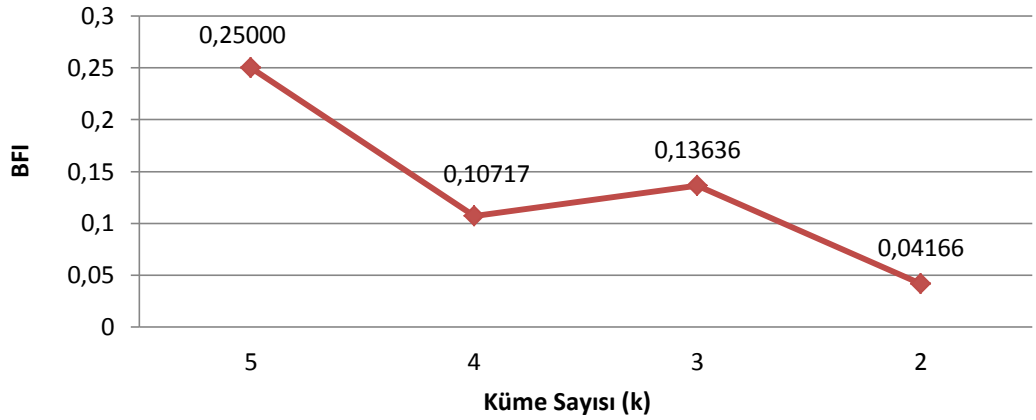
$$TBFI(k) = BFI(k) - BFI(k-1) \tag{4.9}$$

Küme Sayısı (k)	Birleşen Kümeler		BFI
	Küme A	Küme B	
5	2	4	0.25000000000000
4	1	5	0.10714285714285
3	5	3	0.13636363636363
2	4	5	0.04166666666666

Tablo 4.12. Örnek kümelerin BFI ağacı seviyelerine göre BFI değerleri

En iyi küme optimizasyonu için saklanan ve Tablo 4.12'de sunulan değerler grafikleştirip birleştirme farklılık indeks eğrisi saptanmalıdır. Bu doğrultuda küme sayısı – BFI oranı değerlerine bakılmaktadır. Kalan küme sayısı x-eksenini, birleştirme farklılık indeks değeri y eksenini temsil etmektedir. Tablo 4.12'de verilen sonuçların grafikleştirilmiş hali Grafik 4.8'de sunulmuştur.

[85] ing: *merge cost*
[86] ing: *differential merging dissimilarity index*

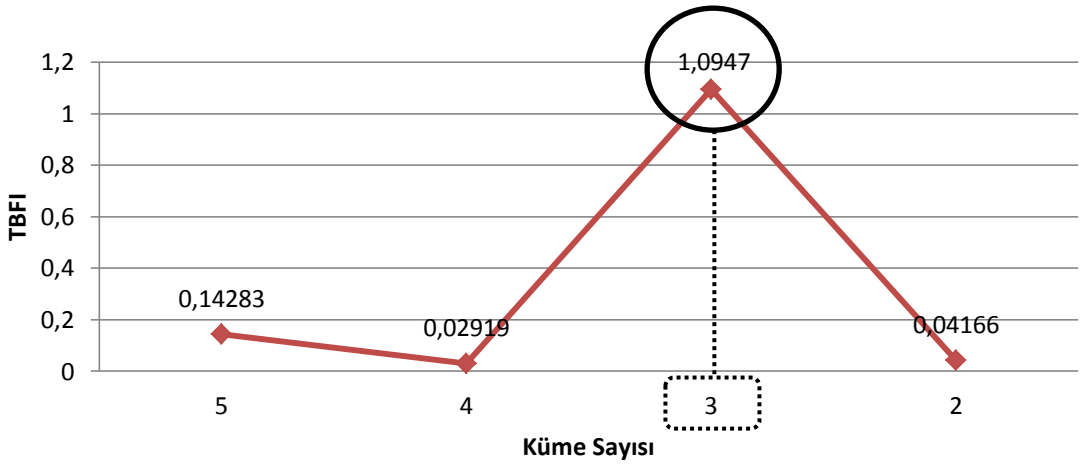


Grafik 4.8. Örnek kümeler için oluşan BFI eğirisi

Örnek haberler için oluşan BFI eğirisi üzeriden 4.9 formülündeki TBFI değerleri hesaplanacak olursa Tablo 4.13'deki sonuçlar elde edilir.

TBFI (5)	BFI(5) – BFI(4)	0,25 - 0,10717	0,14283
TBFI (4)	BFI(4) – BFI(3)	0,10717 - 0,13636	0,02919
TBFI (3)	BFI(3) – BFI(2)	0,13636 - 0,04166	1,0947
TBFI (2)	BFI(2) – BFI(1)	0,04166 - 0	0,04166

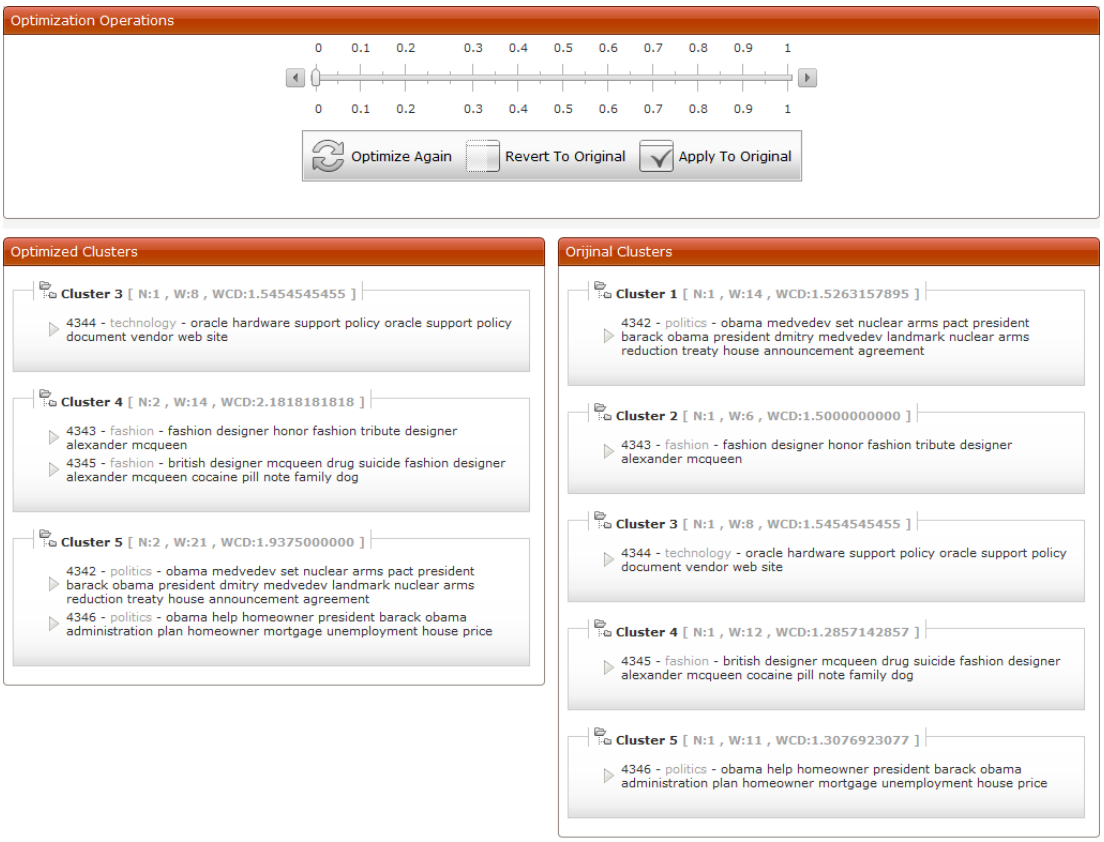
Tablo 4.13. Örnek kümeler için TBFI hesaplama sonuçları



Grafik 4.9. Örnek kümeler için oluşan Türevsel BFI Eğirisi

Beş örnek küme modelinde, türevsel BFI eğrisinin zirve noktası[87] üçtür. Beş küme için en iyi küme sayısı (BKPlot = 3) olacaktır. Küme sayısı büyüdükçe kullanıcıya bağlı bir (destek değeri)[88] θ kullanılabilir.

Sonuç olarak küçük bir TBFI eğrisi, iki komşu parçanın birleşme işleminde benzer kümelerin birleştiği ve yapıyı önemli oranda değiştirmediği anlamına gelir. Büyük bir TBFI eğrisi, birleşme işleminin küme yapısında değişime neden olabildiği anlamına gelir. Bu türevsel BFI değerleri, türevsel BFI eğrisini meydana getirir. Türevsel BFI eğrisi Grafik 4.9'da gösterilmiştir. Türevsel BFI eğrisi, aradığımız optimum aday küme sayısını içeren bir göstergedir. Şekil 4.23'de örnek haberler için küme optimizasyonu sonrası kümelerin son durumu gösterilmiştir.



Şekil 4.23. Küme optimizasyonu sonrası kümelerin durumu

[87] ing: *pick point*
[88] ing: *treshold*

4.7. Anahtar Kelime Çıkarımı

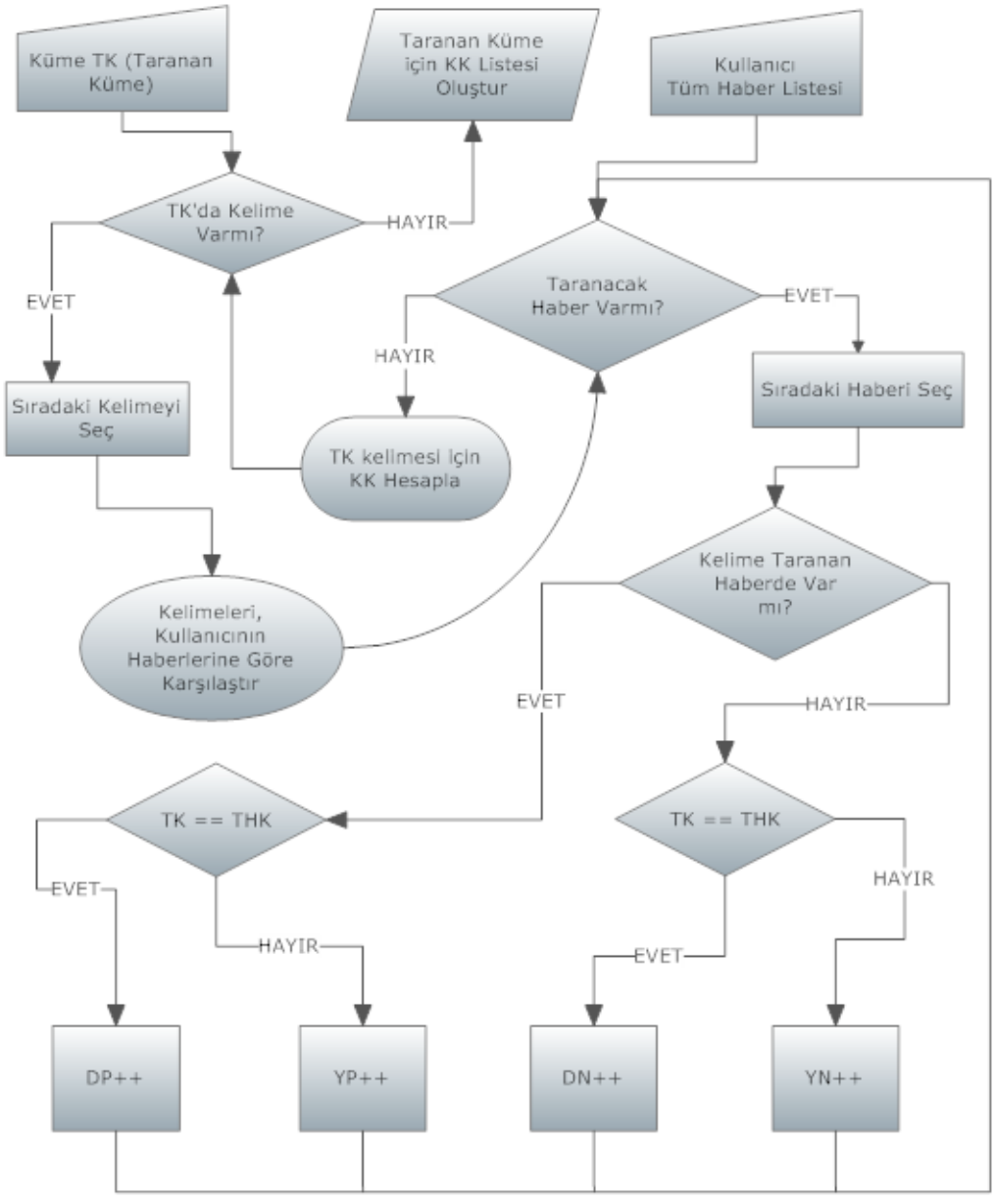
Analistlere ya da dil bilimcilere kümeleme sonuçlarını değerlendirmek için yardımcı olan önemli bir nokta her küme için etiket oluşturmak bir diğer deyişle anahtar kelime çıkarımı yapılmasıdır. Bu işlem özetleme işlemi olarak düşünülebilir. Sağlamlıkları nedeniyle çıkarım tabanlı istatistiksel yaklaşımlar benimsenmektedir.[8] Bu yaklaşımlardan çoğunlukla kullanılan en yaygın yöntem en sık tekrarlanan kelimeleri seçmektir.[8] Terimler kümelerinde kümedeki toplam frekanslarına[89] (KTF) göre sıralanır.[8] Sıralanan kelimeler arasından toplam frekans (TF) diğer bir değişle geçme sıklığı en fazla olan terimler o kümenin etiketi ya da anahtar kelimesi olarak adlandırılırlar. Yöntem çok basit olmasına rağmen kümelere bu frekans terimlerini seçmek ve isimleri temel alarak kümeleri farklılaştırmak risklidir. Çünkü bu yöntemde kümede sıkça yer alan bazı kelimelerin küme içeriği ile ilişkili olmama ihtimali vardır. Tez çalışmasında kelimeleri küme içeriği ilişkisine göre sıralayan korelasyon katsayısı yöntemine dayalı bir çözüm önerilmektedir. 4.10'da[8] T teriminin, C kümesi ile ilişkisini hesaplayan bir anahtar kelime çıkarım yöntemi formülü önerilmiştir.

$$Kk(w,C) = \frac{(DP \times DN - YP \times YN)}{\sqrt{(DP + YN)(YP + DN)(DP + YP)(YN + DN)}} \tag{4.10}$$

Doğru pozitif[90], Yanlış pozitif[91], Doğru negatif[92], Yanlış negatif[93] 'in DP, DN, YP ve YN; C kümesinde yer alan ve almayan ve w kelimesini içeren ve içermeyen haberlere işaret eder, w kelimesinin C kümesine olan korelasyon katsayısını (KK) hesaplar. Şekil 4.24'de 4.10 denkleminin gerçekleştirilen sistem üzerine nasıl uygulanacağını anlatan anahtar kelime çıkarımı akış diyagramı sunulmuştur. Korelasyon katsayıları ile hesaplama yöntemi içlerinde kısa nesne adları içeren birçok küme için etkili bir yöntemdir.[8] Ancak bu yöntem içerisinde birkaç uzun

[89] ing: *cluster total word frequency*
[90] ing: *TP True Positive*
[91] ing: *FP False Positive*
[92] ing: *DN True Negative*
[93] ing: *FN False Neagative*

nesne adı içeren kümeler için yeterince kapsamlı olmayan belirli terimler seçmeye yöneliktir. Çünkü toplam frekansı (TF) hesaba almaz. Bu nedenle korelasyon katsayıları ile hesaplama yönteminde sadece kelime frekansı, küme içindeki kelimelerin sayısını yarısı kadar arttıran terimleri seçilmektedir.



Şekil 4.24. Küme K için anahtar kelime çıkarımı akış diyagramı

Şekil 4.24'de K kümesi için anahtar kelime çıkarımı akış diyagramı verilmiştir. TK taranan kümeyi, THK taranan haberin kümesini temsil etmektedir. Taranan küme anahtar kelimelerin çıkarılacağı yani etiketlenecek olan kümeyi temsil etmektedir.

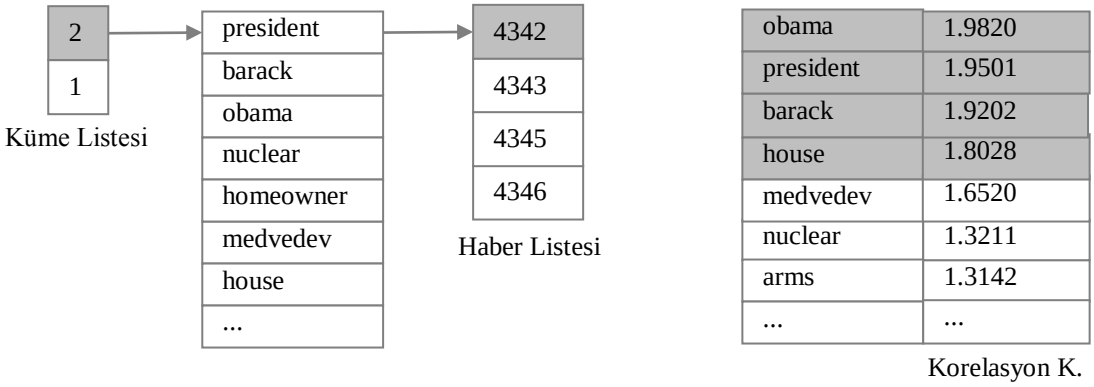
Kullanıcının okuduğu her haberin bağlı bulunduğu bir küme bulunmaktadır. Geliştirilen algoritmaya girdi[94] olarak kullanıcının küme profilleri ve okuduğu tüm haberler verilir. Kullanıcının küme profilleri ayrı ayrı işlenir ve küme kelime listesi çıkartılır. Küme kelime listesi kullanıcının okuduğu tüm haberlerin kelimeleri ile karşılaştırılır. Taranan kümenin kelimeleri okunan haberlerde geçiyor ve haberin kümesi taranan kümeye eşit ise doğru pozitif, eşit değil ise yanlış pozitif artmaktadır. Eğer taranan kümenin kelimeleri okunan haberlerde geçmiyor ve haberin kümesi taranan kümeye eşit ise doğru negatif, eşit değil ise yanlış negatifte artış beklenmektedir. İlgili korelasyon katsayıları taranan kümenin tüm kelimeleri için yapılır. Sonuç olarak her kümenin kelimeleri ve korelasyon değerlerini döndüren bir veri setimiz oluşmaktadır.

$$Kk(w,C) = \frac{(DP \times DN - YP \times YN)}{\sqrt{(DP+YN)(YP+DN)(DP+YP)(YN+DN)}} \times KaliteKatsayısı \qquad (4.11)$$

Akış diyagramında 4.11 formülü ile hesaplanan uyum değeri[95] taranan kümedeki kelimenin o kümeye ne kadar yapışkan olduğunu hesaplar. Buradaki ikilem uyum değerinin küçük çıkması ya da negatif çıkma endişesidir. Bu nedenle bir kalite katsayısı belirleyerek bu durumun önüne geçilmesi önerilmiştir. Kalite katsayı kümedeki taranan kelimenin toplam frekans sayısı[96] (TFS) olarak düşünülmektedir. Bu sayı kümedeki tüm korelasyon katsayıları (KK) ile çarpılmaktadır. Korelasyon katsayısı değeri hesaplandıktan sonra en küçük değer tekrar kontrol edilir. Eğer sıfırdan düşükse tüm liste bu değer ile çıkarılır. Tüm listenin geçici olarak bir kopyası oluşturulur. Daha sonra bu liste sıralanır. Buradaki en büyük dört değer bizim anahtar kelimelerimizi oluşturacaktır. Şekil 4.25'de anahtar kelime çıkarımının

[94] ing: *input*
[95] ing: *coherence value*
[96] ing: *total frequency count*

örnek haber üzerinde incelenmesi gösterilmiştir. Şekil 4.26'de ise örnek olarak kümelenmiş haberler için anahtar çıkarımı ekran görüntüsü verilmiştir.



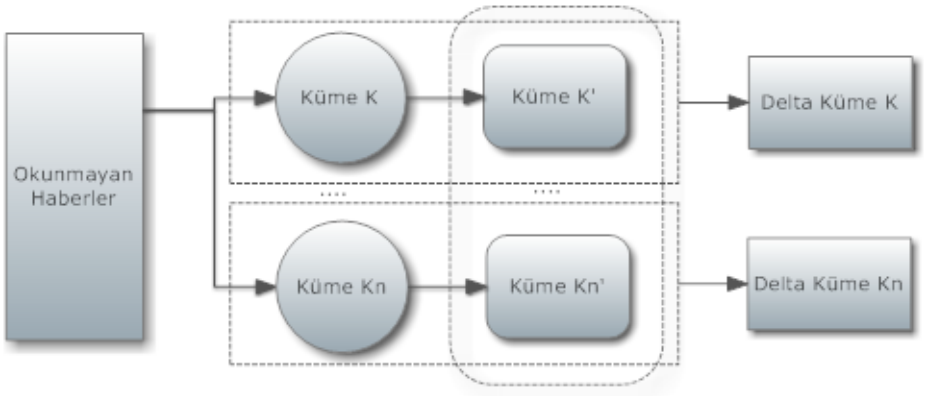
Şekil 4.25. Anahtar kelime çıkarımının örnek haber üzerinde incelemesi

Şekil 4.26. Anahtar kelime çıkarımı ekran görüntüsü

4.8. Haber Tavsiye

Tez çalışmasında önerilen haber tavsiye sisteminin ana amacı, okuyucunun önceden okuduğu haberlere dayanarak, okunmamış haberlerin profilini oluşturarak, kullanıcıya ilgili profillere göre haber tavsiye etmesidir. Gerçekleştirilen sistemi diğer çalışmalardan ayıran en büyük özellik tek profilli değil, çok profilli diğer deyişle kullanıcı bazında çok kümeli tavsiye yapabilmesidir. Kullanıcının kümelerindeki haberler ağırlıklı kapsam yoğunluğu algoritması temeline dayalı olarak işlenmekte ve güncellenmiş haberler arasından en uygun haberler tavsiye edilerek kullanıcıya sunulmaktadır.

Okunmayan haberler arasından tavsiye edilecek haberlerin seçilme işlemi varolan kümelerin yoğunluğuna ve değerlerine göre farklılık göstermektedir. Örneğin "moda" kategorisine ait haberlerin toplandığı bir kümedeki kelimeler genelde "moda, giyim.." vb. kelimeleri içerecektir. Ancak, hava kategorisine ait bir kümede bu kelimeler neredeyse hiç bulunmayacaktır.



Şekil 4.27. Haber tavsiye için delta yönteminin kümeler üzerinde uygulanması

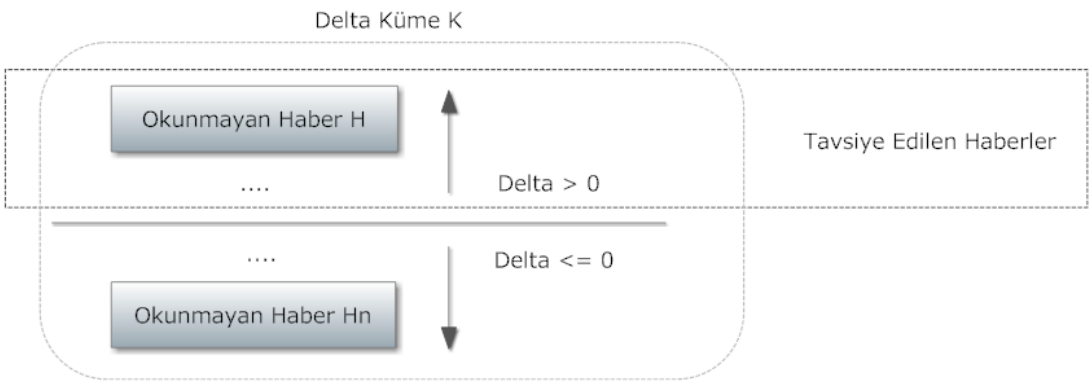
Önerilen tavsiye sisteminde kullanıcı profili ile benzer haberlerin tavsiye işlemi her küme için ayrı ayrı yapılmaktadır. Şekil 4.27'de haber tavsiye için önerilen yöntemin kümeler üzerinde gösterimi sunulmaktadır. İlk aşamada tavsiye edilebilir değeri olan

yani kullanıcının okumadığı tüm haberler kullanıcının kümelerine teker teker eklenir. Bu ekleme işlemi haberin histogramsal verileri ve kümenin histogram verilerinin birleştirilmesi temeline dayanmaktadır. Birleştirme veritabanı üzerinde yapılmaz. Birleştirilen her haber-küme ikilisi için bir yoğunluk değeri hesaplanır. Birleştirme sonrası kümenin kapsam yoğunluk değeri değişecektir. Bu da, Şekil 4.34'de ikinci aşama olarak gösterilen Küme K'yı oluşturacaktır. Son aşama olan delta değerlerini hesaplamak için 4.12'deki delta OH[97] formülü önerilmiştir.

$$\Delta_{OH} = AKYkümeX[OH] - AKYkümeX \tag{4.12}$$

Delta değeri; okunmayan haberin X kümesine aktarıldıktan sonraki kümenin ağırlıklı kapsam yoğunluk değeri (AKYKümeK[OH]) ile kümenin var olan ağırlıklı kapsam yoğunluk değerinin (AKYKümeK) farkı ile bulunur. Bu değer bize okunmayan her haber K kümesine girdiğinde oluşan değişimi vermektedir.

Her küme için delta değerleri listesi oluşturulur ve değer büyükten küçüğe doğru sıralanır. Delta değerinin sıfırdan küçük olduğu durumlarda haber kümenin yoğunluk değerine erişemediği varsayılmaktadır. Bu nedenle sıfırdan küçük deltaya sahip haberler tavsiye dışı olarak değerlendirilmiştir. Şekil 4.28'de K kümesi için delta değerine göre haber eleme şemasal olarak gösterilmiştir.



Şekil 4.28. Haber tavsiye için delta değerine göre haber eleme şemasal gösterimi

[97] *OH: okunmayan haber*

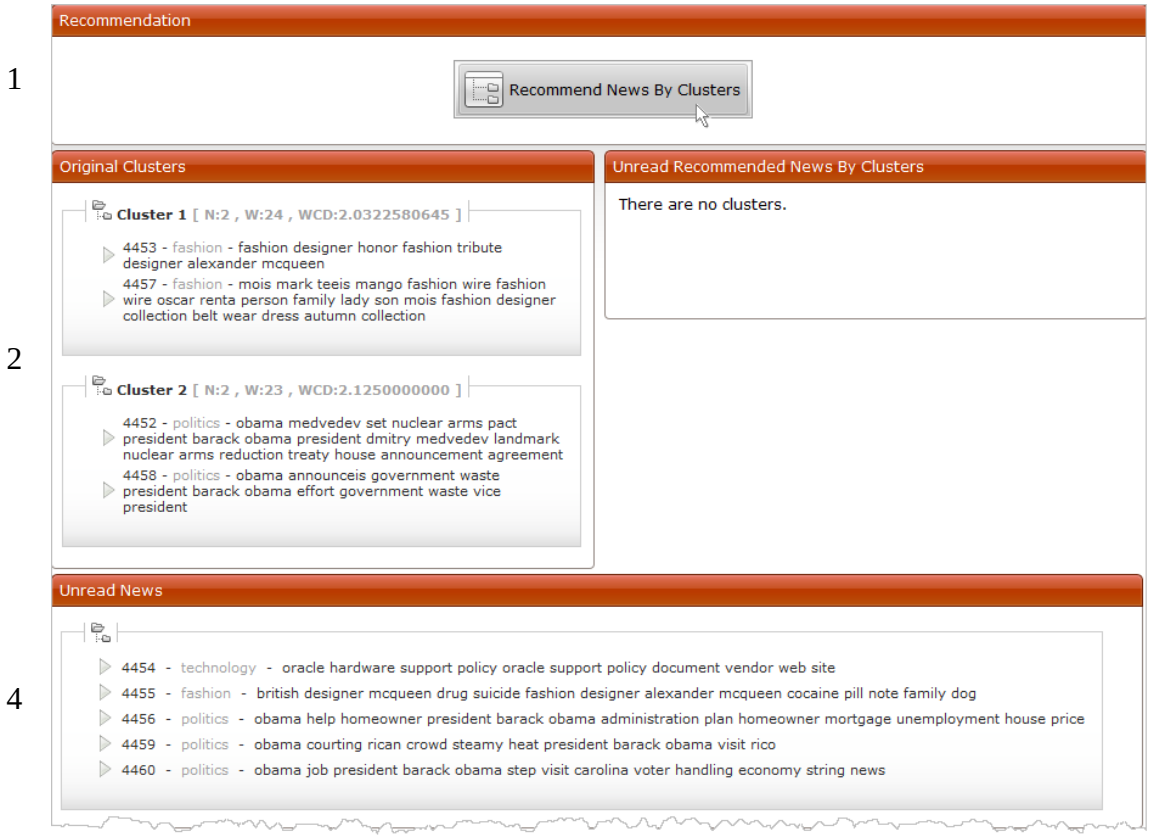
4.8.1. Haber Tavsiyenin Örnek Veriler İle Doğrulanması

Haber tavsiye kullanıcı haber doğrulaması için okutulacak ve okutulmayacak haberler Tablo 4.14'de sunulmuştur. Gri ile işaretlenmiş haberler kullanıcıya okutulan haberleri temsil etmektedir. Okuyucuya politika kategorisinden iki haber, moda kategorisinden iki haber okutulmuştur.

Haber No	Kategori	Haberin işlenmiş kelimeleri	
4452	politika	medvedev obama set nuclear arms pact president barack obama president dmitry medvedev landmark nuclear arms reduction treaty house announcement agreement	
4458	politika	obama announceis government waste president barack obama effort government waste vice president	
4456	politika	obama help homeowner president barack obama administration plan homeowner mortgage unemployment house price	
4459	politika	obama courting rican crowd steamy heat president barack obama visit rico	
4460	politika	job president barack obama step visitobama carolina voter handling economy string news	
4454	teknoloji	oracle hardware support policy oracle support policy document vendor web site	
4453	moda	fashion designer honor fashion tribute designer alexander mcqueen	
4457	moda	mois mark teeis mango fashion wire fashion wire oscar renta person family lady son mois fashion designer collection belt wear dress autumn collection	
4455	moda	british designer mcqueen drug suicide fashion designer alexander mcqueen cocaine pill note family dog	

Tablo 4.14. Haber tavsiye doğrulama için örnek haberler

4452, 4453, 4457 ve 4458 numaralı haberler kullanıcıya okutulduktan sonra bölüm 4.5'de işlenen küme içi haber optimizasyonundan geçirilmiştir. Şekil 4.29'da tavsiye öncesi kümelerin durumu ve okunmayan haberler gösterilmiştir. Şekil 4.29 dikkatli incelendiğinde bir numara ile gösterilen bölüm operasyon işlemlerini taşımaktadır. İkinci bölüm kullanıcının o anki aktif profilini yani kümlerini göstermektedir. Üçüncü bölüm kümelere göre tavsiye edilecek haberleri gösterecek bölümü ve dördüncü bölüm ise okunmayan haberleri gösteren bölümü temsil etmektedir.



Şekil 4.29. Tavsiye öncesi kümelerin durumu

Önerilen sistemde kümelere göre tavsiye işlemi başlatıldığında Şekil 4.30'un dördüncü bölümdeki okunmayan haberler sırası ile birinci küme ve ikinci kümeye sokularak ağırlıklı kapsam yoğunluk değerleri ve buna bağlı delta değerleri hesaplanacaktır. Şekil 4.29'da tüm hesaplama değerleri sunulmuştur. İlk haber olan 4454 birinci kümeye girmeden önceki kümenin AKY değeri 2.0322580645'dir. 4455 numaralı teknoloji kategorisine ait haber kümeye girdiğinde bu değer 1.9047619047619 olmaktadır. Aynı haber ikinci kümeye sokulduğunda ise önceki AKY değeri 2.1250000000 iken yeni değer 1.97674418604651 olmaktadır. Bu durumda delta değerleri şu şekilde hesaplanmaktadır;

$$\Delta_{4454-Küme1} = 2.03225806451 - 1.90476190476 = -0.127496159754$$

$$\Delta_{4454-Küme2} = 2.12500000000 - 1.97674418604 = -0.148255813953$$

4454 numaralı haberin delta hesaplama sonuçlarının ikisi de negatif çıkmaktadır. Bu durumda 4454 numaralı haber her iki küme için iyi bir aday değildir. 4455 Numaralı haber birinci kümeye girmeden önceki kümenin yoğunluk değeri 2.0322580645'dir. Birinci kümeye girdikten sonra değer 2.46666666666667 olmaktadır. 4455 Numaralı haber ikinci kümeye girdiğinde ise kümenin yoğunluk değeri 1.8695652173913'e düşmüştür. Bu durumda gerekli delta hesaplaması yapıldığında;

$$\Delta_{4455-Küme1} = 2.03225806451 - 2.466666666667 = 0.43440860215051$$
$$\Delta_{4455-Küme2} = 2.12500000000 - 1.869565217391 = -0.255434782608$$

4455 numaralı haberin delta küme1 sonuçları pozitiftir. 4455 numaralı haber 0.4344086021 delta değeri ile birinci küme için tavsiye edilir. 4456 numaralı politika haberi birinci küme ile denenmeden önce kümenin yoğunluğu 2.032258064516132 dir. Haber birinci kümeye girdiğinde yoğunluk değeri 1.81818181818182'e düşmüştür. Aynı haber ikinci kümeye girdiğinde ise ağırlıklı kapsam yoğunluk değeri 2.55555555555556'e yükselmiştir. Bu durumda delta değerleri;

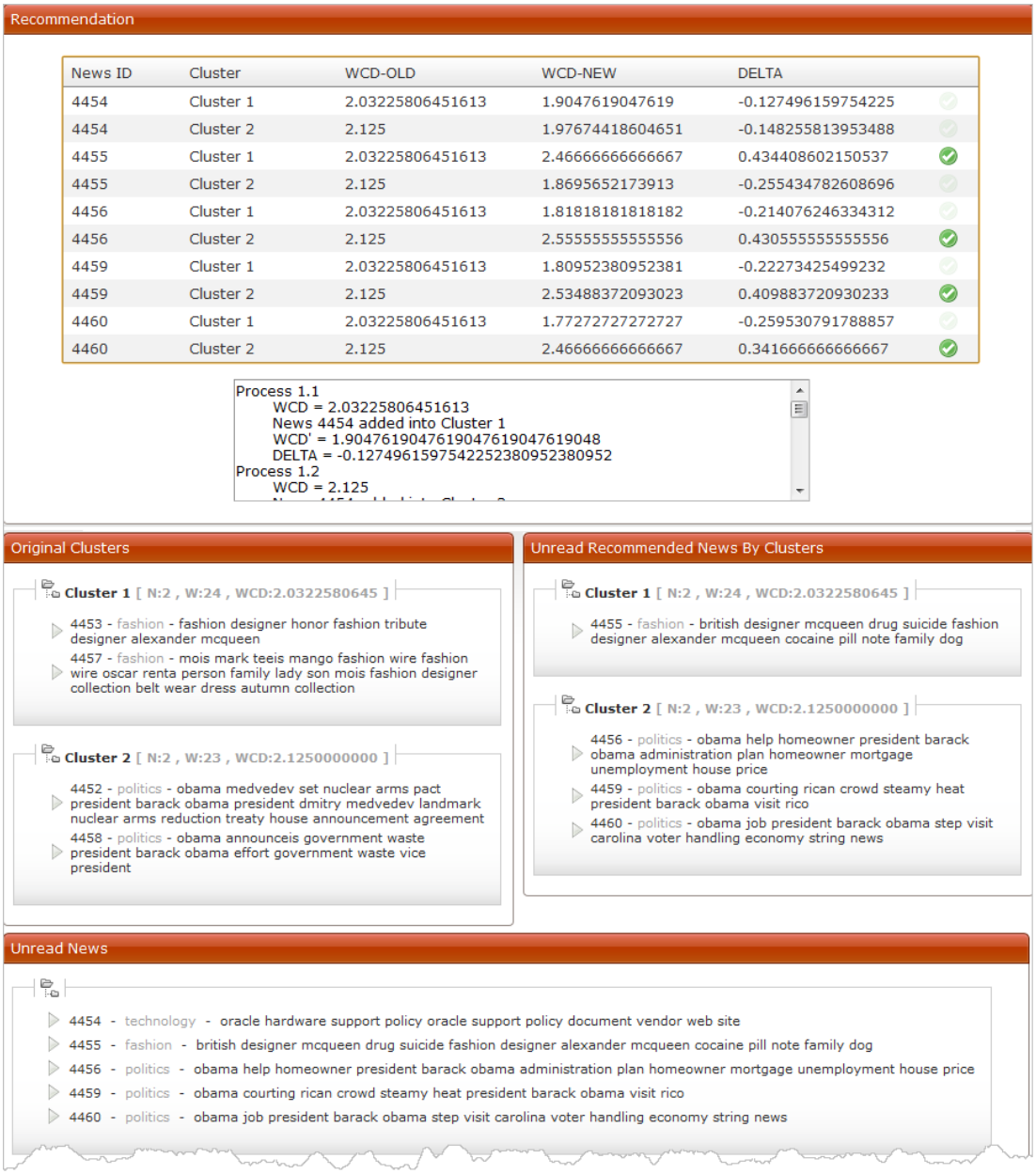
$$\Delta_{4456-Küme1} = 2.03225806451 - 1.818181818181 = -0.2140762463343$$
$$\Delta_{4456-Küme2} = 2.12500000000 - 2.55555555556 = 0.430555555555556$$

4456 numaralı haber birinci kümeye girdiğinde delta değeri negatif ikinci kümeye girdiğinde delta değeri pozitif çıkmaktadır bu durumda 4456 numaralı haber 0.430555555555556 delta değeri ile ikinci küme için uygun bir adaydır. 4459 Numaralı politika haberi birinci kümeye girdiğinde yoğunluğu 1.80952380952381'e düşürmüştür. İkinci kümeye girdiğinde ise yoğunluğu 2.53488372093023'e çıkartmıştır. Bu durumda ilgili haberin delta değerleri şu şekilde hesaplanır;

$$\Delta_{4459-Küme1} = 2.03225806451 - 1.8095238095238 = -0.222734254992$$
$$\Delta_{4459-Küme2} = 2.12500000000 - 2.534883720930 = 0.40988372093023$$

Sonuç olarak 4456,4459 ve 4460 numaralı politika kategorisine ait haberlerin delta değeri ikinci kümelerde pozitif olduğu için delta oranlarına göre yakınlık sıralaması

olarak 4456 ilk, 4459 ikinci ve 4460 üçüncü sırada olmak üzere ikinci küme için tavsiye edilirler. 4455 numaralı moda kategorisine ait haber birinci küme için uygun adaydır. 4454 numaralı teknoloji kategorisine ait haberin delta değeri iki küme içinde negatif çıktığından hiç bir küme için önerilmemektedir. Şekil 4.30'da tüm sonuçların ayrıntılı gösterimi ve kümelerin son hali sunulmuştur.



Şekil 4.30. Tavsiye sonrası kümelerin durumu ve delta sonuçları

BÖLÜM 5

5. DENEYLER

Bu bölümde, tez kapsamında geliştirilen rss tabanlı haber tavsiye sisteminin Yahoo haber veri setleri kullanılarak gerçek kullanıcılar üzerinde test edilmesi, ölçümleri ve sonuçları tartışılacaktır. Bu bölüm şu şekilde organize edilmiştir; ilk olarak Bölüm 5.1'de deneysel verilerde doğruluğu saptamayı sağlayan doğruluk ölçümü yönteminden bahsedilecek. İkinci olarak Bölüm 5.2'de deney sonuçlarını grafik ile yorumlamamızı sağlayacak olan alıcı işletim karakteristiği uzayı hakkında bilgi verilecek daha sonra Bölüm 5.3'de ölçümlerin alınması ve kullanıcı haber tavsiye testi uygulamasından bahsedilecek ve Bölüm 5.4'de ölçümlerden elde edilen verilerin gerçek kullanıcılar üzerindeki test sonuçları aktarılarak bölüm sonlandırılacaktır.

5.1. Doğruluk Ölçümü

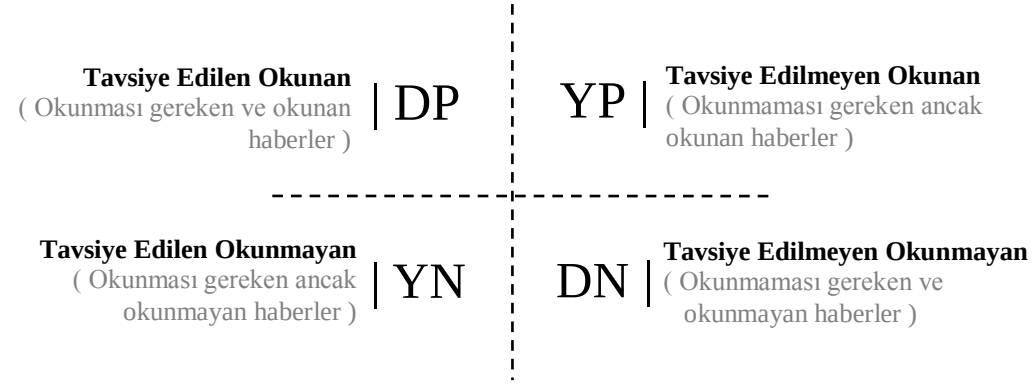
Doğruluk ölçümü sınıflandırıcı sistemlerin performans karşılaştırılmasında yaygın olarak kullanılan bir metrik hesaplama yöntemidir.[31] Tez çalışmasında doğruluk ölçümünün etkin olarak yapılabilmesi için ikili sınıflandırma hata matrisi yöntemi kullanılmıştır. İkili sınıflandırmada[98], çoğunluk sınıfı[99] negatif sınıf olarak ifade edilirken, azınlık sınıfı[100] genellikle pozitif sınıf olarak ifade edilir.

Çizelge 5.1. İkili sınıflandırma hata matrisi sınıflandırma modeli. [31]

		Tahmini Sınıf	
		+	−
Gerçek Sınıf	+	$f++$ (DP)	$f+-$ (YN)
	−	$f-+$ (YP)	$f--$ (DN)

[98] ing: *binary classification*
[99] ing: *majority class*
[100] ing: *rare class*

Çizelge 5.1'de ikili sınıflandırma için bir hata matrisi örneği gösterilmiştir. Hata matrisi sınıflandırma modeliyle, doğru ya da yanlış tahmin edilen örnek miktarlar özetlenir. Hata matrisinde tablolaştırılan hesaplamaları işaret etmek için beş terim kullanılır. Doğru Pozitif[101] (DP) ya da $f++$,sınıflandırma modeliyle doğru tahmin edilen pozitif örnek sayısına karşılık gelir yani sistem tarafından kullanıcıya tavsiye edilen ve okunan haber sayısını temsil eder. Yanlış Negatif[102] (YN)ya da $f+-$, sınıflandırma modeliyle negatif olarak yanlış tahmin edilen pozitif örnek sayısına karşılık gelir yani sistem tarafından kullanıcıya tavsiye edildiği halde okunmayan bir diğer deyişle okunması gereken ancak okunmayan haber kümesini temsil eder. Yanlış Pozitif[103] (YP) ya da $f-+$, sınıflandırma modeliyle pozitif olarak yanlış tahmin edilen negatif örnek sayısına karşılık gelir. Geliştirilen sistem tarafında ise kullanıcıya tavsiye edilmeyen ve okunan yani okunmaması gereken ancak okunan haberler kümesini göstermektedir. Son olarak Doğru Negatif[104] ya da $f--$, sınıflandırma modeliyle doğru tahmin edilen negatif örnek sayısına karşılık gelir. Kullanıcıya tavsiye edilmeyen ve okunmayan yani okunmaması gereken ve okunmayan haber kümesini temsil etmektedir. Önerilen sistem üzerinde işaretlerin özeti Şekil 5.1'de hata matris modeli üzerinde gösterilmiştir.



Şekil 5.1. Hata matris modelinin sisteme uyarlanması

[101] ing: *True Positive* (**TP**)
[102] ing: *False Negative* (**FN**)
[103] ing: *False Positive* (**FP**)
[104] ing: *True Negative* (**TN**)

Hata matrisindeki hesaplamalar, ayrıca yüzde olarak da ifade edilebilir. Doğru Pozitif Oranı[105] (PDO) ya da hassasiyet, doğru tahmin edilen pozitif örnek sayısının oranıdır. Kullanıcıya tavsiye edilen ve okunan haberlerin doğruluk oranını temsil eder. Pozitif doğru oranı hesaplaması 5.1'deki denklem [31] ile hesaplanmaktadır.

$$DPO = \frac{DP}{DP + YN} \tag{5.1}$$

Doğru Negatif Oranı[106] (NDO) ya da keskinlik, doğru tahmin edilen negatif örnek sayısıdır. Diğer deyişle kullanıcıya sistem tarafından tavsiye edilmeyen ve kullanıcının okumadığı haber oranını temsil eder. Negatif yanlış oranı hesaplaması 5.2'deki denklem [31] ile hesaplanmaktadır.

$$DNO = \frac{DN}{DN + YP} \tag{5.2}$$

Yanlış Pozitif Oranı[107] (YPO), pozitif olarak tahmin edilen negatif örneklerin oranıdır. Kullanıcıya tavsiye edilmediği halde okunan haberlerin okunma oranıdır. Yanlış pozitiflerin oranı 5.3'deki denklem [31] ile hesaplanmaktadır.

$$YPO = \frac{YP}{YP + DN} \tag{5.3}$$

Son olarak Yanlış Negatif Oranı[108] (YNO), pozitif olarak tahmin edilen negatif örneklerin oranıdır yani kullanıcıya tavsiye edildiği halde okunmayan haberlerin oranıdır. Yanlış negatif oranı 5.4'deki denklem [31] ile hesaplanmaktadır.

$$YNO = \frac{YN}{YN + DP} \tag{5.4}$$

[105] ing: *True Positive Rate* (**TPR**)
[106] ing: *True Negative Rate* (**TNR**)
[107] ing: *False Positive Rate* (**FPR**)
[108] ing: *False Negative Rate*(**FNR**)

Doğruluk ölçümünde 5.1, 5.2, 5.3 ve 5.4 ‘deki denklemler ile hesaplanan metrikleri haricinde tez çalışmasında doğrulama amaçlı kullanılan iki metrik daha bulunmaktadır. Bunlar hatırlama[109] (r) ve hassasiyet[110] (p) metrikleridir. Hatırlama ve hassasiyet metrikleri sınıflardan birinin başarılı olarak algılanmasının, başka sınıfların algılanmasından daha önemli olarak değerlendirildiği uygulamalarda yaygın olarak kullanılan iki ölçüdür.[31] Metriklerin formülize edilmiş halleri 5.5’deki denklemler ile gösterilmektedir.

$$r = \frac{DP}{DP + YN} \qquad p = \frac{DP}{DP + YP} \tag{5.5}$$

Hassasiyet (p) sınıflandırıcının pozitif sınıf olarak ifade ettiği ve gerçekte pozitif küme içinde olan kayıt oranını belirtir. Hassasiyet arttıkça, sınıflandırıcı tarafından önerilen pozitif yanlış hatalarının sayısı düşer. Hatırlama (r), sınıflandırıcı tarafından doğru tahmin edilen pozitif örneklerin oranını ölçer. Aslında, hatırlama metriği (r) değeri, pozitif doğru oranıyla aynıdır.

Bir ölçüyü maksimize eden bazal modeller yapılandırmak mümkündür. Örneğin pozitif sınıfta olan her kaydı gösteren bir modelde hatırlama (r) değeri mükemmeldir, fakat hassasiyet (p) yetersizdir. Bu durumda zor olan, kümeleme algoritmalarında hem hassasiyet (p) hemde hatırlama (r) değerlerini maksimize eden bir model geliştirmektir. Bu ölçütler için F_1-ölçütü olarak bilinen bir metrik ile özetlenmektedir. F_1 - ölçütü formülü 5.6 denkleminde verilmiştir.

$$F_1 = \frac{2rp}{r + p} = \frac{2 \times DP}{2 \times DP + YP + YN} \tag{5.4}$$

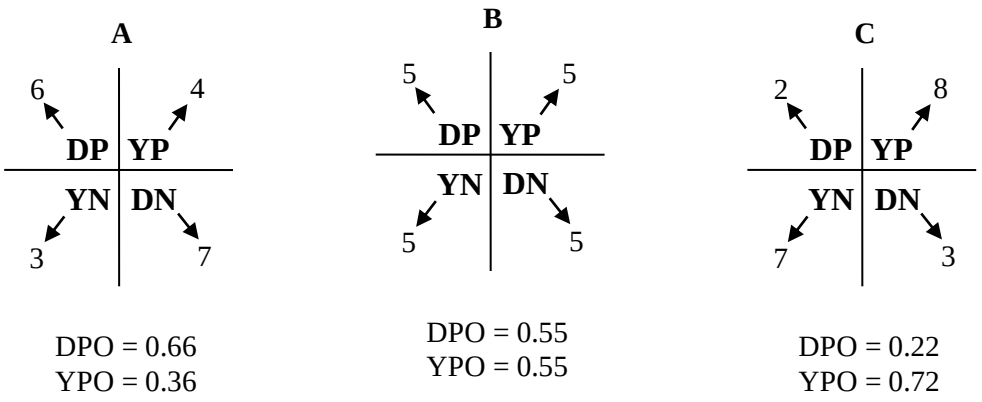
Tez çalışmasında bu bölümde anlatılan tüm metrikler hesaplanmaktadır ve kişi-tarih bazında genel başarı oranı çıkartılmaktadır.

[109] ing: *recall*
[110] ing: *precision*

5.2. Alıcı İşletim Karakteristiği (ROC Uzayı)

Alıcı işletim karakteristiği[111] gerçekleştirilen sistemin doğruluğunu ölçmeyi sağlayan hassasiyetin kesinliğe olan oranıyla oluşturulmaktadır.[33] Daha basit anlamda doğru pozitiflerin, yanlış pozitiflere olan kesri olarak da ifade edilmektedir.[33]

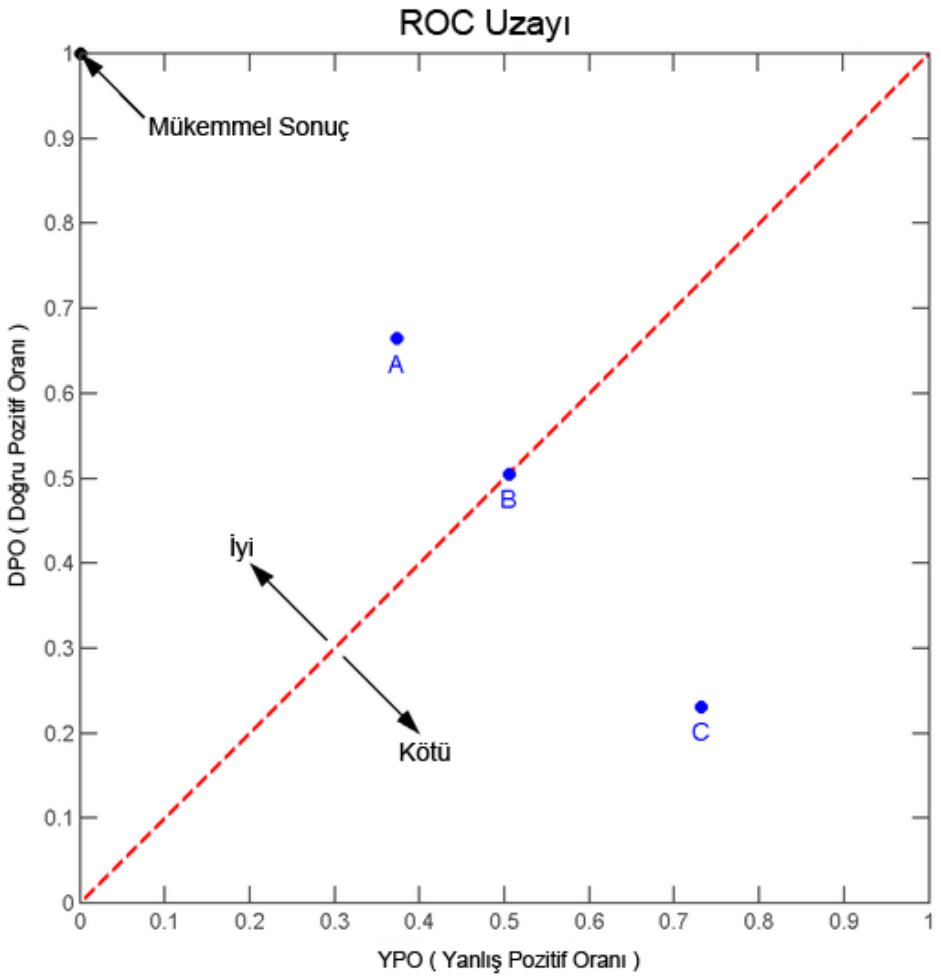
ROC uzayı, x ekseninde YPO, y ekseninde DPO olarak tanımlanır. ROC uzayının amacı her sınıflandırma işleminde yapıldığı gibi, yanlış pozitifleri eleme kabiliyetinin ve doğru pozitifleri tespit etme kabiliyeti arasındaki dengeyi kurmaktır. Her bir tahmin sonucu veya hata matrisindeki bir durum, ROC uzayında bir noktayı temsil eder. ROC uzayında olabilecek en iyi tahmin, grafiğin en üst solundaki noktayı veya koordinat olarak ROC uzayının (0,1) noktasını verir. Bu nokta, %100 hassasiyet (yani sıfır negatif yanlış) ve %100 keskinlik (yani sıfır pozitif yanlış) gösterir. (0,1) noktası ayrıca, mükemmel sonuç olarak da adlandırılır. Köşegen doğrusu, ROC uzayını ikiye böler. Köşegenin üzerindeki noktalar iyi sınıflandırma sonuçlarını, altındaki noktalar kötü sonuçları gösterir. Örnek olarak 10 pozitif ve 10 negatif durum için üç hesaplama sonucu incelenirse Şekil 5.2'deki değerler oluşmaktadır. Örnek değerler ile oluşan ROC uzayı grafiği Şekil 5.3'de verilmiştir.



Şekil 5.2. On pozitif ve on negatif durum için hata matrisi

[111] ing: *receiver operating characteric(ROC)*

Şekil 5.2'de gösterilen örnek verilerin hesaplamaları sonucu oluşan ROC uzay grafiği Şekil 5.3'de gösterilmiştir. Şekil 5.3'de gösterildiği üzere ROC uzayı içinden dört sonuç verilmektedir. A'nın DPO oranı yüksek olduğu için ROC uzayının pozitif tarafında kalmaktadır. B ise nötr durumdadır. C'nin YPO değeri DPO değerine göre yüksek olduğu için ROC uzayında kötü tarafta yer almaktadır.



Şekil 5.3. ROC Uzayı grafik gösterimi [33]

Sonuç olarak, ROC uzayı tez çalışmasında haberlerin doğru tavsiye edilip edilmediğini doğrulamak amaçlı kullanılmaktadır. Bölüm 5.3 ölçümlerin alınmasının ardından Bölüm 5.4'de sonuçlar incelendiğinde ROC uzayının kişi bazlı sonuçları aktarılacaktır.

5.3. Ölçümlerin Alınması

Tez çalışmasında geliştirilen sistemin doğruluğunu kanıtlamak için her kullanıcıya ayrı ayrı sunulmak üzere kullanıcı haber tavsiye testi isimli bir test bölümü hazırlanmıştır. Test aşaması genel olarak üç bölümden oluşmaktadır. İlk bölüm sistemin eğitilmesi, ikinci olarak küme optimizasyonun gerçekleştirilmesi ve son olarak kullanıcıya sunulan haber tavsiye testinin gerçekleştirlmesidir.

5.3.1. Veri Setinin Seçimi

Internet ortamında oldukça fazla sayıda haber kaynağı bulunmaktadır. Sistemin testleri için toplamda seksen kategoriden oluşan ve geniş bir kullanıcı kitlesi tarafından aktif olarak kullanılan Yahoo rss haber kaynağı [32] veri setleri kullanılmıştır. Seçimdeki en büyük etken kullanıcıların web ortamında en çok hangi haber kaynaklarını takip ettiği ve çeşitli rss programlarının son kullanıcıya varsayılan olarak sunduğu rss programları göz önüne alınmıştır.

5.3.2. Test Kullanıcılarının Seçimi

Ölçümlerin alınması için kategori çeşitliliği olması açısından çeşitli yaş aralıklarında on kişi test kullanıcısı olarak seçilmiştir. Tablo 5.1'de test kullanıcıları listelenmiştir.

Kişi No	Kişi Ad	Kişi Özellik	İlgilendiği Kategoriler
1	Alper Ö.	35 Yaş Erkek	A1,D1,D8,D6,G1
2	Beyhan Y.	47 Yaş Erkek	B4,C1,H1,H2,H4
3	Çiğdem D.	56 Yaş Bayan	G3,F1,E3,G1,G7
4	Fisun S.	36 Yaş Bayan	D1,E2,E3,E4,E5
5	Kevser Y.	45 Yaş Bayan	E5,E6,G1,G5,G7
6	Murat D.	58 Yaş Erkek	A1,A3,B1,B2,B5
7	Özgür D.	19 Yaş Erkek	D10,D2,D6,D7,D8
8	Pınar D.	26 Yaş Bayan	C1,D1,G1,H3,G3
9	Umay A.	12 Yaş Bayan	E1,E3,E4,E5,G7
10	Zerrin Y.	27 Yaş Bayan	D2,E2,E4,E5,G1

Tablo 5.1. Doğruluk testi için kullanıcı listesi

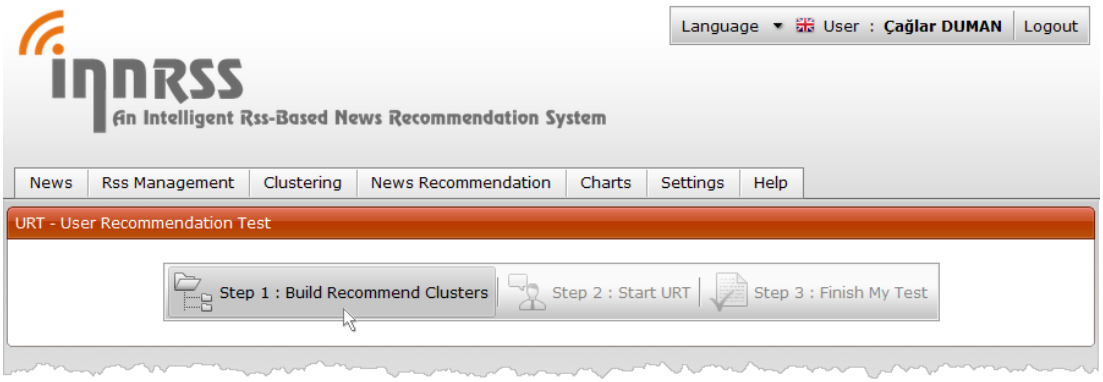
Liste dikkatli incelendiğinde; seçimin geniş yaş aralıklarında yapılması ile haber çeşitliliğinin arttırılması hedeflenmektedir. Tablo 5.1'deki kategoriler Ek-A bölümünde ayrıntılı olarak listelenmiştir.

5.3.3. Sistemin Eğitilmesi

Geliştirilen sisteminin en önemli özelliklerinden olan eğitilebilirlik, kullanıcının haber alışkanlığını öğrenmesini sağlamaktadır. Tablo 5.1'de gösterilen adaylara onbeş gün boyunca her gün test amaçlı veri setlerinden çeşitli kategorilerde istenildiği sayıda haberler okutulmuştur. Okunan haberlerin ilgili profillerdeki küme bütünlüğünün sağlanması için düzenli olarak küme optimizasyonu yapılmıştır. Onbeş gün boyunca sistem kullanıcıların profillerini öğrendiği varsayılmıştır.

5.4. Kullanıcı Haber Tavsiye Testi

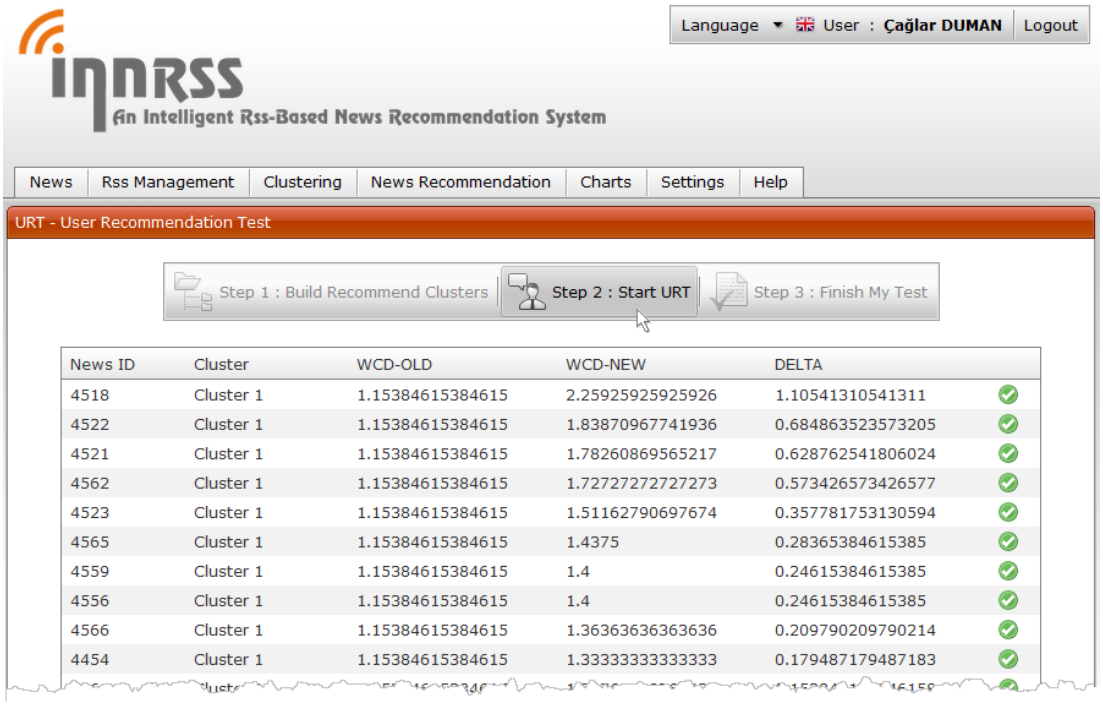
Kullanıcı tavsiye testi[112] tez kapsamında geliştirilen haber tavsiye sisteminin kullanıcılar üzerinde doğruluğunu kanıtlamak amacıyla geliştirilen bir modüldür. Tavsiye testi ekran görüntüsü Şekil 5.4'de gösterilmektedir. Genel olarak test üç aşamadan oluşmaktadır. İlk aşama test verilerini hazırlamayı hedefler. Verilerin hazırlanması için Bölüm 4.8'de aktarılan haber tavsiye sistemi kullanılmaktadır.



Şekil 5.4. Kullanıcı tavsiye test ekranı ekran görüntüsü

[112] ing: *user recommendation test*

Kullanıcı Şekil 5.4'de gösterildiği şekilde tavsiye kümelerini derle[113] bölüme tıklayarak test verilerini hazırlamaya başlamaktadır. Veri hazırlanmasını aktif hale geçirebilmek için kullanıcının en az otuz okunmamış, otuz okunmuş haber verisine ihtiyaç duyulmaktadır. Bu işlem başlatılması için gerekli şartlar sağlanamadığı takdirde uyarı verilip ilgili haberlerin okunması sağlanmaktadır. Test verileri hazırlanması aşamasında önce kullanıcının tüm haberleri sırasıyla tüm kümelere girdirilir ve Bölüm 4.8'de aktarılan haber tavsiye sisteminde olduğu gibi bir delta değer tablosu oluşturulur. Oluşturulan tablonun örnek bir ekran görüntüsü Şekil 5.5'de gösterilmektedir.



News ID	Cluster	WCD-OLD	WCD-NEW	DELTA
4518	Cluster 1	1.15384615384615	2.25925925925926	1.10541310541311
4522	Cluster 1	1.15384615384615	1.83870967741936	0.684863523573205
4521	Cluster 1	1.15384615384615	1.78260869565217	0.628762541806024
4562	Cluster 1	1.15384615384615	1.72727272727273	0.573426573426577
4523	Cluster 1	1.15384615384615	1.51162790697674	0.357781753130594
4565	Cluster 1	1.15384615384615	1.4375	0.28365384615385
4559	Cluster 1	1.15384615384615	1.4	0.24615384615385
4556	Cluster 1	1.15384615384615	1.4	0.24615384615385
4566	Cluster 1	1.15384615384615	1.36363636363636	0.209790209790214
4454	Cluster 1	1.15384615384615	1.33333333333333	0.179487179487183

Şekil 5.5. Kullanıcı tavsiye testi birinci aşama sonu ekran görüntüsü

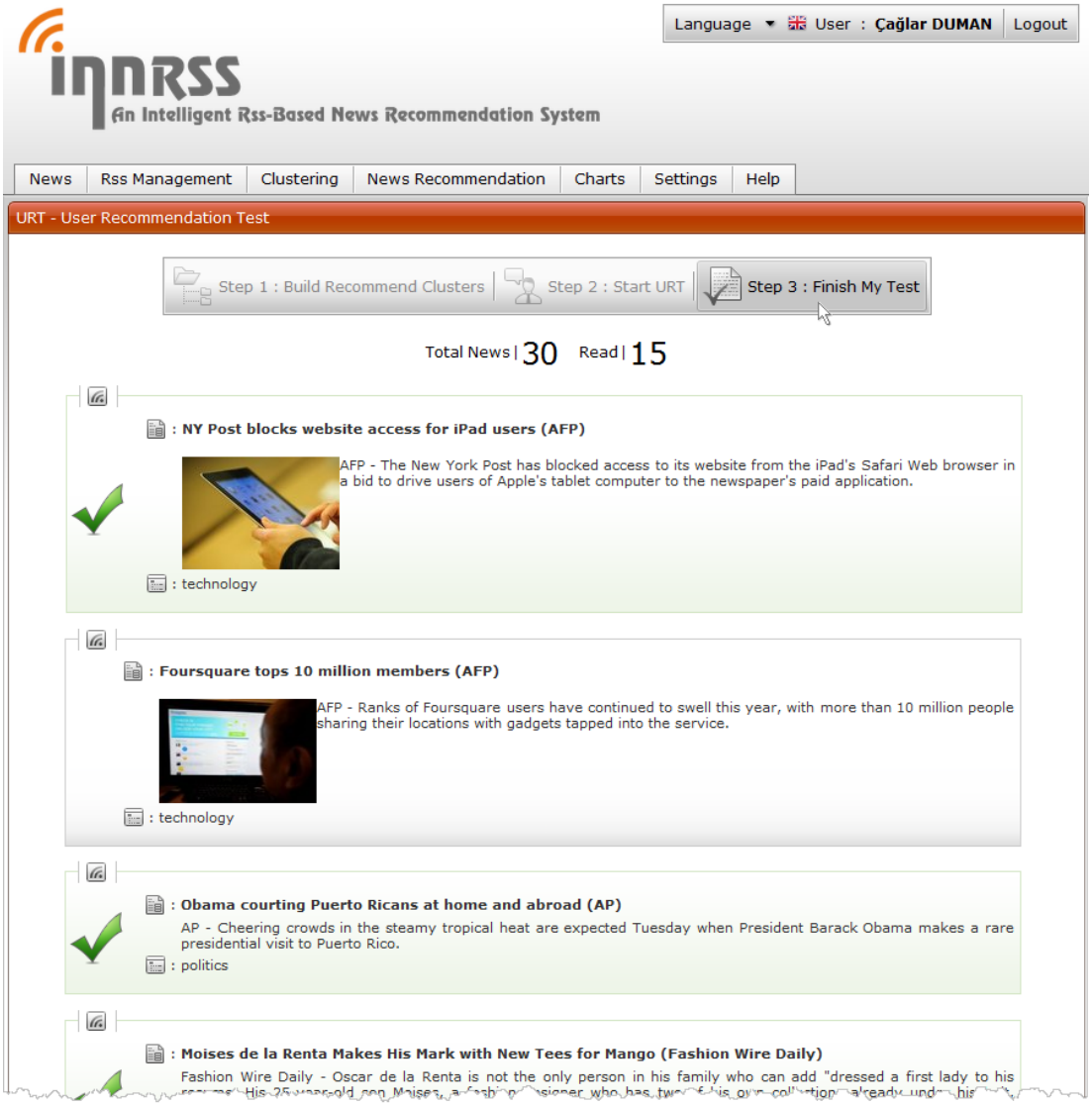
Örnek tablodan ilk satırı incelersek; 4518 numaralı haber birinci kümeye girmeden önce kümenin ağırlıklı kapsam yoğunluk değeri 1.53384615384615'tir. İlgili haber kümeye girdikten sonra bu değer 2.259259259259'a yükselmiştir. Bu durumda 1.10541310541311'lik bir yükselme söz konusudur. Bu delta değeri sıfır'dan büyük

[113] ing: *build recommend clusters*

ve pozitif yönde bir gelişim gösterdiğinden 4518 numaralı haber birinci küme için tavsiye edilmektedir. Bu noktada sistem hangi haberlerin hangi kümeler altında tavsiye edileceği tespit eder. İkinci aşama olarak tavsiye edilen haberler arasından en fazla otuz haber olmak koşulu ile yarısı bizim tavsiye ettiğimiz, yarısıda tavsiye edilmeyen haberler arasından seçilerek kullanıcıya sunulur. Şekil 5.6'da kullanıcı için oluşturulan otuz haber'in ekran görüntüsü sunulmuştur.

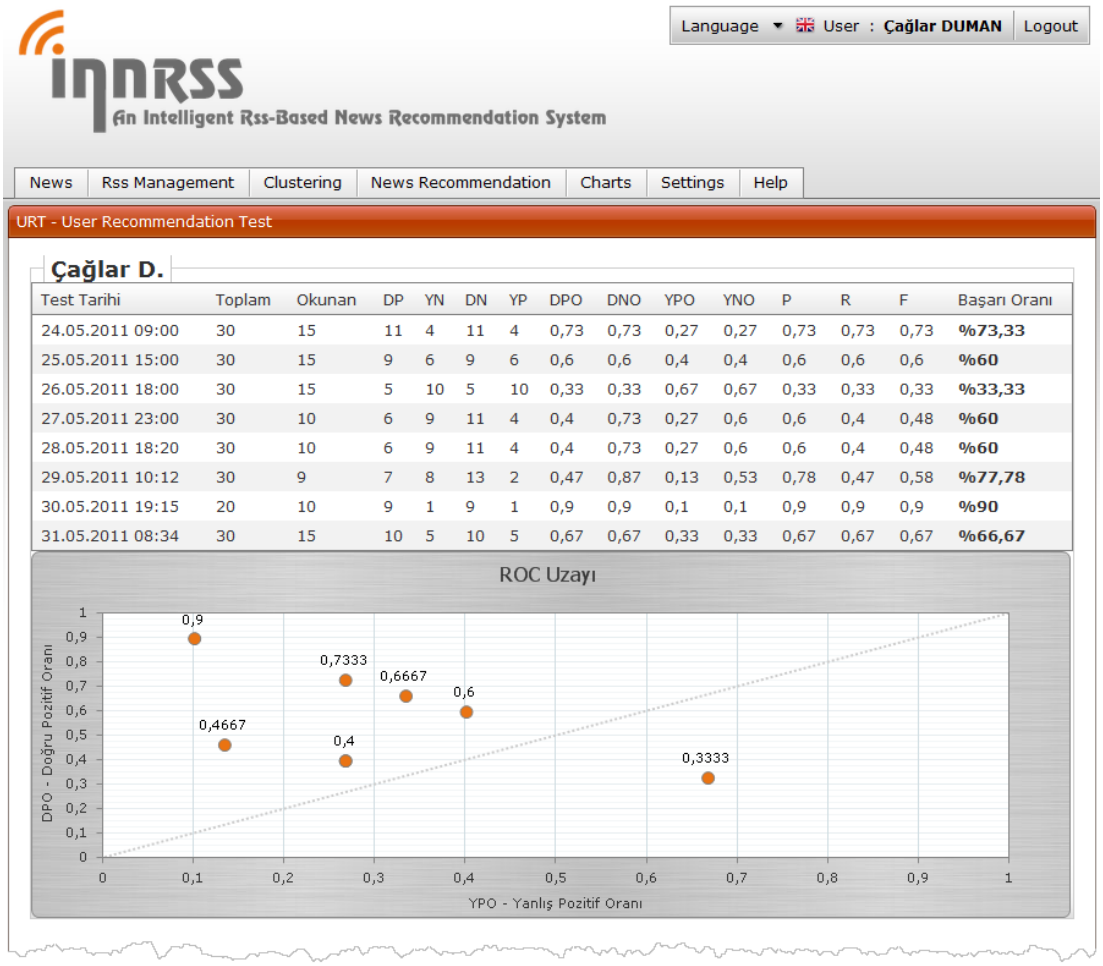
Şekil 5.6. Kullanıcı tavsiye testi ikinci aşama haber okuma ekran görüntüsü

Kullanıcıya sunulan otuz haberin on beşi sistem tarafından delta değerine göre artan olarak tavsiye edilen haberleri, diğer on beşi haber ise kullanıcın hiç bir profilinde (kümesinde) tavsiye edilmeyen haberleri temsil etmektedir. Kullanıcı sunulan haberlerden ilgilendiklerine tıklayıp okuması beklenmektedir. Kullanıcı sunulan haberlerden en fazla yarısı kadarını tercih edebilir. Bunun nedeni tavsiye edilen haberlerin toplamın yarısı olmasıdır. Örneğin otuz haberlik bir tavsiye testinde kullanıcı ilgilendiği on beş haberi okuduğu varsayılırsa okuma sonrası ekran görüntüsü Şekil 5.7'deki gibi olmaktadır.



Şekil 5.7. Kullanıcı tavsiye testi haber okuma sonrası ekran görüntüsü

İkinci aşama olan okuma işlemi de tamamlandığında testin son aşaması yani sonuç değerlendirme aşaması devreye girmektedir. Sonuç değerlendirmede Bölüm 5.1'de aktarılan doğruluk ölçüm teknikleri kullanılarak ilgili hesaplamalar yapılmaktadır. Daha sonra kullanıcının önceden yaptığı testlerde göz önüne alınarak Bölüm 5.2'de aktarılan kullanıcının alıcı karakteristik uzayı diğer değişle ROC uzay grafiği çizilmektedir. Örnek ekran görüntüsü Şekil 5.8'de verilmiştir. Kullanıcının yaptığı her test doğru pozitif oranının yanlış pozitif oranına göre bir değere sahip olur. Bu değer için ROC uzayında bir nokta oluşur. Tez çalışmasında ROC uzay grafiği diyagonalindeki çizgiden yukarda kalan nokta sayısı aşağıda kalan nokta sayısından büyük ise sistem o kullanıcı için doğru haber tavsiye ettiği savunulmaktadır.



Test Tarihi	Toplam	Okunan	DP	YN	DN	YP	DPO	DNO	YPO	YNO	P	R	F	Başarı Oranı
24.05.2011 09:00	30	15	11	4	11	4	0,73	0,73	0,27	0,27	0,73	0,73	0,73	%73,33
25.05.2011 15:00	30	15	9	6	9	6	0,6	0,6	0,4	0,4	0,6	0,6	0,6	%60
26.05.2011 18:00	30	15	5	10	5	10	0,33	0,33	0,67	0,67	0,33	0,33	0,33	%33,33
27.05.2011 23:00	30	10	6	9	11	4	0,4	0,73	0,27	0,6	0,6	0,4	0,48	%60
28.05.2011 18:20	30	10	6	9	11	4	0,4	0,73	0,27	0,6	0,6	0,4	0,48	%60
29.05.2011 10:12	30	9	7	8	13	2	0,47	0,87	0,13	0,53	0,78	0,47	0,58	%77,78
30.05.2011 19:15	20	10	9	1	9	1	0,9	0,9	0,1	0,1	0,9	0,9	0,9	%90
31.05.2011 08:34	30	15	10	5	10	5	0,67	0,67	0,33	0,33	0,67	0,67	0,67	%66,67

Şekil 5.8. Kullanıcı tavsiye testi sonuç ekranı

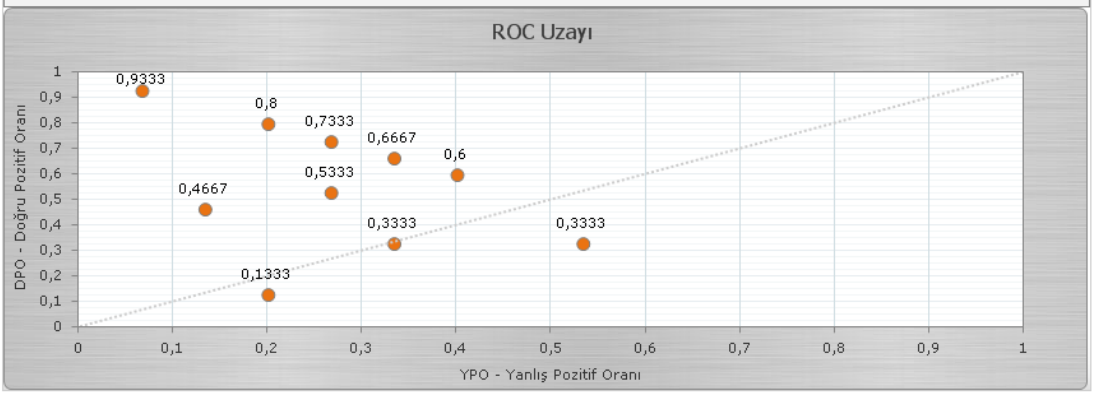
Örneğin Şekil 5.8'de sunulan kişisel ROC uzay grafiğinde altıya birlik oranla sistem doğru tavsiye etmektedir. Bu oran yüzlük sistem diliminde %83.3'e tekabül etmektedir. Son olarak bu durumda belirtilen örnek sonuçlara göre sistemin Çağlar Duman kullanıcısı için %83.3 oranında doğru çalıştığı kanıtlanmaktadır.

5.5. Ölçüm Sonuçları

Bu bölümde Tablo 5.1'de belirtilen kullanıcılar için test sonuçları verilmektedir. Sonuçlar sistemin tavsiye testi sonrası ekran görüntülerinden oluşmaktadır.

Murat D.

Test Tarihi	Toplam	Okunan	DP	YN	DN	YP	DPO	DNO	YPO	YNO	P	R	F	Başarı Oranı
15.05.2011 09:32	30	15	10	5	10	5	0,67	0,67	0,33	0,33	0,67	0,67	0,67	**%66,67**
16.05.2011 10:31	30	9	7	8	13	2	0,47	0,87	0,13	0,53	0,78	0,47	0,58	**%77,78**
17.05.2011 11:00	30	5	2	13	12	3	0,13	0,8	0,2	0,87	0,4	0,13	0,2	**%40**
18.05.2011 08:30	30	15	11	4	11	4	0,73	0,73	0,27	0,27	0,73	0,73	0,73	**%73,33**
19.05.2011 19:51	30	12	8	7	11	4	0,53	0,73	0,27	0,47	0,67	0,53	0,59	**%66,67**
20.05.2011 20:13	30	15	12	3	12	3	0,8	0,8	0,2	0,2	0,8	0,8	0,8	**%80**
21.05.2011 20:13	30	10	5	10	10	5	0,33	0,67	0,33	0,67	0,5	0,33	0,4	**%50**
22.05.2011 19:40	30	13	5	10	7	8	0,33	0,47	0,53	0,67	0,38	0,33	0,36	**%38,46**
23.05.2011 18:20	30	15	14	1	14	1	0,93	0,93	0,07	0,07	0,93	0,93	0,93	**%93,33**
24.05.2011 12:16	30	15	9	6	9	6	0,6	0,6	0,4	0,4	0,6	0,6	0,6	**%60**

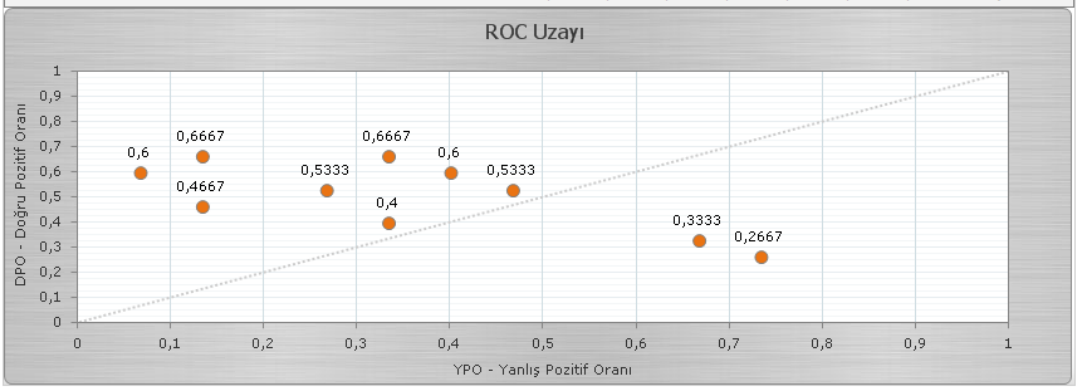


Şekil 5.9. Örnek bir kullanıcı için ölçüm sonuçları

Şekil 5.9'da örnek bir kişinin ölçüm sonuçları incelendiğinde on test verisi üzerinden yedi noktanın çizgi üstünde kaldığı, bir noktanın kararsızlık yaşadığı ve iki noktanın çizgi altı kaldığı görülmektedir. Yüzdelik dilime göre %70 başarı elde edilmiştir. Bu sonucu takiben diğer kişilerin ölçüm sonuçları sırasıyla verilip, tüm sonuçlar Bölüm 5.6'da değerlendirilecektir.

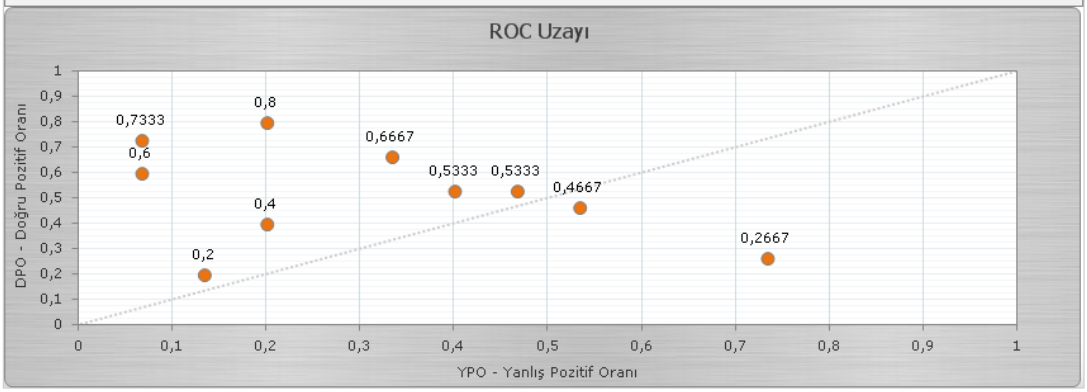
Alper Ö.

Test Tarihi	Toplam	Okunan	DP	YN	DN	YP	DPO	DNO	YPO	YNO	P	R	F	Başarı Oranı
15.05.2011 09:05	30	12	8	7	11	4	0,53	0,73	0,27	0,47	0,67	0,53	0,59	**%66,67**
16.05.2011 18:41	30	15	8	7	8	7	0,53	0,53	0,47	0,47	0,53	0,53	0,53	**%53,33**
17.05.2011 14:25	30	15	9	6	9	6	0,6	0,6	0,4	0,4	0,6	0,6	0,6	**%60**
18.05.2011 19:43	30	15	4	11	4	11	0,27	0,27	0,73	0,73	0,27	0,27	0,27	**%26,67**
19.05.2011 13:20	30	15	5	10	5	10	0,33	0,33	0,67	0,67	0,33	0,33	0,33	**%33,33**
20.05.2011 15:29	30	12	10	5	13	2	0,67	0,87	0,13	0,33	0,83	0,67	0,74	**%83,33**
21.05.2011 08:02	30	11	6	9	10	5	0,4	0,67	0,33	0,6	0,55	0,4	0,46	**%54,55**
22.05.2011 09:07	30	9	7	8	13	2	0,47	0,87	0,13	0,53	0,78	0,47	0,58	**%77,78**
23.05.2011 19:44	30	10	9	6	14	1	0,6	0,93	0,07	0,4	0,9	0,6	0,72	**%90**
24.05.2011 18:40	30	15	10	5	10	5	0,67	0,67	0,33	0,33	0,67	0,67	0,67	**%66,67**



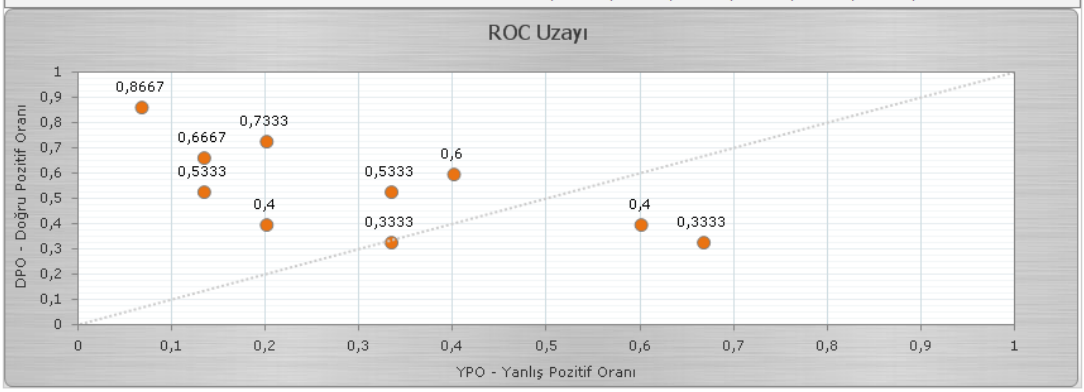
Beyhan Y.

Test Tarihi	Toplam	Okunan	DP	YN	DN	YP	DPO	DNO	YPO	YNO	P	R	F	Başarı Oranı
15.05.2011 12:19	30	10	9	6	14	1	0,6	0,93	0,07	0,4	0,9	0,6	0,72	**%90**
16.05.2011 18:40	30	15	7	8	7	8	0,47	0,47	0,53	0,53	0,47	0,47	0,47	**%46,67**
17.05.2011 22:56	30	15	12	3	12	3	0,8	0,8	0,2	0,2	0,8	0,8	0,8	**%80**
18.05.2011 17:39	30	5	3	12	13	2	0,2	0,87	0,13	0,8	0,6	0,2	0,3	**%60**
19.05.2011 12:16	30	12	11	4	14	1	0,73	0,93	0,07	0,27	0,92	0,73	0,81	**%91,67**
20.05.2011 12:16	30	15	10	5	10	5	0,67	0,67	0,33	0,33	0,67	0,67	0,67	**%66,67**
21.05.2011 17:37	30	9	6	9	12	3	0,4	0,8	0,2	0,6	0,67	0,4	0,5	**%66,67**
22.05.2011 13:22	30	15	4	11	4	11	0,27	0,27	0,73	0,73	0,27	0,27	0,27	**%26,67**
23.05.2011 16:33	30	14	8	7	9	6	0,53	0,6	0,4	0,47	0,57	0,53	0,55	**%57,14**
24.05.2011 13:19	30	15	8	7	8	7	0,53	0,53	0,47	0,47	0,53	0,53	0,53	**%53,33**



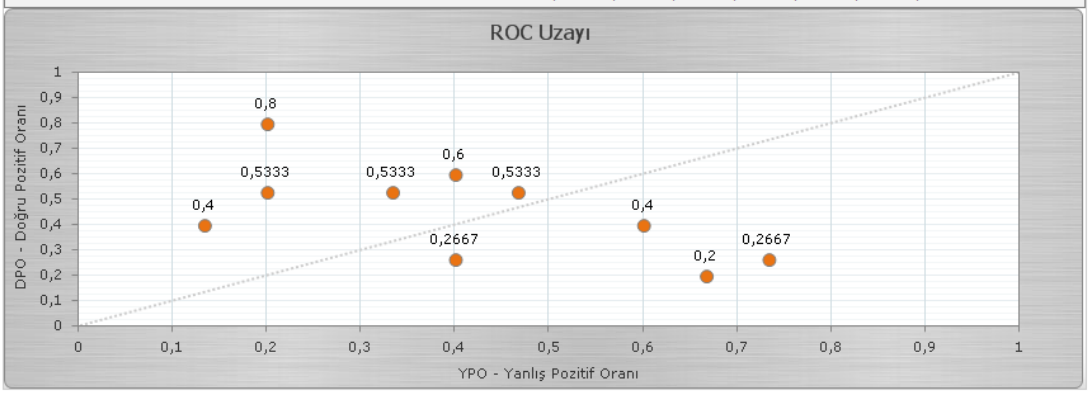
Çiğdem D.

Test Tarihi	Toplam	Okunan	DP	YN	DN	YP	DPO	DNO	YPO	YNO	P	R	F	Başarı Oranı
15.05.2011 13:20	30	10	8	7	13	2	0,53	0,87	0,13	0,47	0,8	0,53	0,64	**%80**
16.05.2011 17:39	30	15	9	6	9	6	0,6	0,6	0,4	0,4	0,6	0,6	0,6	**%60**
17.05.2011 18:42	30	14	11	4	12	3	0,73	0,8	0,2	0,27	0,79	0,73	0,76	**%78,57**
18.05.2011 10:09	30	14	13	2	14	1	0,87	0,93	0,07	0,13	0,93	0,87	0,9	**%92,86**
19.05.2011 12:16	30	15	5	10	5	10	0,33	0,33	0,67	0,67	0,33	0,33	0,33	**%33,33**
20.05.2011 11:13	30	10	5	10	10	5	0,33	0,67	0,33	0,67	0,5	0,33	0,4	**%50**
21.05.2011 14:26	30	13	8	7	10	5	0,53	0,67	0,33	0,47	0,62	0,53	0,57	**%61,54**
22.05.2011 10:09	30	12	10	5	13	2	0,67	0,87	0,13	0,33	0,83	0,67	0,74	**%83,33**
23.05.2011 15:31	30	9	6	9	12	3	0,4	0,8	0,2	0,6	0,67	0,4	0,5	**%66,67**
24.05.2011 10:11	30	15	6	9	6	9	0,4	0,4	0,6	0,6	0,4	0,4	0,4	**%40**



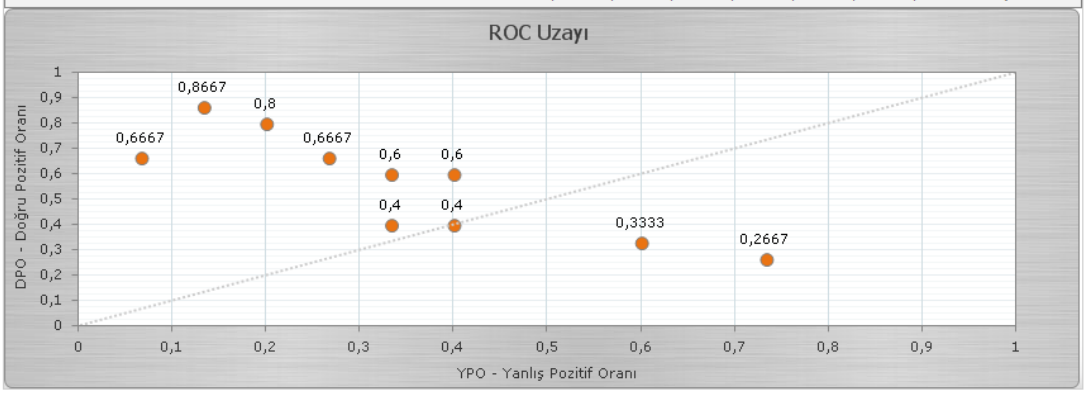
Fisun S.

Test Tarihi	Toplam	Okunan	DP	YN	DN	YP	DPO	DNO	YPO	YNO	P	R	F	Başarı Oranı
15.05.2011 21:54	30	10	4	11	9	6	0,27	0,6	0,4	0,73	0,4	0,27	0,32	**%40**
16.05.2011 15:28	30	15	6	9	6	9	0,4	0,4	0,6	0,6	0,4	0,4	0,4	**%40**
17.05.2011 16:32	30	13	3	12	5	10	0,2	0,33	0,67	0,8	0,23	0,2	0,21	**%23,08**
18.05.2011 09:04	30	15	9	6	9	6	0,6	0,6	0,4	0,4	0,6	0,6	0,6	**%60**
19.05.2011 12:15	30	15	8	7	8	7	0,53	0,53	0,47	0,47	0,53	0,53	0,53	**%53,33**
20.05.2011 10:10	30	13	8	7	10	5	0,53	0,67	0,33	0,47	0,62	0,53	0,57	**%61,54**
21.05.2011 12:17	30	11	8	7	12	3	0,53	0,8	0,2	0,47	0,73	0,53	0,62	**%72,73**
22.05.2011 19:43	30	8	6	9	13	2	0,4	0,87	0,13	0,6	0,75	0,4	0,52	**%75**
23.05.2011 19:44	30	15	4	11	4	11	0,27	0,27	0,73	0,73	0,27	0,27	0,27	**%26,67**
24.05.2011 15:29	30	15	12	3	12	3	0,8	0,8	0,2	0,2	0,8	0,8	0,8	**%80**



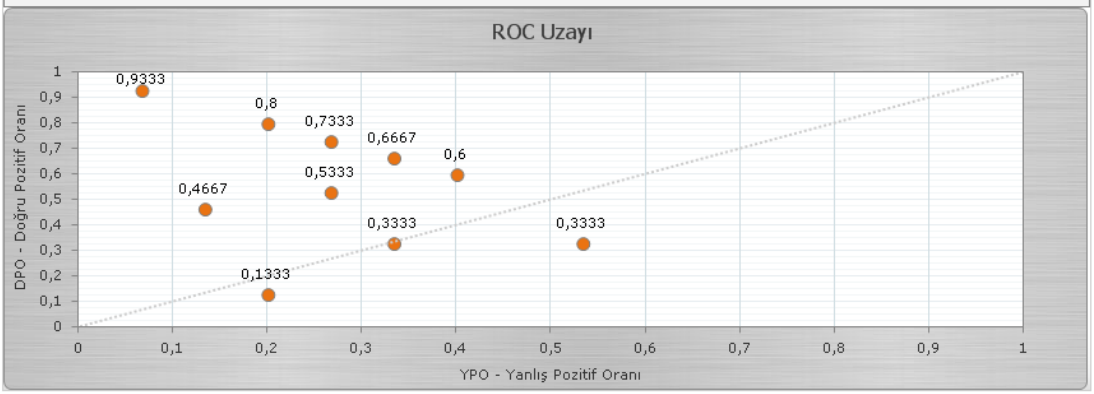
Kevser Y.

Test Tarihi	Toplam	Okunan	DP	YN	DN	YP	DPO	DNO	YPO	YNO	P	R	F	Başarı Oranı
15.05.2011 17:36	30	14	10	5	11	4	0,67	0,73	0,27	0,33	0,71	0,67	0,69	**%71,43**
16.05.2011 12:19	30	15	9	6	9	6	0,6	0,6	0,4	0,4	0,6	0,6	0,6	**%60**
17.05.2011 16:32	30	15	12	3	12	3	0,8	0,8	0,2	0,2	0,8	0,8	0,8	**%80**
18.05.2011 19:43	30	15	4	11	4	11	0,27	0,27	0,73	0,73	0,27	0,27	0,27	**%26,67**
19.05.2011 15:30	30	12	6	9	9	6	0,4	0,6	0,4	0,6	0,5	0,4	0,44	**%50**
20.05.2011 14:26	30	11	6	9	10	5	0,4	0,67	0,33	0,6	0,55	0,4	0,46	**%54,55**
21.05.2011 11:14	30	11	10	5	14	1	0,67	0,93	0,07	0,33	0,91	0,67	0,77	**%90,91**
22.05.2011 15:31	30	15	13	2	13	2	0,87	0,87	0,13	0,13	0,87	0,87	0,87	**%86,67**
23.05.2011 14:25	30	14	5	10	6	9	0,33	0,4	0,6	0,67	0,36	0,33	0,34	**%35,71**
24.05.2011 16:31	30	14	9	6	10	5	0,6	0,67	0,33	0,4	0,64	0,6	0,62	**%64,29**



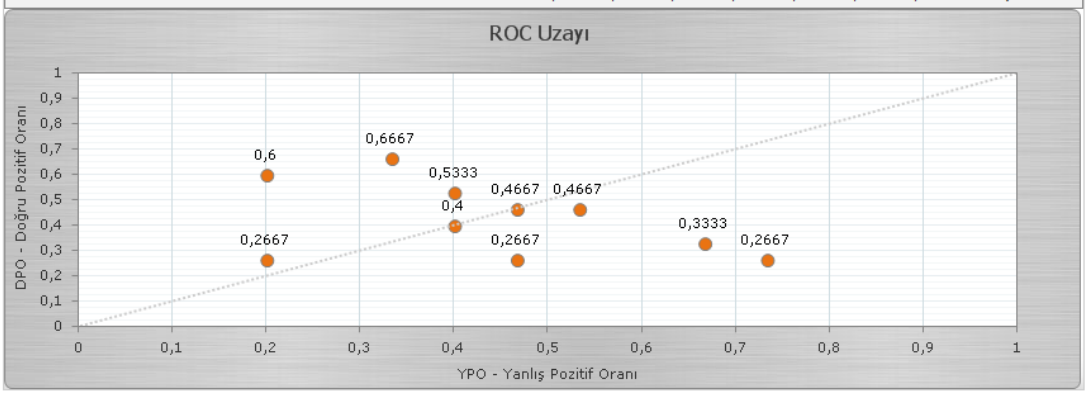
Murat D.

Test Tarihi	Toplam	Okunan	DP	YN	DN	YP	DPO	DNO	YPO	YNO	P	R	F	Başarı Oranı
15.05.2011 09:32	30	15	10	5	10	5	0,67	0,67	0,33	0,33	0,67	0,67	0,67	**%66,67**
16.05.2011 10:31	30	9	7	8	13	2	0,47	0,87	0,13	0,53	0,78	0,47	0,58	**%77,78**
17.05.2011 11:00	30	5	2	13	12	3	0,13	0,8	0,2	0,87	0,4	0,13	0,2	**%40**
18.05.2011 08:30	30	15	11	4	11	4	0,73	0,73	0,27	0,27	0,73	0,73	0,73	**%73,33**
19.05.2011 19:51	30	12	8	7	11	4	0,53	0,73	0,27	0,47	0,67	0,53	0,59	**%66,67**
20.05.2011 20:13	30	15	12	3	12	3	0,8	0,8	0,2	0,2	0,8	0,8	0,8	**%80**
21.05.2011 20:13	30	10	5	10	10	5	0,33	0,67	0,33	0,67	0,5	0,33	0,4	**%50**
22.05.2011 19:40	30	13	5	10	7	8	0,33	0,47	0,53	0,67	0,38	0,33	0,36	**%38,46**
23.05.2011 18:20	30	15	14	1	14	1	0,93	0,93	0,07	0,07	0,93	0,93	0,93	**%93,33**
24.05.2011 12:16	30	15	9	6	9	6	0,6	0,6	0,4	0,4	0,6	0,6	0,6	**%60**



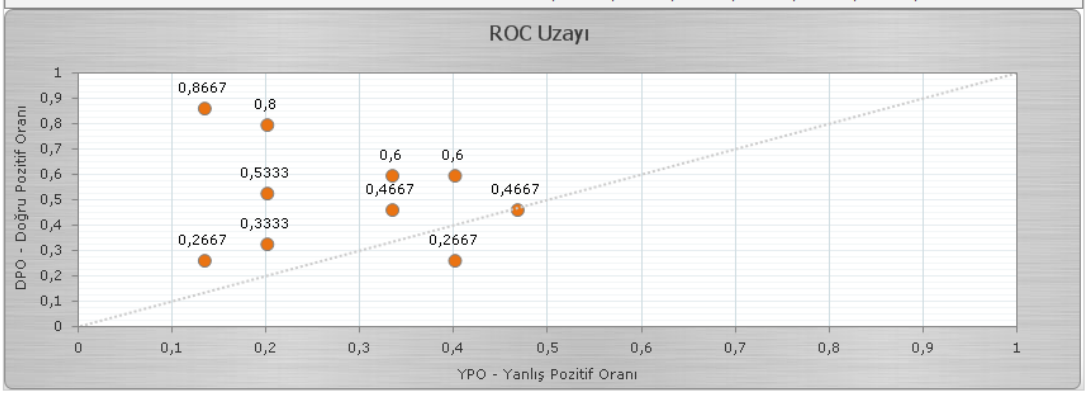
Özgür D.

Test Tarihi	Toplam	Okunan	DP	YN	DN	YP	DPO	DNO	YPO	YNO	P	R	F	Başarı Oranı
15.05.2011 17:36	30	15	10	5	10	5	0,67	0,67	0,33	0,33	0,67	0,67	0,67	**%66,67**
16.05.2011 10:08	30	15	5	10	5	10	0,33	0,33	0,67	0,67	0,33	0,33	0,33	**%33,33**
17.05.2011 14:26	30	14	7	8	8	7	0,47	0,53	0,47	0,53	0,5	0,47	0,48	**%50**
18.05.2011 12:17	30	15	4	11	4	11	0,27	0,27	0,73	0,73	0,27	0,27	0,27	**%26,67**
19.05.2011 14:26	30	11	4	11	8	7	0,27	0,53	0,47	0,73	0,36	0,27	0,31	**%36,36**
20.05.2011 08:00	30	12	6	9	9	6	0,4	0,6	0,4	0,6	0,5	0,4	0,44	**%50**
21.05.2011 10:08	30	14	8	7	9	6	0,53	0,6	0,4	0,47	0,57	0,53	0,55	**%57,14**
22.05.2011 09:04	30	12	9	6	12	3	0,6	0,8	0,2	0,4	0,75	0,6	0,67	**%75**
23.05.2011 13:19	30	15	7	8	7	8	0,47	0,47	0,53	0,53	0,47	0,47	0,47	**%46,67**
26.05.2011 13:20	30	7	4	11	12	3	0,27	0,8	0,2	0,73	0,57	0,27	0,36	**%57,14**



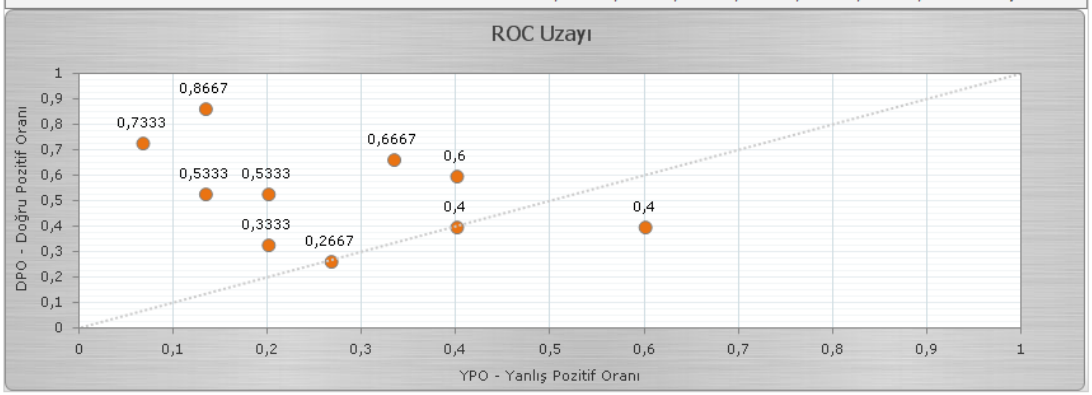
Pınar D.

Test Tarihi	Toplam	Okunan	DP	YN	DN	YP	DPO	DNO	YPO	YNO	P	R	F	Başarı Oranı
15.05.2011 12:18	30	15	12	3	12	3	0,8	0,8	0,2	0,2	0,8	0,8	0,8	**%80**
16.05.2011 15:28	30	15	13	2	13	2	0,87	0,87	0,13	0,13	0,87	0,87	0,87	**%86,67**
17.05.2011 20:49	30	14	7	8	8	7	0,47	0,53	0,47	0,53	0,5	0,47	0,48	**%50**
18.05.2011 13:20	30	12	7	8	10	5	0,47	0,67	0,33	0,53	0,58	0,47	0,52	**%58,33**
19.05.2011 13:22	30	11	8	7	12	3	0,53	0,8	0,2	0,47	0,73	0,53	0,62	**%72,73**
20.05.2011 14:24	30	10	4	11	9	6	0,27	0,6	0,4	0,73	0,4	0,27	0,32	**%40**
21.05.2011 16:34	30	6	4	11	13	2	0,27	0,87	0,13	0,73	0,67	0,27	0,38	**%66,67**
22.05.2011 11:12	30	8	5	10	12	3	0,33	0,8	0,2	0,67	0,63	0,33	0,43	**%62,5**
23.05.2011 13:21	30	14	9	6	10	5	0,6	0,67	0,33	0,4	0,64	0,6	0,62	**%64,29**
24.05.2011 19:46	30	15	9	6	9	6	0,6	0,6	0,4	0,4	0,6	0,6	0,6	**%60**



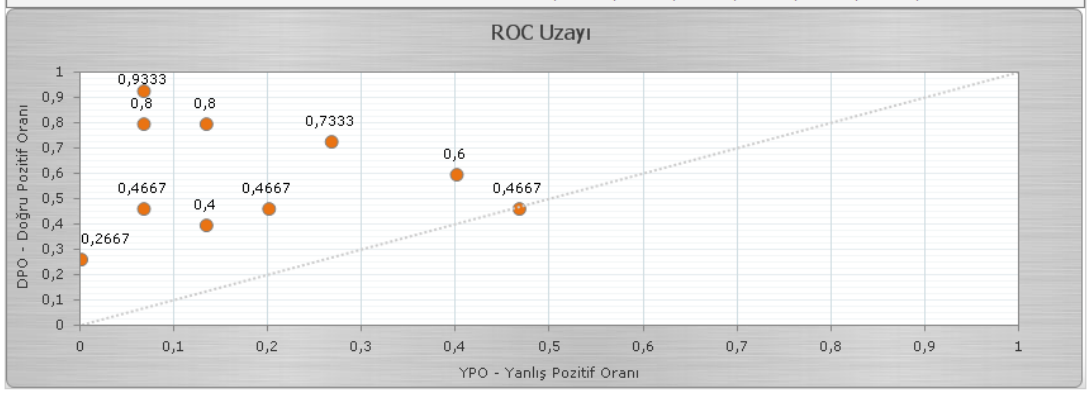
Umay A.

Test Tarihi	Toplam	Okunan	DP	YN	DN	YP	DPO	DNO	YPO	YNO	P	R	F	Başarı Oranı
15.05.2011 13:20	30	15	9	6	9	6	0,6	0,6	0,4	0,4	0,6	0,6	0,6	**%60**
16.05.2011 15:27	30	15	13	2	13	2	0,87	0,87	0,13	0,13	0,87	0,87	0,87	**%86,67**
17.05.2011 18:40	30	12	11	4	14	1	0,73	0,93	0,07	0,27	0,92	0,73	0,81	**%91,67**
18.05.2011 22:56	30	15	6	9	6	9	0,4	0,4	0,6	0,6	0,4	0,4	0,4	**%40**
19.05.2011 17:35	30	12	6	9	9	6	0,4	0,6	0,4	0,6	0,5	0,4	0,44	**%50**
20.05.2011 22:58	30	8	5	10	12	3	0,33	0,8	0,2	0,67	0,63	0,33	0,43	**%62,5**
21.05.2011 17:35	30	15	10	5	10	5	0,67	0,67	0,33	0,33	0,67	0,67	0,67	**%66,67**
22.05.2011 13:22	30	10	8	7	13	2	0,53	0,87	0,13	0,47	0,8	0,53	0,64	**%80**
23.05.2011 08:03	30	8	4	11	11	4	0,27	0,73	0,27	0,73	0,5	0,27	0,35	**%50**
24.05.2011 14:27	30	11	8	7	12	3	0,53	0,8	0,2	0,47	0,73	0,53	0,62	**%72,73**



Zerrin Y.

Test Tarihi	Toplam	Okunan	DP	YN	DN	YP	DPO	DNO	YPO	YNO	P	R	F	Başarı Oranı
15.05.2011 08:03	30	15	14	1	14	1	0,93	0,93	0,07	0,07	0,93	0,93	0,93	**%93,33**
16.05.2011 09:06	30	14	7	8	8	7	0,47	0,53	0,47	0,53	0,5	0,47	0,48	**%50**
17.05.2011 11:13	30	14	12	3	13	2	0,8	0,87	0,13	0,2	0,86	0,8	0,83	**%85,71**
18.05.2011 17:37	30	13	12	3	14	1	0,8	0,93	0,07	0,2	0,92	0,8	0,86	**%92,31**
19.05.2011 20:47	30	15	11	4	11	4	0,73	0,73	0,27	0,27	0,73	0,73	0,73	**%73,33**
20.05.2011 13:20	30	8	6	9	13	2	0,4	0,87	0,13	0,6	0,75	0,4	0,52	**%75**
21.05.2011 19:44	30	10	7	8	12	3	0,47	0,8	0,2	0,53	0,7	0,47	0,56	**%70**
22.05.2011 15:30	30	8	7	8	14	1	0,47	0,93	0,07	0,53	0,88	0,47	0,61	**%87,5**
23.05.2011 14:23	30	4	4	11	15	0	0,27	1	0	0,73	1	0,27	0,42	**%100**
24.05.2011 12:18	30	15	9	6	9	6	0,6	0,6	0,4	0,4	0,6	0,6	0,6	**%60**



5.6. Ölçüm Sonuçlarının Değerlendirilmesi

Tüm ölçüm sonuçları kullanıcı bazında incelenmiş ve Tablo 5.2'de özetlenmiştir. Sonuçlar değerlendirildiğinde; B2,C5,D2,E5,G3 kategorilerini tercih eden Zerrin Y. kullanıcısı en büyük başarı oranına, D2,D6,D7,D8,D10 kategorilerini tercih eden Özgür D. kullanıcısı en düşük başarı oranını sağladığı görülmektedir. Başarı oranı düşük olan kullanıcının kategorileri incelendiğinde hep aynı kategorinin iç kategorilerini tercih ettiği görülmektedir. Bu durum tez kapsamında geliştirilen sistemin ağırlıklı kapsam yoğunluğu yaklaşımının aynı kategorilerdeki çok benzer haberleri tavsiye ettiği ancak spesifik olarak diğer haberlerede benzediği için kullanıcıyı testlerde yanılttığı ortaya konmuştur.

No	Kişi Ad	Pozitif	Nötr	Negatif	%
1	Alper Ö.	8	0	2	%80
2	Beyhan Y.	8	0	2	%80
3	Çiğdem D.	7	1	2	%70
4	Fisun S.	6	0	4	%60
5	Kevser Y.	7	1	2	%70
6	Murat D.	7	1	2	%70
7	Özgür D.	4	2	4	%40
8	Pınar D.	8	1	1	%80
9	Umay A.	7	2	1	%70
10	Zerrin Y.	9	1	0	%90
Ortalama ve Genel Başarı		7,1	0,9	2	%71

Tablo 5.2. Ölçüm sonuçlarının kişi bazlı başarı yüzdeleri

Sonuç olarak tez kapsamında geliştirilen ağırlıklı kapsam yoğunluğu temeline dayalı rss tabanlı haber tavsiye sisteminin on kişi üzerinde tavsiye testi ölçüm sonuçlarına göre genel ortalaması ve başarı oranı %71'dir. Bu rakam bize ortalama olarak sistemin %71 oranında doğru haber tavsiye ettiğini göstermektedir.

BÖLÜM 6

6. SONUÇ

Sonuç olarak günümüzde internetin ve haber kaynaklarının yaygınlaştığı tartışılmaz bir gerçektir. Her gün binlerce haber, web üzerinden yapısal olmayan veri kaynaklarında ve veri tabanlarında tutulmaktadır. Gün geçtikçe bu veri inanılmaz boyutlara ulaşmakta ve işimize yarayan gerçek veri çıkarımı zorlaşmaktadır.

Altı bölümden oluşan bu tez çalışmasında bu problemin çözümü için metin halindeki veriye erişimi kolaylaştıran, zaman ve hız kazandıran metin kategorizasyon tekniği olan ağırlıklı kapsam yoğunluğu ağırlandırma algoritması incelenmiş olup, bu teknik kullanarak rss tabanlı dinamik içerikli, içeriği farklı haber sitelerinin farklı kategorilerinde yer alan haberleri tarayarak, kullanıcının okuduğu benzer haberleri, dinamik olarak kümeleyen, kullanıcının haber alışkanlığını öğrenen, gerekli gruplara göre okuyabileceği haberleri tavsiye eden akıllı bir sistem gerçekleştirilmiştir.

Tez çalışmasının ilk bölümde tezin amacı ve genelinden bahsedilmiştir. İkinci bölümde tez çalışmasının temeli olan ağırlıklı kapsam yoğunluğu algoritması yaklaşımına genel bir bakış yapılmıştır. İşlemsel veri setlerinden, işlemsel veri setlerinin nasıl kümelendiğinden, kapsam yoğunluğu yaklaşımından bahsedilerek, ağırlıklı ve beklenen kapsam yoğunluğu algoritmaları arasındaki farklar ortaya konmuştur. Üçüncü bölümde tez kapsamında kullanılan protokoller, yöntemler ve diğer algoritmalar hakkında genel bilgi aktarılmıştır. Dördüncü bölümde önerilen haber tavsiye sistemi hakkında detaylı bilgi verilmiştir. Sistem mimarisi, haberin okunması ve ön işleme aşamaları, okunan haberlerin kümelenmesi, kümeler arası haberlerin optimize edilerek yeni kümelerin oluşturulması, oluşturulan kümelerin kullanıcılar için optimum sayıda küme sayısının bulunması, anahtar kelime çıkarımı ve haber tavsiye bölümleri ayrıntılı incelenmiştir. Beşinci bölümde ise gerçekleştirilen sistemin aktif kullanıcılar üzerinde test edilmesi, doğruluk ölçümlerinin alınması ve değerlendirilmesi tartışılmıştır.

Beşinci bölümde aktarılan deneysel çalışmada öncelikle sistem on kişi için eğitilmiş ve haber alışkanlıkları öğretilmiştir. Sistemin eğitilmesi tamamen dinamiktir. Sistemin eğitilmesi ardından on beş gün boyunca kullanıcılar teste maruz bırakılmış ve sonuçlar toplanmıştır. Bu sonuçların ayrıntılı ortalamaları son bölümde tartışılmıştır. Sonuç olarak %71 doğruluk oranıyla sistemin kullanıcılara doğru haber tavsiye ettiği kanıtlanmıştır.

Bu çalışma akıllı bir sistem olduğu için yüzde yüz oranda doğru tavsiyelerde bulunması beklenemez çünkü işlenen haberler anlamsal değil kelimesel olarak değerlendirilmektedir. Aynı kelimenin birçok eş anlamlısı olması sistemin insan olmadığı için yanılmasına sebep olacaktır. Anlamsal ağ çalışmaları her ne kadar bu yapıyı kırsa da doğal dil bilimcilerin bu konuda çalışmaları halen devam etmektedir.

Tez kapsamında gerçekleştirilen bu çalışmanın devamında sistem üzerinde kullanıcılar arası profil değerlendirme çalışmaları planlanmaktadır. Ayrıca sistem şu an sadece İngilizce haberler üzerinde doğal dil işleme yapabilmektedir. Ancak sistemin işlediği kelimelerin dili tamamen dinamik olarak tasarlanmıştır. Türkçe haberlerin kullanılamamasının sebebi Türkçe doğal dil işleme algoritmalarının yetersizliğinden kaynaklanmaktadır. İleride Türkçe doğal dil işleme konusunda yeni gelişmeler beklenmekte ve bu paralelliğe yönelik Türkçe haber entegrasyonu da planlanmaktadır.

KAYNAKLAR

[1] Pilavcılar, F.İ., 2007, Metin Madenciliği ile Metin Sınıflama, *Yüksek Lisans Tezi*, Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü, Matematik Mühendisliği, İstanbul.

[2] Carey, P., Creating Web Pages with html, xhtml and xml 2nd edition, Thomson Learing, One Main Street, Cambridge, 2006.

[3] "Really Simple Syndication Rss 2.0 specification feed validator" erişim adresi : http://feed2.w3.org/docs/rss2.html#whatIsRss, erişim tarihi: 12.Ocak 2011.

[4] Yan, H, Chen, K, Liu, L, Efficiently Clustering Transactional Data with Weighted Coverage Density, 15th ACM international conference on Information and knowledge management CIKM'06, Arlington, Virginia, USA, Kasım 2006.

[5] Yan, H, Chen, K, Liu, L, Bae, J, Determining the best K for clustering transactional datasets: A coverage density-based approach, Data & Knowledge Engineering, 68, 28-48, 2009.

[6] Yang, Y., Guan, X, You, J., CLOPE: A Fast and Effective Clustering Algorithm for Transactional Data, 8th ACM international conference on Knowledge discovery and data mining SIGKDD'02, Edmonton, Alberta, Canada, Temmuz 2002.

[7] Chen, K, Liu, L, The "Best K" for Entropy-based Categorical Data Clustering, 17th international conference on Scientific and statistical database management SSDBM'2005, Haziran 2005.

[8] Tseng, Y., Lin, C., and Lin, Y. 2007. Text mining techniques for patent analysis, Information Processing and Management: an International Journal, 43(5), 1216-1247, 2007.

[9] Feldman, R., Sanger, J., The Text Mining Hand Book Advanced Approaches in Advanced Approaches in Analyzing Unstructured Data, Cambridge University Press, Cambridge, England, 2007.

[10] Fayyad, U., Piatetsky, G., Smyth., P., The kdd process for extracting useful knowledge from volumes of data. Communications of ACM, 39(11), 27-34, 1996.

[11] Grabmeier, J., Rudolph., A., Techniques of cluster algorithms in data mining. Data Mining and Knowledge Discovery, 6, 303-360, 2003.

[12] Porter, M.F, An algorithm for suffix stripping, Morgan Kaufmann Multimedia Information and Systems Series Readings in information retrieval, 313 – 316, 1997.

[13] Jackson, P., Moulinier, I., Natural Language Processing for Online Applications, Text Retrieval, Extraction and Categorization, *John Benjamins Publishing Company*, Philadelphia North America, 2007.

[14] "An organizational center for open source projects related to natural language processing OpenNLP Baldridge, J, Bierner, G, Morton, T," , erişim adresi: http://opennlp.sourceforge.net/, erişim tarihi: 12.Şubat.2011.

[15] "Penn Treebank Part-of-Speech Tagging", erişim adresi: http://www.cis.upenn.edu/~treebank/, erişim tarihi: 13.Şubat.2011.

[16] Yüksel, M.E., Turna, Ö.C., Ertürk, A.M., Bilgiye Erişim Sistemlerinde Veri Arama ve Eşleştirme, İstanbul Üniversitesi Bilgisayar Müh. Bölümü, İstanbul, 2000.

[17] "Measures of semantic relatedness using WordNet." erişim adresi: http://wn-similarity.sourceforge.net/; erişim tarihi: 10.Ocak.2011.

[18] Adomavicius, G., Tuzhilin, A., Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions Export, Knowledge and Data Engineering, IEEE Transactions on Knowledge and Data Engineering, 17(6), 734-749, Ocak 2005.

[19] Kayaalp, M., Özyer T, Özyer, T.S., A Collaborative and Content Based Event Recommendation System Integrated with Data Collection Scrapers and Services at a Social Networking Site, International Conference on Advances in Social Networks Analysis and Mining, 113-118, 2009.

[20] Jain, A.K., Murty, M.N., Flynn., P.J., Data clustering: A review. CM Computing Surveys (CSUR), 31(3), 264-323, Temmuz 1999.

[21] Jiang, D., Tang, C., Zhang, A., Cluster analysis for gene expression data: IEEE Transactions on Knowledge and Data Engineering, 16(11), 1370-1386, Kasım 2004.

[22] Hartigan, J., Clustering Algorithms. *John Wiley and Sons Inc.*, New York, NY, 1975.

[23] Modha, D.S., Spangler, W.S., Feature weighting in k-means clustering, Machine Learning, 52(3), 217-237, Temmuz 2003.

[24] Huang, J.Z., Ng, M.K., Rong, H., Li. Z., Automated variable weighting in k-means type clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(5) 657-668, Mayıs 2005.

[25] S. Guha, R. Rastogi, and K. Shim. Cure: An efficient clustering algorithm for large databases, Proceedings of ACM SIGMOD Conference on Management of Data, 73-84, Ocak 1998.

[26] Ben-Dor, A., Shamir, R., Yahkini. Z., Clustering gene expression patterns. Journal of Computational Biology, 6(3), 281-297, Nisan 1999.

[27] Baneld, J.D., Raftery, A.E., Model-based gaussian and non-gaussian clustering. Biometrics, 49,803-821, Nisan 1993.

[28] Kaski, S., Nikkilä, J. and T. Kohonen. Methods for interpreting a self-organized map in data analysis. Proceedings of the European Symposium on Artificial Neural Networks, 185-190, Brussels, Belgium, 1998.

[29] Kim, Y. Lee. S., A clustering validity assessment index. Proceedings of Asian Conference on Knowledge Discovery and Data Mining, pages 602-608, Nisan 2003.

[30] "Converting plural words to singular", erişim adresi : http://www.bennysutton.com/C-sharp/Plural-Singular-Words.aspx, erişim tarihi: 10 Mart 2011.

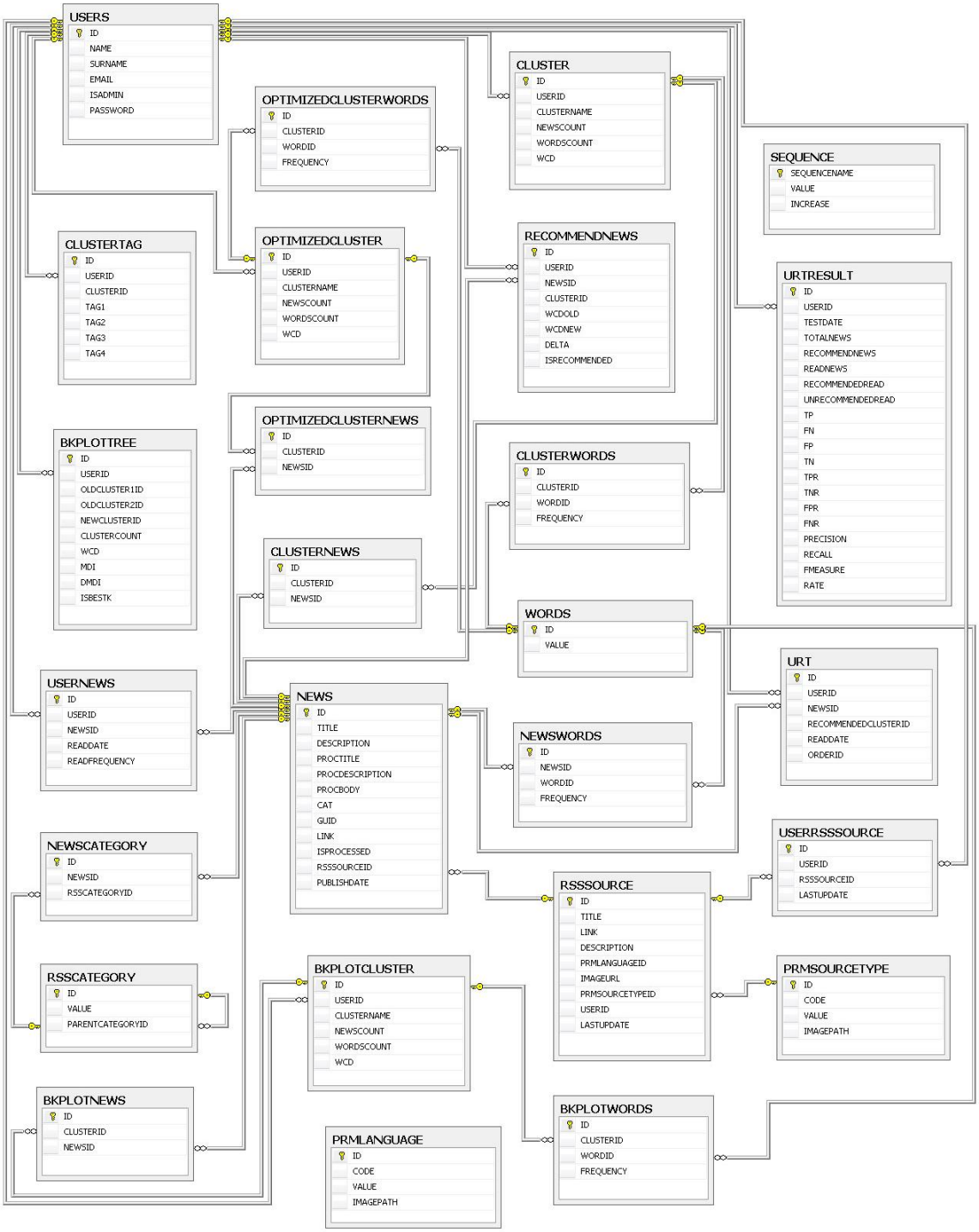
[31] Tan, P.N., Steinbach, M., Kumar, V., Introduction to Data Mining, Pearson Internation Edition, p.295-300, 2006.

[32] Yahoo! News Rss Feed, erişim adresi : http://news.yahoo.com/rss, erişim tarihi: 10 Nisan 2011.

[33] Fogarty, J., Baker,R., Hudson, S., Case studies in the use of ROC curve analysis for sensor-based estimates in human computer interaction, ACM International Conference Proceeding Series, Proceedings of Graphics Interface, Waterloo, Ontario, Canada: Canadian Human-Computer Communications Society, 2005.

EKLER

EK – A: Veritabanı diyagramı



EK – B: Veritabanı yordam listesi

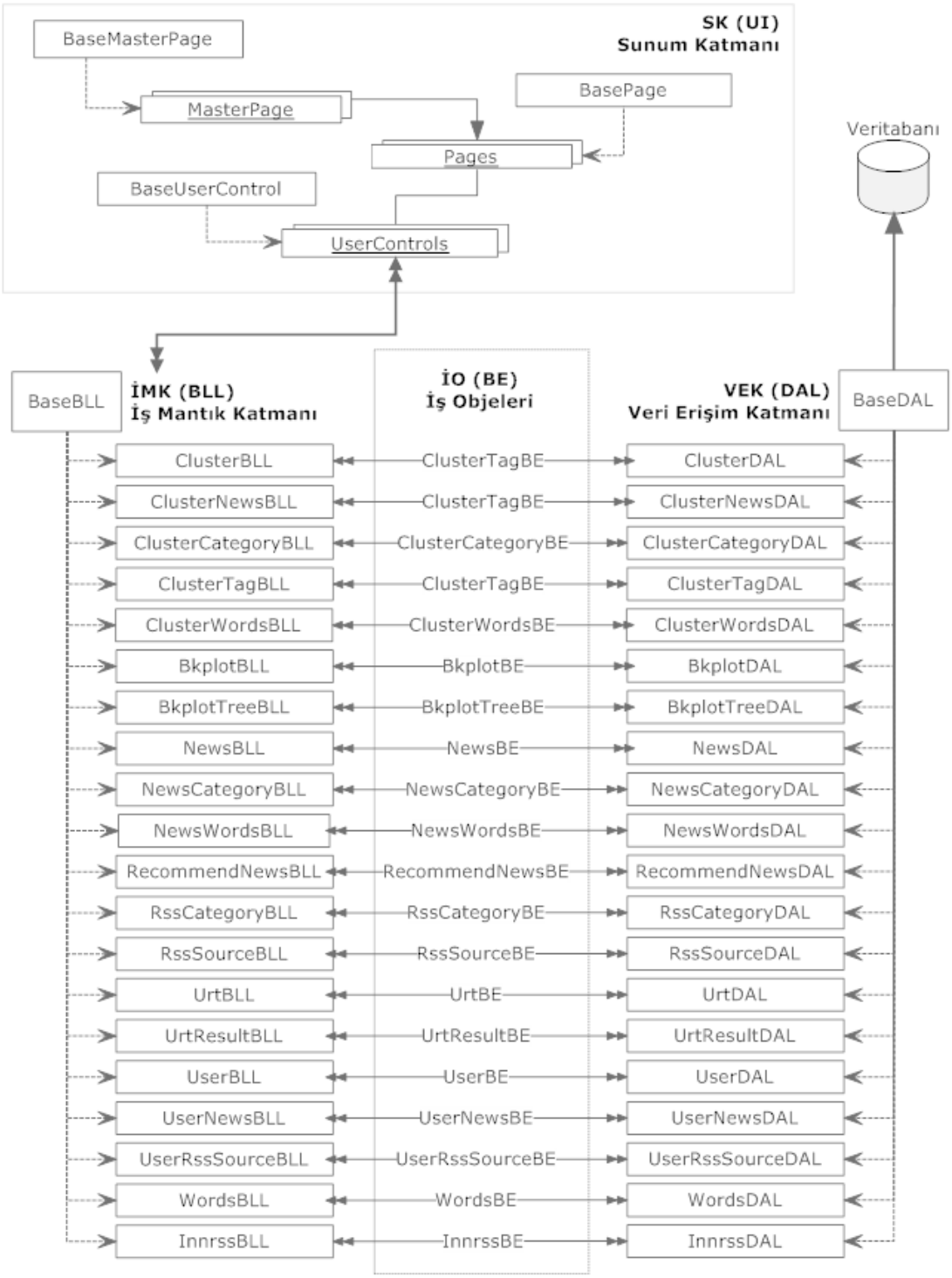
- dbo.APPLYOPTIMIZED2ORIGINAL
- dbo.APPLYORIGINAL2OPTIMIZED
- dbo.COUNTBKPLOTCLUSTERBYUSERID
- dbo.COUNTCLUSTERBYUSERID
- dbo.COUNTOPTIMIZEDCLUSTERBYUSERID
- dbo.COUNTRECOMMENDNEWSBYUSERID
- dbo.DELETEBKPLOTCLUSTER
- dbo.DELETEBKPLOTCLUSTERBYUSERID
- dbo.DELETEBKPLOTNEWS
- dbo.DELETEBKPLOTTREE
- dbo.DELETEBKPLOTTREEBYUSERID
- dbo.DELETEBKPLOTWORDS
- dbo.DELETECLUSTER
- dbo.DELETECLUSTERNEWS
- dbo.DELETECLUSTERTAG
- dbo.DELETECLUSTERTAGBYUSERID
- dbo.DELETECLUSTERWORDS
- dbo.DELETEEMPTYCLUSTERS
- dbo.DELETEEMPTYOPTIMIZEDCLUSTERS
- dbo.DELETENEWS
- dbo.DELETENEWSCATEGORY
- dbo.DELETENEWSWORDS
- dbo.DELETEOPTIMIZEDCLUSTER
- dbo.DELETEOPTIMIZEDCLUSTERBYUSERID
- dbo.DELETEOPTIMIZEDCLUSTERNEWS
- dbo.DELETEOPTIMIZEDCLUSTERWORDS
- dbo.DELETERECOMMENDNEWS
- dbo.DELETERECOMMENDNEWSBYUSERID
- dbo.DELETERSSCATEGORY
- dbo.DELETERSSSOURCE
- dbo.DELETEURT
- dbo.DELETEURTBYUSERID
- dbo.DELETEURTRESULT
- dbo.DELETEUSER
- dbo.DELETEUSERNEWS
- dbo.DELETEUSERNEWSBYRSSSOURCE
- dbo.DELETEUSERRSSSOURCE
- dbo.DELETEUSERRSSSOURCEBYUSERID
- dbo.DELETEWORDS
- dbo.INSERTBKPLOTCLUSTER
- dbo.INSERTBKPLOTNEWS
- dbo.INSERTBKPLOTTREE
- dbo.INSERTBKPLOTWORDS
- dbo.INSERTCLUSTER
- dbo.INSERTCLUSTERNEWS
- dbo.INSERTCLUSTERTAG
- dbo.INSERTCLUSTERWORDS
- dbo.INSERTNEWS
- dbo.INSERTNEWSCATEGORY
- dbo.INSERTNEWSWORDS
- dbo.INSERTOPTIMIZEDCLUSTER
- dbo.INSERTOPTIMIZEDCLUSTERNEWS
- dbo.INSERTOPTIMIZEDCLUSTERWORDS
- dbo.INSERTRECOMMENDNEWS
- dbo.INSERTRSSCATEGORY
- dbo.INSERTRSSSOURCE
- dbo.INSERTURT
- dbo.INSERTURTRESULT
- dbo.INSERTUSER
- dbo.INSERTUSERNEWS
- dbo.INSERTUSERRSSSOURCE
- dbo.INSERTWORDS
- dbo.READALLNEWS
- dbo.READALLNEWSCATEGORY
- dbo.READALLPRMLANGUAGE
- dbo.READALLPRMSOURCETYPE
- dbo.READALLRSSCATEGORY
- dbo.READALLRSSSOURCE
- dbo.READALLURTRESULT
- dbo.READALLUSERS
- dbo.READBKPLOTCLUSTERBYUSERID
- dbo.READBKPLOTNEWSBYUSERID
- dbo.READBKPLOTTREE
- dbo.READBKPLOTTREEBYUSERID
- dbo.READBKPLOTWORDSBYUSERID
- dbo.READCLUSTER
- dbo.READCLUSTERBYUSERID
- dbo.READCLUSTERCATEGORYBYUSERID
- dbo.READCLUSTERNEWS
- dbo.READCLUSTERNEWSBYCLUSTERID
- dbo.READCLUSTERNEWSBYNEWSID
- dbo.READCLUSTERNEWSBYUSERID
- dbo.READCLUSTERRECOMMENDEDBYUSERID
- dbo.READCLUSTERTAG
- dbo.READCLUSTERTAGBYCLUSTERID
- dbo.READCLUSTERTAGBYUSERID
- dbo.READCLUSTERWORDS
- dbo.READCLUSTERWORDSBYCLUSTERID
- dbo.READCLUSTERWORDSBYCLUSTERIDFREQD
- dbo.READCLUSTERWORDSBYUSERID
- dbo.READCLUSTERWORDSBYUSERIDFREQDESC
- dbo.READNEWS
- dbo.READNEWSBKPLOTCLUSTERBYCLUSTERID
- dbo.READNEWSBKPLOTCLUSTERBYUSERID
- dbo.READNEWSBYLINK
- dbo.READNEWSBYUSERID
- dbo.READNEWSBYUSERRSSSOURCEID
- dbo.READNEWSCATEGORY
- dbo.READNEWSCATEGORYBYNEWSID
- dbo.READNEWSCATEGORYBYRSSCATEGORYID
- dbo.READNEWSCLUSTERBYCLUSTERID
- dbo.READNEWSCLUSTERBYUSERID
- dbo.READNEWSOPTIMIZEDBYCLUSTERID
- dbo.READNEWSPROCESSEDBYUSERID
- dbo.READNEWSRECOMMENDBYCLUSTERID
- dbo.READNEWSRECOMMENDEDBYUSERID
- dbo.READNEWSWORDS
- dbo.READNEWSWORDSBYNEWSID
- dbo.READNONRECOMMENDEDNEWSBYUSERID
- dbo.READOPTIMIZEDCLUSTER
- dbo.READOPTIMIZEDCLUSTERBYUSERID
- dbo.READOPTIMIZEDCLUSTERCATEGORYBYUSERID
- dbo.READOPTIMIZEDCLUSTERNEWSBYCLUSTERID
- dbo.READOPTIMIZEDCLUSTERNEWSBYNEWSID
- dbo.READOPTIMIZEDCLUSTERNEWSBYUSERID
- dbo.READOPTIMIZEDCLUSTERWORDSBYCLUSTERID
- dbo.READOPTIMIZEDCLUSTERWORDSBYUSERID
- dbo.READPRMLANGUAGE
- dbo.READPRMLANGUAGEBYCODE
- dbo.READPRMSOURCETYPE
- dbo.READPRMSOURCETYPEBYCODE
- dbo.READREADNEWSBYUSERID
- dbo.READRECOMMENDNEWS
- dbo.READRECOMMENDNEWSBYCLUSTERID
- dbo.READRECOMMENDNEWSBYNEWSID
- dbo.READRECOMMENDNEWSBYRECOMMENDED
- dbo.READRECOMMENDNEWSBYUSERID
- dbo.READRSSCATEGORY
- dbo.READRSSCATEGORYBYNEWSID
- dbo.READRSSSOURCE
- dbo.READRSSSOURCEACTIVEBYUSERID
- dbo.READRSSSOURCEBYLINK
- dbo.READRSSSOURCEBYUSERID
- dbo.READURT
- dbo.READURTBYUSERID
- dbo.READURTCLUSTER
- dbo.READURTNONRECBYUSERID
- dbo.READURTRESULT
- dbo.READURTRESULTBYUSERID
- dbo.READURTRESULTBYUSERIDANDDATE
- dbo.READURTRESULTUSERLIST
- dbo.READUSER
- dbo.READUSERBYEMAIL
- dbo.READUSERNEWS
- dbo.READUSERNEWSBYNEWSID
- dbo.READUSERNEWSBYRSSSOURCEID
- dbo.READUSERNEWSBYUSERID
- dbo.READUSERNEWSCLUSTER
- dbo.READUSERNEWSPROCESSEDBYUSERID
- dbo.READUSERRSSSOURCE
- dbo.READUSERRSSSOURCEBYUSERID
- dbo.READUSERUNREADNEWS
- dbo.READUSERUNREADNEWSBYRSSSOURCEID
- dbo.READWORDS
- dbo.SEQUENCENEXTVAL
- dbo.UPDATEBKPLOTCLUSTER
- dbo.UPDATEBKPLOTNEWS
- dbo.UPDATEBKPLOTTREE
- dbo.UPDATEBKPLOTWORDS
- dbo.UPDATECLUSTER
- dbo.UPDATECLUSTERNEWS
- dbo.UPDATECLUSTERTAG
- dbo.UPDATECLUSTERWORDS
- dbo.UPDATENEWS
- dbo.UPDATENEWSCATEGORY
- dbo.UPDATENEWSWORDS
- dbo.UPDATEOPTIMIZEDCLUSTER
- dbo.UPDATEOPTIMIZEDCLUSTERNEWS
- dbo.UPDATEOPTIMIZEDCLUSTERWORDS
- dbo.UPDATERECOMMENDNEWS
- dbo.UPDATERSSCATEGORY
- dbo.UPDATERSSSOURCE
- dbo.UPDATEURT
- dbo.UPDATEURTRESULT
- dbo.UPDATEUSER
- dbo.UPDATEUSERNEWS
- dbo.UPDATEUSERRSSSOURCE
- dbo.UPDATEWORDS

EK – C: Yahoo haber kaynağı kategori listesi

A	Politika (Politics)
A.1	Genel Politika (General Politics)
A.2	Askeri (Military)
A.3	Meclis (Congress)
B	İş Dünyası (Business)
B.1	Borsa (Stock Markets)
B.2	Ekonomi (Economy)
B.3	Avrupa Ekonomisi (European Economy)
B.4	Şirket Kazançları (Company Earnings)
B.5	Kişisel Finans (Personal Finance)
C	Dünya (World)
C.1	Orta Doğu (Mideast)
C.2	Birleşmiş Milletler (United Nations)
C.3	Afrika (Africa)
C.4	Çin (China)
C.5	Avrupa (Europe)
C.6	Hindistan (India)
C.7	Iran (Iran)
C.8	Japonya (Japan)
C.9	Latin Amerika (Latin America)
C.10	Meksika (Mexico)
C.11	Kuzey Kora (North Korea)
C.12	Rusya (Russia)
D	Teknoloji (Technology)
D.1	İnternet (Internet)
D.2	Kişisel Teknoloji (Personal Technology)
D.3	Linux/Açık Kaynak (Linux/Open Source)
D.4	Mobil ve Kablosuz (Mobile & Wireless)
D.5	Ticari Girişim (Enterprise)
D.6	Yazılım (Software)
D.7	Apple/Macintosh (Apple/Macintosh)
D.8	Bilgisayar Güvenliği (Computer Security)
D.9	Yarı İletken Endüstrisi (Semiconductor Industry)
D.10	Video Oyunları (Video Games)
D.11	Dijital Müzik (Digital Music)
D.12	Portallar ve Arama Motorları (Portals & Search Engines)
E	Eğlence (Entertainment)
E.1	Dedikodu/Kutlama (Gossip/Celebrity)
E.2	Sinema (Movies)
E.3	Televizyon (Television)
E.4	Müzik (Music)

E.5	Moda (Fashion)
E.6	Sanat & Tiyatro (Arts & Stage)
F	Yaşam (Health)
F.1	Beslenme ve Diyet (Weight Loss & Nutrition)
F.2	Cinsel Yaşam (Sexual Health)
F.3	İlaç (Medications/Drugs)
F.4	Ebeveynlik (Parenting)
F.5	İhtiyarlık ve Anti-Aging (Seniors and Aging)
F.6	Hastalıklar ve Durumlar (Diseases & Conditions)
G	Bilim (Science)
G.1	Hava Haberleri (Weather News)
G.2	Uzay ve Astronomi (Space & Astronomy)
G.3	Hayvanlar (Animals & Pets)
G.4	Dinazorlar ve Fosiller (Dinosaurs & Fossils)
G.5	Biyoteknoloji (Biotechnology)
G.6	Enerji (Energy)
G.7	Çevre (Environment)
H	Spor (Sports)
H.1	Futbol (Football)
H.2	Basketbal (Basketball)
H.3	Tenis (Tennis)
H.4	Formula 1 (Formula 1)
H.5	Amerikan Futbol (Rugby)
H.6	Golf (Golf)

EK – D: Innrss katmanlı sınıf diyagramı



EK – E: Haber tavsiye kod örneği

```
try
{

  foreach (UserNewsBE.NEWSRow newsRow in userNewsBE.NEWS)
  {
    NewsWordsBE newsWordsBE = BLL.NewsWordsBLL.ReadByNewsID(newsRow.ID);

    foreach (ClusterBE.CLUSTERRow clusterRow in clusterBE.CLUSTER)
    {
       RecommendNewsBE.RECOMMENDNEWSRow recommendNewsRow =
                                        recNewsBE.RECOMMENDNEWS.NewRECOMMENDNEWSRow();

       recommendNewsRow.WCDOLD = clusterRow.WCD;

       var queryCluster = from DataRow clusterWordsRow in  clusterBE.CLUSTERWORDS.Rows
                          where (int)clusterWordsRow["CLUSTERID"] == clusterRow.ID
                          select clusterWordsRow;


    if (queryCluster.Count<DataRow>() > 0)
    {
       ClusterWordsBE clusterWords = queryCluster.CopyToDataTable<DataRow>();
       ClusterWordsBE mergedWords = BLL.ClusterWordsBLL.Merge(cWords, newsWordsBE.NEWSWORDS);

        recommendNewsRow.WCDNEW = BLL.InnrssBLL.WCD(mergedWords);
        recommendNewsRow.USERID = CurrentUserID;
        recommendNewsRow.NEWSID = newsRow.ID;
        recommendNewsRow.CLUSTERID = clusterRow.ID;
        recommendNewsRow.DELTA = recommendNewsRow.WCDNEW - recommendNewsRow.WCDOLD;
        recommendNewsRow.ISRECOMMENDED = "N";

        if (recommendNewsRow.DELTA > MaxDELTA)
        {
            MaxDELTA = recommendNewsRow.DELTA;
            impClusterID = recommendNewsRow.CLUSTERID;
            impNewsID = recommendNewsRow.NEWSID;
        }

        recommendNewsBE.RECOMMENDNEWS.AddRECOMMENDNEWSRow(recommendNewsRow);
       }
    }

    if (impClusterID > 0 && impNewsID > 0)
    {
        NewsBE maxRecommendNews = recommendNewsBE.RECOMMENDNEWS.Select("USERID = " +
        CurrentUserID + " AND CLUSTERID = " + impClusterID + " AND NEWSID = " + impNewsID);

        if (maxRecommendNews.Length > 0)
        {
             if(maxRecommendNews.DELTA > 0)
                maxRecommendNews.ISRECOMMENDED = DbConstants.YES;
        }
     }
   }

   BLL.RecommendNewsBLL.Flush(recommendNewsBE);

}
catch (Exception ex)
{
    ManageException(ex);
}
```

ÖZGEÇMİŞ

Kişisel Bilgiler

Soyadı, Adı	:	DUMAN, Çağlar
Uyruğu	:	T.C.
Doğum tarihi ve yeri	:	23.02.1983 Ankara
Medeni hali	:	Bekâr
Telefon	:	0 (312) 266 05 99
Email	:	cduman@etu.edu.tr caglarduman@gmail.com

Eğitim

Derece	**Eğitim Yeri**	**Mezuniyet Tarihi**
Lisans	Çankaya Üniversitesi | Bilgisayar Müh.	2007

İş Deneyimi		**Görev**
2011-Halen	Türksat	Yazılım Uzmanı
2010-2011	Esc-Serena Bilgi Teknolojileri A.Ş.	Yazılım Uzmanı
2009-2010	C-Tasarım Web Yazılım Hizmetleri	Yazılım Uzman Yrd.
2008-2008	Entegre Enformasyon Sistemleri	Yazılım Uzman Yrd.

Yabancı Dil

İngilizce