text2arff: Türkçe Metinler İçin Özellik Çıkarım Yazılımı

text2arff: Automatic Feature Extraction Software for Turkish Texts

M.Fatih AMASYALI [1], Feruz DAVLETOV[1], Arslan TORAYEW[1], Ümit ÇİFTÇİ [1]

1. Bilgisayar Mühendisliği Bölümü
Yıldız Teknik Üniversitesi
mfatih@ce.yildiz.edu.tr, fdavletov@gmail.com, atorayew@gmail.com, umitciftci65@gmail.com

Özetçe

Bir kategoriye ait bir metni başka bir kategoriye ait metinden hangi özellikleri ayırt eder? Bu soruya çeşitli cevaplar vererek, metin dosyalarının otomatik sınıflandırılması için uzun zamandır çalışmalar sürmektedir. Bu çalışmada Türkçe metinlerin özelliklerinin çıkarılması işlemini yapan bir yazılım tanıtılmıştır. Yazılım metinden çıkarılacak özellikler olarak geniş seçenekler sunmaktadır. Bunlara örnek olarak; cümle, kelime, ek sayıları, ngramlar, kelimeler, kelime grupları ve saklı anlam indeksi verilebilir. Yazılım metinlere ait özellikleri çıkardıktan sonra elde edilen verilerin kolayca sınıflandırma ya da kümeleme işlemlerinde kullanılabilmesi için WEKA kütüphanesinin dosya formatında (arff) kaydetmektedir.

Abstract

Which features are the most important for the text classification tasks? In the automatic text categorization area, several studies seek answers to this question. In this paper, a feature extraction tool for Turkish texts (Text2arff) is presented. The toolbox automatically extracts several features such as the frequencies of the words and ngrams, word clustering, Latent semantic indexing etc. The features of the texts are saved in arff (WEKA) file format. The arff files can be used easily with WEKA machine learning library.

1. Giriş

Büyük bilgi yığınlarının oluşması, bu yığınların içindeki bilgilere erişimi güçleştirmektedir. Bir arama kataloğuna sahip olmayan bir kütüphane kullanıcılarına ne kadar yararlı olabilir? Metinlerin otomatik olarak sınıflandırılmasında ya da kümelenmesinde metinlerin hangi özellikleri kullanılmalıdır? Literatürde metinlerin hangi özelliklerinin kullanılabileceğine ilişkin çeşitli öneriler bulunmaktadır [1]. Kullanılması gereken özellikler kategori türlerine göre bile farklılık gösterebilir. Kategoriler metinlerin yazarlarıysa başka özellikler, metinlerin konularıysa başka özellikler daha ayırt edici olacaktır.

Farklı metin sınıflandırma / kümeleme problemlerine sahip kullanıcılara hitap eden bir yazılım geliştirme fikrinden hareketle başlayan çalışmamız sonucunda geliştirilen yazılım bu çalışmada tanıtılmıştır.

2. Yazılımın Kullanımı

Programı çalıştıran kullanıcı öncelikle özelliklerinin çıkartılmasını istediği metinlerin yer aldığı klasörü seçer. Metinlerin gerçek kategorileri biliniyorsa, aynı kategoriye sahip metinler tek bir klasörde yer almalı, sisteme her biri içinde farklı kategorilerden metinler içeren klasörleri içeren ana klasör söylenmelidir.

İşlenecek metinlerin belirlenmesinden sonra, çıkarılması istenen özellikler seçilmelidir. Yazılım bu konuda kullanıcıya geniş seçenekler sunmaktadır. 3.bölümde bu özellikler ayrıntılı bir şekilde anlatılmıştır.

Özellik seçimi de yapıldıktan sonra işle butonun basılarak özellik çıkarım işleminin tamamlanması beklenir. Bekleme süresi seçilen özelliklere ve işlenecek metinlerin boyut ve sayılarına göre farklılık göstermektedir. O anda hangi işlemlerin yapıldığı ya da işlemin bittiği ara yüzdeki Bilgi bölümünden takip edilebilmektedir.

Yazılım, özellik çıkarım işlemini sonunda kullanıcının seçtiği özelliklere göre farklı türde ve sayıda dosyalar üretmektedir. Üretilecek dosyaların yerleri ve adları ara yüzden belirlenebilmektedir. Ancak programın temel çıktısı arff formatındaki dosyadır. Metne ait kullanıcının seçtiği tüm özellikler bu dosyada yer almaktadır. Çıkarılan özelliklerin isimleri ise bir txt dosyası olarak kaydedilmektedir. Dosyanın her bir satırında her bir metne ait özellikler bulunmaktadır. Şekil 1'de örnek bir arff dosyası verilmiştir.

```
@ATTRIBUTE kelime_basina_harf_sayisi NUMERIC
@ATTRIBUTE cumle_basina_kelime_sayisi  NUMERIC
@ATTRIBUTE yazari {omer_seyfettin, peyami_sefa}
@DATA
5.1 3.5 omer_seyfettin
4.9 6.0 peyami_sefa
4.7 7.2 omer_seyfettin
4.6 4.1 peyami_sefa
5.0 9.6 peyami_sefa
```

Şekil 1: Örnek arff dosyası.

Şekil 1'deki örnek arff dosyasında 5 adet metne ait 2'şer özellik ve metinlerin ait oldukları kategoriler verilmiştir. Ömer Seyfettin'e ait 2 metne, Peyami Sefa'ya ait 3 metne ilişkin kelime ve harf sayılarıyla ilgili özellikler belirlenmiştir. Öncelikle metinlerin özelliklerinin tanıtımı yapılmış daha sonra @DATA satırından başlayarak metinlerin bu özelliklere ait değerleri verilmiştir. Her bir satırda bir metne ait özellikler verilmiştir. Metinlerin son özellikleri metnin yazarıdır (metnin ait olduğu sınıf/kategori). Metinlerin kategori bilgileri, içinde bulundukları klasörlerin isimleriyle belirlenmektedir.

Arff formatının seçiliş sebebi anlaşılır olması, kolayca güncellenebilir bir metin dosyası olması ve en önemlisi birçok makine öğrenmesi algoritmasını içeren WEKA [2] yazılımını bu formatı direkt kullanabilmesidir. Dolayısıyla geliştirdiğimiz

978-1-4244-9671-6/10/$26.00 ©2010 IEEE

bu yazılım ve WEKA birlikte kullanıldığında otomatik bir metin sınıflandırma ya da gruplama sistemi ortaya çıkmaktadır.

3. Metinlerin Özellikleri

Programın ara yüzünde görülen bölümlerin her birinde metinlere ait farklı özellikler yer almaktadır. Her bir özellik bölümlerdeki işaretleme alanlarının seçilmesiyle aktif hale gelmektedir.

3.1. Kavramlar

Kullanıcıya Kelime Kökleri, Kelime Türleri, Ngramlar, Fonksiyonel Kelimeler, Kelime Ekleri özellik grupları için *TF, TFIDF, TF ve TDIDF* olmak üzere 3 farklı seçenek sunulmuştur. Eşitlik 1'de *t* özelliğinin *d* metnindeki *TFIDF* değeri verilmiştir.

$$TFIDF(t,d) = TF(t,d) \cdot \log \frac{|Tr|}{|Tr(t)|} \quad (1)$$

Eşitlik 1'de *TF(t,d)*, *t* özelliğinin, *d* metninde geçme sayısıdır. *Tr*, toplam metin sayısını, *Tr(t)* içinde en az bir kere *t* özelliği geçen metin sayısını göstermektedir.

Örnek olarak *t* özelliği "gr" 2gramı ise, *TF(t,d)*, "gr" ifadesinin *d* metninde geçiş sayısı olacaktır. *T* özelliği "isim" kelime türü ise *TF(t,d); d* metninin içerdiği isim türündeki kelime sayısı olacaktır. Kullanıcı *TF ve TFIDF*'i seçerse o özelliğe ait hem *TF* hem de *TFIDF* hesaplanarak dosyaya yazılır.

3.2. Sayılar

Metindeki kelime, cümle, harf, ek, cümle başına kelime, cümle başına harf, cümle başına ek, kelime başına harf, kelime başına ek, devrik cümle ve noktalama işaretleri sayısını bulduran 19 adet özellikten oluşan özellik grubudur. Bu gruptaki özelliklerin isimleri txt dosyasına isimleriyle yazılmaktadır.

3.3. Kelime Kökleri

Metinlerin ifade edilmesinde en yaygın kullanılan özellik grubudur. Metinler içerdikleri kelimelerle ifade edilirler. Metinlerin boyutu, tüm metinlerde en az bir kere geçen farklı kelime sayısıdır. Kelimelerin kendilerini kullanmak Türkçe gibi eklemeli dillerde farklı kelime sayısını çok fazla arttırmaktadır. "Ağaç", "ağaçlı", "ağaçlandırma", "ağaçlandırılmış", "ağaçlık" kelimelerinin hepsinin farklı kelimeler ve dolayısıyla farklı özellikler olarak ele alınması buna örnek verilebilir.

Probleme çözüm olarak literatürde kelimelerin kendileri yerine sadece kelime köklerinin kullanılması önerilmektedir. Bu sayede hem aynı anlama işaret eden kelimelerin birleştirilmesi (böylelikle metinler arası benzerliğin daha iyi ifade edilmesi sağlanmaktadır), hem de özellik boyutunun azaltılmasıyla işlem karmaşıklığının ve gürültü veri olasılığının azaltılması sağlanmaktadır. Kelimelerin köklerinin bulunmasında Zemberek [3] kütüphanesinden faydalanılmıştır.

Bu gruptaki özelliklerin isimleri txt dosyasına "KOK" ve frekans türü önekleri eklenerek yazılmaktadır.

3.4. Kelime Türleri

Metinlerdeki geçen kelimelerin türlerini bulduran özellik grubudur. Kelimelerin türlerinin bulunmasında da Zemberek kütüphanesinden faydalanılmıştır. Toplam 15 tane kelime türü (isim, sıfat, zamir, edat vb.) vardır. Bu gruptaki özelliklerin isimleri txt dosyasına "TIP" ve frekans türü önekleri eklenerek yazılmaktadır.

3.5. Ngramlar

Bu özellik grubunda metinler içerdikleri ngramları ile ifade edilirler. Ngramlar N boyutlu karakter çerçeveleridir. Örneğin "gitmek" metni için:
2 gramlar: gi – it – tm – me – ek
3 gramlar: git – itm – tme – mek

Yazılımla metinlere ait 2 ya da 3 gramlar ya da her ikisi birden çıkartılabilir. Daha büyük pencere boyutuna sahip ngramların kullanılmama nedeni çok seyrek frekans matrisleri üretiyor olmalarıdır.

Bu gruptaki özelliklerin isimleri txt dosyasına "2GRAM" ya da "3GRAM" ve frekans türü önekleri eklenerek yazılmaktadır.

3.6. Fonksiyonel Kelimeler

Programla aynı klasörde bulunun fonksiyonel kelimeler dosyasında "fWords.txt" yer alan 620 kelimeye göre metinlerin ifade edilmesinde kullanılan özellik grubudur. Fonksiyonel kelimeler esasen tek başlarına çok fazla anlamları olmayan, ancak yazarların üsluplarının belirlenmesinde önemli oldukları düşünülen kelimelerdir (ve, daha, gibi, de, için vb.).

Bununla birlikte kullanıcı fonksiyonel kelimeler dosyasını istediği gibi değiştirebilir. Bu sayede kullanıcı istediği özel bir kelime grubuna göre metinleri ifade edebilir. Bu gruptaki özelliklerin isimleri txt dosyasına "FONK" ve frekans türü önekleri eklenerek yazılmaktadır.

3.7. Kelime Ekleri

Türkçe eklemeli bir dil olduğundan anlamın oluşmasında eklerin önemi büyüktür. Bu sebeple yazılımda kelimelerin aldıkları eklerin türlerine çıkarılabilmektedir. Kelimelerin eklerinin belirlenmesinde yine Zemberek kütüphanesinden faydalanılmıştır. Zemberek toplam 126 farklı ek türü çıkarmaktadır. Bu gruptaki özelliklerin isimleri txt dosyasına "EK" ve frekans türü önekleri eklenerek yazılmaktadır.

3.8. Kelime Kümeleme

Birbirine yakın anlamlı kelimeleri birleştirip tek bir küme, tek bir özellik haline getirmek fikrine dayanmaktadır [4]. Kelimelerin kendileri yerine köklerinin kullanılması fikrinde olduğu gibi hem boyut sayısı azaltılmakta hem de benzer anlama sahip kelimeler tek bir kavram olarak kullanılmaktadır. Örneğin kelimeler hayvan kavramına ait kelimeler ayrı ayrı özellik değil, tek bir özellik olarak metinlerin ifadesinde kullanılacaktır. Küme "arff" dosyasında tek bir özellik olarak yazılır. Bir kelime kümesinin bir dokümandaki geçiş sayısı, küme içindeki her bir kelimenin o dokümandaki geçiş sayıları toplamıdır. Yazılım 3 kümeleme algoritmasını desteklemektedir.

1.Hiyerarşik: Başlangıçta her bir örneğin ayrı bir küme olduğu, her bir adımda birbirine en çok benzeyen iki kümenin birleştirildiği algoritmadır [5]. Kümelerin birbirine

benzerliklerinin belirlenmesinde kullanıcıya 3 farklı seçenek sunulmuştur. A) En Yakın: İki küme birbirine en yakın elemanları kadar yakındır. B) Ortalama: Yakınlık, iki kümenin her bir elemanının diğer kümenin her bir elemanıyla arasındaki mesafelerin ortalamasıdır. C) En uzak: İki küme birbirine en uzak elemanları kadar yakındır.

2.K-means: Veri dağılımını en iyi temsil edebilecek küme merkezlerinin bulunması fikrine dayanır [6]. Algoritma rasgele atanmış küme merkezleri ile başlar. İterasyonlar ilerledikçe küme merkezlerinin kendi kümelerindeki elemanlara olan ortalama mesafesi azalır.

3.SOM: K-means ile aynı fikre dayanır ancak önerdiği çözüm biraz farklıdır. Algoritmada başlangıçta rasgele atanan küme merkezleri buna ek olarak birbirlerine rasgele bağlanmışlardır. Her bir küme merkezinin hareketinde ona bağlı olan diğer küme merkezleri de hareket eder. Bu sayede hem verileri ifade edecek küme merkezleri hem de verinin topolojisi ortaya çıkarılmaktadır [7].

Kelimeler kümelemede, ya dokümanlarda geçme sayılarıyla, ya TFIDFlerle ya da diğer kelimelerle birlikte geçme sayılarıyla (birlikte geçme matrisi) ifade edilebilmektedir. Kullanıcı bu 3 özellik türünden istediğin seçebilmektedir. Bununla birlikte kullanıcı, kelimelerin indirgenmesini istediği küme sayısını belirleyebilmektedir. K-means ve SOM algoritmaları için seçilen küme sayısı en fazla anlamındadır. Bunun sebebi başlangıçtaki rasgele atamadan kaynaklanan ya da daha sonradan oluşan boş kümelerin oluşmasıdır. Ayrıca yazılımda K-means ve SOM algoritmaları için kullanıcı iterasyon (epoch) sayısını da belirleyebilmektedir.

Kümeleme yapıldıktan sonra her bir kelime kümesinin içinde hangi kelimelerin olduğu otomatik oluşturulan txt dosyasına yazılmaktadır.

3.9. Birlikte Geçme Matrisi

Bu özellik seçildiğinde Birlikte Geçme (Cooccurance) matrisi oluşturulur ve kullanıcının ara yüzde belirttiği yere "coocurance.txt" yazılır. Aynı zamanda simetrik olan matrisin satır ve sütunlarını ifade eden kelimeler de yine kullanıcının belirlediği başka bir dosyaya "kelimeler.txt" yazılır.

Birlikte geçme matrisinin i,j hücresinin değeri, i.kelime ile j. kelimenin metin kütüphanesi içinde çeşitli boyuttaki çerçevelerde birlikte bulunma sayılarını ifade eder. Kullanıcı seçebileceği pencere boyutları 2, 3, 5, aynı dosya içinde ve aynı klasör içinde olmak üzere 5 adettir. Pencere boyutunun bir sayı olması iki kelimenin tüm metinlerde o sayı büyüklüğündeki bir pencere içinde kaç kez birlikte bulunduğunu göstermektedir. Aynı metin ya da klasör içinde olması ise sırasıyla iki kelimenin birlikte bulundukları metin ya da klasör (uygulamamızda klasörler sınıfları temsil etmektedir) sayısını göstermektedir.

3.10. Kelime Filtreleme

Metinlerde çok fazla kullanılan kelimelerin ayır ediciliği azdır. Buna karşın çok az kullanılan kelimeler de gereksiz yere boyut sayısını arttırabilirler. Bu nedenlerle kullanıcı metinlerdeki tüm kelimeler yerine metinlerdeki geçiş sayısı belirli bir aralıkta olan kelimeleri kullanılmak istenirse bu filtre özelliğinden faydalanabilir. Kelimeler kullanıcının belirlediği minimum ve maksimum geçiş sayılarına göre filtrelenir. Bu özellik seçilirse filtreleme işlemi diğer tüm özellik gruplarındaki işlemlerden önce yapıldığı için kelimeleri kullanan tüm özellik gruplarının hesaplanmasına etki eder. Diğer bir deyişle, kelimeleri kullanan tüm özellik gruplarında sadece filtrelenen kelimeler kullanılır.

3.11. Saklı Anlam İndeksleme

Literatürde en yaygın kullanıma sahip özelliklerden biridir. Doküman matrisi (A) üzerinde Tekil Değer Ayrıştırma (Singular Value Decomposition) yaparak metinlerin ifadesinde kullanılan boyut sayısını azaltır [8]. Bunun için A matrisi Eşitlik 2'ye göre özdeğer-özvektör çarpanlarına ayrılır.

$$A_{m\times n} = U_{m\times r}.S_{r\times r}.V^{T}_{n\times r} \quad (2)$$

Bu işlem sonunda S matrisinde, özvektörlerin özdeğerleri diagonalde büyükten küçüğe sıralanır. Kullanıcının seçtiği boyut (k) kadarı işleme alınarak, A matrisi m*n'lik bir matristen (m→metin sayısı, n→ farklı kelime sayısı), m*k'lık (k→ metinlerin yeni boyut sayısı) bir matrise dönüşmüş olur.

Bu gruptaki özelliklerin isimleri txt dosyasına "LSI" öneki eklenerek yazılmaktadır. Tekil Değer Ayrıştırma için LingPipe [9] kullanılmıştır.

3.12. Anlamsal Uzay

Bu özellik grubunda metinlerin anlamsal bir uzaydaki koordinatları bulunur [10]. Metinlerin koordinatları içerdikleri kelimelerin koordinatlarının ortalaması alınarak bulunur. Harris [11], iki kelimenin birlikte geçtiği doküman / cümle / çerçeve sayısının iki kelimenin anlamca benzerliğiyle doğru orantılı olduğunu öne sürmüştür. Bu yaklaşım kullanılarak kelimelerin birbirlerine anlamca benzerliklerine göre konumlandırıldıkları anlamsal uzaydaki koordinatları Birlikte Geçme Matrisi kullanılarak bulunmaktadır. Benzerlikler matrisinden Çok Boyutlu Ölçekleme (Multi Dimensional Scaling) [12] ile koordinatlar elde edilmektedir. Kullanıcı, benzerliklerin ölçülmesinde 5 farklı şekilde oluşturulmuş Birlikte Geçme Matrisi kullanabilmektedir.

Metinlerin koordinatları arff dosyasına, kelimelerin koordinatları otomatik oluşturulan "kelimeMDS.txt" dosyasına yazılmaktadır. Bulunan kelime dolayısıyla metin koordinatlarının boyutu da, diğer bir deyişle kelimelerin ve metinlerin kaç boyutlu bir uzayda ifade edilecekleri, kullanıcı tarafından belirlenebilmektedir. Bu gruptaki özelliklerin isimleri txt dosyasına "MDS" öneki eklenerek yazılmaktadır. Çok Boyutlu Ölçekleme için MDSJ Kütüphanesi [13] kullanılmıştır.

4. Deneysel Sonuçlar

Yazılımın çalışması hakkında bir fikir vermesi amacıyla, hangi özellik gruplarının sınıflandırmada daha başarılı oldukları bir veri kümesi üzerinde çalıştırılarak ölçülmüştür. Kullanılan veri kümesinde 10 yazarın her birine ait 10'ar yazı bulunmaktadır. Diğer bir deyişle 100 örneği 10 sınıfı olan bir sınıflandırma problemi çözülmek istenmiştir. Kullanılan algoritmalar karar ağaçları (C4.5), en yakın komşu (1NN) ve karar destek makineleri (SVM)'dir. Algoritmaların performansları 10'lu çapraz geçerleme ile ölçülmüştür. Tablo 1'de metinlerin ifade edilmesinde kullanılan özellik grupları, parametreleri ve bunlara karşılık oluşturulan arff dosyalarıyla WEKA'da yapılan sınıflandırma denemelerinin sonuçları yüzdelik başarı türünden verilmiştir. Problemde eşit sayıda örnekler içeren 9 sınıfa ait metinler olduğundan rasgele başarı

oranı %11.1'dir. Frekansların ifadesini kullanan tüm özellik gruplarında TF kullanılmıştır.

Tablo 1: Özellik Gruplarının Metnin Yazarını Belirlemedeki Performans Karşılaştırılmaları

Kullanılan Özellik Grubu	**Parametre**	**C4.5**	**1NN**	**SVM**
Sayılar	19 boyut	58.9	60	57.8
Kelime Kökleri	Tüm kelimeler, 4126 boyut	61.1	44.4	94.4
Kelime Kökleri	Frekansı 5-200 arası kelimeler, 1456 boyut	61.1	28.9	90
Kelime Ekleri	111 boyut	43.3	51.1	61.1
Kelime Türleri	15 boyut	38.9	33.3	38.9
Ngramlar	3 gramlar, 12957 boyut	70	40	94.4
Ngramlar	2 gramlar, 2349 boyut	84.4	51.1	91.1
Fonksiyonel Kelimeler	533 boyut	41.1	34.4	76.7
Kelime kümeleme	Küme sayısı=100, Alg: Kmeans	40	34.4	72.2
Kelime kümeleme	Küme sayısı=100, Alg: SOM	28.9	42.2	65.6
Kelime kümeleme	Küme sayısı=100, Alg: Hiyerarşik- En yakın	30	23.3	34.4
Anlamsal Uzay	Pencere boyutu: aynı metin, frekansı 5-200 kelimeler, 100 boyut	50	50	82.2
Hepsi	17647 boyut	63.3	55.6	97.8
Hepsi	CFS ile seçilen 57 boyut	75.6	88.9	90

Tablo 1 incelendiğinde en başarılı özellik gruplarının kelime kökleri, kelime gruplama ve ngramlar olduğu görülmektedir. Tüm özellik grupları birlikte kullanıldığında ise en yüksek başarı elde edilmiştir. Ayrıca, tüm (17657 adet) özelliklerden CFS [14] ile seçilen 57 özellik incelendiğinde; sayılar grubundan 5, kelime türünden 1, kelime köklerinden 3, kelime kümelerinden 4, 2gramlardan 14, 3gramlardan 22, kelime eklerinden 4, anlamsal uzaydan 4 özellik olduğu görülmüştür. Bu sonuçlar, özellik gruplarının birbirini destekler nitelikte olduğunu göstermektedir.

Sadece bir veri tabanıyla yapılan bu deneme, özellik gruplarının birbirlerine göre başarısını genelleştirilmek için yeterli değildir. Başka tür kategorilere (metnin konusu, yazarın cinsiyeti vb.) sahip bir veri tabanında başka özellik grupları ön plana çıkabilir. Yazılımda çeşitli amaçlara hizmet eden uygulamalarda kullanılabilecek çok sayıda özellik grubunu bulunmasının sebebi de budur.

5. Sonuç

Türkçe bugün dünyada milyonlarca insan tarafından konuşulan bir dildir. Buna rağmen üzerinde henüz yeterince çalışma yapılmış, analiz edilmesi için yazılımlar üretilmiş bir dil değildir. Metin işleme yazılımlarına edebiyatçılar, arşivciler, dilbilimciler gibi çeşitli meslek gruplarından ihtiyaç duyulmaktadır. Bu çalışmada da bu ihtiyaçtan yola çıkılarak, metinlerin ifade edilmesinde en çok kullanılan özelliklerin otomatik çıkarımı ve özelliklerin WEKA kütüphanesiyle kullanıma hazır bir formatta kaydedilmesi gerçekleştirilmiştir. Kullanıcıya 9 farklı özellik grubu ve her birine ait çeşitli sayıda ayarlanabilir parametre sunulmuştur.

Yazılımla elde edilen özelliklerin başarılarını ve birbirlerinden farklarını göstermek amacıyla metinlerin yazarlarının bulunmasını amaçlayan örnek bir metin veritabanı üzerinde metinlere ait çeşitli özellik grupları çıkarılmış ve sınıflandırma performansları ölçülmüştür.

Yazılıma http://www.kemik.yildiz.edu.tr/?id=29 adresinden ulaşılabilir. Bu yazılımla oluşturulmuş çeşitli veri kümelerine ise http://www.kemik.yildiz.edu.tr/?id=28 adresinden erişilmektedir.

Yazılıma eklenmesi düşünülen özellikler olarak; kelime ngramları, TF ve TFIDF'e ek olarak yeni kelime frekansı ifade etme yöntemleri, kelimelerin anlamsal ağlar kullanılarak kümelenmesi sayılabilir.

Yazılımın geliştirilmesine yorumlarıyla katkıda bulunan Kemik Doğal Dil İşleme grubuna (http://www.kemik.yildiz.edu.tr) teşekkür ederiz.

6. Kaynakça

[1] Ciya L., Shamim A., Paul D., "Feature Preparation in Text Categorization", *Oracle Text Selected Papers and Presentations ,2001.*

[2] Ian H. Witten and Eibe Frank, *"Data Mining: Practical machine learning tools and techniques"*, 2nd Edition, Morgan Kaufmann, San Francisco, 2005.

[3] http://code.google.com/p/zemberek/

[4] Bekkerman R., Ran El-Yaniv, Naftali T., Yoad W., "Distributional Word Clusters vs. Words for Text Categorization", *Journal of Machine Learning Research*, *pp.1-48, 2002.*

[5] Ethem Alpaydın, "*Introduction to Machine Learning*", The MIT Press, p. 146-148, 2004.

[6] Alpaydın, p. 135-139.

[7] Alpaydın, p. 282-283.

[8] B. Özel, "Küresel k-Ortalamalı Gruplama Yöntemi ile Metinlerin ve Terimlerin Saklı Anlam İndekslenmeleri", *ASYU, Istanbul, Turkey, p. 223-227, 2004.*

[9] Alias-i. 2008. LingPipe 3.8.1. http://alias-i.com/lingpipe

[10] M.Fatih Amasyalı, Aytunç Beken, "Türkçe Kelimelerin Anlamsal Benzerliklerinin Ölçülmesi ve Metin Sınıflandırmada Kullanılması", *SIU 2009.*

[11] Haris Zelig S., "*Mathematical structures of language*", Wiley, pp.12, 1968.

[12] Alpaydın, p. 121-124.

[13] Multidimensional Scaling for Java, University of Konstanz, Department of Computer & Information Science, Algorithmics Group, http://www.inf.uni-konstanz.de/algo/software/mdsj/

[14] Hall, M. A., *"Correlation-based Feature Selection for Machine Learning"*, Doktora Tezi, Hamilton, NZ: Waikato University, Department of Computer Science, 1998.