T.C.
KIRIKKALE ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI
YÜKSEK LİSANS TEZİ

GİZLİ ANLAMSAL ANALİZ İLE METİN SINIFLANDIRMA

Emre DENİZ

EYLÜL 2017

Bilgisayar Mühendisliği Anabilim Dalında Emre DENİZ tarafından hazırlanan GİZLİ ANLAMSAL ANALİZ İLE METİN SINIFLANDIRMA adlı Yüksek Lisans Tezinin Anabilim Dalı standartlarına uygun olduğunu onaylarım.

Prof. Dr. Ali ERİŞEN
Anabilim Dalı Başkanı

Bu tezi okuduğumu ve tezin **Yüksek Lisans Tezi** olarak bütün gereklilikleri yerine getirdiğini onaylarım.

Prof. Dr. Hasan ERBAY
Danışman

Jüri Üyeleri

Başkan	: Yrd. Doç. Dr. Mustafa COŞAR	_____
Üye (Danışman)	: Prof. Dr. Hasan ERBAY	_____
Üye	: Yrd. Doç. Dr. Bülent Gürsel EMİROĞLU	_____

…/…/…

Bu tez ile Kırıkkale Üniversitesi Fen Bilimleri Enstitüsü Yönetim Kurulu Yüksek Lisans derecesini onaylamıştır.

Prof. Dr. Mustafa YİĞİTOĞLU
Fen Bilimleri Enstitüsü Müdürü

ÖZET

GİZLİ ANLAMSAL ANALİZ İLE METİN SINIFLANDIRMA

DENİZ, Emre
Kırıkkale Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı, Yüksek Lisans Tezi
Danışman: Prof. Dr. Hasan ERBAY
Eylül 2017, 49 sayfa

Günümüzde, çoğunluğu metinsel veriler olmak üzere birçok veri kaynağından bilgi elde edilebilmektedir. Spesifik bir konuda aradığımız bilgiyi elde etmek için tüm dokümanları incelemek mümkün değildir. Verileri otomatik olarak sınıflandırmak, istediğimiz verilere ulaşmada önemli bir avantaj sağlar. Gizli Anlamsal Analiz (LSA), Tekil Değer Ayrışımını (SVD) kullanarak bir vektör uzayındaki terimler ve dokümanlar arasındaki gizli yapıyı ortaya çıkaran yöntemlerden biridir. Dokümanların dizinlenmesi, otomatik özetlenmesi ve anahtar kelimelerinin belirlenmesi gibi çalışmalarda kullanılan LSA, yapısı itibari ile metin sınıflandırma alanında da kullanılabilir. Bu çalışmada Reuters veri tabanındaki metinsel veriler kullanılarak LSA ile metin sınıflandırması gerçekleştirilmiştir. Reuters veri tabanından alınan beş sınıfa ait metinsel verilerin terim-sınıf matrisi oluşturulmuştur. Elde edilen terim-sınıf matrisine SVD uygulanarak rank-k yaklaşımına göre anlamsal uzay elde edilmiştir. Bu anlamsal uzaydaki terim ve terimlerin ait olduğu sınıfların konumları temel alınarak sınıfı önceden bilinen dokümanların kosinüs benzerliğine göre ait olabileceği sınıflar listelenmiştir. Yapılan testler sonucunda elde edilen bulgular incelendiğinde

önerilen sınıflama yönteminin büyük oranda doğru sonuçlar çıkardığı gözlemlenmiştir ve mevcut sınıflandırma yöntemlerine alternatif olabileceği görülmüştür.

Anahtar Kelimeler: Metin Madenciliği, Metin Sınıflandırma, Gizli Anlamsal Analiz, Tekil Değer Ayrışımı

ABSTRACT

TEXT CLASSIFICATION WITH LATENT SEMANTIC ANALYSIS

DENİZ, Emre
Kırıkkale University
Graduate School of Natural and Applied Sciences
Department of Computer Engineering, M. Sc. Thesis
Supervisor: Prof. Dr. Hasan ERBAY
August 2016, 49 pages

Today, information can be obtained from many data sources, most of which are textual data. In a specific matter, it is not possible to examine all the documents in order to obtain the information we seek. Classifying the data automatically provides an important advantage in reaching the data we want. Latent Semantic Analysis(LSA) is one of the methods that reveals the latent structure between documents and terms in a vector space using Singular Value Decomposition(SVD). The LSA used in studies such as indexing of documents, automatic summarization and determination of key words documents, can also be used in text classification field by structure. In this study, text classification with LSA was performed using textual data from Reuters database. The term-class matrix of the textual data of the five classes taken from the Reuters database was constructed.The semantic space is obtained according to rank-k approximation by applying SVD to the obtained term-class matrix. Based on the positions of the classes to which the terms and terms in this semantic space belong, the classes to which the previously known documents belong can be classified according to cosine similarity. When the findings obtained from the tests conducted are examined, it is observed that the proposed classification method has resulted in correct results.

Key Words: Text Mining, Text Classification, Latent Semantic Analysis, Singular Value Decomposition

TEŞEKKÜR

Tezimin hazırlanması esnasında yardımlarını esirgemeyen ve büyük destek olan tez yöneticisi hocam, Sayın Prof. Dr. Hasan ERBAY'a, tezimin düzenlemelerini yaparken yardımlarını esirgemeyen Arş. Gör. Fatih VARÇIN, Arş. Gör. Enes AYAN, Arş. Gör. Fahrettin HORASAN'a teşekkürlerimi sunarım.

Maddi ve manevi her zaman yanımda olan desteklerini hiçbir zaman esirgemeyen aileme, özellikle gösterdiği sabır ve verdiği destekten ötürü sevgili eşim, Merve DENİZ'e teşekkür ederim.

İÇİNDEKİLER DİZİNİ

Sayfa

ŞEKİLLER DİZİNİ

ŞEKİL Sayfa

ÇİZELGELER DİZİNİ

ÇİZELGE Sayfa

1. GİRİŞ

Günümüzde birçok veri kaynağından bilgi keşfedilebilmektedir. Bu verilerin büyük bir çoğunluğu metin olarak saklanmaktadır. Artan veri miktarı göz önüne alındığında, tüm dokümanları inceleyerek aranılan bilgiye ulaşmak çok da mümkün görünmemektedir. Verilerin büyük kısmının bilgisayar ortamında saklandığı göz önüne alındığında, verileri otomatik olarak kategorize etmek insanlara çok avantaj sağlamaktadır. Bu sayede zamandan da tasarruf edilebilmektedir. Bu ihtiyaçlardan dolayı, metin sınıflandırma akademik çalışmalarda sıklıkla tercih edilen konulardan bir tanesidir.

Öte yandan, sürekli artan veri miktarı, veri madenciliğinin gelişmesine yol açmıştır. Bu verilerin çoğu metin olarak saklandığı için, veri madenciliğinin bir alt dalı olan metin madenciliği de birçok alanda kullanılmaktadır. Doğal Dil İşleme, Bilgi Çıkarım Sistemleri, Bilgi Erişim Sistemleri ve Doküman Sınıflandırma metin madenciliğinin temel kullanım alanlarıdır [1].

Mevcut verilerin çoğunluğunun metin olarak saklanması neticesinde veri madenciliğine paralel olarak metin madenciliği de hızlı bir şekilde gelişmiştir. Metin sınıflandırma da metin madenciliğinin en önemli parçasını oluşturmaktadır.

Gizli Anlamsal Analiz(LSA), doküman yığınındaki saklı ilişkileri keşfetmede kullanılan matematiksel ve istatiksel bir yöntemdir [2]. LSA daha çok metinsel veriler üzerinde kullanılmaktadır. LSA metinlerin tutarlılığını ölçmek için kullanılan bir tekniktir [3]. LSA ile verilen bir doküman yığınının içeriğine ait ilişkiler bulunarak amaca uygun şekilde kullanıma hazır hale getirilebilir. Metin madenciliği ile veriler üzerinde önişlem yapıp işlenmeye hazır hale getirdikten sonra LSA ile bu verilerdeki gizli bağları bularak birbirine yakın verileri çeşitli kategorilere ayırabiliriz. Bu düşünceden yola çıkarak bu çalışmamızda LSA ile metin sınıflandırma yöntemi geliştirilmiştir.

Tezin geriye kalan kısmı şu şekilde organize edilmiştir. İkinci bölümde metin madenciliği hakkında bilgi verilmektedir. Bu bölümde, metin madenciliğin önemine ve gerekliliğine değinildikten sonra metin madenciliğinin adımları ve uygulama alanları anlatılmıştır.

Tezin üçüncü bölümünde metin sınıflandırma uygulamalarında kullanılan yöntemlerden K-NN algoritması, Destek Vektör Makineleri(SVM) ve Naive Bayes Yöntemi anlatılmıştır. Ayrıca bu alanda yapılmış çalışmalar hakkında literatür bilgisi verilmiştir.

Tezin dördüncü bölümünde çalışma boyunca kullanılan metotlar hakkında bilgi verilmiştir. Bu yöntemler sırası ile Gizli Anlamsal Analiz, Tekil Değer Ayrışımı, Rank-*k* Yaklaşımı, Ağırlıklandırma Yöntemleri ve Kosinüs Benzerliği yöntemleridir.

Tezin beşinci bölümünde metin sınıflandırma için LSA kullanarak gerçekleştirdiğimiz uygulama hakkında bilgi verilmiştir. Uygulama süresince gerçekleştirmiş olduğumuz metin madenciliği işlemleri, veri tabanı işlemleri ve sınıflandırma uygulaması bu bölümde anlatılmıştır.

Tezin altıncı bölümünde tez çalışması sonucunda elde ettiğimiz sonuçlara yer verilmiştir. Bu sonuçlar arasında veri setimizde yer alan test verileri kullanılarak elde ettiğimiz performans sonuçları bulunmaktadır. Ayrıca çalışma sonucu elde ettiğimiz sonuçlar bu bölümde tartışılmıştır.

2. METİN MADENCİLİĞİ

Veri, gerçekleştirilen ve geliştirilmekte olan birçok çalışmanın ana kaynağıdır. İstatistiğin temel noktası olarak da adlandırabileceğimiz veriler, yapılacak birçok çalışmada önemli bir yer kaplamaktadır. Elde edilen verilerin işlenerek anlamlı bir yapıya dönüştürülmesi ile bilgi elde edilir. Gelişen teknoloji ile birlikte verilerin miktarı, kullanım alanı ve depolama yöntemleri de sürekli paralel şekilde gelişme göstermektedir. Artan veri miktarı neticesinde depolama alanı ve dosya sayısı her geçen gün daha da çoğalmaktadır. Bu da aranılan bilginin bulunmasında bize zorluk çıkarmaktadır. Aradığımız bir bilgiye ulaşmak için mevcut tüm dosyalara bakmamız fazla zaman kaybı yaşatacağından pratik bir seçenek olmayacaktır. Bu nedenle, verileri bazı kategoriler altında sınıflandırmak, aradığımız bilgiye ulaşmada bize çok avantaj sağlayacaktır.

Bilgiler, verilere nazaran daha kolay depolanabilmekle birlikte kullanım sıklığı da verilere göre çok daha fazladır. Bilginin temelini oluşturan verileri iki grupta toplayabiliriz. Bunlar yapılandırılmış veriler ve yapılandırılmamış verilerdir.

2.1. Yapılandırılmış Veriler

Yapılandırılmış veriler, ilişkisel veri tabanına düzgün bir şekilde yüklenen ve oldukça organize hale getirilmiş olan verilerdir. Çoğunlukla matrislerde satır ve sütunlar halinde gösterilirler. Yapılandırılmış veriler, arama işlemleri veya algoritmalar yoluyla kolayca işlenebilirler. Bu türdeki verileri depolamak, sorgulamak ve analiz etmek yapılandırılmamış verilere göre daha kolaydır, ancak bunun için satır ve sütundaki alanların net bir şekilde tanımlanması gerekmektedir. Gelişen teknoloji ile birlikte yapılandırılmış verinin elde edilmesi, başka formatlarda temsil edilmesi ve depolanması da kolaylaşmıştır.

Çizelge 2.1'de yapılandırılmış veri örneği görülmektedir.

Çizelge 2.1. Yapılandırılmamış Veri Örneği

Terim	**Sınıf**	**Frekans**
Ayva	1	13
Çay	3	11
Kâğıt	2	9
Armut	1	4
Kahve	3	8
Nar	1	8
Defter	1	10
Şeker	3	12
Elma	1	15
Kalem	2	12
Kitap	2	11
Bardak	3	8
Kivi	1	5
Cetvel	2	7
Kaşık	2	9

Çizelge 2.1'de görüleceği gibi aynı sütundaki veriler aynı tiptendir. Yapılandırılmamış verilerin bu şekilde matris haline dönüştürülerek gösterilmesi, daha kolay bilgi çıkarımı yapılmasını sağlayacaktır.

2.2. Yapılandırılmamış Veriler

Yapılandırılmamış veriler mektup, kitap, eposta gibi yazılı metinlerden ya da görüntü ve seslerden oluşmaktadırlar. Bu veriler, yapısı itibari ile yapısal veri tabanı gösterimine uymamaktadır. Genelde metin madenciliğinin üzerinde çalıştığı veri grubu, yapılandırılmamış verilerden oluşur. Yapılandırılmamış veriler bazı önişlemlerden geçirilerek yapılandırılmış veriler elde edilir. Şekil 2.1'de

Reuters21578 veri setinden alınan yapılandırılmamış veriye ait bir örnek görülmektedir [4].

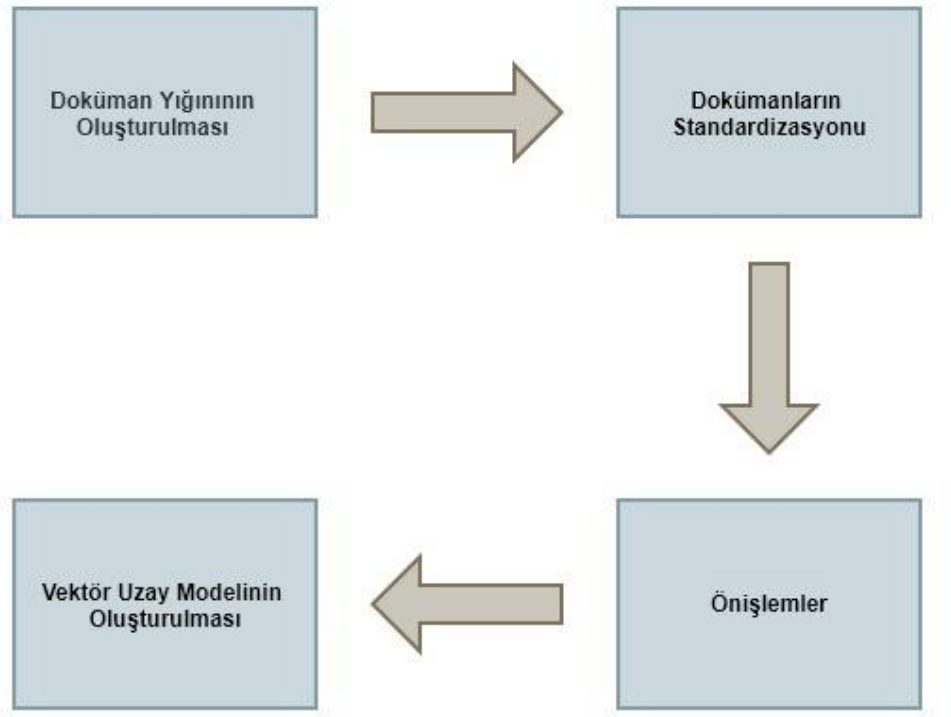
Şekil 2.1. Yapılandırılmamış Veri Örneği

Günümüz teknolojisine bağlı olarak, internet ve bilgisayar kullanımındaki artış doküman yığınının birikmesine neden olmuştur. Hâlihazırda bu dokümanların büyük bir çoğunluğu da yapılandırılmamış verilerdir. Metin madenciliği yapılandırılmamış formattaki verileri işleyerek bu verileri kullanılabilir hale dönüştürür. Böylece yapılandırılmış veri elde edilir. Bazı yapılandırılmış veriler, bilgi olarak da kullanabilir. Ancak her yapılandırılmış veri bilgi olmayabilir. Bunun için bu yapılandırılmış verilerin daha da işlenmesi gerekmektedir.

2.3. Metin Madenciliği Adımları

Veriler üzerinde işlem yapabilmek için, verilerin veri madenciliği metotlarının uygulanabileceği bir biçime dönüştürülmesi gerekmektedir. Verilerin seçilerek, belirli

ölçütlere göre temizlenip istenilen özelliklerin çıkarılması, metin madenciliğinin ana hedefidir. Şekil 2.2'de metin madenciliğinin temel adımları gösterilmiştir.



Şekil 2.2. Metin Madenciliğinin Adımları

2.3.1. Doküman Yığınının Oluşturulması

Metin madenciliği, veri yığınının elde edilmesiyle başlar. Metin madenciliğinin uygulanacağı alana göre değişik doküman yığınları oluşturulabilir. Bu veriler web sitelerinden elde edilebileceği gibi, eposta ile gönderilen mesajlardan ya da mektup ve kitaplardan toplanabilir.

Metin madenciliği tekniklerinin araştırılması ve geliştirilmesi için daha kapsamlı veriler gerekli olabilir. Bunlar 'corpus' ya da 'derlem' olarak adlandırılmaktadır. Literatürde hazır veri setleri olarak sunulan bu verilerden birçok çalışmalarda

faydalanılmaktadır. Reuters veri setleri en çok kullanılan verilerdendir. Bununla birlikte farklı alanlar için birçok hazır veri seti bulunmaktadır.

Ayrıca, bazı devlet kurumları ve şirketler büyük veri koleksiyonlarına sahiptir. Bu kurumlar bazen bu verileri paylaşabilirler. Örneğin, National Institutes of Health tarafından paylaşılan MEDLINE veri seti, metin madenciliği çalışmalarında en çok kullanılan veriler arasındadır [5].

Veriler elde edildikten sonra, hepsinin aynı standartta olup olmadığı kontrol edilmelidir. Birden farklı kaynaktan oluşturulan doküman yığınlarında, standardizasyon sorunu ortaya çıkabilir. Bunu önlemek için tüm verilerin aynı standartta temsil edilmesi gerekmektedir.

2.3.2. İşaretleme

Toplanan veriler aynı standartta biçimlendirildikten sonra, metin işlemlerine başlanabilmektedir. Burada uygulanacak ilk işlem, metni kelimelere ya da belirteçlere dönüştürmektir. Metni kelimelere ayırma süreci, metin içerisindeki gizli bilgileri ve yapıları ortaya çıkarmada kritik bir aşamadır. İşaretleme birçok farklı seviyede yapılabilir. Metni paragraf, cümle ya da kelime düzeyinde belirteçlere ayırabiliriz. Metin madenciliği çalışmalarında en çok kullanılan yaklaşım, metni cümle veya kelimelere ayırmaktır [6].

Karakter akışının işaretlere ayrılması, dil yapısına hâkim biri için basit bir işlemdir. Fakat bu işlemin bir bilgisayar programı tarafından gerçekleştirilmesi daha zor olmaktadır. Bunun nedeni ise belirli karakterlerin kullanıma bağlı olarak, bazı durumlarda işaretin sınırlayıcı olması, bazı durumlarda ise olmamasıdır. Boşluk, sekme ve satırbaşı karakterlerinin hep sınırlayıcı olduğu, işaret olarak sayılmadığı varsayılmaktadır. Ancak “( )”, “< >”, “!” ve “?” karakterleri her zaman sınırlayıcı karakterlerdir. Ayrıca “.”, “,”, “:” ve “-” karakterleri de kullanımına bağlı olarak sınırlayıcı olabilir veya olmayabilirler.

Öte yandan, kullanılan işaretleme algoritmasının net bir şekilde oluşturulması gerekmektedir. Cümle ya da kelime seviyesinde işaretleme yapılacağını düşünüldüğünde, her biri için de belirteçleri belirleyecek karakterleri ayrı ayrı tanımlanmalıdır. Örneğin, her bir cümle için işaretleme yapmak istenildiğinde, her cümlenin sonunun hatasız şekilde bulunması gerekmektedir. Ardından bir sonraki cümlenin başlangıcı da tespit edilebilmelidir. Böylece cümle düzeyinde işaretleme gerçekleştirilebilir.

2.3.3. Gövdeleme

Doküman yığınından alınan bir veri belirteçlere ayrıldıktan sonra, her bir belirtecin aynı standart forma dönüştürülmesi gerekmektedir. Gövdeleme olarak bilinen bu işlemler literatürde ‘stemming’ ya da ‘lemmatization’ olarak adlandırılır [7]. Bu süreç metin madenciliği uygulamalarında zorunlu olmadığı gibi, yapılan çalışmaya göre tercih edilebilir. Metin sınıflandırma çalışmalarında gövdeleme işlemi avantaj sağlamaktadır [7]. Kelimelerin kök halinin elde edilmesi, terim sayısını azaltmanın yanısıra aynı köke sahip terimlerin frekanslarının daha doğru hesaplanmasına da yardımcı olacaktır.

Türkçe, İngilizce ya da diğer dillerde kelimeler değişik farklı formlarda görülebilirler. Birbirinden farklı görünen bu kelimeler, aslında ortak kök yapısına sahip olabilirler. Bu aşama Türkçe kelimeler üzerinde şu şekilde açıklanabilir. ‘Gel’ kelimesi kök halinde bir fiildir. Bu kelimeye birçok yapım eki ya da çekim eki getirerek yeni kelimeler elde edilebilir. ‘Geldim’, ‘gelecek’ ve ‘gelebilecek’ kelimeleri incelenirse, hepsi de birbirinden farklı birer belirteç olarak karşımıza çıkmaktadır. Gövdeleme işlemi, bu kelimeleri tek bir form haline, yani ‘gel’ fiil köküne indirgemiş olur.

Çizelge 2.2’de bazı Türkçe terimler ve bunların gövdeleme sonucu bulunan kök halleri gösterilmektedir.

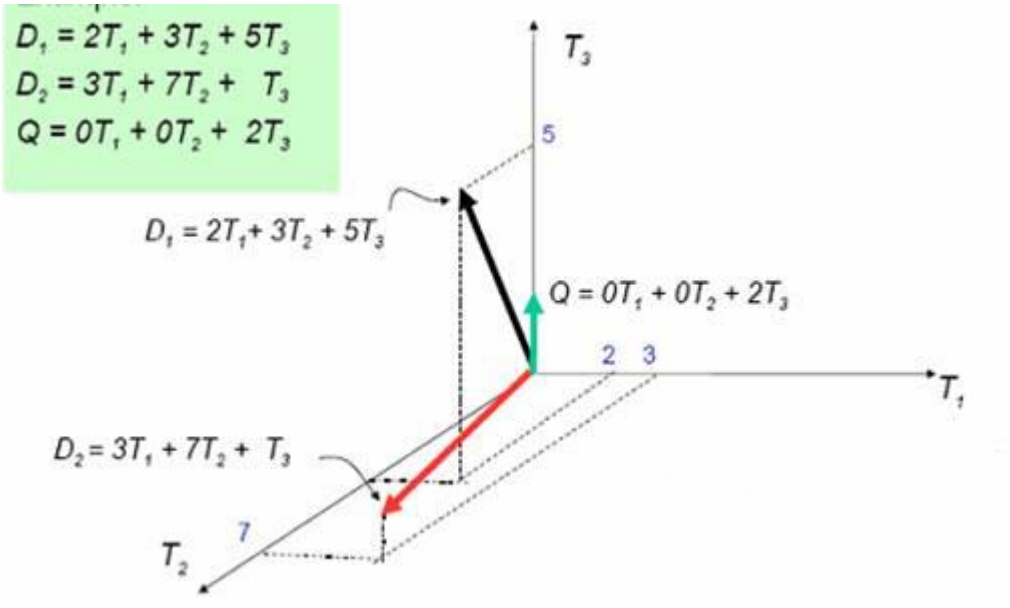
Çizelge 2.2. Gövdeleme Örneği

Terim	**Kök Hali**
Kalemlik	Kalem
Gelecek	Gel
Sızıntı	Sız
Meyveler	Meyve
Uçurum	Uç

Öte yandan gövdeleme işleminin beraberinde getirdiği bazı dezavantajlar da bulunmaktadır. Kelimeler üzerinde gerçekleştirilen gövdeleme işlemi ile anlam kayıpları yaşanabilmektedir. Örneğin, ‘gelecek’ kelimesini ele alırsak, bu kelime gövdeleme işleminden sonra ‘gel’ köküne indirgenecektir. Elde edilen bu kök, sadece fiil anlamında olan ‘gelmek’ kelimesini ifade etmektedir. Ancak kelimenin asıl hali olan ‘gelecek’ ise hem ileri bir zamanı niteleme anlamında, hem de ‘gel’ fiil kökünün ‘-ecek’ ekini almış hali şeklinde kullanılabilir. Gövdeleme sonucu kelimenin asıl hali kaybedileceği için, cümlenin anlamı da kaybedilmektedir.

2.3.4. Vektör Uzay Modelinin Oluşturulması

Doküman yığını üzerinde gerçekleştirilen gövdeleme ve işaretleme işlemleri önişlem olarak adlandırılmaktadır. Yapılacak olan çalışmanın gereksinimlerine göre bu önişlem süreci detaylandırılabilir. Bununla birlikte, bu veriler üzerinde analiz yapılabilmesi için metinlerin sayısal forma dönüştürülmesi ve matris formunda gösterilmesi gerekmektedir. Bu süreç doküman gösterimi olarak da bilinir. En yaygın yöntem ise vektör uzay modelidir. Bu modelde her bir doküman bir vektör olarak temsil edilir. Vektör uzay modeli bilgi çıkarımı, bilgi filtreleme, indeksleme ve doküman sınıflandırma gibi alanlarda kullanılan cebirsel bir yöntemdir. Doğal dil belgelerinin çok boyutlu uzayda özel bir anlamını simgelemektedir [8].

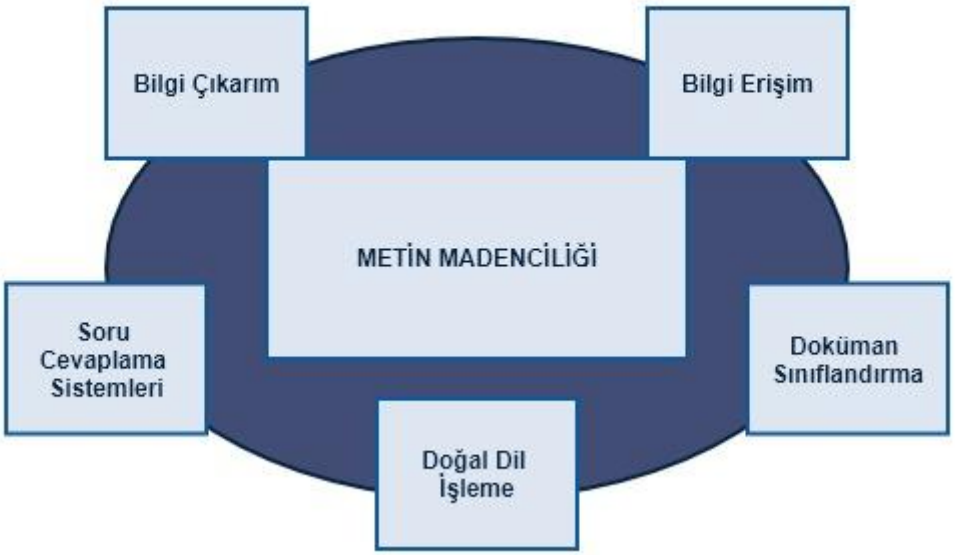


Şekil 2.3. Vektör Uzay Modeli

Dokümanlar şekil 2.3'de görüldüğü gibi kelimelerin vektörleri olarak ifade edilirler. T'ler ise kelimeleri ifade etmektedir. Vektörler arasındaki açının kosinüsü hesaplanıp karşılaştırma yapılarak vektörlerin benzerlikleri karşılaştırılabilir [9].

2.4. Metin Madenciliği İle İlişkili Alanlar

Yapılan çalışmalar incelendiğinde, birçok alanda metin madenciliğinden faydalanıldığı görülmektedir. Metinsel verinin bulunduğu her alanda metin madenciliği uygulamaları geliştirilebilmektedir. Metin madenciliği ile ilişkili olan başlıca alanlar Şekil 2.4'de gösterilmiştir [1].



Şekil 2.4. Metin Madenciliği İle İlişkili Alanlar

Daha önce verilerin miktarından ve tutulma şekillerinden bahsedilmişti. Bu verilerle başa çıkmak çoğu zaman mümkün olmamaktadır. Bir insanın doküman yığınlarındaki verileri faydalı bir şekilde kullanabilmesi için bu verilerin işlenmeye hazır hale getirilmesi gerekmektedir. Önişlem olarak da adlandırılan bu süreç genel olarak Doğal Dil İşleme olarak bilinir. Doğal Dil İşleme, metin madenciliğinin öncül aşamalarındandır [10]. Bu sayede bilgisayarın verileri daha iyi analiz etmesi sağlanabilmektedir. Ayrıca insanlara da bu verileri kullanırken kolaylık sağlamaktadır. Verileri standart forma getirme, belirteçlere ayırma, kelimeleri köklerine ayırma ve etkisiz kelimeleri verilerden kaldırma Doğal Dil İşlemenin temel aşamalarıdır. Tüm verileri sade, standart ve kullanılabilir bir biçime getirmeyi amaçlayan Doğal Dil İşleme, metin madenciliğinin diğer kullanım alanlarının da temelini oluşturmaktadır [10].

Gelişen teknoloji ve bilgisayar kullanımındaki artış ile veriler daha çok bilgisayarlar üzerinde depolanmaya başlamıştır. Depolama yönteminin değişmesi de beraberinde

bu verilere erişmek için yeni yöntem ihtiyaçları doğurmuştur. Böylece bilgi erişim sistemleri geliştirilmiştir. Yapılan çalışmalar neticesinde arama motorları tasarlanmış ve verilere ulaşmak daha kolay hale gelmiştir. Ancak aşırı yoğun bir şekilde artan veri miktarı sonucu, aranılan verilere erişmek sanıldığı kadar da kolay bir işlem olmayabilir. Veriye erişimi kolaylaştırmak için bilgi erişim sistemlerinin veriye en hızlı biçimde getirecek şekilde geliştirilmesi gerekmektedir. Burada amaç, kullanıcı sorgularını iyi bir şekilde analiz ederek doküman yığınındaki bu sorgulara en yakın verileri bulup kullanıcılara getirmektir. Bu kapsamda birçok arama motoru geliştirilmiştir. Google, Yahoo! Search, Yandex ve Altavista en çok bilinen arama motorlarıdır.

Metin madenciliğinin ilgilendiği en önemli alanlardan bir diğeri ise bilgi çıkarımıdır. Doküman yığınındaki verilerden en faydalı bilgileri çekerek bilgi keşfi yapmak bu verilerin kullanımı açısından insanlara birçok kolaylık sağlamaktadır. Bilgi çıkarım sistemleri genellikle yapılandırılmamış veriler üzerinde çalışmaktadır. Bu veriler üzerinde metin madenciliği uygulanarak veriler yapılandırılıp bilgi keşfi gerçekleştirilebilir. Bu sistemler anahtar tabanlı ya da benzerlik tabanlı çalışmaktadır. Burada girilen anahtar kelimeler dikkate alınarak doküman yığınındaki benzer veriler çekilip kullanıcının karşısına sunulmaktadır. Bilgi erişimi ve bilgi çıkarımı benzer gibi gözükmektedir ancak bazı temel noktalarda birbirlerinden ayrılmaktadır. Bilgi erişim sistemleri kullanıcıdan aldığı sorgu neticesinde bu sorguya yakın verileri kullanıcıya verir. Bilgi erişim sistemleri ise, girilen anahtar kelimelere göre ilgili veriler tam olarak tespit edilip yapılandırılmış bir biçimde kullanıcıya sunulmaktadır.

Metin madenciliğinin bir diğer kullanım alanı da doküman sınıflandırmadır. Depolanılan veriler birbirinden farklı birçok konuda yazılmış olabilirler. Hem bilgi erişimini, hem de bilgi çıkarımını kolaylaştırmak için bu verileri sınıflandırmak önemli bir işlemdir. Bu aşamada temel metin madenciliği işlemleri uygulanarak doküman sınıflandırılması gerçekleştirilebilmektedir. Verilerin karmaşıklığı ve çeşitliliği göz önüne alındığı zaman, sadece metin madenciliği yöntemi ile doküman sınıflandırması yapılması çok iyi sonuçlar doğurmamaktadır. Bu aşamada diğer metotlardan da faydalanarak daha etkili ve doğru sonuçlar veren sınıflandırma yöntemleri geliştirilmiştir.

2.5. Metin Madenciliğinin Fonksiyonları

Metin madenciliğinin fonksiyonları, en karmaşığı en altta ve en basiti en yukarıda yer almak üzere Şekil 2.5'de gösterildiği gibidir [11].

Şekil 2.5. Metin Madenciliğinin Fonksiyonları

Şekil 2.5 incelendiğinde, doğal dil işlemenin en karmaşık metin madenciliği fonksiyonu olduğu, anahtar kelimenin ise en basit metin madenciliği fonksiyonu olduğu anlaşılmaktadır.

3. METİN SINIFLANDIRMA YÖNTEMLERİ

Metin sınıflandırma, akademik çalışmalarda popüler olan bir alandır [12]. Bu konuda gerçekleştirilen çalışmalar metinleri konu, yazar, stil ve türlerine göre sınıflandırmayı amaçlamaktadır. Metin sınıflandırma ile ilişkilendirilebilecek ilk çalışmalar yazar özelliği çıkarımında bulunan [13] ve üslup analizi gerçekleştiren [14] çalışmalar olarak belirtilebilir. Makine öğrenmesi ve doğal dil işleme tekniklerinin gelişmesi, metin sınıflandırma yöntemlerine yeni yaklaşımlar kazandırmıştır [15]. Craig tarafından diskriminant analizi ve çapraz entropi ile yazar ve üslup analizi [16], Ramsay tarafından karar ağacı ile yapılan tür analizi sınıflandırmaları [17] örnek gösterilebilir.

Metin sınıflandırma yöntemlerinin çoğalması, deneysel değerlendirmelerin öneminin artmasına yol açmıştır. Gerçekleştirilen bazı çalışmalarda, aynı veri setleri kullanılarak mevcut teknikler karşılaştırılmıştır [18-20]. Ancak bu çalışmalarda kullanılan veri setleri haberler ve web belgeleri ile sınırlı kalmıştır. Daha yaratıcı yazılar ya da edebi metinlere ait sınıflandırmalar gerçekleştirilmemiştir. Çalışmaların gelişmesi neticesinde asıl amaç, konu sınıflandırma olmuştur. Bu bağlamda dokümanlar konularına göre sınıflandırılmaya başlanmıştır.

Sınıflandırma yöntemlerini genel olarak denetimli ve denetimsiz sınıflandırma olarak ikiye ayrılmaktadır [18]. Denetimsiz sınıflandırmada eğitim verisi bulunmamakla birlikte sınıflandırılacak dokümanlar için küme ya da sınıf sayısı da belirli değildir. Belirlenen algoritma ile dokümanlar ya da metinler sınıflandırılır. Denetimli sınıflandırmada ise eğitim verileri mevcuttur. Ayrıca dokümanların sınıf ya da küme sayısı belirlidir. Bu eğitim verilerinden elde edilen bulgulara göre sınıflandırılmak istenen metinler ya da dokümanlar sınıflandırılır. Başta herhangi bir bilgi olmadığı için, denetimsiz sınıflandırmanın denetimli sınıflandırma kadar bir doğruluk oranı yakalaması beklenmemektedir [18].

En çok kullanılan metin sınıflandırma teknikleri Naive Bayes ve SVM'dir. Varolan çalışmalar incelendiğinde, SVM'in en başarılı metin sınıflandırma yöntemi olduğu görülmektedir [19, 20]. Naive Bayes basit ancak oldukça etkili bir öğrenme metodudur

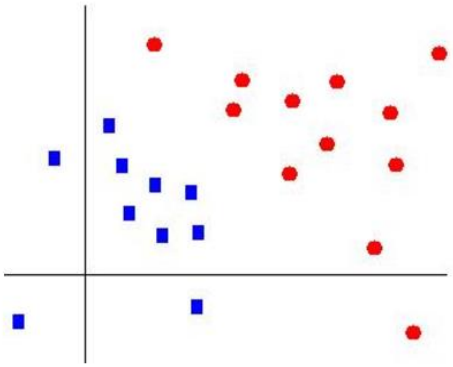
[21]. Ayrıca K-NN komşuluk yöntemi de metin sınıflandırma çalışmalarında kullanılmaktadır. Tezin bu bölümünün geriye kalan kısmı boyunca, K-NN yakın komşuluk, Naive Bayes ve SVM hakkında bilgi verilmiştir.

3.1. K-NN Yakın Komşuluk

K-NN(K-nearest neighborhood), yakın komşuluk kavramını dikkate alan, hem sınıflandırma hem de regresyon tahmini problemlerinde kullanılan bir yöntemdir. Geniş ölçüde, sınıflandırma problemlerinde tercih edilmektedir. Yorumlaması kolay ve hesaplama süresinin düşük olmasından dolayı yaygın olarak kullanılmaktadır.

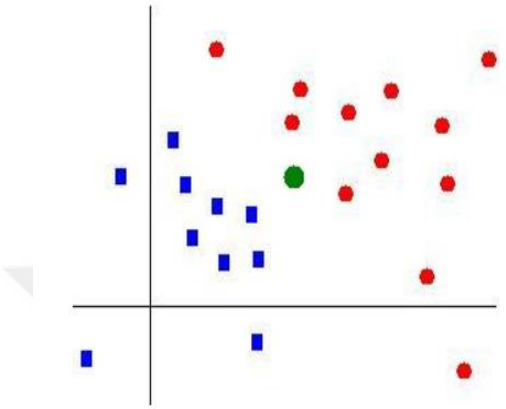
Bir vektör uzayında temsil edilen veriler incelendiğinde, birbirine yakın olan nesnelerin aynı kategoride sınıflandırılabileceği düşünülebilir. K-NN algoritmasının mantığı da bu düşünceye dayanmaktadır. Algoritmanın amacı, mevcut sınıflandırılmış verilerden faydalanarak, en yakın komşuluğa göre yeni bir veriyi sınıflandırmaktır. Algoritmanın temelinde veriler ikiye ayrılmaktadır. Bunlar sınama ve öğrenme örnekleri olarak adlandırılır. Daha önceden sınıfı belirli olan veriler öğrenme örneklerini temsil ederler.

K-NN algoritmasının çalışma mantığını aşağıda detaylı şekilde açıklanmıştır.



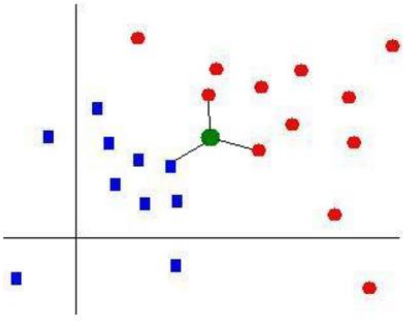
Şekil 3.1. K-NN Örneği-1

Şekil 3.1'de iki boyutlu koordinat sisteminde iki sınıfa gruplandırılmış öğrenme verileri temsil edilmektedir. K-NN algoritmasının amacı, sınıfını bilmediğimiz bir verinin bu düzlemdeki yerinden yola çıkarak sınıfını belirlemektir. K-NN algoritmasındaki k değeri, incelenecek komşu miktarını belirtmektedir. Şimdi bu algoritmanın nasıl çalıştığı incelenecektir.



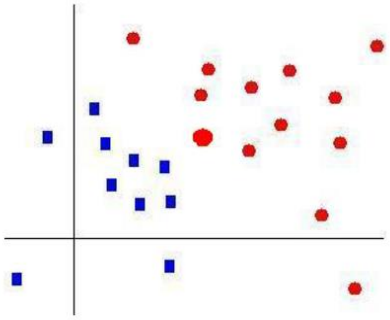
Şekil 3.2. K-NN Örneği-2

Şekil 3.2'de görüldüğü üzere, düzleme yeni eklenen yeşil noktanın sınıfı bilinmemektedir. K-NN algoritması için $k = 3$ alındığında sınama örneğine en yakın komşular Şekil 3.3'de görüldüğü gibi olmaktadır.



Şekil 3.3. K-NN Örneği-4

Görüldüğü üzere $k = 3$ alındığında en yakın üç komşu değer alınır ve incelenir. Bu en yakın üç komşudan çoğunluk olan kırmızı olduğu için, sınama örneğinin sınıfı da kırmızı olarak bulunmuş olur ve o sınıfta temsil edilir. Şekil 3.4'de görüldüğü üzere sınama örneği ilgili sınıfa eklenmiştir.



Şekil 3.4. K-NN Örneği-4

K-NN algoritmasında seçilen k değerine göre sonuç değişiklik gösterebilir. k değeri, eşitlik durumu oluşmaması için genellikle tek sayı olarak seçilir.

K-NN algoritması metin sınıflandırma alanında ilk olarak Masan ve arkadaşları tarafından kullanılmıştır [22]. Buradaki fikir, belirli bir sorgunun sınıfının doküman uzayında kendisine en yakın k miktarda belgenin kategorisine bakarak bulunabileceğidir.

3.2. Bayes Modelleme ve Naive Bayes Yöntemi

Bayes modelleme, bilgisayar bilimlerinde veri modelleme ve durum geçişi ifade etmek için kullanılan yöntemlerden birisidir. Literatürde bayes ağları olarak da geçen yöntem, istatistik tabanlı sınıflandırma yapmaktadır. Bayes modellemesi bayes teorisini kullanan istatistiksel bir sınıflandırıcıdır. Bayes yaklaşımında veriler

hakkında istatistiksel tahminler yürütülmektedir. Naive Bayes yöntemi, bayes yaklaşımının gerçekleştirilmiş bir örneğidir.

Naive Bayes yöntemi, oldukça pratik bir Bayes öğrenme metodudur. Hedef değer göz önüne alındığında, özellik değerlerinin koşullu olarak bağımsız olduğunu varsayar ve bu nedenle hesaplama maliyetini önemli ölçüde azaltır [23]. Basit fakat etkili bir yöntem olan Naive Bayes yöntemi, metin sınıflandırma uygulamalarında sıkça kullanılmaktadır [19].

Naive Bayes algoritması birçok farklı formülasyonlarla uygulanabilmektedir. Metin sınıflandırma alanında en çok kullanılan iki Naive Bayes yöntemi ise Çok Değişkenli Bernoulli Modeli ve Çok Terimli Model'dir [24]. Çok Değişkenli Bernoulli Modeli özellik değerleri olarak kelime varlığını ya da yokluğunu kullanmaktadır. Çok Terimli Model ise özellik değeri olarak kelime frekansını kullanmaktadır. Geçmiş çalışmalar incelendiğinde az miktarda kelime içeren veri setleri için Çok Değişkenli Bernoulli Modelinin, daha çok miktarda kelime içeren veri setlerinde ise Çok Terimli Modelin metin sınıflandırma uygulamalarında daha iyi sonuçlar verdiği gözlemlenmiştir.

3.3. Destek Vektör Makineleri

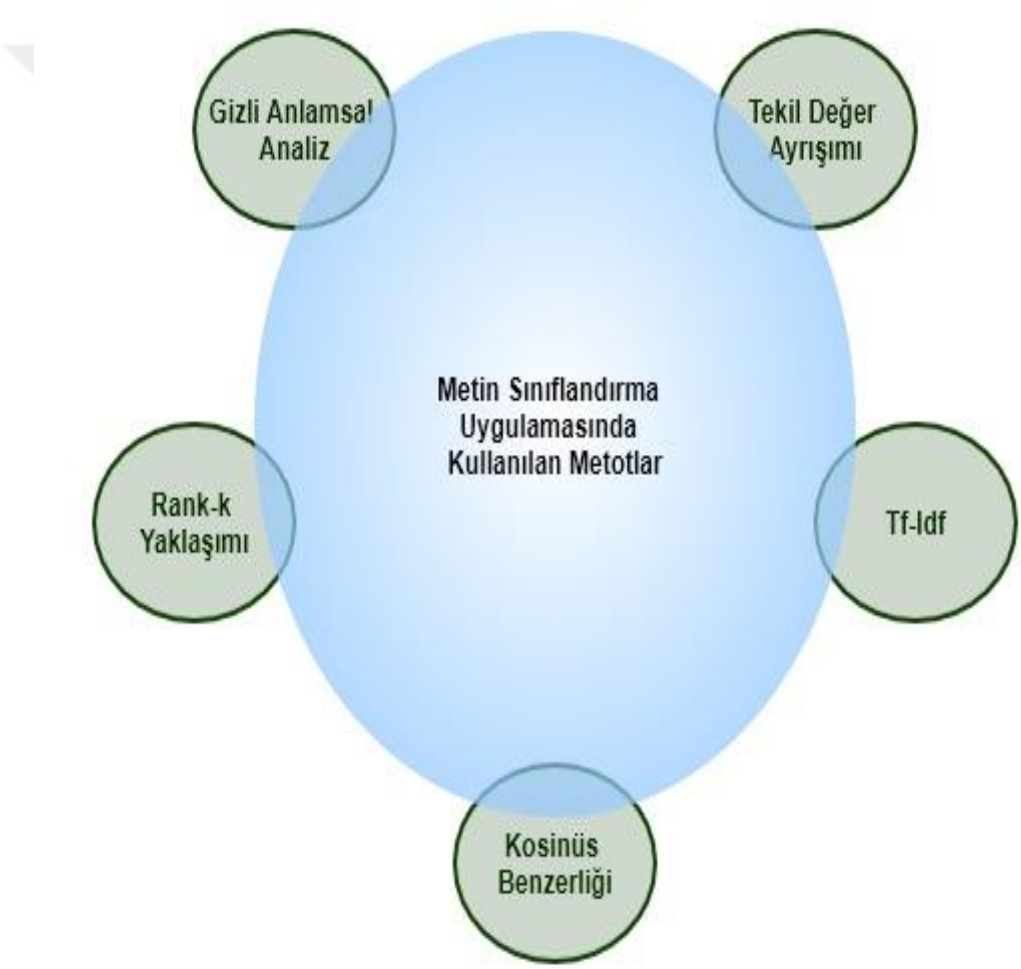
Destek Vektör Makineleri(SVM), istatistiksel öğrenme kuramından yapısal risk azaltma ilkesine dayalı olarak geliştirilen denetlemeli öğrenme yöntemidir [25]. Doğrusal sınıflandırıcılar olarak SVM, iki karar sınırları arasındaki maksimum marjlarla veri noktalarını ayıran hiperdüzlemleri bulmayı hedeflemektedir. Ayrıca SVM, genelleme hatasını en aza indirgemeyi amaçlar. Böylece gereğinden fazla uyma olasılığını azaltma avantajına sahiptir. SVM, metin sınıflandırma alanında çokça tercih edilen bir yöntemdir. Karşılaştırmalı çalışmalar incelendiğinde, SVM'in diğer metotlardan daha iyi performans gösterdiği görülmektedir [19].

SVM algoritması, terim frekansı ölçümünde birçok varyasyona izin vermektedir. Bunlar sırası ile 'svm-bool', 'svm-tf', 'svm-ntf' ve 'svm-tfidf''dir. 'svm-bool' özellik değeri olarak kelimelerin varlığını ya da yokluğunu kullanır. 'svm-tf' ise özellik değeri

olarak kelime frekansını kullanmaktadır. ‘svm-ntf’ ise özellik değeri olarak normalize edilmiş kelime frekansını kullanır. Son olarak ‘svm-tfidf’ ise ters doküman frekansı ile ağırlıklandırılmış terim frekansını kullanır.

4. KULLANILAN METOTLAR

Tezin üçüncü bölümünde, metin sınıflandırma alanında kullanılan temel yöntemler hakkında bilgi verilmiştir. Bu bölümde ise, tez çalışmasında kullanılan yöntemler hakkında bilgi verilecektir. Tez kapsamında metin sınıflandırma yöntemi olarak kullanılan temel metot LSA'dır. Kullanılan tüm metotlar şekil 4.1'de gösterilmiştir.



Şekil 4.1. Kullanılan Metotlar

4.1. Gizli Anlamsal Analiz

LSA(Latent Semantic Analysis), bilgisayar modellemesinde ve metinlerin analizinde kullanılan matematiksel bir yöntemdir. Birçok farklı alanda bu matematiksel metottan faydalanılmaktadır. Bilgi çıkarımı, örüntü tanıma ve sınıflandırma problemlerinde sıkça kullanılmaktadır. Gizli Anlamsal İndeksleme(LSI) olarak da bilinmektedir. LSA'da temel düşünce, kelimelerin birbirlerine olan benzerliğini gösteren bir yapı oluşturmaktır [26].

LSA bize bir dokümanın vektör temsilini oluşturma olanağı sunmaktadır. Sahip olduğumuz bu vektör uzayı, bize doküman yığını içinde bulunan dokümanlar arasında karşılaştırma yapma imkânı sunmaktadır. Yani, vektörler arasındaki mesafeyi ölçerek vektörler arasındaki benzerlik ilişkisini tanımlayabiliriz. Bu sayede dokümanları konularına göre sınıflandırabiliriz.

Diğer bir deyişle, verilerin saklandığı doküman yığını içerisinde bazı gizli ilişkiler mevcut olabilir. LSA, bu ilişkileri tespit etmek için kullanılan yöntemlerden biridir. Daha çok metinsel veriler üzerinde kullanılan LSA ile bu metinsel veriler içerisindeki gizli bağlantılar ortaya çıkarmayı hedefler. Esasında LSA metinler arası böyle bir gizli ilişkinin varlığını kabul eder. LSA yönteminden faydalanabilmek için, yapılandırılmamış verilerimizi yapılandırılmış biçime getirmemiz gerekmektedir. LSA sözdizimsel ve gramer yapısı temizlenen her doküman yığınına uygulanabilir [27]. Bunun için doküman yığınındaki veriler $m \times n$ boyutlu terim-doküman matrisine dönüştürülmelidir. Terim-doküman matrisi elde edildikten sonra, bu veri grubu içerisindeki gizli anlamsal yapıların var olup olmadığını kontrol etmek için rank-k yaklaşımı gerçekleştirilir. Rank-k yaklaşımı uygulanırken uyulması gereken bazı zorunluluklar vardır. Bu aşamada $k \ll \min(m,n)$ koşulu yerine getirilmelidir. Rank-k yaklaşımı hesaplanırken faydalanılan temel yöntem Tekil Değer Ayrışımı -Singular Value Decomposition(SVD)'dır. Terim-doküman matrisindeki en büyük k tekil değeri ve ona karşılık gelen sağ ve sol tekil vektörler kullanılarak doküman ve terimler düşük ranklı olarak temsil edilir. Bu işlemin ardından sorgu vektörü oluşturulur ve k vektör uzayında düşük ranklı temsilcisi elde edilir. Ardından bu dokümanları temsil eden vektörlerle benzerlik ilişkisi incelenir.

4.2. Tekil Değer Ayrışımı

Bir matrisi ortogonal çarpanlarına ayırmak için kullanılan temel yöntem SVD'dir. Google'ın geliştirdiği PageRank algoritmasında, veri sıkıştırma tekniklerinde, insan yüzü modellemede, gen analizinde ve bilgi çıkarımı gibi birçok değişik alanda SVD'den faydalanılmaktadır. $m \geq n$ olmak üzere verilen $m \times n$ boyutlu ve r ranklı terim-doküman matrisi A'nın tekil değer ayrışımı

$$A = USV^T \tag{4.1}$$

biçimindedir. Burada U ve V matrisleri ortogonal matrislerdir. U matrisi $m \times m$ boyutundadır ve sütunları sol tekil değer vektörleri olarak da adlandırılır. V^T matrisi ise $n \times n$ boyutundadır ve sütunları sağ tekil değer vektörleri olarak anılırlar. S matrisi ise A matrisi gibi $m \times n$ boyutunda köşegen matristir. Yani, $S = \text{diag}(\sigma_1, \sigma_2, \cdots, \sigma_n)$ biçiminde ifade edilir ve köşegen elemanları A'nın tekil değerleri olarak isimlendirilir. Bu tekil değerler

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > \sigma_{r+1} = \cdots = \sigma_n = 0 \tag{4.2}$$

eşitsizliğini sağlarlar. Eğer $\delta > 0$ eşik değeri alınırsa

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_k > \delta \geq \sigma_{k+1} \geq \cdots \geq \sigma_r > \sigma_{r+1} = \cdots = \sigma_n = 0 \tag{4.3}$$

k değeri sayısal rank anılır. Yukarıdaki eşitsizliğin tatmin edici olması için σ_k ve σ_{k+1} arasında anlamlı bir boşluk olması gerekir [28].

A matrisinin tekil değerlerini hesaplamak için AA^T ve $A^T A$ matrislerinin öz değerlerini ve öz vektörlerini bulmamız gerekmektedir. $A^T A$'nın öz vektörleri V matrisinin sütunlarını oluşturur. AA^T'nın öz vektörleri ise U matrisinin sütunlarını

oluşturur. S matrisinde temsil edilen tekil değerler, AA^T ve A^TA'nın öz değerlerinin pozitif kareköküdür. Tekil değerler, S matrisinin diyagonal girdileri olup azalan düzende düzenlenmişlerdir.

Kelime kullanımına bağlı olarak oluşturulacak $m \times n$ boyutlu matrisimizin boyutu da değişiklik gösterecektir. Genellikle bu boyutun büyük boyutlarda olduğu gözlemlenmiştir. Bu matrisin tekil değer ayrışımının hesaplama karmaşıklığı $O\left(m^2 n\right)$ dir.

4.3. Terim Frekansı Ağırlıklandırma Yöntemi

Terimleri matrislerde temsil ederken, ağırlıklarını belirlemek için kullanılan yöntemlerden bir tanesi de Terim Frekansı Ağırlıklandırma Yöntemidir. Genel olarak Terim Frekansı (term frequency) olarak adlandırılmaktadır. Terim Frekansı yönteminde her bir terim, doküman yığınındaki bir kelimeyi temsil etmektedir. Ağırlıklandırma yapılırken, her bir terimin dokümanlardaki sıklığı göz önüne alınır. Yani her bir terimin ilgili sınıftaki frekansı alınarak terim sınıf vektöründeki değeri olarak atanır. Diğer ağırlıklandırma yöntemleri ile karşılaştırıldığında, terim frekans ağırlıklandırma yönteminin daha basit bir yöntem olduğu görülmektedir. Bu özelliğinden dolayı işlem kolaylığı sağlaması için terim frekansı yöntemi tercih edilmiştir.

4.4. Ters Doküman Frekansı Ağırlıklandırma Yöntemi

Ters Doküman Frekansı Ağırlıklandırma, bir doküman yığınında herhangi bir terimin önemini ölçmek için kullanılan istatistiksel bir yöntemdir. Genel olarak Ters Doküman Frekansı (Inverse Document Frequency, IDF) olarak adlandırılmaktadır. Bu yöntemde, her bir terimin diğer dokümanlarda var olup olmadığı incelenir. Terimler, bulundukları doküman sayısı ile orantılı olacak şekilde ağırlıklandırılırlar. Terimin bulunduğu doküman sayısı, bize o terimin fark yaratan terim olduğunu göstermektedir.

Bununla orantılı olarak, bir terim ne kadar az dokümanda temsil edilmişse sınıflandırma için bize o kadar yardımcı olmaktadır. Aksi şekilde, çok fazla farklı sınıfa ait dokümanlarda temsil edilen terimler sınıflandırma için çok belirleyici olmamaktadır.

4.5. Rank-k Yaklaşımı

Doküman yığını incelendiğinde, mevcut terimlerin tüm dokümanlarda bulunmadığı gözlemlenmektedir. Bundan dolayı terim-doküman matrisi seyrek bir matristir. Mevcut olan terim doküman matrisimizin büyüklüğü, üzerinde işlem yapılmasını zorlaştırmaktadır. Terim-doküman matrisinin çok büyük olması, işlem yükünü artırmaktadır. Bununla birlikte, anlamsal yapıda katkısı bulunmayan kelimeleri de barındırmaktadır. Dokümanlar arasında sınıflandırma işlemi gerçekleştirilirken, farklılık yaratacak kelimeleri tespit etmemiz ve bize faydası olmayan kelimeleri de önemsememiz gerekmektedir. Bu sebeplerden dolayı, bir boyut düşürme işlemi olan rank-k yaklaşımı uygulanmıştır. Matris ayrışımı uygulandıktan sonra rank-k yaklaşımı gerçekleştirilir. Rank-k yaklaşımı sayesinde gizli anlamsal yapıyı bozan ve gürültü olarak adlandırılan kısım yok edilir [27].

4.6. Kosinüs Benzerliği

Vektör uzayında temsil edilen dokümanlar arasında karşılaştırma yapabilmek için kullanılan tekniklerden bir tanesi kosinüs benzerliği(Cosine Similarity)'dir. Kosinüs benzerliği ile farklı dokümanlar arasındaki benzerliğin trigonometrideki kosinüs fonksiyonu kullanılarak tespit edilmesi amaçlanmaktadır. Kosinüs benzerliği metodu sınıflandırma ve dokümanlar arasındaki benzerliği bulmada en çok kullanılan benzerlik ölçüm metodudur [29]. İki doküman arasındaki kosinüs benzerlik ölçümü aşağıdaki formül ile hesaplanmaktadır.

$$SIM_C\left(D_1, D_2\right) = \frac{D_1 . D_2}{\left\|D_1\right\|_2 . \left\|D_2\right\|_2} \tag{4.4}$$

Kosinüs benzerliği teoreminde bir eşik değeri kullanılmaktadır. Bu eşik değeri, kosinüs benzerlik değerinden büyük ise iki dokümanın birbiri ile benzer olduğunu söyleyemeyiz. Ancak bu eşik değeri kosinüs benzerlik değerinden küçük olursa, iki dokümanın birbirine benzer olduğunu söyleyebiliriz [29].

Gerçekleştirdiğimiz uygulamada, test verilerinin sınıflandırılması yapılırken kosinüs benzerliği yöntemi kullanılmıştır. Test dokümanlarının, eğitim verilerine göre alınan kosinüs benzerliklerine göre hangi konuya ait oldukları belirlenmiştir.

5. LSA İLE DOKÜMAN SINIFLANDIRMA

Bu bölüm, tez kapsamında geliştirilen sınıflandırma uygulamasına ayrılmıştır. Uygulamanın çalışmasını anlatan akış şeması Şekil 5.1'de gösterilmiştir.

Şekil 5.1. Akış Şeması

Doküman sınıflandırma işlemini gerçekleştirebilmek için, ilk olarak temel metin madenciliği adımlarının gerçekleştirilmesi gerekmektedir. Metin madenciliğinin dokümanların toplanması ile başladığına değinilmişti. Mevcut olan derlemler ve veri setleri incelendikten sonra, Reuters21578 veri setinin kullanımına karar verilmiştir. 116 farklı sınıfta metin içeren Reuters veri seti, bu verilerin hem 'training' hem de 'test' verilerini kullanıcıya sunmaktadır. Uygulamada kullanmak üzere bu veri setleri arasında seçim yapılmış ve dört adet sınıfa ait metinler 'training' verisi olarak alınmıştır. Bu sınıflar sırası ile 'coffee', 'wheat', 'ship' ve 'gold' dur. Bu sınıflara ait veri setinde bulunan 'training' ve 'test' metin sayısı aşağıda belirtildiği gibidir.

Çizelge 5.1. Eğitim Ve Test Verisi Sayıları

Sınıf	**Training Veri Sayısı**	**Test Veri Sayısı**
Coffee	112	28
Wheat	212	71
Ship	197	90
Gold	94	31

Bu veriler üzerinde önişleme başlamadan önce, tüm verilerin aynı standart bir hale getirilmesi gerekmektedir. Bu bağlamda tüm veriler '.txt' formatına dönüştürülerek hepsinin aynı standartta olduğu teyit edilmiştir.

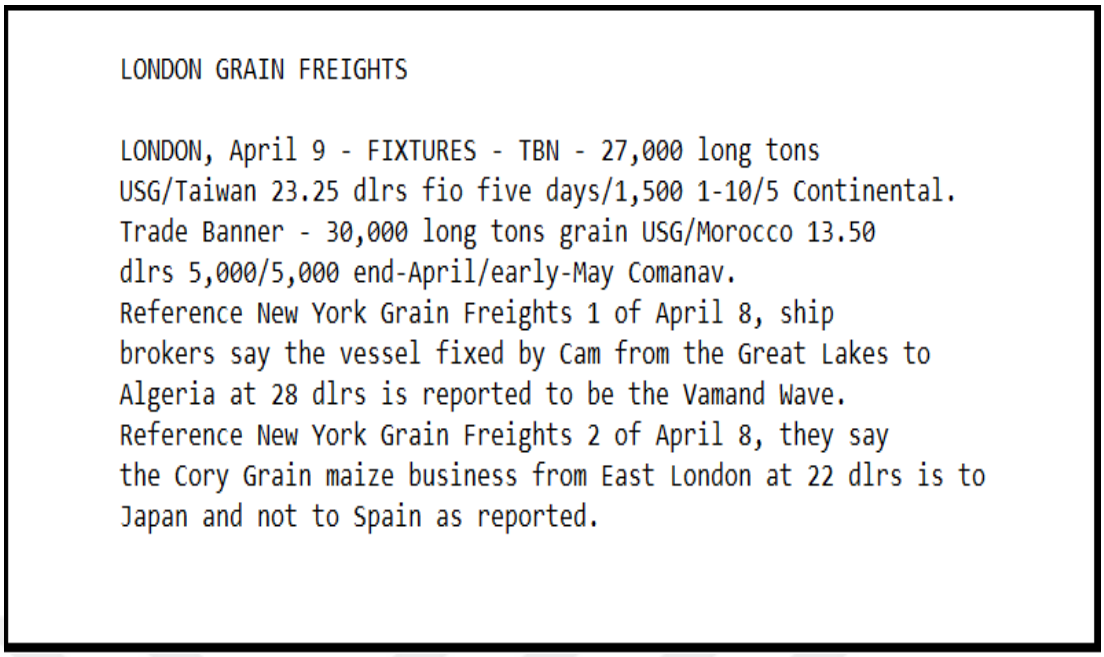
Aşağıda, dört veri sınıfına ait eğitim verilerden birer adet örnek gösterilmiştir.

```
COLOMBIA BUSINESS ASKED TO DIVERSIFY FROM COFFEE

BOGOTA, April 8 - A Colombia government trade official has
urged the business community to aggressively diversify its
activities and stop relying so heavily on coffee.
Samuel Alberto Yohai, director of the Foreign Trade
Institute, INCOMEX, said private businessmen should not become
what he called "mental hostages" to coffee, traditionally
Colombia's major export.
The National Planning Department forecast that in 1987
coffee will account for only one-third of total exports, or
about 1.5 billion dlrs, with oil and energy products making up
another third and non-traditional exports the remainder
```

Şekil 5.2. Coffee Sınıfına Ait Eğitim Verisi

```
CCC ACCEPTS EXPORT BID FOR WHEAT FLOUR TO IRAQ

WASHINGTON, April 8 - The Commodity Credit Corporation has
accepted a bid for an export bonus to cover a sale of 12,500
tonnes of wheat flour to Iraq, the U.S. AGriculture Department
said.
The bonus awarded was 113.0 dlrs per tonne and will be paid
to Peavey Company in the form of commodities from CCC stocks.
The wheat flour is for delivery May 15-June 15, 1987, the
department said.
An additional 162,500 tonnes of wheat flour are still
available to Iraq under the Export Enhancement Program
initiative announced January 7, 1987, USDA said.
```

Şekil 5.3. Wheat Sınıfına Ait Eğitim Verisi



Şekil 5.4. Ship Sınıfına Ait Eğitim Verisi

Şekil 5.5. Gold Sınıfına Ait Eğitim Verisi

5.1. Önişlem

Örnek verilerden de görüldüğü üzere, dört sınıfa ait veri seçilmiş ve standart hale getirilmiştir. Bu veriler üzerinde sınıflandırma çalışması yapabilmemiz için, metinleri önişlemlerden geçirerek en sade hallerine dönüştürerek kelimelere ayırmalıyız. Bunun için sırasıyla bazı önişlemler uygulanmıştır. Bu adımlar noktalama işaretlerini kaldırma, metinleri belirteçlere ayırma, etkisiz kelimelerin tespit edilerek filtrelenmesi ve son olarak da gövdeleme işlemleridir. Şimdi gerçekleştirilen bu önişlemleri detaylandıralım.

5.1.1. Noktalama İşaretlerinin Kaldırılması

Önişleme alınan metne uygulanan ilk işlem noktalama işaretlerinin temizlemesidir. Bu aşamada tüm noktalama işaretleri ve sayısal değerler tespit edilerek kaldırılmıştır. Ayrıca tüm harfler küçük harf olacak şekilde dönüştürülmüştür. Böylece her kelimenin aynı standartta gösterilmesi sağlanmıştır. Noktalama işaretlerinin kaldırılması, metindeki cümle bütünlüğünü bozabilir, ancak uygulamamızda kelime tabanlı belirteçleme ya da işaretleme yapacağımız için herhangi bir kayıp olmayacaktır.

```
ico producers to present new coffee proposal

london feb   international coffee organization ico
producing countries will present a proposal for reintroducing
export quotas for  months from april  with a firm
undertaking to try to negotiate up to september  any future
quota distribution on a new basis ico delegates said
distribution from april  would be on an unchanged basis as
in an earlier producer proposal which includes shortfall
redistributions totalling  mln bags they said
resumption of an ico contact group meeting with consumers
scheduled for this evening has been postponed until tomorrow
delegates said
```

Şekil 5.6. Noktalama İşaretlerinden Temizlenmiş Doküman

Görüldüğü üzere metin içerisindeki kelimelerin harfleri küçük harf olarak düzenlenmiş ve noktalama işaretleri ile tüm rakamlar metinden temizlenmiştir.

5.1.2. İşaretleme

Metindeki tüm kelimeleri aynı standartta temsil ettikten sonra yapılacak ilk işlem, karakter akışını işaretlere ya da belirteçlere bölmektir. Daha önceden paragraf, cümle ya da kelime tabanlı işaretleme yapılabileceğinden bahsetmiştik. Uygulamanın işaretleme bölümünde, kelime tabanlı işaretleme tercih edilmiştir. Noktalama işaretlerinden ve rakamlardan temizlenen yapılandırılmamış veri, kelimelerine ayrılmıştır. İşaretleme işleminin tamamlanmasının ardından yapılandırılmamış veri kaynağının kelimelerinden oluşan bir kelime dizisi elde edilmiştir.

```
['ico', 'producers', 'to', 'present', 'new', 'coffee', 'proposal', 'london',
'feb', 'international', 'coffee', 'organization', 'ico', 'producing', 'countries',
'will', 'present', 'a', 'proposal', 'for', 'reintroducing', 'export', 'quotas',
'for', 'months', 'from', 'april', 'with', 'a', 'firm', 'undertaking', 'to', 'try',
'to', 'negotiate', 'up', 'to', 'september', 'any', 'future', 'quota', 'distribution',
'on', 'a', 'new', 'basis', 'ico', 'delegates', 'said', 'distribution', 'from', 'april',
'would', 'be', 'on', 'an', 'unchanged', 'basis', 'as', 'in', 'an', 'earlier', 'producer',
'proposal', 'which', 'includes', 'shortfall', 'redistributions', 'totalling', 'mln',
'bags', 'they', 'said', 'resumption', 'of', 'an', 'ico', 'contact', 'group', 'meeting',
'with', 'consumers', 'scheduled', 'for', 'this', 'evening', 'has', 'been', 'postponed',
'until', 'tomorrow', 'delegates', 'said']
```

Şekil 5.7. İşaretleme Sonucu Elde Edilen Kelime Dizisi

Şekil 5.7'de kelime tabanlı işaretleme sonucu elde edilen kelime dizisi görülmektedir.

5.1.3. Etkisiz Kelimelerin Kaldırılması

Belirteçlere ayrılma işlemi tamamlandıktan sonra, bir kelime dizisi elde edilmiştir. Bu kelimeler üzerinde sırası ile bazı temizleme işlemlerinin uygulanması gerekmektedir.

İlk olarak etkisiz kelimeler bu kelime dizisinden çıkarılmıştır. Etkisiz kelimeler, cümleye hiçbir anlam katmayan, edat, bağlaç vb. kelimelerden oluşan ve ‘stopwords’ olarak adlandırılan kelimeler grubudur. Bu temizleme işlemini yapmadan önce, ‘etkisiz kelimeler’ listesi oluşturulmuştur. Oluşturulmuş olan ‘etkisiz kelimeler’ listesindeki bazı örnek kelimeler aşağıdaki tabloda verilmiştir.

Çizelge 5.2. Etkisiz Kelime Listesi

a	beforehand	getting	instead	later
able	begin	give	into	latter
about	beginning	given	invention	latterly
above	beginnings	gives	inward	least
abst	begins	giving	is	less

Çizelge 5.2’de görüldüğü üzere, bu kelimelerin anlam üzerinde etkisi olmayacağı için kelime dizisinden ayrılması gerekmektedir. Böylece metin madenciliği işlemleri daha kolay gerçekleştirilebilir. Bu çalışmada kullanılan ‘etkisiz kelimeler’ listesi toplamda 670 adet etkisiz kelime içermektedir. Belirteçlere ayırma işlemi sonucu elde edilen kelime dizisindeki kelimeleri inceleyerek, rastlanılan tüm etkisiz kelimeler kelime dizisinden çıkarılarak kelime dizisi daraltılmış olunur.

```
['ico', 'producers', 'coffee', 'proposal', 'london', 'feb', 'international',
'coffee', 'organization', 'ico', 'producing', 'countries', 'proposal',
'reintroducing', 'export', 'quotas', 'months', 'april', 'firm', 'undertaking',
'negotiate', 'september', 'future', 'quota', 'distribution', 'basis', 'ico',
'delegates', 'distribution', 'april', 'unchanged', 'basis', 'earlier', 'producer',
'proposal', 'includes', 'shortfall', 'redistributions', 'totalling', 'bags',
'resumption', 'ico', 'contact', 'group', 'meeting', 'consumers', 'scheduled',
'evening', 'postponed', 'tomorrow', 'delegates']
```

Şekil 5.8. Etkisiz Kelimelerin Kaldırılması Sonucu Elde Edilen Kelime Dizisi

Şekil 5.8'de etkisiz kelimelerin, kelime dizisinden çıkarıldıktan sonra kelime dizisinin son hali görülmektedir. Kelime dizisi ilk başta 93 kelimeden oluşmaktaydı. Etkisiz kelimelerin diziden çıkarılması sonucu, dizideki kelime adedinin 52'ye düştüğü görülmüştür.

5.1.4. Gövdeleme

Etkisiz kelimeleri, kelime dizisinden temizledikten sonra gövdeleme işlemi kelime dizisindeki bütün kelimelere uygulanmıştır. Bu aşamada, kelime dizimizde bulunan tüm kelimeler en sade haline getirilmiştir. Kelimeler her zaman kök halinde kullanılmamaktadır. Çeşitli yapım ekleri ve çekim ekleri alarak farklı yapılarda da kullanılabilmektedir. Bu durum, kelimeleri farklı göstererek metin madenciliğinde bize engel olarak karşımıza çıkmaktadır. Bu sebeple, kelimeleri kök haline getirmek bizim için zorunluluk arz etmektedir. Kelimelerin kök hali bulunurken, çeşitli algoritmalardan yararlanılmıştır. Mevcut algoritmalar karşılaştırıldığında, en optimum sonuca Porter Stemmer ile ulaşıldığı tespit edilmiştir. Çalışmamızda gövdeleme işlemini gerçekleştirirken Porter Stemmer kütüphanesinden faydalanılmıştır. Porter Stemmer kütüphanesi ile kelime dizimizde bulunan tüm kelimelerin kök hali bulunarak, kelime dizimiz en yalın hale getirilmiştir.

Porter Stemmer algoritması test edildiğinde elde edilen sonuçlar tabloda gösterilmiştir.

Çizelge 5.3. Gövdeleme Örneği

Kelime	**Bulunan Kök Hali**
Coffee	coffe
Producers	produc
Proposal	propos
Exporter	export
Meeting	meet

Tablo bize Porter Stemmer'ın bazı kelimelerde bulmuş olduğu kökleri göstermektedir. Porter Stemmer'ın her kelime için tam doğru sonuç bulamadığını söyleyebiliriz. Ancak aynı kelimeler için, aynı sonuçları bize döndüreceğinden dolayı ilerde yapacağımız işlemlerde herhangi bir olumsuz sonuca neden olmayacaktır.

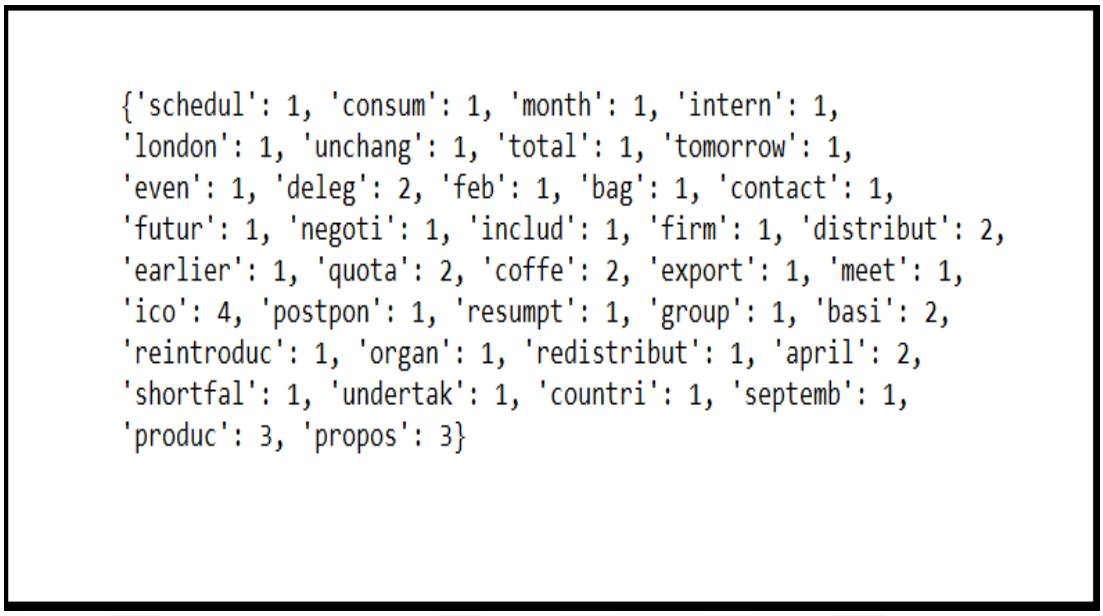
Şekil 5.9'de Porter Stemmer algoritmasının kelime dizisine uygulanmasının ardından elde edilen sonuçlar görülmektedir.

```
['ico', 'produc', 'coffe', 'propos', 'london', 'feb',
'intern', 'coffe', 'organ', 'ico', 'produc', 'countri',
'propos', 'reintroduc', 'export', 'quota', 'month',
'april', 'firm', 'undertak', 'negoti', 'septemb', 'futur',
'quota', 'distribut', 'basi', 'ico', 'deleg', 'distribut',
'april', 'unchang', 'basi', 'earlier', 'produc', 'propos',
'includ', 'shortfal', 'redistribut', 'total', 'bag',
'resumpt', 'ico', 'contact', 'group', 'meet', 'consum',
'schedul', 'even', 'postpon', 'tomorrow', 'deleg']
```

Şekil 5.9. Gövdeleme Sonucu Elde Edilen Kelime Dizisi

5.2. Veri Tabanı İşlemleri

Kelime dizimizdeki kelimeleri en sade haline getirdikten sonra, dizideki kelimeleri kontrol edilerek frekanslarının hesaplanması aşaması gelmektedir. Örneğin, 'elma' kelimesi dizimizde 5 adet geçmişse, bu değer bu kelimenin frekansı olarak kaydedilmektedir. Bu işlem gerçekleştirilirken Python'da bulunan 'dictionary' yapısı kullanılmıştır. Kelimeleri 'dictionary' yapısında saklarken, 'key' bölümüne kelimeyi, 'value' bölümüne ise frekans kaydedilmiştir.



Şekil 5.10. Sözlük Yapısı

Şekil 5.10'da sözlük yapısı görülmektedir. Her bir kelimenin, ilgili dokümanda kaç adet geçtiğini bize göstermektedir. Bu sözlük yapısı, veri tabanı işlemlerinden bize kolaylık sağlamaktadır. Ayrıca ileride terim frekansı hesaplamasında da avantaj sağlayacaktır.

Kelime dizimiz üzerindeki işlemleri bu şekilde tamamlayarak önişlem süreci tamamlanmıştır. Bu aşamadan itibaren veri tabanı işlemlerine geçilmiştir. Veri tabanı kabaca üç adet tablodan oluşmaktadır. Bunlar sırası ile 'class', 'term' ve 'term-class' tablolarıdır.

'Class' tablosunda sınıf verileri tutulmaktadır. Sırası ile 'class_id' ve 'class' verilerini içerir. Bu sınıflar, metin sınıflandırmasında ayıracağımız sınıfları içermektedir. Bu sınıflar 'coffee', 'wheat', 'ship' ve 'gold' sınıflarıdır. Projede kullanılan 'class' tablosu örneği Çizelge 5.4'deki gibidir.

Çizelge 5.4. Class Tablosu

id_class	**class**
1	Coffee
2	Wheat
3	Gold
4	Ship

‘Term’ tablosunda terim verileri tutulmaktadır. Sırası ile ‘id_Term’, ‘term’, ‘frequency’ ve ‘total’ alanlarını içerir. ‘id_Term’ terimlerin ‘id’ numarasını göstermektedir. ‘term’ ise kelimeyi belirtir. ‘frequency’ sütunu, o terimin tüm metinlerde geçen toplam miktarını gösterir. ‘total’ ise o terimin kaç farklı dokümanda geçtiğini göstermektedir. Örnek ‘term’ tablosu Çizelge 5.5’deki gibidir.

Çizelge 5.5. Term Tablosu

id_Term	**term**	**frequency**	**total**
1	schedul	32	27
2	consum	96	40
3	month	179	114
4	intern	142	105
5	london	134	90
6	unchang	23	14
7	total	145	91
8	import	126	55
9	splinter	6	4
10	modifi	2	2
11	discuss	52	41
12	present	8	8
13	formal	6	6
14	inform	4	3

‘term-class’ tablosu ise terimler ile sınıflar arasındaki ilişkiyi göstermektedir. Sırası ile ‘id’, ‘term_id’, ‘class_id’ ve ‘term_class_frequency’ sütunlarından oluşur. Buradaki ‘term_class_frequency’ o terimin ilgili sınıftaki frekansını, yani bulunma miktarını temsil etmektedir. ‘term-class’ tablosu örneği Çizelge 5.6’da gösterilmiştir.

Çizelge 5.6. Term-Class Tablosu

id	**term_id**	**class_id**	**term_class_frequency**
1	1	1	43
2	2	2	104
3	3	3	101
4	4	1	203
5	5	1	142
6	6	2	22
7	7	5	38
8	8	4	23
9	9	3	4
10	10	2	87
11	11	1	3
12	12	2	81
13	13	3	22
14	14	4	99
15	15	1	114
16	16	2	38
17	17	3	47
18	18	1	27
19	19	2	67
20	20	3	159
21	21	1	289
22	22	2	201
23	23	1	135

Doküman sınıflandırma çalışmamızda, önişlem ve veri tabanı işlemleri paralel şekilde yürütülmüştür. Çalışmada kullanılan eğitim verileri ön işlemden geçirildikten sonra veri tabanına kaydedilmiştir. Bu işlem, tüm eğitim veri seti için sırasıyla gerçekleştirilmiştir. Çalışmamızda kullandığımız dört sınıfa ait eğitim veri setleri için bu önişlem ve veri tabanına ekleme işlemleri gerçekleştirilerek veri tabanımız son halini almıştır. Böylece veri tabanımız, doküman sınıflandırmada kullanılabilecek hale gelmiştir.

5.3. Terim-Sınıf Matrisinin Oluşturulması

LSA, girdi olarak terim-doküman matrisi kabul etmektedir. Bu uygulamada, terim-doküman matrisine benzer olacak şekilde terim-sınıf matrisi oluşturulmuştur. Veri setinde kullandığımız benzer gruptaki dokümanlar, belirli bir sınıfı temsil etmektedirler. Amacımız, bu sınıfa benzer diğer dokümanları sınıflandırmak olduğu için, terim-doküman matrisi yerine terim-sınıf matrisi oluşturmamız daha doğru olacaktır.

Terim-sınıf matrisini,

$$A = \left[a_{ij} \right] \tag{5.1}$$

şeklinde tanımlarsak a_{ij} değeri i'nci terimin j'inci sınıftaki ağırlığını göstermektedir. Terim-sınıf matrisimizin özelliği, mevcut terim ve sınıf miktarına göre değişiklik gösterir. Dokümanlarımız m adet terim ve n adet sınıf içeriyorsa, elde edeceğimiz terim-sınıf matrisimiz $m \times n$ boyutlu olacaktır. Her terim, her sınıfta temsil edilemeyebilir. Bu sebepten dolayı terim-sınıf matrisimizi seyrek bir matris olarak niteleyebiliriz [30]. Oluşturduğumuz $m \times n$ boyutlu terim-sınıf matrisinin bir benzeri aşağıdaki tabloda gösterildiği gibidir.

Çizelge 5.7. Terim-Sınıf Matrisi

Term	**Coffee_Class**	**Wheat_Class**	**Ship_Class**	**Gold_Class**
Elma	3	4	5	0
Kalem	2	0	2	5
Armut	3	8	1	4
Kitap	4	12	13	11
Okul	5	15	0	14

Tabloda görüldüğü üzere, sınıf sütunları terimlerin her bir sınıf içinde hangi miktarda bulunduğunu temsil etmektedir. Terim-sınıf matrisi oluşturulurken, terim frekansı ağırlıklandırma yöntemi kullanılmıştır.

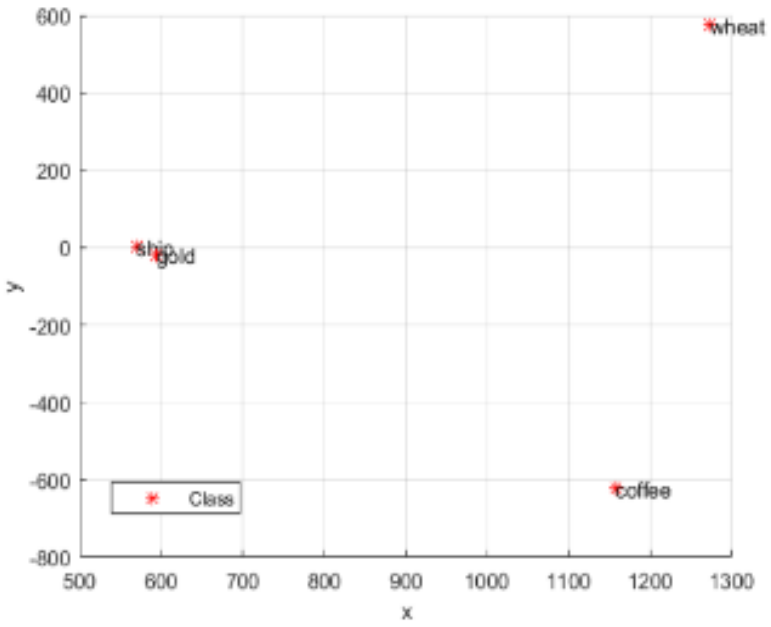
5.4. Matris Ayrışımının Uygulanması

Terim-sınıf matrisinde her birim terimin ağırlığı belirlendikten sonra, elde edilen A terim-sınıf matrisine matris ayrışımı uygulanır. Bu çalışmada, SVD matris ayrışımı kullanılmıştır. SVD, bir matris ayrışımında kullanılan başarılı bir yaklaşımdır ve metin madenciliğinde oldukça kullanılan bir yöntemdir. Uygulayacağımız tekil değer ayrışımı, LSA'nın ilk aşamasını oluşturmaktadır. İlk aşamada terim-sınıf matrisinin tekil değerleri hesaplanır.

Uygulamanın bu aşamasında, $m \times n$ boyutlu A terim-sınıf matrisine SVD uygulanmıştır. SVD sonucu, yine A matrisimizle aynı boyutta olan S matrisi elde edilir. Daha önce A matrisinin çok büyük boyutlu olduğu belirtilmişti. SVD sonucu elde edilen bu matris, ekonomik yapısından dolayı yapılacak işlemlerde kolaylık sağlayacaktır. Ayrıca yine SVD sonucu elde edilen U ve V matrisleri, terim ve sınıf verilerinin tekil değerlerini temsil etmektedir.

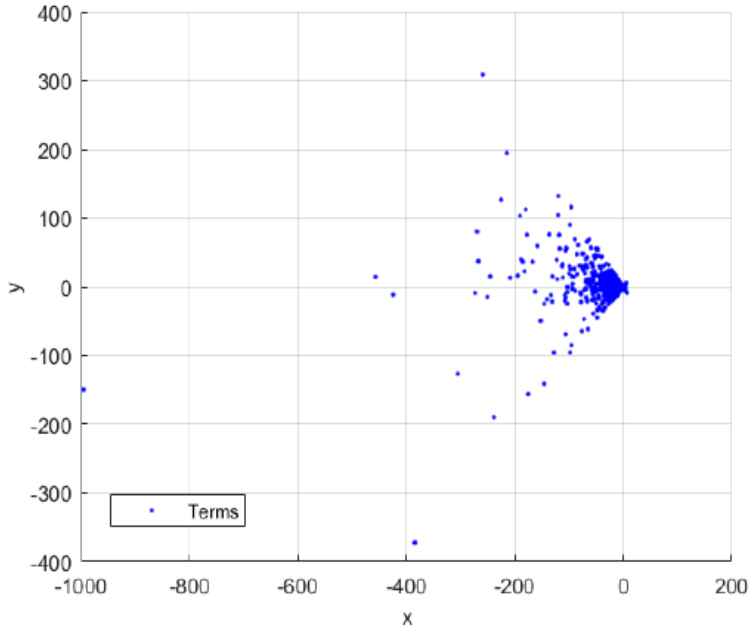
Şekil 5.10'da SVD uygulandıktan sonra elde edilen sınıf dağılımı görülmektedir. Burada sınıfların SVD sonrası elde edilen koordinatları görülmektedir. Sınıf

vektörlerinin daha rahat görülebilmesi için, $k = 2$ alınarak iki boyutlu bir gösterim sağlanarak sınıfların dağılım x ve y ekseininde gösterilmiştir.



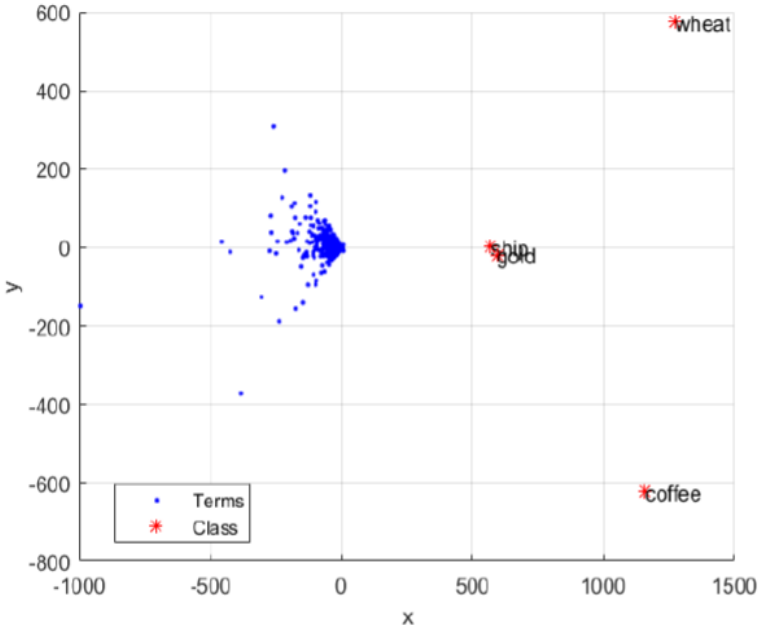
Şekil 5.11. SVD Sonrası Sınıf Dağılım Grafiği

Şekil 5.11'de SVD uygulandıktan sonra elde edilen terim dağılımı gösterilmiştir. Yine $k = 2$ alınarak x ve y ekseninde terimlerin dağılımı görülmektedir. Eğer $k = 3$ alınırsa, grafikte ek olarak üçüncü bir boyut olan z ekseni görülecektir.



Şekil 5.12. SVD Sonrası Terim Dağılım Grafiği

Şekil 5.13'de, SVD sonrası elde edilen terim-sınıf dağılımı görülmektedir.



Şekil 5.13. SVD Sonrası Terim-Sınıf Dağılım Grafiği

5.5. Test Verilerinin Sınıflandırılması

Terim-sınıf matrisi olarak isimlendirilen A matrisine SVD uygulandıktan sonra elde edilen $\sum, U$ ve V matrislerine rank-k yaklaşımı uygulanır. Sonuçta

$$A_k = U_k \sum_k V_k^T \tag{5.2}$$

kesik SVD elde edilir. Burada U_k ve $\sum_k$ matrislerinin çarpımları terimlerin vektörlerini, $\sum_k$ ve V_k^T matrislerinin çarpımları ise sınıfların vektörlerinin yer aldığı sırasıyla terim ve sınıf temsilci matrisleri olarak adlandırılan X ve Y matrislerini oluşturur.

$$X = U_k \sum_k, Y = \sum_k V_k^T \tag{5.3}$$

aldığımızda X matrisinin satırları terim sınıf matrisi olarak adlandırılan A matrisinde yer alan aynı indisteki vektörünü temsil ederken Y matrisinin her bir sütunu da A matrisinde yer alan aynı sütün indisindeki sınıfa ait vektörü temsil etmektedir. Böylece her bir terim ve sınıf aynı vektör uzayında sembolize edilebilir. Elde edilen bu vektör uzayında sınıfı bilinmedik doküman ya da doküman gruplarının terim-sınıf matrisinde yer alan sınıf vektörlerinin belirttiği konuma benzerliği dikkate alınarak sınıfları belirlenir. Doküman dizinleme işlemlerinde sorgulama olarak adlandırılan işlem bu uygulamadaki sınıflama işleminde sınıfların belirlenmesi işlemi olarak benzer şekilde kullanılmaktadır.

Sınıf belirleme sürecinde sınıfı bilinmeyen doküman için, terim sınıf matrisinde yer alan terimlerin ağırlığı A matrisinin oluşturulması sürecindeki aynı ağırlıklandırma tekniğine göre hesaplanır. Böylece elde edilen $m \times 1$ boyutlu c vektörü kullanılarak sınıfı bilinmeyen dokümanın A matrisinden elde edilen vektör uzayındaki konumu

$$\hat{c} = c^T X \sum_k^{-1} \tag{5.4}$$

formülüne göre bulunur. Elde edilen $\hat{c}$ sınıfı bilinmeyen dokümanın vektör uzayındaki temsilcisidir. Böylece bu dokümanın diğer sınıflara olan benzerliği vektör uzayındaki diğer sınıfların konumları dikkate alınarak anlamsal yakınlığını ölçen kosinüs benzerliği yöntemiyle belirlenir. Sınıfı bilinmeyen dokümanın, diğer sınıflara benzerliği karşılaştırılarak en çok benzeyen sınıf tespit edilmiştir. Tespit edilen sınıf, ilgili dokümanın sınıfı olarak atanmıştır. Test verileri üzerinde yapılan testlerin sonuçları bir sonraki bölümde anlatılmıştır.

6. ARAŞTIRMA BULGULARI VE SONUÇ

Vektör uzayının elde edilmesinden sonra, veri setinde bulunan test verileri sınıflandırma yönteminin performansını değerlendirmek amacıyla sınıflandırma testine tabi tutulmuştur. Test aşamasında 'coffee' sınıfına ait 28, 'wheat' sınıfına ait 71, 'ship' sınıfına ait 90 ve 'gold' sınıfına 31 adet metinden faydalanılmıştır.
Test aşamasında her bir sınıf için elde edilen kosinüs benzerlik değerleri aşağıdaki gibidir.

k	Kosinüs Threshold Değeri			
	coffee	wheat	ship	gold
2	0,924	0,822	0,988	0,768
3	0,781	0,689	0,928	0,859
4	0,711	0,593	0,683	0,768

Şekil 6.1. Sınıflara Göre Kosinüs Eşik Değerleri

Daha önce kosinüs benzerliğini anlatırken, bir eşik değerinin kullanıldığına değinilmişti. Kosinüs benzerliği uygulandığında elde edilen sonuç bu eşik değerinden büyükse, test edilen dokümanı eşik değerinden büyük olduğu sınıfa sınıflandırılabilir. Şekilde her bir sınıfın değişik k değerleri için eşik değerleri gösterilmiştir. Görüldüğü üzere, 3 farklı k değeri için test işlemi gerçekleştirilmiştir. Buradaki k değerleri, rank-k yaklaşımında alınan k değerini temsil etmektedir. Kosinüs değerleri dikkatlice

incelendiğinde ‘coffee’, ‘wheat’ ve ‘ship’ sınıfları için eşik değerlerinin $k = 2$ alındığında en yüksek olduğu görülmektedir. ‘gold’ sınıfı için ise en yüksek eşik değeri $k = 3$ alınınca ortaya çıkmıştır. Bu k değerlerini göz önüne alırsak, ilk üç sınıf için en optimum sonuçların k =2 alındığında elde edildiğini, ‘gold’ sınıfı için ise en başarılı sınıflandırmanın k =3 değerinde yapılacağını öngörülebilir.

Şimdi elde edilen bu eşik değerlerinin anlamlarını detaylandıralım. Bir test verisini sınıflandırma işlemine tabi tutarken, rank-k yaklaşımı esnasında k değerini k =2 alındığını varsayalım. Teste tutulan dokümanın her bir sınıfa ait kosinüs benzerlik değeri kosinüs teoremi ile hesaplanmaktadır. Test verisinin ‘coffe’ sınıfına ait kosinüs benzerlik değeri 0,924’den büyük ise, test verisinin sınıfı ‘coffee’ olarak atanır. Bu şekilde test verisinin tüm sınıflara ait kosinüs değeri elde edilerek karşılaştırılır. Yapılan karşılaştırma sonucu uygun bir eşleşme bulunmuş ise, ilgili sınıf test verisinin sınıfı olarak atanmaktadır.

Test verilerinin sınıflandırılması sonucu elde edilen neticeler Şekil 6.2’de görüldüğü gibidir.

k	Doğruluk Oranı(%)			
	coffee	wheat	ship	gold
2	85,7	71,8	82	13,6
3	78,6	78,9	7,8	93,3
4	57,1	69,1	7,9	76,6

Şekil 6.2. Doküman Sınıflandırma Sonucu Elde Edilen Başarı Performansı

Şekilde 3 farklı k değerine ait her bir sınıf için bulunan doğruluk oranları görülmektedir. Yukarıdaki şekil incelendiğinde, değişik 3 adet k değeri göze çarpmaktadır. Bu k değeri, rank-k yaklaşımındaki k değerini temsil etmektedir. k değeri değişince her bir sınıfa ait doğruluk oranlarının da değiştiği gözlemlenmiştir. Kosinüs eşik değerlerini gözlemlerken, 'coffee', 'wheat' ve 'ship' sınıfları için en uygun sonuçların k=2 iken çıkacağını öngörmüştük. 'gold' sınıfı için ise en uygun değerin k=3 olduğunu dile getirmiştik. Görüldüğü üzere 'coffee' ve 'ship' sınıfları için en uygun k değerinin 2, 'wheat' ve 'gold' sınıfı için en uygun k değerinin 3 olduğu tespit edilmiştir.

Test verileri sınıflandırılırken, k değerlerine bağlı olarak birçok performans verisi elde edilmiştir. En başarılı sonuçlar k=3 alındığında %93,3 başarım ile 'gold' sınıfı için elde edilmiştir. 'gold' sınıfına ait diğer sonuçlar gözlemlendiğinde, k=2 alındığında %13,6 oranında doğru sınıflandırma yapılmıştır. Aynı sınıfa ait k değeri 4 olduğunda başarı oranı %76,6'ya çıkmıştır.

Rank-k yaklaşımında alınan k değerinin sınıflandırmaya etkisi net bir şekilde görülmektedir. k=2 alındığında, gürültüler işleme dâhil edilmediği için daha iyi sonuçlar elde edilmiştir. k değeri yükseldiğinde gürültüler de dahil olduğundan dolayı başarı oranında azalma gerçekleşmiştir. Doküman ya da sınıf miktarını artırırsak, en optimum sonucu veren k değerinin de artacağını düşünebiliriz. Doküman sınıflandırma uygulamasında biz LSA'yı terim-sınıf matrisi üzerinde gerçekleştirdik. Normalde LSA, terim-doküman matrisleri üzerinde uygulanmaktadır. Kullandığımız terim-sınıf matrisi, literatürde kullanılan terim-doküman matrisine nazaran oldukça küçük boyutta bir matristir. Eğer sınıf miktarının da terim-doküman matrisindeki doküman adedi gibi yüksek değerlerde olduğunu varsayarsak, en doğru sınıflandırmayı yapabilmemiz için almamız gereken k değeri 100 ile 300 arasında olmalıdır [31].

İlerleyen çalışmalarda aynı veri setleri üzerinde K-NN en yakın komşuluk, SVM ve Naive Bayes ile metin sınıflandırma işleminin yapılması planlanmaktadır. Elde edilen sonuçlar LSA ile elde edilen başarı oranları ile karşılaştırılacaktır.

KAYNAKLAR

[1] A.-H. Tan, «Text mining: The state of the art and the challenges,» %1 içinde *Proceedings of the PAKDD 1999 Workshop on Knowledge Disocovery from Advanced Databases*, 1999.

[2] F. Varcin, H. Erbay ve F. Horasan, «Latent Semantic Analysis Via Truncated ULV Decomposition,» %1 içinde *Signal Processing and Communication Application Conference (SIU), 2016 24th*, 2016.

[3] P. W. Foltz, W. Kintsch ve T. K. Landauer, «The Measurement Of Textual Coherence With Latent Semantic Analysis,» *Discourse Processes*, Cilt %1 / %22-3, no. 25, pp. 285-307, 1998.

[4] Anonim, «daviddlewis,» [Çevrimiçi]. Available: http://www.daviddlewis.com/resources/testcollections/reuters21578/. [Erişildi: 29 10 2016].

[5] Anonim, «nlm.nih,» [Çevrimiçi]. Available: https://www.nlm.nih.gov/bsd/pmresources.html. [Erişildi: 20 5 2017].

[6] R. Feldman ve J. Sanger, The text mining handbook: advanced approaches in analyzing unstructured data, Cambridge University Press, 2007.

[7] S. Weiss, N. Indurkhya ve T. Zhang, PredictiveMethodsforAnalysing Unstructured Information, 2005.

[8] İ. F. Pilavcılar, *"Metin Madenciliği ile MeTin Sınıflandırma" , Yıldız Teknik Üniv. FBE, Yüksek Lisans,* İstanbul, 2007.

[9] M. W. Berry ve M. Castellanos, Survey of text mining II, Springer, 2008.

[10] T. W. Miller, Data and text mining: A business application approach, Prentice-Hall, Inc., 2004.

[11] M. Dunham, Data mining: Introductory and advanced topics, Pearson Education India, 2006.

[12] J. Unsworth, «Scholarly Primitives: what methods do humanities researchers have in common, and how might our tools reflect this?,» %1 içinde *Symposium on Humanities Computing: Formal Methods, Experimental Practice. King's College*, London, 2000.

[13] F. Mosteller ve D. Wallace, «Inference and disputed authorship: The Federalist,» 1964.

[14] D. I. Holmes , «The evolution of stylometry in humanities scholarship,» *Literary and linguistic computing,* cilt 3, no. 13, pp. 111-117, 1998.

[15] S. Argamon ve M. Olsen, «Toward meaningful computing,» *Communications of the ACM,* cilt 4, no. 49, pp. 33-35, 2006.

[16] H. Craig, «Authorial attribution and computational stylistics: If you can tell authors apart, have you learned anything about them?,» *Literary and Linguistic Computing,* cilt 1, no. 14, pp. 103-113, 1999.

[17] S. Ramsay, «In praise of pattern. In 'The Face of,» %1 içinde *3rd Conference of the Canadian Symposium on Text Analysis (CaSTA),* 2004.

[18] P. S. Szczepaniak, A. Tomczyk ve M. Pryczek, «Supervised web document classification using discrete transforms, active hypercontours and expert knowledge,» %1 içinde *WImBI2006,* 2006.

[19] S. Dumais, J. Platt, D. Heckerman ve M. Sahami, «Inductive learning algorithms and representations for text categorization,» %1 içinde *Proceedings of the seventh international conference on Information and knowledge management*, 1998.

[20] Y. Yang ve X. Liu, «A re-examination of text categorization methods,» %1 içinde *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, 1999.

[21] P. Domingos ve M. Pazzani, «On the optimality of the simple Bayesian classifier under zero-one loss,» *Machine learning,* cilt 2, no. 29, pp. 103-130, 1997.

[22] B. Masand , G. Linoff ve D. Waltz, «Classifying news stories using memory based reasoning,» %1 içinde *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, 1992.

[23] T. Mitchell, «Machine Learning, McGraw-Hill Higher Education,» *New York,* 1997.

[24] A. McCallum ve K. Nigam, «A comparison of event models for naive bayes text classification,» %1 içinde *AAAI-98 workshop on learning for text categorization*, 1998.

[25] V. N. Vapnik ve S. Kotz, Estimation of dependences based on empirical data, Springer-Verlag New York, 1982.

[26] T. K. Landauer, P. W. Foltz ve D. Laham, «An introduction to latent semantic analysis,» *Discourse processes,* Cilt %1 / %22-3, no. 25, pp. 259-284, 1998.

[27] S. Deerwester, S. Dumais, G. W. Furnas , T. K. Landauer ve R. Harshman, «Indexing by latent semantic analysis,» *Journal of the American society for information science,* cilt 6, no. 41, p. 391, 1990.

[28] Å. Björck, Numerical methods for least squares problems, SIAM, 1996.

[29] R. Baeza-Yates ve B. Ribeiro-Neto, Modern information retrieval, ACM press New York, 1999.

[30] G. W. O'Brien, *Information management tools for updating an SVD-encoded indexing scheme.MS Thesis,* University of Tennessee, Knoxville, 1994.

[31] S. T. Dumais, «Improving the retrieval of information from external sources,» *Behavior Research Methods, Instruments, & Computers,* cilt 2, no. 23, pp. 229-236, 1991.

[32] T. Joachims, «Text categorization with support vector machines: Learning with many relevant features,» *Machine learning: ECML-98,* pp. 137-142, 1998.