Araştırma Makalesi / Research Article

Ağırlıklandırılmış Çizgelerde Tf-Idf ve Eigen Ayrışımı Kullanarak Metin Sınıflandırma

Taner UÇKAN[1*], Cengiz HARK[2], Ebubekir SEYYARER[1], Ali KARCI[3]

[1]*Van Yüzüncü Yıl Üniversitesi, Bilgisayar Teknolojileri Bölümü, Van*
[2]*Bitlis Eren Üniversitesi, Enformatik Bölümü, Bitlis*
[3]*İnönü Üniversitesi, Bilgisayar Mühendisliği Bölümü, Malatya*
(ORCID:0000-0002-5190-3504) (ORCID: 0000-0001-5385-6775)
(ORCID: 0000-0002-8981-0266) (ORCID:0000-0002-8489-8617)

Özet
Günümüzde gerek metin gerekse cümle sınıflandırma problemleri üzerinde yoğunlukla çalışılmaktadır. Metin sınıflandırma işlemlerinde en önemli problemlerden biri sınıflandırılacak metinlerin yapısal olmamasıdır. Belli bir formata sahip olmayan metinlerin öncelikle bir önişlemden geçirilmesi gerekmektedir. Bu çalışmada metinleri sınıflandırma işleminde öncelikle sınıflandırılacak metinlerin önişlemini yapmak amacıyla KUSH (Karci-Uçkan-Seyyarer-Hark) adında bir önişleme aracı geliştirildi. Sonrasında elde edilen işlenmiş metinlerin sınıflandırılmasında çizge tabanlı matematiksel bir yaklaşım sunulmaktadır. Yapılan çalışmada Türkiye'de iyi bilinen 6 haber portalından ve 6 farklı alandan elde edilen metinleri içeren TTC-3600 veri seti kullanılmaktadır. Sınıflandırılacak metinler Tf (Terim frekansı) ve Idf (Ters doküman Frekansı) değerleri dikkate alınarak çeşitli önişlemlerden geçirildikten sonra kenar ve düğümlerden oluşan bir ağırlıklı çizge oluşturulmaktadır. Ağırlıklandırılmış çizgeler kullanılarak sınıflandırma işleminin etkililiği ve matematiksel verimliliği arttırılmıştır. Elde edilen çizgeyi ifade eden Komşuluk Matrisi ve Derece Matrisi kullanılarak Laplace Matrisi elde edilmektedir. Laplace Matrisinin özdeğer ayrışımı sonucunda elde edilen özdeğer ve özdeğer vektörleri ile metinler sınıflandırılmaktadır. Yapılan testler sonucunda sınıflandırma oranlarında dikkate değer bir doğruluk değerine ulaşıldığı görülmektedir.

Anahtar kelimeler: Çizge Bölümleme, Metin Sınıflandırma, Öz Vektörler, TTC-3600, Tf-Idf.

Text Classification Using Tf-Idf and Eigen Decomposition in Weighted Graphs

Abstract
Today, both text and sentence classification problems are studied intensively. One of the most important problems in the text classification process is that the texts to be classified are not structural. Texts that do not have a specific format must first be pre-processed. In this study, a preliminary processing tool called KUSH (Karci-Uçkan-Seyyarer-Hark) was developed in order to pre-process the texts to be classified first. Afterwards, a graph based mathematical approach is presented in the classification of processed texts. Studies in six including well-known news portals and obtained the text from 6 different areas in Turkey TTC-3600 data sets are used. Texts to be classified are subjected to various pre-treatments taking into consideration the Tf (Term frequency) and Idf (Reverse document frequency) values, and then a weighted graph consisting of edges and nodes is formed. By using weighted charts, the efficiency and mathematical efficiency of the grading process were increased. By using the matrix of the neighborhood matrix and the degree matrix, the Laplace matrix is obtained. The eigenvalue and eigenvalue vectors and texts derived from the eigenvalue decomposition of the Laplace matrix are classified. As a result of the tests performed, it is seen that a significant accuracy value is reached in the classification rates.

Keywords: Graph partitioning, Text Classification, Eigenvectors, TTC-3600.

[*]Sorumlu yazar: taneruckan@yyu.edu.tr
Geliş Tarihi: 22.02.2019, Kabul Tarihi: 01.07.2019

1. Giriş

Günümüzde bilgilerin işlenmesi ve analiz edilmesi çoğu zaman bilgisayarlar olmadan imkânsız hale gelmektedir. Bununla birlikte bilgi işlem kaynaklarının hızla artan gücüne rağmen birçok görevin tamamen bilgisayarlaştırılması da oldukça zor olmuştur. Çeşitli kaynaklardan elde edilen bilgilerin büyük bir oranını metin dosyaları oluşturmaktadır. Bu dosyalar genellikle yapılandırılmamış bilgilerden oluşmakta ve bu dosyalardan gerekli bilgilerin elde edilmesi için analiz edilmeleri gerekmedir. Çok sayıda bilgi varlığının getirdiği sayısız fayda ile beraber ortaya çıkan bazı sorunların da çözülmesi gerekmektedir. Bilgiye erişmenin (aranılan bilginin bulunabilirliğinin) kolay olması oldukça önemli bir konudur. Bu bağlamda ortaya çıkan sorunlardan bir tanesi de elektronik ortamdaki metinlerin sınıflandırılması sorunudur. Metin sınıflandırma sorunu en genel anlamı ile eldeki bir metnin önceden belirlenen sınıflardan hangisine ya da hangilerine girdiğinin belirlenmesi demektir [2]. Yapısal olmayan metinlerden oluşan verilere örnek olarak yakından ilgili olan ancak birbiriyle aynı olmayan haber makaleleri verilebilir. Bu tür verileri analiz etme görevini bilgisayarlaştırmak için kümelenme problemini çözmek ve veri birimlerini benzerlik ölçüsüne göre gruplamak gereklidir [1]. Metin sınıflandırma ve kümeleme, belge alma, web'de arama, spam filtreleme, makale veya köşe yazılarının hangi yazara ait olduğunu bulmak [2] gibi birçok uygulamada önemli bir rol oynar. Bu uygulamaların merkezinde, lojistik regresyon veya K- araçları gibi makine öğrenme algoritmaları yer almaktadır. Bu algoritmalar tipik olarak metin girişinin sabit uzunluklu bir vektör olarak temsil edilmesini gerektirir. Metinler için en yaygın sabit uzunluklu vektör gösterimi, sadeliği, etkinliği ve çoğu zaman şaşırtıcı doğruluğu nedeniyle *bag of words* veya n-gram *bag of words* yöntemidir [3]. Metin sınıflandırmada en yaygın gösterim şekli olan vektör uzayı modeli *bag of Word* yöntemi üzerine kurulmuştur. Bu modelin en büyük avantajı sınıflandırma algoritmaları tarafından kolaylıkla kullanılabilmesidir. Ancak bu model ile metinlerin temsili yapıldığında sadece kelimelerin veya cümlelerin frekansları dikkate alınmaktadır; bunun yanında yapısal ve anlamsal özellikler tamamen görmezden gelinmektedir[4]. Yapılandırılmamış metinlerin yanı sıra elde edilen verilerin bir kısmı yarı yapılandırılmış verilerden oluşmaktadır. Yarı yapılandırılmış verilerin benzerliklerini bulabilmek için şimdiye kadar birçok benzerlik ölçümleri sunulmuştur ancak bu yöntemler düz metinden oluşan dokümanların benzerliğini bulmada kullanılamamıştır. Çünkü düz metinler yarı yapılandırılmış metinler gibi metinlerin yapıları hakkında açık bilgiler sunmamaktadır. İki nesne arasındaki benzerliği değerlendirmek için en popüler yol bu nesneleri karşılaştırmak ve bu karşılaştırma sonucunda bilgi elde etmektir. Verilerin benzerliğini ölçmek için çeşitli benzerlik yöntemleri kullanılmaktadır. Bu yöntemler sırasıyla RNA yapılarında, alan kavramlarında, dokümanlarda ve kural tabanlı sistemlerde kullanılmıştır [5]. Verilerin benzerliklerinden yola çıkarak bu verileri sınıflandırma işlemleri yapılmaktadır. Benzer veriler aynı sınıf içerisinde gösterilirken farklı yapıdaki veriler farklı sınıflar altında toplanmaktadır. Metin sınıflandırma yapılırken çeşitli veri madenciliği algoritmaları kullanılmakla beraber daha matematiksel altyapıya sahip ve uygulanabilirliği yüksek olan çizge yapılarından da faydalanılmaktadır. Diğer metin sınıflandıra algoritmaları da olduğu gibi metinlerin çizgelerle gösteriminde de *bag of words* yaklaşımından faydalanılmaktadır fakat sonrasında yapılan çeşitli matematiksel işlemler sonrasında standart *bag of words* yaklaşımından çok daha etkili bir belge kodlamasına olanak sağlamakta ve doğruluğu daha yüksek bir sınıflandırma sonucu üretmektedir [4]. Metinlerin çizgeler olarak gösterimindeki en önemli dezavantajı matematiksel modellemesinin karmaşık bir yapıya sahip olmasıdır. Bu dezavantaj çizge gösteriminin sahip olduğu ifade gücünün tam olarak kullanılmasını önler [4]. Çizgeler, anlamsal ağlar, belge işleme, görüntü analizi, biyometrik tanımlama, bilgisayarla görme ve video analizi gibi birçok uygulama alanındaki yapısal nesnelerin tanımlanması için kullanılan güçlü ve çok yönlü bir araçtır [6]. Farklı bir yapı olan Çizgeler; nesneler arasındaki karmaşık yapısal bilgiyi temsil etmek için yaygın olarak kullanılmaktadır. Çizgelerdeki düğümler nesneleri temsil ederken kenarlar nesne çiftleri arasındaki bağlantıları göstermektedir. Güçlü esneklik, etkilenebilirlik ve boyut kısıtlaması olmayan karakteristik özelliklerinden dolayı çizge veri madenciliği, biyoinformatikten (örneğin, DNA protein dizisinin değişip değişmediğinin belirlenmesi), kimya (örneğin bilinmeyen kimyasal bileşiklerin toksik olup olmadığının tespit edilmesi), sosyal ağlara (örneğin içyapı özelliklerine göre sosyal grupların sınıflandırılması) gibi birçok alanda kullanılmıştır. Yapısal veri analizine olan talep artışı ve çizge veri tabanlarının popülerleşmesiyle birlikte, otomatik bir çizge sınıflandırma modeline büyük bir ihtiyaç vardır ve böylelikle bilinmeyen çizgelerin sınıfları tahmin edebilir veya farklı sınıflar arasındaki karmaşık yapıları anlaşılabilir [7]. Günümüzde Çizge sınıflandırma ile ilgili ana araştırmalar ikili

sınıflandırma işlemine yani pozitif sınıf ve negatif sınıf üzerine odaklanmaktadır. Bir eğitim çizge seti ve bir test çizge seti göz önüne alındığında çizge sınıflamanın amacı test çizge setindeki iki kategoriyi ayırt etmek için eğitim çizge setine göre bir sınıflandırma modeli oluşturmaktır [7]. Bu çalışmada Spektral Çizge Bölümleme yöntemi ile sınıflandırma işlemi yapılmıştır. Spektral Çizge Bölümleme, bir grafiği iki alt bölüme ayırma yöntemidir; öyle ki alt bölümler neredeyse eşit sayıda köşeye sahip olurken iki alt bölüm arasındaki kenar sayısını da en aza indirilir. Spektral Çizge Bölümleme kullanılacak benzeşim matrisinin spektrumu kullanılmaktadır. Bir matrisin spektrumu, o matrisin özdeğerlerini ifade etmektedir; bu nedenle bir Çizgenin spektral bölümlenmesi yapılırken Çizge ile ilişkili olan bir matrisin özdeğerleri kullanılır [8]. Spektral Çizge Bölümlemede Fiedler's Vektörü adı verilen özdeğer vektörü kullanılmaktadır. Fiedler's vektörü Laplacian ayrışımı sonucunda elde edilen özdeğer vektörlerinin sıralanması sonucunda bulunan ikinci en küçük vektörü ifade etmektedir. Bu vektör değerleri negatif ve pozitif değerlerden oluşmaktadır. Vektörde bulunan negatif değerler bir gruba pozitif değerler ise farklı bir gruba konularak ikili sınıflandırma işlemi yapılmaktadır.

Bu çalışmada, verilen bir dokümanın içermiş olduğu cümlelerin ikili sınıflandırması yapılmıştır. Bu işlem yapılırken Fiedler'in[8] Spektral çizge bölmeleme yöntemi kullanılmıştır. Çizge bölümleme işlemine gelinmeden önce verilen dokümanların bir ön işleme tabi tutulması gerekmektedir. Bu bağlamda KUSH adında bir önişleme aracı geliştirildi. Bu önişleme aracının diğer önişleme araçlarından farkı olarak önişleme yapılırken herhangi bir dil köken kütüphanesine ihtiyaç duyulmaması ve kendi kendini besleyen bir araç olmasını gösterilebilir. KUSH ile verilen dokumalar öncelikle parçalama işlemlerine tabi tutularak frekans değerleri hesaplanmakta ve sonrasında dokümanlar içerisinde bulunan anlam olarak aynı alanı ifade eden fakat farklı yapım ve çekim eki almış olan kelimelerin değiştirilme işlemi yapılmaktadır. Bu işlemin ardından KUSH ile elde edinilen yeni dokümanların içerdikleri kelimelere ait *Tf* ve *Idf* değerleri hesaplanarak komşuluk matrisi elde edilmektedir. Ön işleme işlemi ile ilgili ayrıntılı bilgi "Metinlerin Çizge Olarak Temsil Edilmesi" bölümünde verilmektedir.

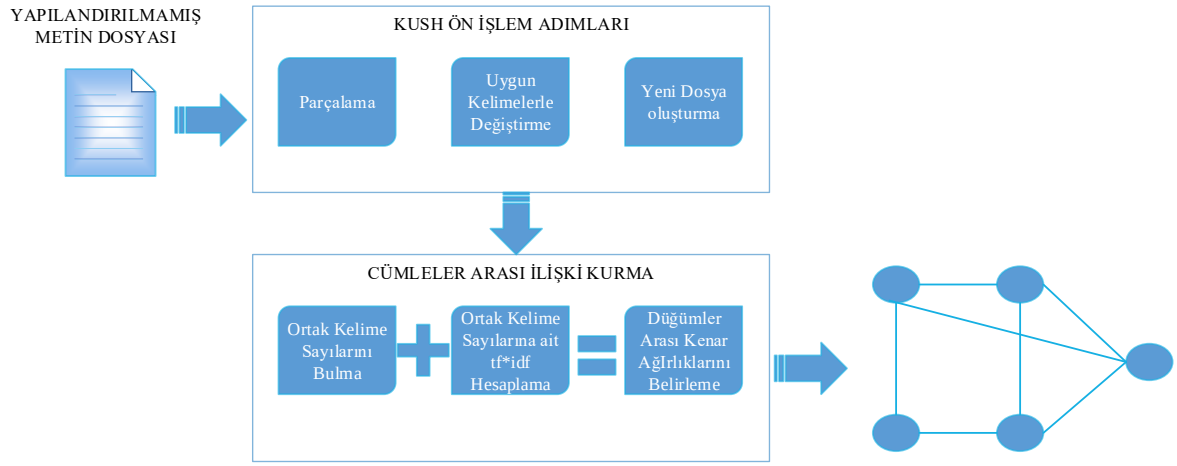
Ön işleme işlemi yapıldıktan sonra spektral çizge bölmeleme yöntemi uygulanmaktadır. Bu yöntemi uygulamak için öncesinde elde edilen çizgeden terim frekans değerleri ile ağırlıklandırılmış ve terim frekans değerlerine ek olarak tf *df değerinin eklenmesinden oluşan ağırlıklı çizgeler oluşturulmaktadır.

Makalenin devamında metinlerin birer çizge olarak temsil edilmesi anlatılmıştır. Üçüncü bölümde önerilen yöntem hakkında bilgi verilmiş ve çalışmada kullanılan ağırlıklandırma yöntemi ayrıntılı olarak açıklanmıştır. Dördüncü bölümde ise çalışmada kullanılan Spektral çizge bölmeleme yöntemi kullanılarak çizgenin bölmelenmesi anlatılmış ve devamında beşinci bölümde yapılan çalışmaların test aşamasında kullanılan veri setinden bahsedilmiş olup bu veri seti üzerinde yapılan testlerde kullanılan ölçüm değerleri ve ölçüm sonuçları altıncı bölümde ayrıntılı olarak açıklanmıştır.

2. Metinlerin Çizge Olarak Temsil Edilmesi

Metin Sınıflandırma, verilen metin setlerinin birbirinden bağımsız iki alt kümeye ayırma işlemidir. Her bir alt kümelerin kendi elemanları arasındaki benzerliğin mümkün olduğunca yüksek aynı zamanda kümeler arasındaki benzerliğin mümkün olduğunca düşük olması gerekmektedir [1]. Yapılan çalışmada metinlerin çizgelere dönüştürme işlemleri Şekil 1'de adım adım gösterilmektedir.

Bu çalışmada, Türkçe metin sınıflandırma çalışmalarında kullanılan [9, 10] TTC-3600 veri seti kullanılmaktadır. Bu veri setinde bulunan bütün metinler yapılandırılmamış verilerden oluşmaktadır. Bu yüzden sınıflandırma işlemi öncesinde bunların bir ön işleme tabi tutulması gerekmektedir. Geliştirilen KUSH yazılımı ile dosyadan okunan yapılandırılmamış metinler belirli ön işleme adımlarından geçirildikten sonra çizgeye dönüşümü yapılacak olan yeni bir metin dosyası oluşturmaktadır. Bu çalışmaya özgü geliştirilen KUSH yazılımının metin ön işleme algoritması Tablo 1'de gösterilmektedir.



Şekil 1. Metinden Çizgeye Dönüştürme Adımları

Tablo 1. KUSH Metin Ön İşleme Algoritması

1. input *D=Ham_Dosya;*

2. output *O=İslenmiş_Dosya;*

3. if Lenght*(D)>0*

*4. Cumleler[]=****String.Empty;***

*5. Kelimeler[]=****String.Empty;***

*6. Uygun_Alternatifler[]=****String.Empty;***

7. En_Uygun_Alternatif= ***String.Empty;***

*8. Cumleler[]=D.****Split****(‘.’);*

9. for all *c € Cümleler* ***do***

10. *Cümleler[c]=temizle(c);*

11. end for

12. *Kelimeler[]=Cümleler.****Split****(‘ ’);*

13. for all *k € Kelimeler* ***do***

14. *Uygun_Alternatifler[k]=Uygun_Alternatifler_Bul(Kelimeler[k],Kelimeler);*

15. for *u € Uygun_Alternatifler* ***do***

16. *En_Uygun_Alternatif=En_Uygun_Alternatif_Bul(k,Uygun_Alternatifler)*

17. end for

18. *Kelimeler[k]=En_Uygun_Alternatif;*

19. end for

20. end if

21. for all *k € Kelimeler* ***do***

22. *O+=k+”” ”;*

23. end for

24. return *O;*

Tablo 2. Komşuluk Matrisi ve Derece Matrisi Oluşturma Algoritması

```
input D=işlenmiş_Dosya
output Komşuluk_Matrisi,Derece_Matrisi
Komsuluk_matrisi[,]
Derce_Matrisi[,]
Laplacian_Matris[,]
Kelimeler[]
Cümleler[]=D.Split('.')
for i in Cümleler:
        Kelimeler[]=Cümleler[i].Split(" ")
        Kelimeler=GerkesizKelimeleriKaldır()
for j in range(0,Lenght(Kelimeler)-1) do
        for k in range(0,Lenght(Kelimeler)-1) do
                if Lenght(Kelimeler[j] ∩ Kelimeler[j] ) >0 then
                        Komsuluk_değeri=Lenght(Kelimeler[j] ∩ Kelimeler[k])
                        Terim_frekansi=Terim_Frekansı_Hesapla(Kelimeler[j])
                        Ters_Terim_Frekansı=Ters_Terim_Frekansı_Hesapla(Kelimeler[j])
                        Tf_Idf_degeri=Terim_frekansi * Ters_Terim_Frekansı
                        Komsuluk_değeri=Komsuluk_değeri+Tf_Idf_degeri
                        Derece_degeri+=1
                else
                Komsuluk_değeri=0
                end if
                Komsuluk_matrisi[j,k]=Komsuluk_değeri
                Derce_Matrisi[j,j]=Derece_degeri
         end for  end for
return Komsuluk_matrisi, Derce_Matrisi
```

Dokümanlara uygulanan ön işlem sonrasında elde edilen yeni dokümandaki cümleler birer düğüm olarak temsil edilmekte ve cümlelerin içerdikleri ortak kelime sayıları kenarların ağırlıkları olarak belirlenmektedir. Bu ağırlıklandırma işlemi ile cümleler arasındaki ilişkinin gücü belirlenmektedir. Böylelikle dokümanların temsili artık numerik değerler ile yapılmakta ve matematiksel işlemlere uygun hale gelmektedir. Cümleler arasındaki ortak kelime sayılarının belirlenmesinin ardından KUSH yazılımı ile oluşturulan yeni doküman kullanılarak cümleler arasındaki ortak her bir kelimenin Tf, Idf ve Tf*Idf değerleri hesaplanarak yani kelimelerin taşıdığı anlam miktarı belirlenerek kenar ağırlıklarına eklenmektedir. Böylelikle cümleler arasındaki bağlılık derecesi güçlendirilmektedir. Metinlerden yola çıkarak düğümleri, ayrıtları ve bu ayrıtların ağırlıklarını belirlemek için geliştirilen algoritma Tablo 2'de gösterilmektedir.

3. Önerilen Yöntem

Bu çalışmada verilen bir dokümanda bulunan cümleleri ikili (Binary) sınıflandırma işleminin yapılması amaçlanmaktadır. Bu işlem yapılırken Spektral Çizge Bölmeleme yöntemi kullanılmaktadır. Literatürde daha önce farklı alanlarda Spektral Çizge Bölmeleme yöntemleri kullanılmıştır [12]–[15]. Fakat yapılan araştırmalar sonucunda metin benzerlikleri üzerinde kullanımına rastlanılmamıştır. Yapılan bu çalışmada çizge oluşturulmadan önce çalışmaya özgü bir ön işleme yapılmakta ve sonrasında da çizgelerin ağırlıklandırılması için de yeni bir ağırlıklandırma yöntemi sunulmaktadır.

3.1. Terim Frekansı ve Ters Doküman Frekansı (Tf-Idf)

Bilgiye erişmekte kullanılan en yaygın yöntemlerden biri terim frekansı ve ters terim frekansıdır [16]. *Tf-Idf*, en temel belge temsil şeklidir ve kabul edilen üç temsil yöntemi arasında en uzun tarihe sahiptir [17]. *Tf-Idf*, bir dokümanda bulunan kelimeler üzerinde işlem yapmaktadır yani *bag of Word* yöntemine dayanmaktadır [18]. Terim Frekansı bir terimin dokümanda tekrarlanma sayısını temsil etmektedir. Bir dokümanda bir terimden ne kadar fazla tekrar var ise o terimin frekans değeri daha yüksek olduğu kabul edilmektedir.

Ters terim frekansı ise bir terimin birden fazla dokumanda bulunma sıklığını belirtir. Terim frekansının tersine bir terim ne kadar az dokümanda bulunursa o kadar değerli olduğu kabul edilir. Buna örnek olarak bütün cümlelerde bulunabilecek "ve" ,"ile","bu","şu" gibi terimleri örnek verebiliriz. Bu terimler genellikle birçok dokümanda bulunmakta ve dokümanın yapısı veya sınıfı ile ilgili herhangi bir bilgi vermemektedir. Bu yüzden bu tür terimlerin bilgi taşımadığı kabul edilmektedir. *Idf* hesaplama yöntemine göre bir kelime ne kadar az dokümanda bulunursa o kadar çok anlam taşımaktadır. *Tf*Idf* değerinin hesaplanması Denklem*(1)* 'de gösterilmektedir.

tf_{ij} : j dokümanı içerisinde i teriminin bulunma sayısı.
df_i : i teriminin en az bir defa bulunduğu doküman sayısı.
N : Toplam doküman sayısı.

$$tf_{i,j} * Idf_{i,j} = tf_{i,j} x \log\left(\frac{N}{df_i}\right) \qquad (1)$$

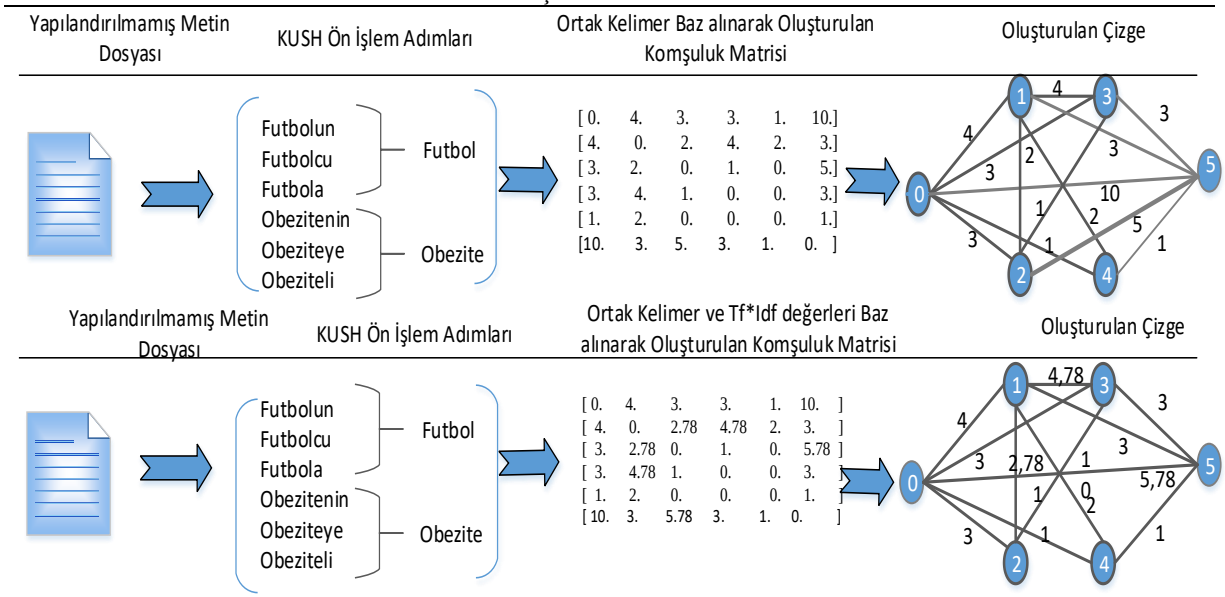
3.2. Çizge Ağırlıklandırma

Bu çalışmada çizge ayrıtları ağırlıklandırılırken iki temel unsur dikkate alınmaktadır. Birinci durumda cümleler arasında bulunan ortak terim (kelime) sayısı ayrıt ağırlığına eklenmektedir. İkinci durumda ise cümleler arasında ortak olan her bir terimin *Tf * Idf* değerleri ayrı ayrı hesaplanarak ayrıt ağırlığına eklenmektedir. Çizgelerin ayrıt ağırlıkları belirlendikten sonra bu çizgelere ait komşuluk matrisleri elde edilmektedir. Komşuluk Matrisi düğümlerden düğümlere olan bağlantıyı gösteren bir kare matristir. Komşuluk matrisinin elemanları $a_{i,j}$ olmak üzere G_{dd} ile gösterilen çizgeye ait komşuluk matrisi denklem (2) de gösterilmektedir.

$$a_{ij} = \begin{cases} 1, & Eğer(d_i, d_j) \in K \text{ ise} \\ 0, & Diğer\ Durumlar \end{cases} \qquad (2)$$

Aşağıda yapılandırılmamış 6 cümleden oluşan bir paragraf için hem ortak kelime sayısı ile hem de ortak kelime sayısına ek olarak ortak her kelimeye ait *Tf * Idf* değerlerinin eklenmesi sonucunda oluşturulan ağırlıklandırmalar ile elde edilen komşuluk matrisi ve çizge tabloda gösterilmektedir.

Tablo 3. Cümleler Arası Ortak Kelime Sayıları ile Bu Kelimelerin Tf*İdf Çarpımları Sonucunda Çizge Oluşturma Adımları



3.3. Laplace Matrisi ve Spektral Çizge Bölmeleme

Admittance matrix, Kirchhoff matrix veya Discrete Laplacian olarakta isimlendirilen Laplacian matrisi, çizgelerin matris olarak temsil edilmesinde ayrıca matrisler hakkında önemli bilgiler elde edilmesi amacı ile kullanılmaktadır [19]. Bir çizgenin Laplacian matrisi tıpkı komşuluk matrisi gibi çizgeler hakkında birçok bilgiyi taşır ancak kendine özgü birçok kullanımı ve farklı özellikleri mevcuttur [20]. Yapılan çalışmada oluşturulan çizgeler yönsüz ve ağırlıklandırılımış çizgelerden oluşmaktadır. Ön işleme sonrasında oluşturulan yeni dökümandan komşuluk matrisleri ve derece matrisleri elde edilmektedir. Derece matrisi bir çizgede bulunan her bir düğüme gelen ve düğümden çıkan kenar sayılarının toplamından oluşmaktadır. Derece matrisleri birer diagonal matris olarak gösterilmektedir. Bu çalışmada Fiedlerin Spektral Çizge Bölmeleme safhasına gelmeden önce laplacian matrisi elde edilmektedir. Laplacian matrisinin elde edilmesinde kullanılan çeşitli yöntemler bulunmaktadır. Bu yöntemler aşağıdaki gibi özetlenebilir:

3.3.1. Basit Laplacian Matrisi

Verilen basit bir G çizgesi için D matrisi derece matrisini, K matrisi komşuluk matrisini temsil etmektedir. Bu durumda Basit laplacian matrisi Denklem *(3)*'te ki gibi hesaplanmaktadır.

$$L = D - K \tag{3}$$

L matrisine ait değerler denklem *(4)*'te ki gibi verilmektedir.

$$L_{i,j} := \begin{cases} \deg(v_i) & \text{eğer i=j} \\ -1 & \text{eğer i} \neq \text{j ve } v_i \text{ ile } v_j \text{ komşu ise} \\ 0 & \text{diğer durumlar} \end{cases} \tag{4}$$

3.3.2. Simetrik Normalize Laplacian Matrisi

Verilen bir çizgede D derece matrisi, K komşuluk matrisi olmak üzere, *Simetrik Normalize Laplacian* matrisi Denklem *(3)*'te ki gibi hesaplanmaktadır.

$$L^{sym} := D^{-1/2} L D^{-1/2} = I - D^{-1/2} K D^{-1/2} \tag{5}$$

L matrisine ait değerler denklem *(6)*'deki gibi verilmektedir.

$$L_{i,j}^{sym} := \begin{cases} -1 & \text{eğer i=j ve } \deg(v_i) \neq 0 \\ -\dfrac{1}{\sqrt{\deg(v_i)\deg(v_j)}} & \text{eğer i} \neq \text{j ve } v_i \text{ ile } v_j \text{ komşu ise} \\ 0 & \text{diğer durumlar} \end{cases} \tag{6}$$

3.3.3. Rasgele Yürüyüş Normalize Laplacian Matrisi

Verilen bir çizgede D derece matrisi, K komşuluk matrisi olmak üzere *Rasgele Yürüyüş Normalize Laplacian* matrisi Denklem *(7)*'te ki gibi hesaplanmaktadır

$$L^{rw} := D^{-1} L = I - D^{-1} K \tag{7}$$

L matrisine ait değerler denklem (8)'deki gibi verilmektedir.

$$L_{i,j}^{rw} := \begin{cases} 1 & \text{eğer i=j ve } \deg(v_i) \neq 0 \\ -\dfrac{1}{\sqrt{\deg(v_i)}} & \text{eğer i} \neq \text{j ve } v_i \text{ ile } v_j \text{ komşu ise} \\ 0 & \text{diğer durumlar} \end{cases} \quad (8)$$

3.3.4. Genelleştirilmiş Laplacian

Verilen bir çizgede D derece matrisi, K komşuluk matrisi olmak üzere tanımlanan Q matrisine ait değerler Denklem*(9)*'daki gibi verilmektedir.

$$\begin{cases} Q_{i,j} < 0 & \text{eğer i} \neq \text{j ve } v_i \text{ ile } v_j \text{ komşu ise} \\ Q_{i,j} = 0 & \text{eğer i} \neq \text{j ve } v_i \text{ ile } v_j \text{ komşu değil ise} \\ diğer & \text{diğer durumlar} \end{cases} \quad (9)$$

Yapılan çalışmada yukarıda verilen denklemlerden Basit Laplacian Matris denklemi kullanılmaktadır. Basit Laplacian matrisi, derece matrisi ile komşuluk matrisinin farkından elde edilmektedir. Bu laplacian matrisi ile Spektral Çizge Bölmeleme yöntemi kullanılarak başlangıçta tek parçadan oluşan çizge, iki ayrı alt çizgeye ayırılarak sınıflandırma işlemi yapılmaktadır.

4. Spektral Çizge Bölmeleme

Spektral çizge bölmeleme 1970'lerin başlarında ortaya çıkan etkili bir çizge bölmeleme yöntemidir [21]. Verilen bir A Çizgesi düğüm(V) ve kenarlardan(E) oluşmaktadır [8]. Kenarlar düğümleri birbirlerine bağlamaktadır ve $(u,v) \in E$ olmak üzere bu iki düğüm arasında bir bağlantı bulunuyor ise $(u,v) \in E$ olduğu kabul edilmektedir. *G=(V,E)* şeklinde tanımlanan bir *G* çizgesine ait komşuluk matrisi $K(G)=(k_{i,j})$ olarak tanımlanmaktadır. *G=(V,E)* Çizgesine ait komşuluk matrisine ait değerler Denklem*(10)*'da ki gibi tanımlanmaktadır.

$$k_{i,j} = \begin{cases} 1, & (i,j) \in E \\ 0, & (i,j) \notin E \end{cases} \quad (10)$$

G=(V,E) şeklinde tanımlanan bir *G* çizgesine ait tanımlanmış olan ve derece matrisini belirten $D(G)=(d_{i,i})$ diagonal matrisi Denklem *(11)* 'deki gibi tanımlanmaktadır.

$$d_{i,i} = \begin{cases} d(i), & i = j \\ 0, & i \neq j \end{cases} \quad (11)$$

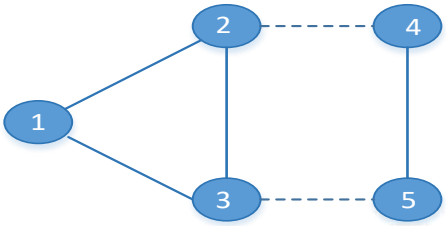
Bir kare matris olan $A \in R^{nxn}$ ve x vektörü $A \in R^n$ 'de tanımlı olmak üzere *Ax* çarpımı R^n, de yeni bir vektör oluşturmaktadır. A matrisi ile x vektörünün çarpımı sonucunda oluşan yeni vektör, x vektörü ile aynı yönde fakat farklı büyüklükte olmaktadır. Yani x vektörü bir λ değeri ile yeniden ölçeklendirilmiş olmaktadır. Bu durumdu *Ax* çarpımındaki *x* değerlerine, A matrisinin öz vektör değerleri ve aynı zamanda λ değerlerine ise her bir vektör değerinin öz değeri denilmektedir. Matematiksel olarak $Ax=\lambda x$ şeklinde gösterilir. *nxn* Boyutlu bir *A* matrisinin öz değerlerini hesaplamak için I birim matris olmak üzere $\det(A-\lambda I_n)x=0$ eşitliğinin çözülmesi gerekmektedir [8].

Yapılan çalışmada Fiedler tarafından sunulan Spektral Çizge Bölmeleme yöntemi kullanılmaktadır. Bu yönteme göre *G=(V,E)* şeklinde tanımlı bir çizge en uygun noktalardan iki ayrı alt çizgeye bölünebilir. Bu yöntemde çizgeler bölünürken Laplacian matrisi kullanılmaktadır. *G* matrisinden elde edilen komşuluk matrisi ve derece matrisinin basit laplacian dönüşümü sonucunda elde

edilen laplacian matrisine ait öz değer ve öz vektör değerleri hesaplanmaktadır. Fielder'e göre laplacian matrisinden elde edilen bütün öz değerler $\lambda_1 \le \lambda_2 \le .... \le \lambda_n$ şeklinde küçükten büyüğe doğru sıralanmakta ve elde edilen dizilimde en küçük ikinci öz değere (λ_2) sahip öz vektör değerleri G çizgesinin hangi düğümlerden bölüneceğini göstermektedir. Ayrıca Literatürde λ_2 ye ait öz vektör değerlerine Fiedler vektörü'de denilmektedir[8, 22]. Tablo 4 'te Spektral çizge bölmelemenin bir örneği gösterilmektedir.

Tablo 4. Spektral Çizge Bölmeleme Adımları

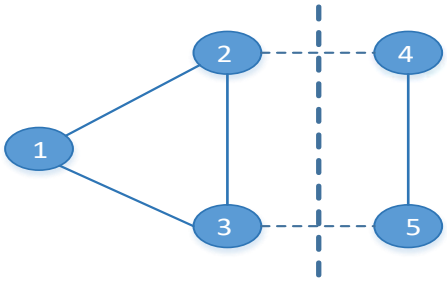
TEK PARÇALI ÇİZGE



Laplacian Dönüşümü L=D-K (Denklem *(3)*)

Düğüm No	1	2	3	4	5
1	2	-1	-1	0	0
2	-1	2	-1	0	0
3	-1	-1	2	0	0
4	0	0	0	1	-1
5	0	0	0	-1	1

BÖLÜNMÜŞ ÇİZGE



Özdeğer Ayrışımı E=eig(L)

Düğüm No	1	2	3	4	5
1	0.57	0	0	0.26	0.7
2	0.57	0	0	-0.8	-0.15
3	0.57	0	0	0.5	-0.6
4	0	-0.7	-0.7	0	0
5	0	-0.7	0.7	0	0

λ_2 **=[0 0 0 -0.7 - 0.7]= {D1, D2, D3,},{ D4, D5}**

5. Veri Seti

Yapılan çalışmada test veri seti olarak TTC-3600 veri seti kullanılmıştır. Kullanılan veri seti Türkiye'de iyi bilen 6 haber potalından ve 6 farklı alandan elde edilen metinleri bulundurmaktadır. Bu veri seti metin madenciliği çalışmalarına uygun olarak hazırlanmıştır [9]. Spor, teknoloji, kültür-sanat, ekonomi, sağlık ve siyaset alanlarını içeren ve her birinden 600 adet dokümanın bulunduğu bir veri setidir. Bu veri setinde bulunan bütün dokümanlar tamamen yapısal olmayan ham metinlerden oluşmaktadır.

6. Deneysel Çalışma

Bu çalışmada yapısal olmayan metinlerin çizgelere dönüştürülmesi ve bu metinlerin temsil ettiği çizgelerin spektral çizge bölmeleme yöntemi kullanılarak sınıflandırlması amaçlanmıştır. Sınıflandırma sürecinde Türkçe metinler üzerinde yapılacak veri madenciliği çalışmalarında kullanılması amacı ile geliştirilmiş olan TTC-3600 veri seti kullanılmıştır. Bu çalışmada veri setinde bulunan metinler KUSH ön işleme aracı sayesinde çizge oluşturmaya daha elverişli hale getirilmiştir. Ön işleme sonucunda elde edilen metinler iki farklı yöntemle çizgelere dönüştürülmüş ve elde edilen sonuçlar karşılaştırılmıştır. Bahsedilen her iki yöntemde de çizgeler oluşturulurken metinlerde bulunan cümleler, düğümleri temsil

etmektedir. Düğümler arasındaki ayrıt ağırlıklarını belirlemede aşağıda gösterilen iki farklı yol izlenmiştir.

- ✓ Düğümler arasındaki ayrıt ağırlıklarını cümleler arasında bulunan ortak kelime sayısı olarak belirlemesi.
- ✓ Düğümler arasındaki ayrıt ağırlıklarını cümleler arasında bulunan ortak kelime sayılarına ek olarak bu kelimelerin *Tf * Idf* çarpım değerlerinin de eklenmesi.

Yapılan deneysel çalışmalar sonucunda cümleler arasında bulunan ortak kelime sayısına ek olarak bu kelimelerin *Tf * Idf* değerlerinin eklenmesi sonucunda elde edilen sınıflandırma başarısında ortalama %1ile %5 arasında bir artışın olduğu gözlemlenmiştir. Yapılan çalışmada sınıflandırma başarı ölçütü olarak karmaşıklık matrisi kullanılmıştır. Karmaşıklık matrisinde doğruluk(*accuracy)*, *Hassasiyet(precision), Duyarlılık(sensitivity), Özgünlük(specificity), Hatırlama(recall)* değerleri dikkate alınmaktadır. Karmaşıklık matrisi gerçek değerlerin bilindiği bir test verisi üzerinde sınıflandırma modelinin doğruluğunu test etmek amacı ile kullanılan bir matristir. Çalışmada kullanılan karmaşıklık matrisi 2x2 boyutunda bir matristir. Bu matrisin boyutu yapılacak sınıflandırmaya göre uzman görüşüne dayalı olarak değişkenlik gösterebilmektedir. Bu çalışmada ikili sınıflandırma yapıldığından dolayı her değer bir sınıfa aittir veya değildir şeklinde yorumlanmaktadır. Bu nedenle 2x2 boyutundaki bir matris yeterli olmaktadır. Karmaşıklık matrisi oluşturulurken kullanılan ölçüm değerleri Tablo 3'te gösterilmektedir.

Tablo 3. Karmaşıklık Matrisi Ölçüm Değerlerinde Kullanılan Parametreler

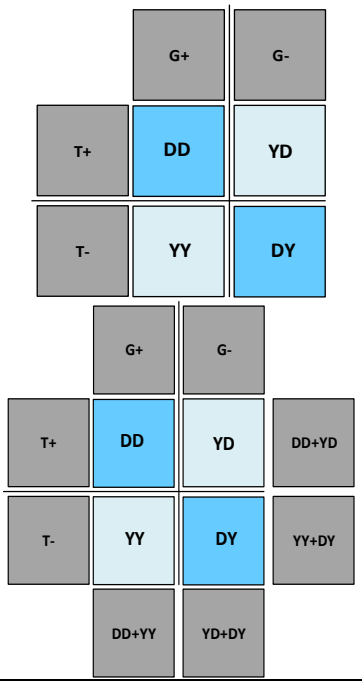
Ölçüm Değeri	Açıklama
True Positive (DD)	Gözlem olumludur ve olumlu olduğu tahmin edilmektedir.
False Negative (YY)	Gözlem olumludur fakat olumsuz olarak tahmin edilmektedir
True Negative (DY)	Gözlem olumsuzdur ve olumsuz olarak tahmin edilmektedir
False Positive (YD)	Gözlem olumsuzdur fakat olumlu olarak tahmin edilmektedir.

Karmaşıklık matrisinden elde edilen yukarıdaki değerlerin farklı matematiksel formülazasyonları sonucunda elde edilen ve sınıflandırma doğruluğunu belirlemek için kullanılan ölçüm değerleri Tablo 4'te gösterilmektedir.

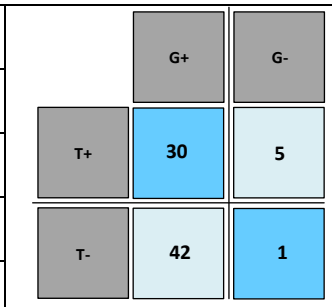
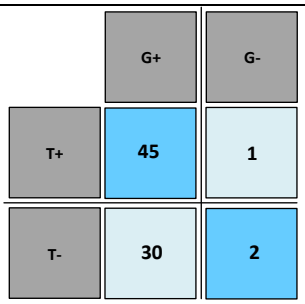
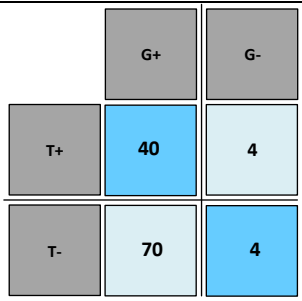
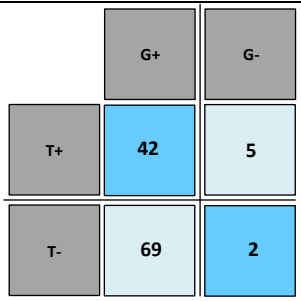
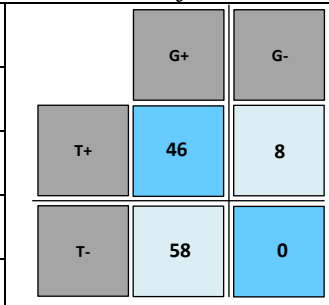
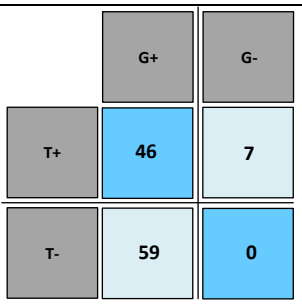
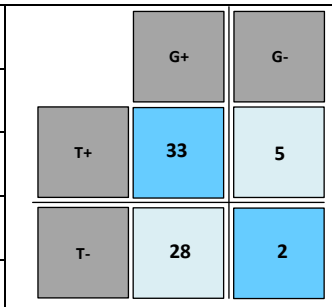
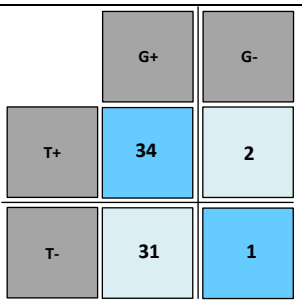
Yapılan sınıflandırma testlerinde Ekonomi-Kültür Sanat, Sağlık Siyaset, Siyaset- Teknoloji, Teknoloji-Ekonomi olmak üzere dört farklı alandan farklı boyutlarda yapısal olmayan metinler iki farklı yöntem kullanılarak sınıflandırılmıştır. Sınıflandırma soncuda elde edilen karmaşıklık matrisi ölçüm değerleri Tablo 5' te gösterilmektedir.

Yapısal olmayan veriler üzerinde yapılan sınıflandırma sonuçları ile bu sonuçlara ait karmaşıklık matrisi ölçüm değerleri ve *Tf * Idf* çarpımının sınıflandırmaya kazandırdığı doğruluk sonuçları Tablo 5' te gösterilmektedir. Yapılan işlemin sonucunda sınıflandırma doğruluğunda kayda değer bir artış olduğu gözlenmektedir.

Tablo 4. Karmaşıklık Matrisi Ölçüm Değerleri

Karmaşıklık Matrisi	Ölçüm Değeri	Formül	Tanım
	Doğruluk(Acc)	$\frac{DD+DY}{TOPLAM}$	Sınıflandırmanın doğruluk oranını belirler.
	Hassasiyet(Prec)	$\frac{DD}{DD+YD}$	Pozitif olarak tahmin edilen bir durumdaki başarıyı gösteren durum
	Duyarlılık(Sn)	$\frac{DD}{DD+YY}$	Doğru olarak belirlenmiş gerçek doğruların oranını belirler.
	Özgünlük(Sp)	$\frac{DY}{YD+DY}$	Doğru olarak belirlenmiş gerçek negatiflerinin oranını belirler
	Hatırlama Oranı (Rc)	$\frac{DD}{DD+YY}$	Pozitif durumların ne kadar başarılı tahmin edildiğini gösterir.
	F Puanı(Fs)	$2*\frac{Prec*Rc}{Prec+Rc}$	Önerilen modelin Doğruluğunun bir ölçüsüdür. En iyi durumda 1 en kötü durumda 0 değerini alır.

Tablo 5. Modelin Karşılaştırmaları ve Performans Farkları

	Ortak Kelime Sayıları Baz Alınarak Sınıflandırma Ölçümleri			Ortak Kelime Sayılarına ek olarak *Tf * Idf* çarpımının eklenmesi sonucunda oluşan Sınıflandırma Ölçümleri		
Ekonomi –Kültür Sanat	Acc	**0.9230**		Acc	**0.9615**	
	Prec	0.857		Prec	0.9782	
	Sn/Rc	0.9677		Sn/Rc	0.9574	
	Sp	0.8936		Sp	0.9677	
	Fs	0.9090		Fs	0.9677	
	Sınıflandırma Doğruluğu farkı: %3,85					
Sağlık-Siyaset	Acc	**0.9322**		Acc	**0.9406**	
	Pre	0.9090		Pre	0.8936	
	Sn/Rc	0.9090		Sn/Rc	0.9545	
	Sp	0.9459		Sp	0.9324	
	Fs	0.9090		Fs	0.9230	
	Sınıflandırma Doğruluğu farkı: %0,8					
Siyaset-Tekknoloji	Acc	**0.9285**		Acc	**0.9375**	
	Prec	0.8518		Prec	0.8679	
	Sn/Rc	1.0		Sn/Rc	1.0	
	Sp	0.8787		Sp	0.8939	
	Fs	0.92		Fs	0.9292	
	Sınıflandırma Doğruluğu farkı: %0.9					
Teknoloji - Ekonomi	Acc	**0.8970**		Acc	**0.9558**	
	Prec	0.8684		Prec	0.9444	
	Sn/Rc	0.9428		Sn/Rc	0.9714	
	Sp	0.8484		Sp	0.9393	
	Fs	0.9041		Fs	0.9577	
	Sınıflandırma Doğruluğu farkı: %5,88					

5. Sonuçlar ve Öneriler

Yapılan çalışmada yapısal olmayan verilerden oluşan TTC-3600 veri seti ile Spektral Çizge Bölmeleme yöntemi kullanılarak metin sınıflandırma yapılmıştır. Metin sınıflandırma işleminde ön işleme aracı olarak kullanılan KUSH adını verdiğimiz ve .Net ortamında yazılan bir araç geliştirildi. İlk etapta sadece cümleler arasında bulunan ortak kelime sayısını ilişki ağırlığı olarak kabul edilmekte ve sınıflandırma yapılmaktadır. Sonrasında sınıflandırma doğruluğunu arttıracağı düşünülerek cümlelerin içerdikleri ortak kelimelerin terim frekans değerleri ve ters terim frekans değerleri hesaplanarak cümeleler arası ilişki ağırlığına eklenmektedir. Böylelikle cümlelerin içerdikleri ortak kelimeler niceliklerinin yanında nitelikleri ile de temsil edilmektedirler. Yani *Tf * Idf* çarpım değerleri ile ortak kelimelerin temsil edildikleri alana ait ne kadar bilgi taşıdıkları da cümleler arsındaki ilişki ağırlılığına eklenmektedir. Çalışmanın ilk bölümünde %95 değerlerine varan yüksek sınıflandırma doğruluğu elde edilmekte ve ayrıca uygulamış olduğumuz ikinci ağırlıklandırma ile de doğruluk değerinde %1 ile %5 aralığında daha iyi performanslar elde edildiği yapılan testlerle gözlemlenmiştir. Bundan sonraki aşamalarda yapılması düşünülen çalışmalar sırasıyla çalışmanın tutarlılığını test etmek amacı ile bilimsel araştırmalarda çoğunlukla kullanılan uluslararası veri setleri üzerinde test yapmak ve geliştirilen KUSH aracını birden fazla dilde ön işleme yapabilecek kabiliyetler kazandırma olarak düşünülmektedir.

Kaynaklar

[1] Mikhina E.K., Trifalenkov V.I. 2018. Text clustering as graph community detection. Procedia computer science, 123: 271-277.

[2] Aydemir, E. Türkçe Köşe Yazılarında Yapay Sinir Ağlarıyla Yazar ve Gazete Tahmin Etme. DÜMF Mühendislik Dergisi, 10 (1): 45-56.

[3] Le Q., Mikolov T. 2014. Distributed representations of sentences and documents. In International conference on machine learning (pp. 1188-1196).

[4] Jiang C., Coenen F., Sanderson R., Zito M. 2010. Text classification using graph mining-based feature extraction. In Research and Development in Intelligent Systems XXVI (pp. 21-34). Springer, London.

[5] Wan X. 2007. A novel document similarity measure based on earth mover's distance. Information Sciences, 177 (18): 3718-3730.

[6] Zhao G., Luo B., Tang J., Ma J. 2007. Using eigen-decomposition method for weighted graph matching. In International Conference on Intelligent Computing (pp. 1283-1294). Springer, Berlin, Heidelberg.

[7] Ma T., Shao W., Hao Y., Cao J. 2018. Graph classification based on graph set reconstruction and graph kernel feature reduction. Neurocomputing, 296: 33-45.

[8] Slininger B. 2013. Fiedlers Theory of Spectral Graph Partitioning.

[9] Kılınç D. 2016. The Effect of Ensemble Learning Models on Turkish Text Classification. Celal Bayar Üniversitesi Fen Bilimleri Dergisi, 12 (2): 215-220 .

[10] Kılınç D., Özçift A., Bozyigit F., Yıldırım P., Yücalar F., Borandag E. 2017. TTC-3600: A new benchmark dataset for Turkish text categorization. Journal of Information Science, 43 (2): 174-185.

[11] Shang T., Xia X., Zheng J. 2018. MIME-KNN: Improve KNN Classifier Performance Include Classification Accuracy and Time Consumption. DEStech Transactions on Computer Science and Engineering, (csse).

[12] Barrett W., Francis A., Webb B. 2017. Equitable decompositions of graphs with symmetries. Linear Algebra and its Applications, 513: 409-434.

[13] Pothen A., Simon H.D., Liou K.P. 1990. Partitioning sparse matrices with eigenvectors of graphs. SIAM journal on matrix analysis and applications, 11 (3): 430-452.

[14] Naumov M., Moon T. 2016. Parallel spectral graph partitioning. NVIDIA Technical Report, NVR-2016-001.

[15] Wang Q., Guo S., Hu J., Yang Y. 2018. Spectral partitioning and fuzzy C-means based clustering algorithm for big data wireless sensor networks. EURASIP Journal on Wireless Communications and Networking, 2018 (1): 54.

[16] Alupoaie S., Cunningham P. 2013. Using tf-idf as an edge weighting scheme in user-object bipartite networks. arXiv preprint arXiv:1308.6118.

[17] Robertson S. 2004. Understanding inverse document frequency: on theoretical arguments for IDF. Journal of documentation, 60 (5): 503-520.
[18] Kim D., Seo D., Cho S., Kang P. 2019. Multi-co-training for document classification using various document representations: TF–IDF, LDA, and Doc2Vec. Information Sciences, 477: 15-29.
[19] Bapat R.B. 2010. Graphs and matrices (Vol. 27). London: Springer.
[20] Barrat A., Barthelemy M., Vespignani A. 2008. Dynamical processes on complex networks. Cambridge university press.
[21] Dhillon I.S. 2001. Co-clustering documents and words using bipartite spectral graph partitioning. In Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 269-274). ACM..
[22] Chung F.R. 1996. Lectures on spectral graph theory. CBMS Lectures, Fresno, 6: 17-21.