Sekans Etiketleme Uygulamaları için Makine Öğrenmesi Yöntemlerinin Karşılaştırılması

Comparison of Machine Learning Methods for the Sequence Labelling Applications

Mehmet Fatih AMASYALI
Bilgisayar Mühendisliği Bölümü
Yıldız Teknik Üniversitesi
İstanbul, Türkiye
mfatih@ce.yildiz.edu.tr

Metin BİLGİN
Bilgisayar Mühendisliği Bölümü Doktora Öğrencisi
Yıldız Teknik Üniversitesi
İstanbul, Türkiye
f0111301@ce.yildiz.edu.tr

***Özetçe*—Bu çalışmada, Yapay verisetleri üzerinde Koşullu Rastgele Alanlar (Condition Random Fields-CRF) ve Klasik Makine Öğrenmesi (KMÖ) yöntemlerinin karşılaştırılması yapılmıştır. Çalışmanın ilk bölümünde Yapay Veri seti üzerinde CRF ve KMÖ yöntemlerinin performansı ölçülmüştür. Yapılan çalışmalar sonucunda Klasik Makine Öğrenmesi yöntemleri Naive Bayes (NB) dışında CRF'den daha yüksek performans göstermiştir. NB ve CRF'nin başarısı çıkışların tek bir dağılımdan meydana geldiği durumlarda yüksek, diğer durumlarda düşük kalmaktadır. Ayrıca bu çalışmada eğitim setinin boyutunun başarıya etkisi ölçülmüştür. İkinci çalışma bu durumu test etmek için yapılmıştır.**

Anahtar Kelimeler — Koşullu Rastgele Alanlar, Sekans Etiketleme

Abstract— In this study, on artificial data sets, it was compared condition random fields(CRF) and classical machine learning(CML) types. First part of this study, the performances of CRF and CML types were measured on artificial data sets. As the result of studies, CML types, except Naive Bayes, performanced higher than CRF. The success of NR and CRF is high when the outputs consist of one distribution, in other case it stays low. Besides in this study, it was evaluated the effect of education set size on success. The second study was made to test this situation.

Keywords — Conditional Random Fields, Sequence Labeling

I. GİRİŞ

Sekans etiketleme, Makine öğrenmesinde kullanılan örüntü tanımanın bir çeşididir. Sekans Etiketlemenin kullanım alanları, Varlık İsmi Tanımlama(NER), Konuşma Tanıma(Speech Recognition), POS tagging vb. kullanım alanlarına sahiptir [1].

Tablo 1'de görüldüğü gibi İstanbul kelimesi bazen yer ismi olarak etiketlenirken, bazen de kurum ismi olarak etiketlenmiştir. CRF gibi olasılıksal çizge modelleri etiketlenmiş bu verisetleri içindeki geçiş durumlarını hesaplayarak etiketsiz olarak verilmiş kümedeki kelimeye karşılık üretilebilecek en yüksek olasılıklı etiketi atamaya çalışır. Tablo 1'de Varlık İsmi Tanımlama (NER), Tablo 2'de Makine Çevirisi (ML) ve Tablo 3'de Cümle Öğelerine Ayırma(Role Labeling) için sekans etiketleme örnekleri verilmiştir.

Giriş	İstanbul	önemli	finans	merkezi	.	
Çıkış	T_Yer	T	T	T	Noktalama	
Giriş	İstanbul	Üniversitesi	önemli	bir	lokasyondadır	.
Çıkış	Bas_Kurum	Bit_Kurum	T	T	T	Noktalama

.(T_Yer-Tekil Yer, Bas_Kurum-Başlangıç Kurum,Bit_Kurum-Bitiş Kurum,T-Tekil)

Tablo 1. Varlık İsmi Tanımlama (NER)

Giriş 1	Kelimeler	I	walk	fast office
Giriş 2	Kelime indeksi	1	2	3
Çıkış	Düzenlemeden sonraki indeksi	1	3	2

Tablo 2. Makine Çevirisi (Machine Translate)

Giriş	Bugünlerde	hava	çok	sıcak	.
Çıkış	Zarf Tümleci	Özne	Zarf Tümleci	Yüklem	Noktalama

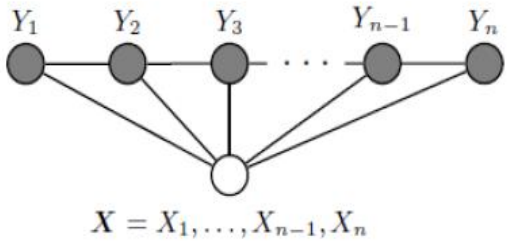
Tablo 3. Cümle Öğelerine Ayırma(Role Labeling)

Bir diziyi etiketleme de kullanılan algoritmalar olasılığa dayanan istatistiksel çıkarsama algoritmalarıdır. Dizi etiketleme de kullanılan en yaygın istatistiksel Çizge(Graf) tabanlı modeller HMM, MEMM ve CRF sistemleri dir. Yapılan çalışmalar sonucunda CRF' nin graf tabanlı modeller içinde en iyisi olduğunu göstermiştir. Şekil 1'de kullandığımız CRF'ye ait graf görülmektedir.

CRF makine öğrenmesi ve örüntü tanımada, yapılandırılmış veride kullanılan istatistiksel bir sınıflandırma yöntemidir. CRF Doğal Dil İşleme problemlerinde, giriş dizileri için etiket dizilerini tahmin etmede sıklıkla kullanılır [2].

CRF, Lafferty ve arkadaşları [3] tarafından önerilen istatistiksel dizilim sınıflandırmasına dayanan bir makine öğrenmesi yöntemidir. Dizilim sınıflandırıcıları bir dizilim içerisindeki her birime bir etiket atamaya çalışırlar. Olası etiketler üzerinde bir olasılık dağılımı hesaplar ve en olası etiket dizilimini seçerler.

Buna göre CRF modeli P(y, x) olasılığını hesaplamak üzere geliştirilmiş bir olasılık modeli olarak tanımlanabilir. Burada y =y1,...,yn olası çıktı etiketlerini belirtirken, x =x1,..,xn giriş verilerini belirtir. Buna göre CRF modeli Şekil 1 ve (1) ile gösterilebilir [2].



Şekil 1. CRF Modelinin Grafı

$$P(y, x)=\frac{1}{Z_\theta(x)}\exp\left(\sum_j \lambda_j t_j (y_{t-1}, y_i, x, i)+ \sum_k \mu_k s_k (y_i, x, i)\right) \quad \textbf{(1)}$$

Denklem (1)' de görüleceği üzere nitelik fonksiyonu parametreleri t. etiket (y_t) ve t-1. etiket (y_{t-1}) ve sözcük dizilimi x olan bir fonksiyonudur. Nitelik fonksiyonları makine öğrenmesinde kullanmak istenilen nitelikleri belirleyen fonksiyonlardır. Bu bağıntıda j, sekanslar arasındaki geçiş sayısını, k ise sekanstaki kelime sayısını ifade eder. $\sum_j \lambda_j t_j (y_{t-1}, y_i, x, i)$, sekanslar arasındaki geçişleri, $\sum_k \mu_k s_k (y_i, x, i)$, girişe karşılık gelen çıkışları ifade eder. IV.kısımdaki yaptığımız çalışmada ki tekil durumların çıkış kümesi üzerindeki etkisinin bağıntıdaki bu kısımdan kaynaklanmaktadır.

Sekans etiketlemede genelde çizge tabanlı yöntemler kullanılmaktadır. Bu çalışmada KMÖ yöntemlerinin bu uygulamalara uygunluğunu araştırılmıştır. Bunun içinde Yapay veri kümeleri üzerinde bu 2 temel bakış açısı karşılaştırılmıştır.

Yapay veri kümelerini kullanma sebebimiz, algoritmaların hangi durumlarda başarılı-başarısız olduğunu anlayabilmek için veri kümesinin tüm parametrelerinin kontrol altında olması gerektiğindendir. Gerçek veri kümelerinde bu parametreleri kontrol etmek mümkün değildi.

Yaptığımız ilk çalışmada Yapay veri setleri üzerinde sekans etiketleme işlemi için bir çalışma yaptık. Burada KMÖ yöntemleri, CRF den daha iyi sonuçlar elde etti. Ardından CRF' nin ne gibi durumlarda daha kötü sonuçlar elde ettiğini anlamak için yeni bir çalışma gerçekleştirdik.

Yapılan son çalışmalar gösterdi ki CRF' nin özellikleri bağımsız varsaymasının başarı üzerinde olumsuz sonuçlara yol açtığını bulduk. KMÖ yöntemlerinden aynı varsayıma sahip olan Naive Bayes' de aynı şekilde CRF gibi kötü performansa sahip olduğunu görüldü.

Çalışmanın 2. Bölümünde Yapay Veri Seti oluşturma, 3. Bölümünde çalışmada kullanılan yöntemler, 4. Bölümünde ilk yapılan çalışma ile ilgili bilgiler, 5. Bölümünde CRF ve NB' nin kötü performansının sebebini anlamak için yapılan çalışmaya ait bilgiler anlatılmaktadır.

II. SEKANS ETİKETLEME İÇİN YAPAY VERİ SETİ OLUŞTURMA

Bu bölümde yapay veri seti oluşturmak için izlenen yol anlatılmaktadır. Bunun için öncelikle kurallar tanımlanmalı ve ardından bu kuralların çalıştırılması ile yapay veri seti istenen boyutta oluşturulmaktadır.

A. Kural Oluşturma

Çalışmanın bu aşamasında yapay veri setleri oluşturmak için kurallar tanımlanmıştır. 4 farklı kural tanımlama şekli belirlenerek 4 farklı yapay veri seti oluşturulmuştur. Çalışmanın bir sonraki araştırma alanı gerçek veri setleri üzerinde Varlık İsmi Tanımlama belirlendiğinden veri setleri 2 girişli ve 1 çıkışlı olacak şekilde tasarlanmıştır.

1.VERİSETİNİ OLUŞTURMAK İÇİN KURALLAR

1.Veri setini oluşturmak için elimizde ;

Giriş Kümesi-1 {g1[0], g1[1], g1[2]}

Giriş Kümesi-2 {g2[0], g2[1], g2[2]}

Çıkış Kümesi E{x, y, z, t, u}, şeklinde kümelerin olduğu varsayımından yola çıkarak Tablo 4'deki kural oluşturma şablonu kullanılarak veri seti oluşturulmuştur.

Kural 1- Eğer $\mathbf{x_{1(t)}}$=**g1[1]** ve $\mathbf{x_{2(t)}}$= **g2[1]** ise **Çıkış=x**
Kural 2- değilse Eğer $\mathbf{x_{1(t)}}$= **g1[2]** ve $\mathbf{x_{2(t)}}$= **g2[1]** ise **Çıkış=y**
Kural 3- değilse Eğer $\mathbf{x_{1(t)}}$= **g1[0]** ve $\mathbf{x_{2(t)}}$= **g2[0]** ise **Çıkış=z**
Kural 4- değilse Eğer $\mathbf{x_{1(t)}}$= **g1[1]** ve $\mathbf{x_{2(t)}}$= **g2[1]** ise **Çıkış=t**
Kural 5- değilse Eğer $\mathbf{x_{1(t)}}$= **g1[2]** ve $\mathbf{x_{2(t)}}$= **g2[0]** ise **Çıkış=u**
Kural 6- Eğer $\mathbf{x_{1(t-1)}}$= **g1[2]** ve $\mathbf{x_{2(t)}}$= **g2[2]** ise **Çıkış=u**
Kural 7- değilse Eğer $\mathbf{x_{1(t-1)}}$= **g1[0]** ve $\mathbf{x_{2(t-1)}}$= **g2[1]** ise **Çıkış=t**
Kural 8- Eğer $\mathbf{x_{1(t+1)}}$= **g1[0]** ve $\mathbf{x_{2(t)}}$= **g2[1]** ise **Çıkış=y**
Kural 9- Eğer $\mathbf{x_{1(t+1)}}$= **g1[1]** ve $\mathbf{x_{2(t+1)}}$= **g2[0]** ise **Çıkış=x**
Kural 10- Eğer $\mathbf{x_{1(t+1)}}$= **g1[2]** ve $\mathbf{x_{2(t)}}$= **g2[2]** ise **Çıkış=z**
Kural 11- değilse Çıkış=(Rastgele üretilen (x,y,z,t,u)'dan biri)

Tablo 4. 1.Kural Kümesi Şablonu

2.VERİSETİNİ OLUŞTURMAK İÇİN KURALLAR

Bu veri setini oluşturmak için 3 farklı kural sistemi kullanılmıştır. Bu veri setini CRF ve NB' in neden diğer yöntemlerden kötü sonuçlar elde ettiğini göstermek için kullanacağız. Bunun için 4'er kurallı 2 ve 6 kurallı bir kural seti kullanılmıştır. Bunlarla ilgili kural tablosu Tablo 5' de görülmektedir.

2.Kural Kümesi Listesi
Kural 1- Eğer $\mathbf{x_{1(t)}}$= **g1[0]** ve $\mathbf{x_{2(t)}}$= **g2[0]** ise **Çıkış=z**
Kural 2- Eğer $\mathbf{x_{1(t)}}$= **g1[0]** ve $\mathbf{x_{2(t)}}$= **g2[1]** ise **Çıkış=t**
Kural 3- Eğer $\mathbf{x_{1(t)}}$= **g1[1]** ve $\mathbf{x_{2(t)}}$= **g2[0]** ise **Çıkış=x**
Kural 4- Eğer $\mathbf{x_{1(t)}}$= **g1[1]** ve $\mathbf{x_{2(t)}}$= **g2[1]** ise **Çıkış=y**
3.Kural Kümesi Listesi
Kural 1- Eğer $\mathbf{x_{1(t)}}$= **g1[0]** ve $\mathbf{x_{2(t)}}$= **g2[0]** ise **Çıkış=y**
Kural 2- Eğer $\mathbf{x_{1(t)}}$= **g1[0]** ve $\mathbf{x_{2(t)}}$= **g2[1]** ise **Çıkış=x**
Kural 3- Eğer $\mathbf{x_{1(t)}}$= **g1[1]** ve $\mathbf{x_{2(t)}}$= **g2[0]** ise **Çıkış=x**
Kural 4- Eğer $\mathbf{x_{1(t)}}$= **g1[1]** ve $\mathbf{x_{2(t)}}$= **g2[1]** ise **Çıkış=y**
4.Kural Kümesi Listesi
Kural 1- Eğer $\mathbf{x_{1(t)}}$= **g1[0]** ve $\mathbf{x_{2(t)}}$= **g2[0]** ise **Çıkış=x**
Kural 2- Eğer $\mathbf{x_{1(t)}}$= **g1[0]** ve $\mathbf{x_{2(t)}}$= **g2[1]** ise **Çıkış=y**
Kural 3- Eğer $\mathbf{x_{1(t)}}$= **g1[0]** ve $\mathbf{x_{2(t)}}$= **g2[2]** ise **Çıkış=y**
Kural 4- Eğer $\mathbf{x_{1(t)}}$= **g1[1]** ve $\mathbf{x_{2(t)}}$= **g2[0]** ise **Çıkış=x**
Kural 5- Eğer $\mathbf{x_{1(t)}}$= **g1[1]** ve $\mathbf{x_{2(t)}}$= **g2[1]** ise **Çıkış=x**
Kural 6- Eğer $\mathbf{x_{1(t)}}$= **g1[1]** ve $\mathbf{x_{2(t)}}$= **g2[2]** ise **Çıkış=y**

Tablo 5. 2-3-4. Kural Şablonları

B. Verisetleri oluşturma

A Kısmında türetilen kurallar yardımıyla 4 farklı veri seti oluşturulmuştur. Bu veri setlerini CRF ve KMÖ yöntemlerinde deneyebilmek için uygun formata çevirmek gerekmektedir. CRF için test setinde çıkışlar silinmiş, KMÖ yöntemleri için ise Weka programı kullanılacağından ötürü veri setleri arff dosya formatına çevrilmesi gerekmektedir. Bunun için Tablo 6' da gerekli olan dönüşümü yapmak için kullanılan şablon görülmektedir. Pencere boyutu olarak 3 seçilmiş ve mevcut kelimenin bir öncesine ve bir sonrasına bakılmıştır.

g1[0]	**g1[1]**	**g1[2]**	…
g2[0]	**g2[2]**	**g2[1]**	….
x	**z**	**x**	

Öz1	Öz2	Öz3	Öz4	Öz5	Öz6	Çıkış
?	**g1[0]**	**g1[1]**	?	**g2[0]**	**g2[2]**	x
g1[0]	**g1[1]**	**g1[2]**	**g2[0]**	**g2[2]**	**g2[1]**	z

Tablo 6. Sekansın 3 boyutlu pencere ile ifade edilmesi

III. KULLANILAN YÖNTEMLER

Bu kısımda çalışmada kullanılan yöntemler kısaca açıklanacaktır.

CRF ile ilgili detaylı bilgiler Giriş bölümünde verilmiştir.

Naive Bayes(NB) teoremi, olasılık kuramı içinde incelenen önemli bir konudur. Bu teorem bir rastlantısal değişken için olasılık dağılımı içinde koşullu olasılıklar ile marjinal olasılıklar arasındaki ilişkiyi gösterir. NB, hedef degişkenle bağımsız değişkenler arasındaki ilişkiyi analiz eden tahminci ve tanımlayıcı bir sınıflama algoritmasıdır [4]

IBK, kNN algoritması olarak ta bilinir. En yakın komşu yöntemine "tek bağlantı kümeleme yöntemi" ismi de verilmektedir. İlk anda tüm gözlem değerleri birer küme olarak değerlendirilir. Aşamalı olarak bu kümeler birleştirilerek yeni kümeler elde edilir. Bu yöntemde öncelikle gözlemler arasındaki mesafe belirlenir. K-en yakın komşu (k-nearest neigbour - kNN) algoritması benzerlik yoluyla öğrenme teknikleri kullanan algoritmalardan biridir [5].

Bagging, eğitim verisi setlerinin küçük boyutları ile çalışır. Özgün eğitim seti N adet alt kümelere bölünmüştür. Bu alt kümelerin her biri eğitim seti olarak kullanılır. Her bir alt küme aynı zamanda bir sınıflandırıcı oluşturur. Bu sınıflandırıcıları bir birleşik sınıflandırıcı bileştirir. Bu nedenle Bagging (torbalama) ismi verilmiştir [6].

J48, Bu algoritma karar ağacı tekniklerini kullanır. Yukarıdan aşağıya doğru böl ve yönet şeklinde bir ağaç yapısı oluşturur. Öznitelikler bu algoritmada düğüm noktası oluşturacak şekilde yerleştirilir ve eğitim verisine göre yapraklar meydana gelir. Yapraklar aynı zamanda sınıf etiketlerini belirtir [7].

IV. İLK ÇALIŞMA

Bu çalışmada Tablo 1' deki kurallar yardımıyla türetilen kurallar kullanılarak 50,100,500,2500 ve 5000 cümlelik eğitim ve 2500 cümlelik (52500 sekans) test seti kullanılmıştır. Yapılan çalışmaya ait sonuçlar Tablo 7' de görülmektedir.

Başarı (%)	CRF	NB	IBK	Bag.	J48
50 cümle	68.56	64.37	80.93	79.79	78.68
100 cümle	70.74	67.11	82.29	81.35	80.21
500 cümle	74.07	66.63	82.55	82.56	82.68
2500 cümle	75.28	66.72	82.78	82.78	82.83
5000 cümle	75.17	66.69	82.87	82.9	82.84

Tablo 7. İlk çalışma sonuçları

	x	y	z	t	u	Nokta
x	1467	744	0	1040	0	0
y	0	1434	796	867	0	0
z	815	814	1644	0	0	0
t	605	0	750	1907	0	0
u	0	0	0	0	1617	0
Nokta	0	0	0	0	0	725

Tablo 8. İlk Çalışma - CRF Karışım Matrisi
(Eğitim Seti 5000 cümle)

Tablo 7' den de görüldüğü üzere CRF, Naive Bayes dışındaki makine öğrenmesi yöntemlerinden daha başarısızdır.

CRF' nin sadece "u" çıkışı için %100 başarı göstermesinin nedeni, Eğer x1(t)= g1[1] ve x2(t)= g2[1] ise Çıkış=u kuralının dışında hiçbir kuralın u çıktısı üretmemesidir. Aynı durum Nokta çıktısı içinde söylenebilir. Dolayısıyla başka bir olasılık olmadığından girişler kuralı sağladığı durumda çıkış "u" olarak işaretlenir. Bölüm V' de yapılan denemeler bunu test etmek için yapılmıştır.

Eşitlik 1'de verilen denklemden hareketle, giriş1(X1) ve giriş2(X2) ve Çıkış (Y) etiketlerinden, girişlere göre çıkış etiketleri olasılıkları hesaplanıyor. Eşitlik 1' deki $\sum_k \mu_k s_k (y_i, x, i))$ kısmından görülebildiği üzere, girişlerimiz çıkışlarımız üzerinde direk etkilidir ve çıkış kümesine etiket atanırken girişlere göre olasılığı en fazla olan etiket atanmaktadır. Dolayısıyla çıkışlar girişlere bağımlı olduğundan elimizdeki eğitim setinin içinde barındırdığı tekil durumlar sonuç kümesi üzerinde etkili oluyor. Bu çalışmada da görüldüğü üzere, tekil durumlarda sistem doğruluğu yüksek iken tekillik bozulduğunda sistem başarısı düşük oluyor.

V. CRF VE NB' İN KÖTÜ PERFORMANSININ SEBEBİNİ ANLAMAK İÇİN YAPILAN DENEYLER

Bu kısımda Tablo 5' de verilen kural setleri kullanılarak oluşturulan veri setleri kullanılmış ve Tablo 9' daki sonuçlara ulaşılmıştır. Eğitim seti 9555 sekans, test seti 15750 sekanstır.

	CRF	NB	IBK	Bag.	J48
2.Kural Kümesi	100	100	100	100	100
3.Kural Kümesi	52	52	100	100	100
4.Kural Kümesi	100	100	100	100	100

Tablo 9. İkinci Çalışma Sonuçları(Doğruluk %)

	g2[0]	g2[1]
g1[0]	z	t
g1[1]	x	y

Tablo 10. 2.Kural Kümesi Oluşturma Matrisi

	g2[0]	g2[1]
g1[0]	y	x
g1[1]	x	y

Tablo 11. 3.Kural Kümesi Oluşturma Matrisi

	g2[0]	g2[1]	g2[2]
g1[0]	x	y	y
g1[1]	x	x	y

Tablo 12. 4.Kural Kümesi Oluşturma Matrisi

	x	y	Nokta
x	3774	3820	0
y	3715	3691	0
Nokta	0	0	750

Tablo 13. 3.Kural Kümesi - CRF Karışım. Matrisi

	x	y	Nokta
x	3788	3806	0
y	3752	3654	0
Nokta	0	0	750

Tablo 14. 3.Kural Kümesi - NB Karışım Matrisi

Tablo 10 ve Tablo 12' de görüldüğü gibi çıkışlar tek bir dağılımdan geldiğinde CRF ve NB, KMÖ yöntemleriyle benzer başarı gösterirken, Tablo 11' deki çıkışların birden fazla dağılımdan geldiğinde CRF ve NB' nin başarıları düşmektedir.

VI. DEĞERLENDİRME VE İLERİ UYGULAMALAR

Yapılan çalışmalar sonucunda, CRF' nin NB dışındaki makine öğrenmesi yöntemlerinden daha geride kaldığı görülmüştür. Bunda özellikleri bağımsız kabul etmesinin rolu olabileceği varsayımı ile yaptığımız deneyler tıpkı özellikleri bağımsız kabul eden NB'de benzer şekilde olması tezimizi kanıtlamaktadır.

Eğitim setinin boyutunun artırılmasının başarıya kademeli olarak artırdığı eğitimin bittiği noktadan itibaren ufak salınımlar yaptığı görülmüştür.

Bir sonraki aşama da yapay veri seti üzerinde ki bu çalışmayı gerçek veri setleri üzerinde deneyerek sonuçlarını gözlemlemek istiyoruz.

Ayrıca CRF nin çıkışların tek bir dağılımdan geldiği varsayımına sahip olmayan bir versiyonunun geliştirilmesi planlanmaktadır.

KAYNAKÇA

[1] A. Mccallum, K. Rohanimanesh, and C. Sutton. Dynamic Conditional Random Fields for Jointly Labeling Multiple Sequences. NIPS Workshop on Syntax, Semantics and Statistics, 2003.

[2] Lafferty, J., McCallum, A., Pereira, F. C. N., (2001) "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data".

[3] Lafferty, J., McCallum, A. ve Pereira, F., 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data, Proc. 18th International Conf. on Machine Learning, Morgan Kaufmann, San Francisco, CA, s.282–289.

[4] Hudairy, H., "Data mining and decision making support ın the governmental sector", Master Thesis, Louisville University, Kentucky, 1-5 (2004).

[5] Özkan Y. Veri Madenciliği Yöntemleri. 1st ed. Çölkesen R, editor. İstanbul, Türkiye: Papatya Yayıncılık; 2008.

[6] Machova K, Barcak F, Bednar P. A Bagging Method using Decision Trees in the Role of Base Calssifiers. Acta Polytechnica Hungarica. 2006; 3(2).

[7] Kargupta H, Han J, Yu PS, Motwani R, Kumar V. Next Generation of Data Mining Minnesota: Chapman & Hall/CRC; 2008.