METİN SINIFLANDIRMADA SEZGİSEL ARAMA ALGORİTMALARININ PERFORMANS ANALİZİ

Ahmet HALTAŞ[1], Ahmet ALKAN[2], Mustafa KARABULUT[1]
[1] Gaziantep Üniversitesi, Teknik Bilimler Meslek Yüksekokulu, Gaziantep
[2] Kahramanmaraş Sütçü İmam Üniversitesi, Elektrik-Elektronik Müh. Bölümü, Kahramanmaraş
haltas@gantep.edu.tr, aalkan@ksu.edu.tr, mkarabulut@gmail.com

(Geliş/Received: 08.12.2014; Kabul/Accepted: 23.06.2015)

ÖZET

Metin sınıflandırma problemlerinde en büyük sorun, veri uzayının büyük boyutta olması ve başarı oranını düşürmesidir. Sezgisel arama algoritmaları literatürde pek çok alanda kapsamlı bir şekilde kullanılıyor olmalarına rağmen metin sınıflandırma alanında yaygın olarak kullanılmamaktadır. Bunun en önemli sebebi, bu algoritmaların özellik seçimi için kullanıldığında oldukça çok vakit ve hesaplama gücüne ihtiyaç duymalarıdır. Bu çalışmada bu algoritmaları metin sınıflandırmada kullanabilecek bir yöntem benimsenmiş ve popüler dört sezgisel arama algoritması (Genetik Arama, Parçacık Sürü Optimizasyon Arama, Evrimsel Arama, TABU Arama) bu amaçla test edilmiştir. Elde edilen sonuçlara göre, bahsi geçen algoritmalar özellik seçimi amaçlı kullanılarak metin sınıflandırma performansını artırmaktadırlar. Az da olsa TABU arama algoritması diğerlerine göre daha iyi sonuç vermiştir.

Anahtar Sözcükler: Metin sınıflandırma, sezgisel algoritma, özellik seçimi

PERFORMANCE ANALYSIS OF HEURISTIC SEARCH ALGORITHMS IN TEXT CLASSIFICATION

ABSTRACT

One of the most important problems in text categorization tasks is that the data space is very high dimensional which significantly diminishes the classification performance. Although, heuristic search algorithms are broadly used in many fields in the literature, they are not widely used in text categorization field. One important reason behind this fact is that these algorithms require high computational power and time to process the data for attribute selection purpose. In this study, a method to utilize such algorithms as a part of text categorization task is adopted and four popular heuristic search algorithms (Genetic Algorithm, Particle Swarm Optimization, Evolutionary Search and TABU Search) are tested. Obtained results show that heuristic search algorithms can be used effectively to increase the classification performance. Also, TABU algorithm has shown a slight performance advantage over the others.

Keywords: Text classification, heuristic algorithm, feature selection

1. GİRİŞ (INTRODUCTION)

Günümüz teknolojinin gelişmesi ve internetin yaygınlaşması, elektronik ortamda oluşturulan belge sayısının artmasına sebep olmuştur. Belge sayının artması, faydaları ile birlikte bazı sorunlarda ortaya çıkarmıştır. Metin yığınları içinde bilgiler kaybolurken, değerli bilgiye ulaşmak için ilgili dokümanların içeriğinin tanımlanması ve işlenmesi yani sınıflandırılmasına ihtiyaç duyulmuştur. Metin sınıflandırma, belgeleri daha önceden belirlenmiş kategorilere ayırma işlemi olarak tanımlanır [1]. Çok sayıda metin arasından analizler sonucu bilgiyi ortaya çıkartan metin madenciliği 1960'lı yıllarda başlamış, 2000'li yıllardan sonra daha da önem kazanmıştır [2].

Metin sınıflandırmanın birçok farklı uygulama alanı bulunmaktadır. Örneğin metin süzme (text filtering), istenmeyen e-posta (spam) filtrelenmesi, metinlerden yazarın ve metin dilinin tanınması, kategorilerin insan

eliyle yapıldığı kütüphane organizasyonu gibi ortamlarda, kategori atanmasında yardımcı olmak gibi birçok uygulama metin sınıflandırma uygulamalarıdır.

Sınıflandırma veya kategorizasyon işlemi sürecinde, metinlerdeki kelime kökleri alınarak tüm metinlerden kelime torbası (Bag-of-Words) oluşturulur. Kelime torbasında tüm metinlerin kelime kökleri özellik sayısı olarak gösterildiğinden metin, yüksek boyutta özellik vektörü ile gösterilir. Metin sınıflandırma problemlerinde en büyük sorun, veri uzayının büyük boyutta olması ve başarı oranını düşürmesidir. Literatür çalışmalarında bu problemi çözmek için genellikle bir ön işlem adımı olarak özellik seçimi yapılmaktadır. Önem sırasına göre boyut azaltma işlemi yapan özellik seçim işlemi hem başarı performansını yükseltmekte, hem de hesaplama süresini azaltmaktadır. Kullanılan verinin yüksek boyutta özellik vektörü olması sınıflandırma işlemini zorlaştırdığından boyut azaltma işlemi çok önemlidir.

Literatürde metin sınıflandırma problemini özellik seçimine odaklanarak iyileştirmeye çalışan pek çok uygulama bulunmaktadır [3][4]. Bu çalışmalarda en çok kullanılan özellik seçim algoritmaları Information Gain (IG), Principal Component Analysis (PCA) ve Ki-Kare (CHI) algoritmalarıdır. Genel itibariyle, şimdiye kadar yapılan kıyaslama çalışmalarını da içeren literatürü baz aldığımızda, henüz tüm metin sınıflandırma problemlerini tek başına kolaylaştırabilecek bir özellik seçim algoritması üretilememiştir. Bu yüzden, özellik seçimi üzerine yapılan çalışmalar önemini korumaktadır. Bahsedilen algoritmalara ek olarak, daha yeni bir yönelim özellik seçimi olarak sezgisel arama algoritmalarının kullanılmasıdır. Fakat bu algoritmaların doğası gereği çok yüksek boyuttaki özelliklerin seçimi amaçlı kullanılmalarında yüksek bellek, işlemci gücü ve zamana ihtiyaç duymakta olduklarından diğer algoritmalara göre literatürde daha az kullanılmışlardır [5]. Var olan az sayıda çalışmalardan birisi olan H. Uğuz'un çalışmasında [6] Reuters-21578 ve Classic3 veri kümelerini kullanılmış. Information Gain (IG), Genetik Algortima (GA), Principal Component Analysis (PCA) özellik çıkarım yöntemi ile iki aşamalı akış modeli uygulanmıştır. Boyut azaltmanın etkinliğini değerlendirmek için k-NN ve C4.5 sınıflandırıcıları kullanılmış ve IG - PCA ile boyut indirgendiğinde başarımın yüksek olduğu değerlendirmiştir. Başka bir çalışmada, M. Karabulut Reuters- 21578 ve Ohsumed veri kümelerini, Information Gain ve Parçacık Sürü Optimizasyonu (GPSO) kullanarak işlemiştir.

Bu çalışmada, metin madenciliğinde diğer uygulama alanlarına göre daha az kullanılan sezgisel arama algoritmalarının, yani Genetik Arama (GA), Parçacık Sürü Optimizasyon Arama (PSO), Evrimsel Arama (EA), TABU Arama (TA) algoritmalarının, metin madenciliğinde özellik seçmek için efektif kullanılmasına dair bir uygulama geliştirilmiş ve bu algoritmaların performanslarını analiz etmeyi amaçlanmıştır. Yazarların bilgisi dahilinde olan literatüre bakıldığında, bu algoritmaların birbirleriyle kıyaslanmasına dair diğer alanlarda pek çok çalışma bulunmasına rağmen, metin madenciliği alanında kıyaslanmalarına dair bir çalışma henüz bulunmamaktadır. Bu anlamda bu belge, kıyaslama yaparak ilgili literatüre katkı sağlamayı da amaçlamaktadır.

Bu çalışmada sezgisel algoritmaları kullanmak için iki aşamalı bir özellik seçim stratejisi izlenmiştir. İlk etapta nispeten daha hızlı çalışan IG algoritması ile özellikler daha küçük bir sayıda terime (örneğin 100 terim) indirgenip; ikinci etapta, elde edilen bu özellik alt kümesi üzerinde sezgisel algoritmalar çalışılmış ve aranan özellik çıkartılmıştır. Elde edilen özellikler ile metin sınıflandırılma işlemi devam ettirilip, başarım test edilmiştir. Çalışmayı gerçeklemek için Reuters-21578 ve Ohsumed veri kümeleri kullanılmıştır. Bu veri kümeleri literatürde yaygın olarak kullanıldıklarından, başka çalışmaların da bu makale sonuçlarıyla kıyaslanabilmelerinin önü açık olacaktır. Sezgisel arama algoritmalarından en yaygın bilinen dört tanesi (GA, PSO, EA, TA) seçilmiş, bahsedilen veri setleri üzerinde denenmiş ve kıyaslanmıştır.

2. VERİ KÜMESİNİN BELİRLENMESİ (DATASET SELECTION)

Metin sınıflandırma problemlerinde yaygın olarak kullanılan "Reuters veri seti" Reuters haber ajansının 1987 yılında haber metinlerinden derlediği bir veri kümesidir [7]. Bu veri seti, 118 kategoride, 21578 belge içermektedir. Bu veri setine ait pek çok farklı sürüm bulunmaktadır; bu çalışmada ModApte-10 sürümünün ikili sınıflandırmaya (binary classification) adapte edilmiş bir versiyonu kullanılmıştır. Bu versiyonda veriseti içindeki 10 kategoriye ait veriler ayrı ayrı veriseti yapılarak, her veriseti dosyasındaki örnekler ilgili kategoriye aidiyeti evet(yes) ve hayır (no) olarak ayarlanmıştır. İlgili veri kümesi dosyalarının özellikleri Tablo-1'de gösterilmiştir.

Tablo 1. Reuters-21578 veri kümesi özellikleri (Reuters-21578 dataset properties)

Veri Kümesi	Özellik Sayısı	Örnek Sayısı	Pozitif Örnek Sayısı	Negatif Örnek Sayısı
earn	9500	12897	3964	8933
acq	7495	12897	2369	10528
money-fx	7757	12897	717	12180
grain	12473	12897	582	12315
crude	14466	12897	578	12319
trade	7600	12897	486	12411
interest	10458	12897	478	12419
ship	9990	12897	286	12611
wheat	8626	12897	283	12614
corn	8302	12897	237	12660

Çalışmada kullanılan ikinci veri seti olan OHSUMED veri kümesi ise, MEDLİNE veri tabanından alınan metinlere ait bir veri kümesidir. 1991 yılında 50216 doküman veri setinin ilk 20000 kaydı kullanılmıştır, 23 tıbbi konu başlığı ile oluşturulmuştur. OHSUMED veri kümesi özellikleri Tablo 2.'de gösterilmiştir.

Tablo 2. OHSUMED veri kümesi özellikleri (OHSUMED dataset properties)

Veri Kümesi	Özellik Sayısı	Örnek Sayısı	Pozitif Örnek Sayısı	Negatif Örnek Sayısı
c01	7624	13929	929	13000
c04	6667	13929	2630	11299
c06	7112	13929	1220	12709
c08	6784	13929	1073	12856
c10	7102	13929	1562	12367
c12	6757	13929	1039	12890
c14	6811	13929	2550	11379
c20	7304	13929	1220	12709
c21	7492	13929	1263	12666
c23	6026	13929	3952	9977

3. ÖN İŞLEM AŞAMASI (PREPROCESSING PHASE)

Ön işlem aşamaları veri kümesi üzerinde analiz yapabilmek için yapılan işlemler dizinidir. Çalışmada işlem akışı Şekil-1'de özetlenerek adımlar ve başlıklar detayları ile Bölüm 3 içinde verilecektir.

1) Veri Kümesinin Belirlenmesi
   a) Reuters- 21578 ve OHSUMED veri kümeleri belirlendi.
2) Ön İşlem Aşaması
   a) İşaretleme (Tokenization)
   b) Kök Bulma (Stemming)
   c) Durak Kelimeleri Çıkarma
   d) Terim Ağırlıklandırma
   e) Terim Ayıklama
   f) Binary Vektör oluşturma
3) Özellik Seçimi
   a) Information Gain
   b) Sezgisel Arama Algoritmaları (Genetik Arama, Parçacık Sürü Optimizasyon Arama, Evrimsel Arama, TABU Arama)
4) Sınıflandırma
   a) Naive Bayes
5) Sonuçların Değerlendirilmesi

Şekil 1. Önerilen metin sınıflandırma yapısı (The proposed text classification structure)

3.1 İşaretleme (Tokenization)

İşaretleme metin içindeki terimleri; Simgelere, noktalama işaretlerine veya kelimelere ayırmak için kullanılır. Belgeler bölüm, paragraf, cümle, kelime ve hecelere ayrılabilir. En sık rastlanan durum ise kelimelere ayrılmasıdır. Bu çalışmada da kelimelere ayırarak işaretleme yapılmıştır.

3.2 Kök Bulma (Stemming Lemmatization)

Farklı ek almış kelimelerin köke indirgenerek aynı kelime kökleri ile temsil edilmesi sağlanır. Bu şekilde hem özellik sayısı azalacak hem de aynı kelime köküne sahip kelimelerin frekansı daha doğru bir şekilde hesaplanacaktır [2]. Çalışmamızda kök bulma algoritmalarından Porter's stemming algoritması kullanılmıştır.

3.3 Durak Kelimeleri (Stop-Words)

Edat, Bağlaç ve zamir gibi çok sık kullanılan ve tek başına anlam ifade etmeyen kelimeler metinden çıkartılır. Örneğin İngilizcedeki "a", "the", "as", "at" gibi kelimeler bu gruba girer. Kelime sayısının azaltılması hem hız hem de performansı iyileştirebilir. Çalışmada, 571 kelimeden oluşan mevcut stop-word-list kelime listesi (http://www.unine.ch/Info/clef/) veri kümesinden çıkarılmıştır.

3.4 Terim Ağırlıklandırma (Term Weighting)

Benzer literatür çalışmalarında olduğu gibi, bu çalışmada da, sınıflandırılacak tüm belgeler, kesikli metin uzayından, sürekli sayısal uzaya dönüştürülmüştür. Bu dönüştürme işlemi sonucu her belge bir vektörle temsil edilecek şekle getirilmiştir. Her belge ön işleme adımlarından geçirilerek sayısal bir forma dönüştürülür. Tüm belge vektörleri birleştirilerek matris oluşturulur.

Ağırlık değerlerinin belirlenmesi iki yönteme dayanır. Birinci yöntem bir dokümanda, bir terim çok geçiyorsa, ilgili kategoriye atanması o kadar etkili olur. İkinci yöntem ise birden çok dokümanda, aynı terim bulunuyorsa, o terimin ayırt edici özelliği azalır [8]. Terim ağırlıklandırma yöntemlerinden terim frekans yönetimi kullanıldı (TF=Term Frequency). Terim frekans yönteminde metinler içerdikleri kelimenin frekansı ile ifade edilir. Terim frekansı $T_i=(w_{i1},w_{i2},w_{i3},...,w_{in})$ şeklinde gösterilir. Burada *T* vektörü, *i* dizi sayısını, *n* terimlerin toplam sayısını, *w* kelime frekansını göstermektedir.

3.5 Terimleri Ayıklama (Term Extraction)

Boyut azaltma yöntemlerinden, stop-word yönteminden başka frekansı eşik değerinden az olan terimleri kaldırarak da boyut azaltma yapabiliriz. Düşük frekanslı kelimelerin, metin bağlamında daha az önemli olduğu kabul edilerek bu kelimeler çıkarılabilir. Bu çalışmada boyut azaltmak için eşik kelime frekansını 3 olarak belirlenir. Kelime frekansı üçten az olan kelimeler kelime torbasından çıkarılır.

3.6 Binary Vektör Oluşturma (Binary Vector Representation)

Bu yöntem ile metinsel veriler eşit derecede önem veren ikili gösterim ile ifade edilmektedir. Doküman içinde geçen kelimelerin, kelime torbasında varlıklarına göre D_i={0,1,….,1} şeklinde gösterilir.

4. ÖZELLİK SEÇİMİ (FEATURE SELECTION)

Özellik seçimi büyük boyutlu veri kümesini, daha küçük boyutta temsil edebilecek nitelikte, alt küme olarak tanımlanır. Özellik seçimi ile az zamanda ve daha başarılı sınıflandırma performansı gerçekleştirilebilir. Veri kümesi özellik seçiminden önce nitelikli ve niteliksiz terimlerin oluşturduğu yüksek boyutlu terimlerdir. Sınıflandırma yapmadan önce nitelikli terimleri ayırmak için iki aşamalı özellik azaltma yöntemi kullanılırmıştır.

İlk aşamada veri kümeleri (IG) ile 100 terime düşürülür. İkinci aşamada ise 100 terime düşürülen alt veri kümesinde incelenmek istenilen sezgisel algoritmaların performans başarımı, boyut azaltma başarımları ve işlem süresi incelenerek analiz edilir.

4.1 Bilgi Kazancı (Information Gain)

IG değişken değer ölçüsü olarak tanımlanan istatistiki bir değer olarak hesaplanır [9]. Bu yöntem terim azaltma işlemlerinde çok sık kullanılan bir yöntemdir.

Kategori tahmini için olası kümeler {c1, c2, ... ,cm} olmak üzere her terim (t) için eşitlik (1) ile hesaplanır.

$$IG(t,c_i)=\sum_{c\in\{c_i,\bar{c}_i\}}\sum_{t'\in\{t,\bar{t}\}}P(t'|c).log\frac{P(t'|c)}{P(t').P(c)} \quad (1)$$

P(t, c_i)=t'nin c_i'ye üyelik olasılığı
P(t, $\bar{c}_i$)=t'nin c_i'ye üye olmama olasılığı
P($\bar{t}$, c_i)=t'nin değilinin c_i 'ye olma olasılığı
P($\bar{t}$, $\bar{c}_i$)= t'nin değilinin c_i 'ye olmama olasılığı

Terimlerin IG değerlerine göre belli eşik değerin altında kalan terimler elenerek yüksek frekanslı değerlerin özellik uzayı içinde kalması sağlanır.

4.2 Evrimsel Arama (Evolutionary Search)

Optimizasyon problemlerinin çözümünde evrimsel algoritmalar geniş yer almaktadır. Evrimsel algoritma Darwin'in doğal ayıklama prensibine dayanmaktadır. Bireylerin oluşturduğu genler, popülasyonu oluşturmaktadır. Evrimsel algoritmaların avantajı, karmaşık matematiksel işlemler yerine basit işlemler kullanmasıdır [10].

Evrimsel algoritma dört bileşenden oluşur. Bunlar başlatma, mutasyon, değerlendirme ve seçim bileşenleridir. İlk bileşen olan başlatma aşamasında değerler evrimsel hesaplama yaklaşımlarında olduğu gibi rasgele atanır. Çeşitliliğin artırılması için mutasyon operatörü, sonraki nesil de birey seçimi için seçim operatörü kullanılmaktadır. Değerlendirme operatörü ve uygunluk fonksiyonu ile hatalar belirlenerek en iyileme yapılır.[11]

4.3 Parçacık Sürü Optimizasyonu (Particle Swarm Optimization)

PSO 1995 yılında Russell Eberhart ve James Kennedy tarafından kuş sürülerinin hareketlerinden esinlenerek geliştirilen popülasyon tabanlı bir arama tekniğidir [12]. Kuş ve balık sürülerinin tek başlarına altından kalkamayacakları yiyecek bulma ve tehlikeden kaçış gibi işler, sürülerin toplu hareketlerinden esinlenerek geliştirilmiştir.

PSO parametre sayısının azlığı gibi avantajlara sahiptir [13]. PSO kuş sürülerinin konum ve zaman olarak iki boyutlu davranışlarının benzetimidir. Kuşların bulundukları ortamda, yiyecek yerini aramalarına çözüm bulmaya benzetilir. Parçacıklar olarak adlandırılan her bir çözüm, arama uzayında bir kuşa benzetilir. Parçacık hareket ettiğinde, konum bilgisi fonksiyonda değerlendirilerek uygunluk değeri hesaplanır. Parçacık olarak, konum, hız ve elde ettiği en iyi uygunluk değerlerini (p_{best}) hatırlamalıdır. Bu parçacıktaki değerler sürüdeki optimum en yakın parçacığın pozisyonuna göre ayarlanır. Her iterasyonda parçacık hafızaları tüm parçacıkların en iyi çözümü (g_{best}) ve her parçacığın kendi en iyi çözümü (p_{best}) kullanılarak güncellenir. Ana fikir bireyler arasındaki bilgi paylaşımı ile en iyi stratejiyi geliştirmektir. Belli iterasyon sonucunda sürünün en iyi uygunluk değeri problemin çözümü olur.

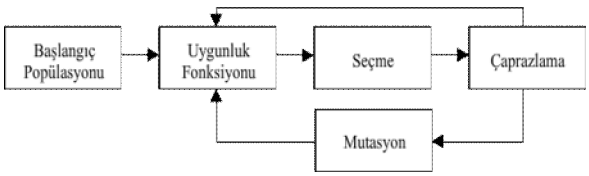
```
BEGIN
popülasyon Tane Parçacığa Hız ve Konum Değerleri Ata
REPEAT
        FOR =1 TO popülasyon
        Uygunluk değerini hesapla;
        Pbest değerini güncelle;
        gbest değerinin güncelle;
        Konum ve Hız değerlerini Güncelle;
        END FOR
UNTIL
END
```

Şekil 2. PSO algoritmasının yapısı (PSO algorithm structure) [13]

4.4 Genetik Arama (Genetic Search)

Charles Darwin tarafından ortaya atılmış olan Evrim teorisine dayanan en iyinin yaşaması kuralına uygun olarak arama yöntemiyle çözüm üretme prensibine göre çalışmaktadır.1975 yılında John Holland tarafından temelleri oluşturuldu [14]. GA'da çözülmesi istenen problemin değişkenleri kromozomlarla temsil edilirler. Bu kromozomların

rasgele bir kısım seçilerek çözüm oluşturulur. Oluşturulan çözümün kromozomlar üzerinde operatörler ile değişiklikler yapılarak en iyi çözüm değeri elde edilmeye çalışılır. Bu operatörler seçme, çaprazlama ve mutasyondur.



Şekil 3. Genetik arama algoritma akış şeması (Genetic Search Algorithm Flowchart) [15]

4.5 Tabu Arama (Tabu Search)

1986 yılında Fred Glover tarafından optimizasyon problemlerinin çözümü için geliştirilmiş sezgisel arama algoritmasıdır [16]. Rasgele değerler ile herhangi bir çözüm bulunur. Problemin amacı yerelde minumun değeri elde etmektir. Bir değer elde edildikten sonra komşu çözümler hesaplanır. Bir çözümden başka çözüme gitmeye hareket denir. Hareket sonucunda daha iyi sonuçlar hafızaya alınır, daha kötü sonuç ise tabulaştırılır. Tabu bazı durumlara engel olmak için yasaklar listesi tutulur. Yeni çözümler elde edildikçe en iyi çözüme odaklanılır.

Adım 1: Bir başlangıç çözümü al.
Adım 2: Parametreleri belirle algoritmada kullanılacak tabu listesi uzunluğu, durdurma kriteri, tabu yıkma vs.) değerlerini ata
Adım 3: Komşu çözümler üretilerek, bu çözümler ile başlangıç çözümü arasından tabu listesinde olmayan tüm çözümler arasından en iyiyi seç.
Adım 4: Seçilen en iyi değer ile mevcut çözüm değiştirilerek tabu listesi yenile.
Adım 5: Durdurma kriteri sağlanıncaya kadar Adım 3 ve 4'ü tekrar et.

Şekil 4. Tabu arama algoritmasının adımları (Tabu search algorithm steps)

5. NAİF BAYES (NAIVE BAYES)

Metin sınıflandırma alanında en çok kullanılan algoritmalar Naive Bayes, Destek Vektör Makinesi, K-En Yakın Komşuluk, C 4.5 algoritmalarıdır[17]. Başarılı ve çok kullanılan algoritma olmasından dolayı Naive Bayes algoritması bu çalışmada kullanılmıştır.

Naive Bayes, uygulanabilirliği kolaylığı ve başarılı performansıyla metin sınıflandırmada en çok tercih edilen metotlardan biri olmuştur [18][19]. Problemde olması muhtemel bağımlı durumların olasılıkları eşitlik (2) ile hesaplanır. P(A|B) ifadesi B durumuna göre, A durum olasılığını temsil edilir.

$$P(A|B) = \frac{P(A \cap B)P(A)}{\mathrm{P(B)}} \quad (2)$$

Naive Bayes algoritmasında olası çıkış durumları içerisinde en yüksek olasılıklı durum hedef sınıf seçilir. Matematiksel ifadesi eşitlik (3) ile gösterilir.

$$Y' = \arg\max_{y_j \in Y} P(Y=y_j|X) \quad (3)$$

Eşitlik (3)'de Y' ifadesi hedef sınıfı, y_j ifadesi j.çıkış durumu, X tespit edilecek giriş sınıfını temsil edilir. Giriş vektörü birden çok olduğu durumlarda formül tüm nitelikler için koşullu olasılıkların çarpımı olarak eşitlik (4)'e dönüşür.

$$P(x_1, x_1, x_{1,\ldots,} x_1|y_j) = \prod_{i=1}^{m} P(Y=(x_j|y_i) \quad (4)$$

Eşitlik (4)'te x giriş özelliklerini, m ise nitelik adedini temsil edilir. Bayes ile Naive Bayes arasındaki en önemli fark, olasılık değeri yerine hedef sınıf olasılığının bulunmasıdır. Bayes teoremindeki paydada bulunan değer tüm durumlarda ortak olduğundan Naive Bayes sınıflandırmada ihmal edilebilir. Bu durumda eşitlik (4), çoklu nitelik girişi olan eşitlik (5)'e dönüşür.

$$Y' = \arg\max_{y_j \in Y}(P(Y=y_j) \prod_{i=1}^{m} P(X=x_i|Y=y_j)) \quad (5)$$

6. PERFORMANS DEĞERLENDİRME KRİTERLERİ (PERFORMANCE EVALUATION CRITERIA)

Genel olarak sınıflandırma görevlerinin performansını ölçmek için kullanılan ait bu kriterler, metin sınıflandırma performansını ölçmek için de kullanılır. Bu kriterler kesinlik (precision), anma (recall), F-ölçütü (F-measure) gibi ölçütlerdir. Modelin başarısı, doğru veya yanlış sınıflara atanan örnek sayıları ile alakalıdır. Bu bilgiler hata matrisinde, gerçekler satırlarla, tahminler ise sütunlarla gösterilerek ifade edilir. Ölçümlerde kullanılan terimler hata matrisinde gösterilir. Hata matrisinde gösterilen, TP(True Positive) doğru sınıflandırılmış pozitif örnek sayısı, TN(True Negative) doğru sınıflandırılmış negatif örnek sayısı, FP(False Positive) yanlış sınıflandırılmış pozitif örnek sayısı, FN(False Negative) yanlış sınıflandırılmış negatif örnek sayılarını ifade eder [20].

Doğru sınıflandırılmış pozitif örnek sayısının, pozitif sınıflandırılmış örneklerin sayısına oranı kesinlik olarak ifade edilir.

$$\text{Kesinlik, } \pi_i = \frac{\mathrm{TP}i}{\mathrm{TP}i + \mathrm{FP}i} \quad (6)$$

Tablo 3. Hata matris gösterimi (The error matrix representation)

Hata Matrisi		Öngörülen Sınıf	
		Sınıf=a	Sınıf=b
Doğru Sınıf	Sınıf=a	TP(True Positive)	FN(False Positive)
	Sınıf=b	FP(False Negative)	TN(True Negative)

Doğru sınıflandırılmış pozitif örnek sayısının, toplam pozitif örnek sayısına oranı anma olarak ifade edilir.

$$\text{Anma}, \rho_i = \frac{\text{TP}i}{\text{TP}i + \text{FN}i} \quad (7)$$

Kesinlik ve anma ölçütleri, anlamlı sonuçlar için yeterli olmayabilir. Bunun için her iki ölçütün birlikte kullanıldığı daha doğru sonuçlar verebilecek değerlendirme kıstası olarak f-ölçütü kullanılır. Bu tanımlama kesinlik ve anma değerlerinin harmonik ortalaması alınarak hesaplanır. Kesinlik ve anma değerlerinin her ikisinin de etkisini görebileceğimiz ölçüt olarak kullanılır.

$$\text{F} - \text{Ölçütü}, \text{F}_1 = \frac{2 \text{ x } \rho \text{ x } \pi}{\rho + \pi} \quad (8)$$

Genel başarım iki farklı şekilde, makro ve mikro ortalama olarak hesaplanabilir. Makro ortalama tüm kategorinin toplamını alarak, kategori sayısına bölünmesi ile elde edilir (Eşitlik (9)) .

$$\textbf{F1 (Makro Ortalama)}, \textbf{Makro F}_1 = \frac{\sum_{i=1}^{n} \mathbf{F1}_i}{\mathbf{k}} \quad (9)$$

7. DENEY SONUÇLARI VE DEĞERLENDİRME (EXPERIMENTAL RESULTS AND DISCUSSION)

Bu çalışmada metin sınıflandırmada kullanılan boyut azaltma yöntemlerinden sezgisel algoritmaların sınıflandırma başarımına etkisi Reuters-10 ve Ohsumed veri kümeleri kullanılarak analiz edilmiştir. Deneyler üç parametrede incelenmiştir. Birinci parametre olarak performans değerlendirmeleri karşılaştırılmıştır. İkinci olarak boyut azaltma başarımları karşılaştırılmıştır. Üçüncü olarak algoritmaların toplam işlem süreleri analiz edilmiştir. Tüm deneyler 8 çekirdek ve 8 GB RAM ile 2.3 GHz işlemciye sahip bir bilgisayarda çalıştırılmıştır. Yeni Zelanda Waikato Üniversitesi'nde açık kaynak lisansı ile geliştirilen Java tabanlı bir yazılım olan WEKA aracı kullanılmıştır. Ölçümlerde 10 kat çapraz doğrulama ile sınıflandırma performansı ölçülmüştür.

Sınıflandırma performansını ölçmek için yapılan deneylerin hepsinde gaussian kernel density estimator kullanan Naive Bayes implementasyonu kullanılmıştır. Sınıflandırma yapılmadan önce, Bölüm 3'de ön işlemler yapılarak veri setleri sınıflandırma algoritmasına sunulacak formata getirilmiştir. Fakat bu aşamada veri setlerini sınıflandırma algoritmasına hazırlamak amacıyla özellik seçimi yapılması gereklidir. Bu makalede iki aşamalı bir özellik seçme işlemi gerçekleştirilmiştir. Birinci aşamada IG algoritması ile sütun sayısı 100 özelliğe indirgenmiş, bir sonraki aşamada ise PSO, GA, EA ve TA algoritmaları teker teker çalıştırılarak, her bir veri setinin dört farklı kopyası oluşturulmuştur. Elde edilen bu yeni veri setleri sırasıyla Naive Bayes algoritmasına verilmiş ve sınıflandırma performans sonuçları Tablo 4'de sunulmuştur.

Sezgisel arama algoritmalarının kıyaslanabilmesi amacıyla, bu algoritmaların parametrelerine mümkün olduğu kadar aynı değerler verilmiştir. EA, GA, PSO parametreleri crossover olasılığı, Crossover Probability 0,6, mutasyon meydana gelme olasılığı, Mutation Probability 0,01, popülasyon boyutu, Population Size 20 alınmış, TA parametreleri ise çeşitlendirme olasılığı, Diversification Probability 1, başlangıç boyut değeri, Initial Size -1, en iyi çözüm için kontrol değeri, Neighborhood -1 olarak verilmiştir.

Tablo 4. (a) ve (b) incelendiğinde Reuters ve Ohsumed veri kümelerinde F-measure değerlerinde çok büyük farklılıklar olmamasına rağmen TA algoritmasıyla daha başarılı sonuçlar elde edilmiştir. Bu başarılı sonuç TA algoritmaların tek çözüme bağlı sezgisel algoritma ve diğer sezgisel algoritmaların (PSO, EA, GA) ise birden fazla çözümü aynı anda bulmaya çalışan algoritmalar olduğu gerçeği ile ilişkilendirilebilir. Dolayısıyla, özellik seçiminde, tek bir çözümü iyileştirmek, birden fazla iyi çözümü aynı anda arama tekniğine göre daha verimli sonuç üretebilmektedir.

Tablo 5. (a) ve (b) her bir özellik seçim algoritmasının kendilerine IG algoritması sonucu verilen 100 adet özelliği en az kaç özelliğe indirgediğini sunmaktadır. Bu sonuçlara göre, her iki veri kümesinde de boyut indirgeme başarımı olarak TA algoritmasının incelenen diğer algoritmalara göre yaklaşık yarı yarıya farklılık vardır. TA algoritmalarında ortalama 18,8 terim sayısı ile boyut indirgeme sonucunda daha başarılı bir sonuç çıkarılmıştır. Dolayısıyla, TA algoritmasının çıkarttığı özelliklerle yapılan sınıflandırmanın performansının diğerlerine göre daha başarılı olmasının ardında, TA algoritmasının daha az özellik üretmesi bulunduğu gözlemlenmektedir. Daha farklı ifade edilirse, TA algoritması metinlerde bulunan ayırıcı özellikleri barındıran en küçük alt kümeyi, diğerlerine göre daha başarılı bir şekilde seçebilmiştir. Örneğin Tablo 5 (a)'da PSO her veri kümesi için ortalama 44,7 kelime seçerken, TA aynı veri kümesinde ortalama 18,8 kelime seçerek, veri setini daha az kelimeyle ayırt edebilmiştir.

Tablo 4. Reuters ve Ohsumed veri kümesi F-measure sonuçları (Results for Reuters and Ohsumed datasets in terms of F-measure)

Reuters	PSO	GA	EA	TA
(1) corn	0,977	0,970	0,972	0,991
(2) crude	0,971	0,973	0,975	0,980
(3) acq	0,894	0,886	0,893	0,897
(4) earn	0,960	0,951	0,953	0,947
(5) money-fx	0,938	0,934	0,932	0,948
(6) grain	0,969	0,962	0,962	0,988
(7) interest	0,946	0,950	0,943	0,958
(8) ship	0,986	0,987	0,986	0,990
(9) trade	0,949	0,945	0,950	0,963
(10) wheat	0,981	0,980	0,980	0,995
Ortalama	0,957	0,954	0,955	0,966

(a)

Ohsumed	PSO	GA	EA	TA
(1) c01	0,933	0,936	0,930	0,941
(2) c04	0,912	0,911	0,903	0,913
(3) c06	0,925	0,924	0,922	0,917
(4) c08	0,932	0,931	0,934	0,934
(5) c10	0,874	0,879	0,869	0,875
(6) c12	0,949	0,948	0,950	0,948
(7) c14	0,900	0,904	0,903	0,900
(8) c20	0,921	0,923	0,921	0,938
(9) c21	0,919	0,920	0,921	0,918
(10) c23	0,683	0,687	0,691	0,696
Ortalama	0,895	0,896	0,894	0,898

(b)

Tablo 5. Reuters ve Ohsumed veri kümesi için boyut azaltma başarım sonuçları (Dimension reduction performance results in Reuters and Ohsumed datasets)

Reuters	PSO	GA	EA	TA
(1) corn	23	28	47	12
(2) crude	36	28	31	11
(3) acq	65	54	65	35
(4) earn	36	30	41	22
(5) money-fx	53	62	48	28
(6) grain	40	36	51	15
(7) interest	45	39	57	19
(8) ship	53	42	46	18
(9) trade	51	52	36	18
(10) wheat	45	37	32	10
Ortalama	44,7	41	45	19

(a)

Ohsumed	PSO	GA	EA	TA
(1) c01	60	55	59	28
(2) c04	48	37	55	21
(3) c06	58	39	47	28
(4) c08	43	33	37	13
(5) c10	59	60	56	42
(6) c12	46	37	35	14
(7) c14	59	40	48	17
(8) c20	35	38	39	16
(9) c21	41	29	39	17
(10) c23	60	44	54	45
Ortalama	50,9	41	47	24

(b)

Tablo 6. Reuters ve Ohsumed veri kümesi boyut azaltma işlem süresi (sn) (Comparison of consumed time in seconds to process dimension reduction task in Reuters and Ohsumed datasets)

Reuters	PSO	GA	EA	TA
(1) corn	14	14	14	13
(2) crude	12	12	12	12
(3) acq	18	20	18	18
(4) earn	20	22	19	20
(5) money-fx	16	19	16	16
(6) grain	15	17	15	15
(7) interest	17	17	17	17
(8) ship	12	13	12	12
(9) trade	14	16	14	14
(10) wheat	14	14	14	14
Ortalama	15,2	16	15	15

(a)

Ohsumed	PSO	GA	EA	TA
(1) c01	12	13	11	11
(2) c04	16	16	16	15
(3) c06	12	12	12	12
(4) c08	10	10	10	10
(5) c10	13	13	13	14
(6) c12	13	13	12	12
(7) c14	15	15	16	15
(8) c20	15	14	14	14
(9) c21	14	14	14	14
(10) c23	16	16	16	17
Ortalama	13,6	14	13	13

(b)

Algoritmaların çalışma sürelerinin içeren Tablo 6. (a) ve (b) incelendiğinde her iki veri kümesinde de birbirini benzer değerler elde edilmiştir. Sezgisel algoritmalar ile boyut indirgeme işlemi süreleri karşılaştırıldığında TA ve EA algoritmalarının işlem hızı aynı çıkmış ve en iyi olarak gözlemlenmiştir. TA algoritmasının boyut azaltma başarımı daha iyi olmasına rağmen işlem süresi olarak daha hızlı olduğu görülmektedir. Tablo 7. (a) ve (b) incelendiğinde veri kümelerinin NB ile sınıflandırma işlem süreleri karşılaştırılmış, Tablo 4. (a), (b)'deki boyut indirgeme başarımı ile ilişkili sonuçlar gözlemlenmiştir.

Burada boyut indirme başarısı yüksek olan yani boyutu daha az terime indirebilen TA algoritmalarının işlem hızı en başarılı olarak saptanmıştır.

Tablo 7. Reuters ve Ohsumed veri kümesi NB ile sınıflandırma işlem süresi (sn) (Comparison of consumed time in seconds to classify Reuters and Ohsumed datasets by NB algorithm)

Reuters	PSO	GA	EA	TA
(1) corn	4	5	7	3
(2) crude	5	5	5	3
(3) acq	10	9	10	6
(4) earn	6	6	7	4
(5) money-fx	8	10	7	5
(6) grain	6	6	7	3
(7) interest	7	6	8	4
(8) ship	7	6	6	4
(9) trade	7	8	6	4
(10) wheat	7	5	5	3
Ortalama	6,7	6,6	6,8	3,9

(a)

Ohsumed	PSO	GA	EA	TA
(1) c01	8	8	8	5
(2) c04	8	6	8	4
(3) c06	8	6	7	5
(4) c08	6	5	6	3
(5) c10	9	9	8	7
(6) c12	7	6	5	3
(7) c14	9	8	7	4
(8) c20	6	6	6	3
(9) c21	6	5	6	4
(10) c23	10	7	8	7
Ortalama	7,7	6,6	6,9	4,5

(b)

Tablo 8. Reuters ve Ohsumed veri kümesi IG- 100 Genel Sonuçları (Classification results of Reuters and Ohsumed datasets after 100 features are selected by IG)

Reuters	F-Measure	IG Süre	NB Süre
(1) corn	0,955	117	13
(2) crude	0,966	211	13
(3) acq	0,897	126	15
(4) earn	0,963	178	16
(5) money-fx	0,924	114	14
(6) grain	0,952	180	15
(7) interest	0,930	148	18
(8) ship	0,977	137	15
(9) trade	0,928	115	15
(10) wheat	0,970	118	15
Ortalama	0,946	144,4	14,9

(a)

Ohsumed	F-Measure	IG Süre	NB Süre
(1) c01	0,925	146	13
(2) c04	0,893	148	15
(3) c06	0,924	140	13
(4) c08	0,931	131	13
(5) c10	0,874	153	14
(6) c12	0,937	137	13
(7) c14	0,896	170	14
(8) c20	0,906	173	14
(9) c21	0,910	175	15
(10) c23	0,683	150	15
Ortalama	0,888	152,3	13,9

(b)

Tablo 9. Reuters ve Ohsumed veri kümesi IG- 200 Genel Sonuçları (Classification results of Reuters and Ohsumed datasets after 200 features are selected by IG)

Reuters	F-Measure	IG Süre	NB Süre
(1) corn	0,950	127	28
(2) crude	0,946	250	26
(3) acq	0,916	148	31
(4) earn	0,960	208	33
(5) money-fx	0,904	124	28
(6) grain	0,943	195	27
(7) interest	0,917	168	29
(8) ship	0,966	154	26
(9) trade	0,917	128	28
(10) wheat	0,965	131	27
Ortalama	0,938	163,3	28,3

(a)

Ohsumed	F-Measure	IG Süre	NB Süre
(1) c01	0,926	170	26
(2) c04	0,887	163	30
(3) c06	0,920	171	28
(4) c08	0,930	168	28
(5) c10	0,875	190	29
(6) c12	0,934	182	33
(7) c14	0,889	199	34
(8) c20	0,897	184	30
(9) c21	0,901	196	37
(10) c23	0,682	162	35
Ortalama	0,884	178,5	31

(b)

Tablo 8 ve 9'da sezgisel arama algoritmaları kullanılmadan sadece IG metodu ile 100 ve 200 özellik seçilerek yapılan sınıflandırma işleminin her veriseti için sonuçları verilmiştir. Bu sonuçlar incelendiğinde, sınıflandırma performansının sezgisel arama algoritmaları ile yapılan özellik seçimi sonucu üretilen verisetleri üzerindeki sınıflandırma performansından daha düşük olduğu gözlemlenmiştir.

Bu durumda sezgisel arama algoritmalarının genel olarak özellik seçimi performansını iyileştirdiği söylenebilir.

Tablo 10. Reuters ve Ohsumed veri kümesi Özellik Seçimi Olmadan (Ö.S.O.) sınıflandırma sonuçları (Reuters and Ohsumed classification results without feature selection)

Reuters	F-measure	İşlem Süresi
(1) corn	0,938	350
(2) crude	0,884	626
(3) acq	0,932	317
(4) earn	0,956	486
(5) money-fx	0,889	339
(6) grain	0,924	593
(7) interest	0,912	415
(8) ship	0,943	384
(9) trade	0,880	309
(10) wheat	0,953	361
Ortalama	0,921	418

(a)

Ohsumed	F-measure	İşlem Süresi
(1) c01	0,906	467
(2) c04	0,870	416
(3) c06	0,901	433
(4) c08	0,892	407
(5) c10	0,852	371
(6) c12	0,914	342
(7) c14	0,838	351
(8) c20	0,863	367
(9) c21	0,870	379
(10) c23	0,679	307
Ortalama	0,859	384

(b)

Tablo 11. Reuters ve Ohsumed veri kümesi genel sonuçları (General results with Reuters and Ohsumed datasets)

Reuters	F-measure	Boyut	Süre1	Süre2
PSO	0,957	44,7	15,2	6,7
GA	0,954	40,8	16,4	6,6
EA	0,955	45,4	*15,1*	6,8
TA	*0,966*	*18,8*	*15,1*	*3,9*
IG-100	*0,946*	*100*	-	*14,9*
IG-200	*0,938*	*200*	-	*28,3*
Ö.S.O.	0,921	12897	-	418

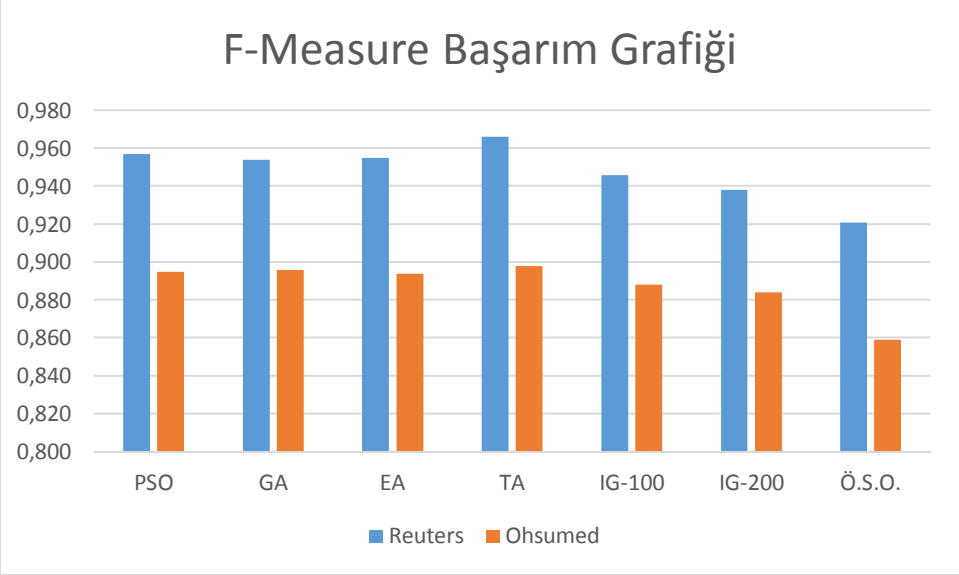
(a)

Ohsumed	F-measure	Boyut	Süre1	Süre2
PSO	0,895	50,9	13,6	7,7
GA	0,896	41,2	13,6	6,6
EA	0,894	46,9	*13,4*	6,9
TA	*0,898*	*24,1*	*13,4*	*4,5*
IG-100	*0,888*	*100*	-	*13.9*
IG-200	*0,884*	*200*	-	*31*
Ö.S.O.	0,859	13929	-	384

(b)

Son olarak, sezgisel arama algoritmaları tabanlı özellik seçiminin sınıflandırma performansını artırdığını gözlemlemek amacıyla, özellik seçimi olmadan veri setlerinin ham halleri Naive Bayes algoritmasına verilerek, sınıflandırma yapması istenilmiştir. İlgili sonuçları içeren Tablo 10. (a) ve (b) incelendiğinde Reuters veri kümesinde ortalama F-measure değeri özellik seçimi yapıldığında ortalama %95,8 gibi bir değer iken, özellik seçimi yapılmadığında %92,1'e düşmüş ve bu esnada sınıflandırma yapmak için geçen sürenin de arttığı gözlenmiştir. Benzer durum Ohsumed veri kümesi için de gözlenmiş, ortalama F-measure değeri özellik seçimi yapıldığında ortalama %89,6 iken, özellik seçimi yapılmadığında %85,9'e düşmüş ve yine işlem süresi aynı şekilde artmıştır. Bu son deneyle, sezgisel algoritmaların özellik seçimi için kullanılabilir ve verimli olduğu sonucuna varılmıştır.

Tablo 11. (a),(b) ve Şekil 5'de ise genel sonuçlar düzenlenmiş ve her iki veri kümesinde de benzer sonuçlar elde edilmiştir. Tablo 11. (a), (b)'de belirtilen Süre1 sütunu sezgisel arama algoritması için özellik seçimi işlem süresini, Süre2 sütunu ise Naive Bayes ile sınıflandırma işlem süresi olarak temsil edilmiştir. Burada Boyut indirgeme başarımları, sınıflandırma işlem sürelerini etkilemekte aynı zamanda performans başarımları da yüksek çıkmaktadır. Boyut indirmede terim sayısı yüksek olan algoritmalarda işlem süresi yüksek ve performans başarımları daha düşük gözlemlenmiştir. Bu farklılığın sezgisel algoritmaların farklı çözüm sayılarına göre işleyişinden kaynakladığı düşünülmektedir. Tek çözüme dayalı algoritmalardan olan TA algoritması, topluluğa dayalı algoritmalara göre daha başarılı olduğu saptanmıştır. Son olarak, TA algoritmasının yapılan deneylerde daha iyi olduğu gözlenen sonuçlarının istatistiksel olarak anlamlı olup olmadığı incelenmiştir. Bunun için boş hipotez Ho: μtabu = μdiğer olarak alınırken, alternatif hipotez Ha: μtabu > μdiğer kabul edilecektir. Boş hipotezin doğruluğunu %95 güven aralığında test etmek için eşleştirilmiş t-test (paired two-sample t-test) kullanılmıştır. Bunun için TA algoritmasının Reuters ve Ohsumed verisetleri için ürettiği F-Measure performans değerleri (μ_{tabu}) ve TA dışındaki diğer 3 algoritmanın sonuçlarının ortalaması ($\mu_{diğer}$) alınmıştır. Sonuçlar Tablo 12'de verilmiştir.



Şekil 5. Sonuç Performans Başarım Grafiği (General performance chart)

Tablo 12. TA ve diğer algoritmaların performans kıyasının t-test değerleri(Performance comparison of T-test values of TA and other algoritms)

Veriseti	**t_{kritik}**	**t_{stat}**
Reuters	2,262	3,629
Ohsumed	2,262	1,507

Tablo 12'de $t_{stat} > t_{kritik}$ olan durumda, H_0 reddedilir, yani TA algoritmasının Reuters veri setindeki performansı istatistiksel olarak anlamlıdır. Fakat Ohsumed veri tabanındaki sonuçlara bakıldığında boş hipotezi reddedecek yeterli kanıt bulunmadığı görünmektedir. Bu durumda, Reuters veriseti için TA algoritmasının üstün performansı istatistiksel olarak da anlamlı çıkarken, Ohsumed veriseti için bu durum istatistiksel olarak ispatlanamamıştır.

8. SONUÇLAR (CONCLUSIONS)

Özellik seçim aşaması, metin sınıflandırma sürecinin en önemli adımlarındandır. Bu belgede, popüler arama algoritmalarının (GA, PSO, EA, TA) sınıflandırma performansı, boyut azaltma başarımı ve işlem süreleri analiz edilmiştir. Genel olarak, ilgili literatürde, sezgisel algoritmalar, yüksek boyutlu özellik içeren veri kümelerinde işlem sürelerinin fazla olması sebebiyle tercih edilmemektedir. Çok boyutlu özellik sayısına sahip veri tabanlarında sınıflandırma işlemleri uzun sürede tamamlanır. Örneğin tipik bir metin madenciliği veri setinde sütun sayısı binler civarındadır. Bu makalede kullanılan veri tabanları olan Reuters-10 veri kümesinde 12897, Ohsumed veri kümesinde ise 13929 özellik bulunmaktadır. Sezgisel arama algoritmalarına veri setinin ham hali verildiğinde, sonuçları elde etmek çok daha uzun sürebilmektedir. Bu yüzden sorunu 2 aşamalı olarak azaltma yöntemi kullanılmıştır. İlk aşamada veri kümeleri IG yöntemi ile özellik sayısı azaltılarak 100 terime indirilmiş, ikinci aşamada ise sezgisel algoritmalar ile özellik seçimi yapılmıştır. İkinci aşamada GA, PSO, EA, TA algoritmaları tek tek kullanılarak veri setlerinin özellik seçimi yapılmış hali elde edilmiş ve bu veri kümeleri NB ile sınıflandırılarak performansları karşılaştırılmıştır.

Deneysel sonuçlar, performans, boyut azaltma başarısı, işlem süresi şeklinde üç parametrede incelenmiştir. Sonuç olarak, her iki veri setinde de TA algoritmasının çok büyük farklılık olmamasına rağmen boyut sayısını daha az boyuta indirgeyerek daha hızlı çalıştığı ve performans başarımı daha yüksek olduğu gözlenmiştir. Bu başarım algoritmaların aramada kullandıkları çözüm sayısına göre; tek çözüme dayalı TA arama algoritmasının, topluluğa dayalı PSO, EA, GA arama algoritmalarına göre daha başarılı olduğu deney sonuçlarına göre söylenebilir.

Genel olarak sınıflandırma ve özellik seçim amaçlı kullanılan algoritmaların herhangi birisinin tek başına diğerlerinden tüm durumlarda ve tüm uygulama alanlarında iyi olması beklenmemektedir ve henüz böyle bir durum gözlenmemiştir. Bu yüzden, literatürde çeşitli algoritmaların birleştirilerek yeni algoritmaların üretilmesi (hybrid algorithms) veya algoritmaların bir arada sonuca ortak karar vermesi (ensemble algorithms) gibi yeni yöntemler kullanılmaktadır. Bir sonraki çalışmada, sezgisel arama algoritmalarının daha iyi bir sonuç üretmek için bahsedilen yöntemlerle birleştirilmesi konusu üzerinde uygulanması düşünülmektedir. Bu şekilde, var olan algoritmalardan daha iyi sonuç üretecek hibrit bir algoritma üretile bilinmektedir.

KAYNAKLAR (REFERENCES)

1. Joachims, T., "A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization", **Proceedings of the Fourteenth International Conference on Machine Learning, San Francisco**, CA, USA, 143–151, 1997.
2. Oğuzlar, A., "Metin Madenciliği Nedir?", **Temel Metin Madenciliği**, Bursa, Dora Basım, 2011.
3. Yang Y. ve Pedersen J. O., "A Comparative Study on Feature Selection in Text Categorization", **Proceedings of the Fourteenth International Conference on Machine Learning**, San Francisco, CA, USA, 412–420, 1997.
4. Zheng Z., Wu X., ve Srihari R., "Feature Selection for Text Categorization on Imbalanced Data", **SIGKDD Explor Newsl**, Cilt 6, No. 1, 80–89, Haziran 2004.
5. Karabulut M., "Fuzzy unordered rule induction algorithm in text categorization on top of geometric particle swarm optimization term selection", **Knowl.-Based Syst.**, Cilt 54, 288–297, Aralık 2013.
6. Uğuz H., "A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm", **Knowl.-Based Syst.**, Cilt 24, No. 7, 1024–1032, 2011.
7. Sebastiani F., "Machine Learning in Automated Text Categorization", **ACM Comput. Surv**., Cilt 34, sayı 1, 1–47, Mar. 2002.
8. Lahtinen T., **Automatic indexing: an approach using an index term corpus and combining linguistic and statistical methods**, Tez, University of Helsinki, Helsinki, 2000.
9. Kök V., Kuloğlu N., "Sollama Esnasında Taşıt Ve Yol İle İlgili Faktörlerin Karar Ağacı Yöntemiyle İrdelenmesi", **Erciyes Üniversitesi Fen Bilim. Enstitüsü Derg.**, No. 21(1–2), 180–188, 2005.
10. Talbi E.G., "Metaheuristics: From Design to Implementation" **Wiley Publishing**, 2009.
11. Engelbrecht A. P., "Computational intelligence: an introduction", 2nd ed. Chichester, England , Hoboken, NJ, **John Wiley & Sons**, 2007.
12. Kennedy J. ve Eberhart R., "Particle swarm optimization", **IEEE International Conference on Neural Networks**, Cilt 4, 1942–1948, 1995.
13. Ortakcı Y. ve Göloğlu C., "Parçacık Sürü Optimizasyonu İle Küme Sayısının Belirlenmesi", **Akademik Bilişim**, Uşak, 335–341, 2012.
14. Haupt R. L. ve Haupt S. E., Practical Genetic Algorithms. **John Wiley & Sons**, 2004.
15. Nabiyev V. V., **Yapay zeka: insan-bilgisayar etkileşimi**, Ankara, Seçkin Yayıncılık, 2012.
16. Czapiński M., "An effective Parallel Multistart Tabu Search for Quadratic Assignment Problem on CUDA platform", **J. Parallel Distrib. Comput.**, Cilt 73, No. 11, 1461–1468, Kasım 2013.
17. Sebastiani F., "Machine Learning in Automated Text Categorization", **ACM Comput Surv**, Cilt 34, No. 1, 1–47, Mar. 2002.
18. Alpaydin E., **Introduction to machine learning**, 2nd ed. Cambridge, MIT Press, 2010.
19. Aggarwal C. C. ve Zhai C., "A Survey of Text Classification Algorithms", Mining Text Data, Eds. **Springer US**, ss. 163–222, 2012.
20. Yang Y., "An Evaluation of Statistical Approaches to Text Categorization", **Inf Retr**, Cilt 1, No. 1–2, ss. 69–90, May 1999.
21. "Machine Learning Project at the University of Waikato in New Zealand." **[Çevrimiçi]**: http://www.cs.waikato.ac.nz/ml/. [Erişim: 24-Mart-2015].