Topluluk Öğrenmesi ve Karma Tipli Öznitelikler ile Film Derecelendirme Puanı Tahmini

Movie Rating Prediction Using Ensemble Learning and Mixed Type Attributes

Ayşegül ÖZKAYA EREN ve Mustafa SERT
Bilgisayar Mühendisliği Bölümü
Başkent Üniversitesi
Ankara, Türkiye
21610279@mail.baskent.edu.tr, msert@baskent.edu.tr

***Özetçe*— Günümüzde kullanıcılar izledikleri bir film hakkında duygu ve düşüncelerini, internet aracılığıyla anında paylaşabilir hale gelmişlerdir. Kullanıcı derecelendirme puanlarının otomatik yöntemlerle tahmin edilmesi, gişe hasılatını tahmin edebilmek açısından sinema sektörü için çok önemli bir göstergedir. Bu nedenle film derecelendirme puanı tahmini, makine öğrenmesi alanında üzerinde çalışılan popüler konulardan biri olmuştur. Mevcut çalışmaların çoğunda veri kümeleri içerisindeki sayısal öznitelikler kullanılırken, sayısal olmayan özniteliklerin kullanımı görece kısıtlıdır. Bu çalışmada, film derecelendirme puanı tahmini için, sayısal ve sayısal olmayan özniteliklerin bir arada kullanımına ve topluluk öğrenmesi (ensemble learning) yaklaşımına dayalı bir yöntem önerilmektedir. Önerilen yöntemin etkinliği ve başarımı Internet Movie Database (IMDb) performans veri kümesi üzerinde, literatürdeki farklı yöntemlerle karşılaştırmalı olarak doğrulanmıştır. Elde edilen sonuçlar, karma öznitelik kullanımının, topluluk öğrenmesi yöntemi ile puan tahminini iyileştirdiğini göstermektedir.**

Anahtar Kelimeler — Film derecelendirme puanı tahmini; topluluk öğrenmesi; IMDb.

***Abstract*— Nowadays, audience can easily share their rating about a movie on the internet. Predicting movie user ratings automatically is specifically valuable for prediction box office gross in the cinema sector. As a result, movie rating prediction has been a popular application area for machine learning researchers. Although most of the recent studies consider using mostly numerical features in analyses, handling nominal features is still an open problem. In this study, we propose a method for predicting movie user ratings based on numerical and nominal feature collaboration and ensemble learning. The effectiveness and the performance of the proposed approach is validated on Internet Movie Database (IMDb) performance dataset by comparing with different methods in the literature. Results show that, using mixed data types along with the ensemble learning improves the movie rating prediction.**

Keywords — Movie Rating prediction; ensemble learning; IMDb.

I. GİRİŞ

Film derecelendirme puanı (FDP) tahmini üzerine pek çok araştırmacı, farklı veri kümesi ve farklı öğrenme algoritmaları kullanarak çalışmışlardır. Internet Movie Database (IMDb) veri kümesi bu alanda çok kullanılan veri kümelerinden biridir ve kullanıma açık, ücretsiz olarak dağıtılmaktadır. IMDb veri kümesi, filmlerin yönetmenleri, oyuncuları, dili, süresi, bütçesi gibi sayısal ve nominal (sayısal olmayan) pek çok öznitelik içerir ancak öğrenme algoritmalarının çoğu sayısal değerleri kullanabiliyorken, nominal değerlerin bir ön işlemden geçirilmesi gerekir [1].

Literatürde yapılan çalışmaların çoğunda araştırmacılar veri kümesinin sayısal veya nominal özniteliklerine ayrı çalışmalar uygulamışlardır. Bu öznitelikleri birlikte kullanan çalışmalar ise, algoritmalarına göre veri tiplerini tek bir veri tipine çevirerek verileri ön işlemden geçirmişlerdir. Bu alandaki araştırmalar, kullandıkları yaklaşımlara göre içerik tabanlı, işbirlikçi ve karma yöntemler olmak üzere üç grupta incelenebilir [2]. İçerik tabanlı yöntemler, puanı kestirilmek istenen filmin özelliklerine benzer filmleri bulmaya yönelik bir model oluştururken, işbirlikçi yöntemler bir kullanıcının verdiği puana göre benzer filmleri izlemiş diğer kullanıcıların seçili filme kaç puan vereceğini bulmaya çalışır. Karma yöntemler ise içerik tabanlı yöntemler ile işbirlikçi yöntemlerin sonuçlarını birleştirerek yeni bir model ortaya koyar. En yakın komşu algoritması (K-NN), karar ağaçları, yapay sinir ağları ve bazı özel algoritmalar yukarıdaki yöntemler esas alınarak bu alandaki çalışmalarda sıklıkla kullanılmıştır.

Hsu ve arkadaşları doğrusal kombinasyon ile derecelendirme puanı tahmini yapmıştır [3]. Bu yöntemde IMDb veri kümesi ve bu veri kümesinin yönetmen, aktör, tür gibi nominal öznitelikleri kullanılmıştır. Sayısal değer olan süre özniteliği ise kategorik değerlere dönüştürülerek kullanılmıştır. Rapor edilen tahmin mutlak hata (Prediction Absolute Error-PAE) oranı, uyguladıkları 3 farklı algoritma için 0.82'nin altındadır.

978-1-5090-6494-6/17/$31.00 ©2017 IEEE

Debnath ve arkadaşları, özniteliklere farklı ağırlıklar vererek IMDb veri kümesini kullanmışlar ve doğrusal kombinasyon yöntemiyle elde ettikleri öznitelik ağırlıklarını birleştirmişlerdir. Sayısal ve nominal öznitelikler bir arada kullanılmış, nominal özniteliklere kategorik değerler atanmıştır. Çalışmanın sonucunda özniteliklerin ağırlıklarını doğru bir şekilde değiştirince, duyarlılık oranının arttığı tespit edilmiştir [4].

Wernard Schmit ve Sander Wubben Twitter içeriği kullanarak FDP tahmin etmeye çalışmışlardır [5]. Twitter'da konusu geçen filmlerin, kullanıcıların yorumlarını içerdiğinden yola çıkmışlar ve Twitter içeriğini işleyerek, Destek Vektör Makinesi (DVM) öğrenme yöntemini kullanmışlardır. Biramane ve arkadaşları tarafından yapılan bir çalışmada ise filmler hakkında sosyal medyadan elde edilen bilgiler ile filmlerin aktör, yönetmen vb gibi öznitelikleri arasındaki ilişki bulunmaya çalışılmış ve sosyal medya bilgilerinin (Youtube, Wikipedia) de film derecelendirme puanı tahmini yapmakta önemli bilgiler içerdiğini göstermişlerdir [6].

Jing Gao ve arkadaşları ise NetFlix veri kümesi üzerinde işbirlikçi yöntemler ile topluluk öğrenme algoritması üzerine çalışmışlar ve ortalama Karesel Hatanın Karekökü (Root Mean Squared Error-RMSE) 0.87 olarak hesaplamışlardır [7].

Bu bildiride, karma tipteki özniteliklerin ayrı ayrı ve birlikte kullanımının farklı tahmin algoritmalarındaki başarımları içerik tabanlı öğrenme yöntemi esas alınarak analiz edilmiştir. Benzer çalışmalardan farklı olarak, farklı veri tiplerindeki öznitelikler ve topluluk algoritması bir arada kullanıldığında hata oranının azalacağı öngörülmüştür. Bu amaçla, analizlere 3 farklı öğrenme algoritması dahil edilmiş, algoritmalardan 2 tanesi için veriler arası dönüşüm işlemi uygulanırken, diğer algoritmada dönüşüm işlemi uygulanmamıştır. Böylelikle, sayısal ve nominal karma veriler olarak adlandırdığımız özniteliklerin tek bir algoritmada kullanılması sağlanmıştır. Çalışma kapsamında öğrenme algoritmaları olarak K-NN ve karar ağaçları algoritmaları kullanılmıştır. Karar ağaçları algoritmaları için birden fazla karar ağacı ile topluluk öğrenmesi algoritması oluşturan TreeBagger (Bag of Decision Trees) algoritması [8] ile sayısal, nominal ve ilişkisel veri tiplerini bir arada kullanabilen Trestle Tree algoritması seçilmiştir [9].

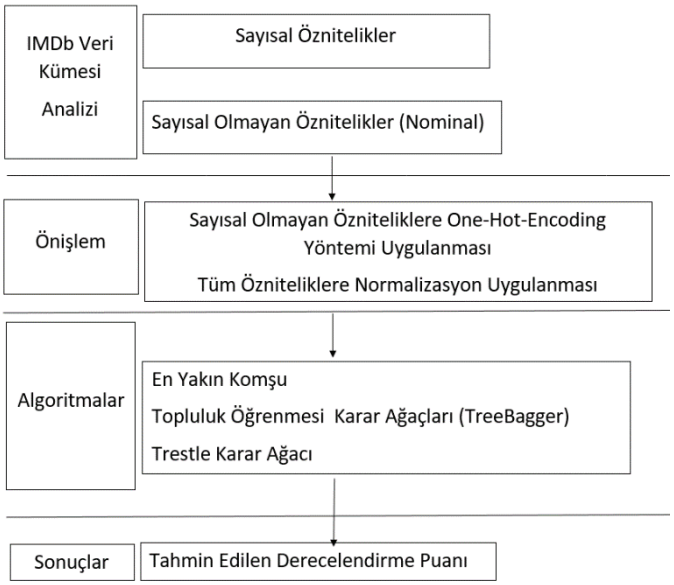
Bu bildirinin diğer bölümleri şu şekilde düzenlenmiştir: 2. bölümde, önerilen sınıflandırma algoritmaları ve öznitelik analizleri tanıtılmıştır. Bölüm 3'te, deneyler ve değerlendirme sonuçları, Bölüm 4'te ise çalışmanın sonuçları, kısıtları ve gelecek çalışma planları sunulmuştur.

II. YÖNTEM

Önerilen FDP sistemi iki aşamadan oluşmaktadır. İlk aşamada, veri kümesine ön işlem uygulanarak öznitelik analizleri gerçekleştirilmektedir. İkinci aşamada, elde edilen öznitelik temsilleri kullanılarak tahmin yapılmaktadır. Önerilen sistemin genel yapısı Şekil 1'de sunulmuştur.

A. Veri Kümesi Analizi

Kullandığımız IMDB veri kümesi 5044 veri ve 27 öznitelikten oluşmaktadır. IMDb veri kümesinin içerdiği 27 öznitelikten



Şekil 1: Sistem Blok Şeması

bazıları "film adı", "yapılan yorum sayısı", "bütçe" "yönetmen adı", "1. Oyuncu adı", "1. Oyuncuya yapılan facebook beğeni sayısı" şeklindedir. Bu öznitelikler içerisinden en çok katkısı olan özniteliklerin seçimi için bilgi kazançları hesaplanmış ve buna göre öznitelikler arasında eleme yapılmıştır. Bu sonuçlar dikkate alınarak 12 öznitelik elde edilmiştir. Şekil 2'de öznitelikler ve bilgi kazançları, Tablo 1'de ise seçilen öznitelikler ve veri tipleri hakkında bilgi verilmiştir. Bilgi kazancı değerleri (1) ve (2) [10] kullanılarak hesaplanmıştır.

$$\text{Entropi}(S) = \sum_{i=1}^{c} -p_i \log_2 p_i \quad (1)$$

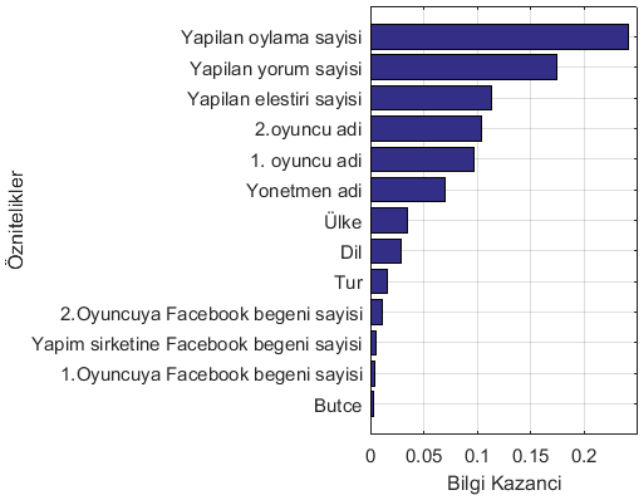
$$\text{Kazanç}(S,A) = \text{Entropi}(S) - \sum_{v \in \text{Değerler}(A)} (|S_v| / |S|)\, \text{Entropi}(S_v) \quad (2)$$

Denklem (1) ve (2)'de S örnek kümesini, p S kümesinin i sınıfına ait kısmının oranını, c hedef sınıfın kaç değer alabileceğini, A seçili özniteliği, *Değerler(A)* A özniteliğinin olası değerlerini, S_v ise A özniteliğinin v değerine sahip olduğu alt kümeleri temsil etmektedir.

B. Sınıflandırma Algoritmaları

Öznitellik seçiminden sonra, veri kümesine sistem blok şemasında görüldüğü gibi, 3 farklı algoritma uygulanmıştır. K-NN ve TreeBagger algoritmaları MatLab [8] kullanılarak geliştirilmiş, Trestle Tree algoritması ise Python programlama dili ile geliştirilmiştir. K-NN algoritmasında Euclidean uzaklığı yöntemi veri kümesine uygulanmıştır.

TreeBagger algoritması topluluk öğrenme algoritması olan karar ağacı yöntemidir. Birden fazla karar ağacı oluşturularak,



Şekil 2: Öznitelikler Bilgi Kazancı

Tablo 1 Veri Kümesi Öznitelikleri

Öznitelik Adı	**Veri Tipi**
Bütçe	Sayısal
Yapılan yorum sayısı	Sayısal
Oylayan kullanıcı sayısı	Sayısal
Yapılan eleştiri sayısı	Sayısal
Yapım şirketine yapılan Facebook beğeni sayısı	Sayısal
1.Oyuncuya yapılan Facebook beğeni sayısı	Sayısal
2.Oyuncuya yapılan Facebook beğeni sayısı	Sayısal
Yönetmen adı	Nominal
1. Oyuncu adı	Nominal
2. Oyuncu adı	Nominal
Dil	Nominal
Ülke	Nominal
Tür	Nominal

her ağaçtan farklı bir sonuç elde edilir ve sonuçlar birleştirilir. Karar ağacı sayısı ve sınıflandırma metodu (sınıflandırma-regresyon) parametreleri ile verilen sayıda ağaç oluşturur. Literatürde yapılan topluluk algoritmalarının başarı oranı yüksektir [8], bu nedenle karma veri tipi deneylerinden biri için topluluk algoritması seçilmiştir.

Trestle Tree algoritması da bir karar ağacı algoritmasıdır [9]. Bu karar ağacı algoritmasında, insan beyninin önceki tecrübelerden yararlanarak, her tip veriyi işleyerek ve veriler arası ilişki kurarak öğrendiğinden yola çıkılarak bir algoritma oluşturulmuştur. FDP tahmini yapılırken de, veri kümesindeki tüm verilerin aynı anda değerlendirilebilmesi amacıyla bu yöntem çalışmamıza uygulanmıştır. Trestle Tree ilk olarak bir eğitim oyunu için geliştirilmiştir. Elde edilen sonuçların başarı oranının önceki çalışmaları yakaladığı ortaya konmuştur.

III. DENEYSEL ÇALIŞMALAR VE DEĞERLENDİRMELER

A. Veri Kümesi

Veri analizi yapıldıktan sonra elde edilen öznitelikler üzerinden, veri kümesinde eksik bilgi içeren kayıtlar veri kümesinden çıkarılmıştır. Bu işlemden sonra, elde edilen 4492 veri, 3499 tanesi eğitim ve 993 tanesi test veri kümesi olmak üzere 2 parçaya bölünmüştür.

K-NN ve TreeBagger karar ağacı algoritmaları sadece sayısal değerlerle çalıştığı için, nominal değerlerin kullanımı için One-Hot-Encoding (OHE) yöntemi kullanılmıştır. Örnek olarak yönetmen adlarının her biri bir öznitelikmiş gibi ele alınmış ve veri kümesindeki her bir veri için bu özniteliğe sahip olup olmadığına bakılmıştır. Her bir öznitelik bir bit ile gösterilmiştir. Filmin yönetmeni, öznitelik olarak belirtilen yönetmene eşit olduğu durumda bit 1 ile gösterilirken, eşit olmadığı durumlar için 0 ile gösterilmiştir. Bu uygulama ile öznitelik sayısı, yönetmen adı sayısı kadar artmıştır. Trestle Tree algoritması ise sayısal ve nominal değerleri aynı anda kabul ederek çalışabilir. Bu nedenle nominal değerlere ayrıca bir işlem uygulanmamıştır. Tüm değerler, tüm algoritmalarda [-1,1] değer aralığında normalize edilmiştir.

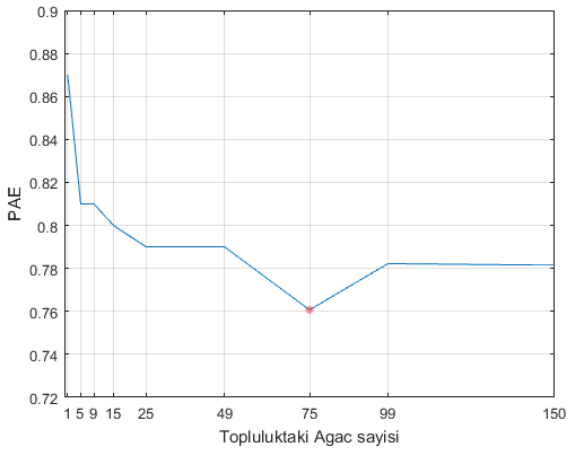
B. Deneysel Sonuçlar

Çalışmanın sonuçlarını değerlendirmek için literatürde bu alanda kullanılan değerlendirme yöntemlerinden biri olan Tahmin Mutlak Hata (PAE) yöntemi [3] seçilmiştir. PAE ve ortalama PAE değerleri (3) ve (4) kullanılarak hesaplanmıştır. K-NN için komşu sayısı 7 seçildiğinde elde edilen sonuçlar K-NN için en yüksek sonucu vermiştir (Tablo 1). TreeBagger algoritması ise farklı ağaç sayıları ile denenmiş, ağaç sayısına bağlı olarak elde edilen PAE grafiği Şekil 3'te sunulmuştur. Topluluktaki ağaç sayısı 75 olduğunda en iyi sonuç alınmış, ağaç sayısı büyüdükçe hata oranı sabit kalmıştır.

$$\text{PAE} = |\,\text{Tahmini Puan} - \text{Gerçek Puan}\,| \tag{3}$$

$$\text{Ortalama PAE} = \frac{\sum_1^n \text{PAE}_n}{n} \tag{4}$$

Uygulanan algoritma sonuçları ve literatürde aynı değerlendirme yöntemini kullanan bir çalışmanın sonuçları Tablo 2 ve Tablo 3'te karşılaştırılmıştır. Tablo 2 ve Tablo 3'te verildiği üzere, en iyi sonuçlar karma veri tipi ile topluluk öğrenme algoritmasından elde edilmiştir.



Şekil 3: Topluluktaki Ağaç sayısı ve PAE Oranları

Tablo 2 PAE yöntemi ile Çalışma sonuçları ve Literatür Sonuçları

Algoritma	**Kullanılan Öznitellik Tipi**	**PAE**
K-NN	Sayısal	0.95
	Nominal	1.06
	Sayısal + Nominal	0.88
TreeBagger	Sayısal	0.81
	Nominal	0.94
	Sayısal + Nominal	**0.76**
Trestle Tree	Sayısal	1.04
	Nominal	0.97
	Sayısal + Nominal	0.94
Linear Prediction [3]	Sayısal + Nominal	0.73
Multiple Linear Regression [3]	Sayısal + Nominal	0.81
Neural Networks [3]	Sayısal + Nominal	0.69

Tablo 3 PAE yöntemi ile belli eşik değerleri için çalışmanın doğruluk oranları ve Literatür Sonuçları

Algoritma	**PAE**	**PAE<1**	**1<=PAE<2**	**2<=PAE<3**	**3<=PAE<4**
K-NN	0.92	%61.4	%28.4	%8.05	%1.9
Trestle Tree	0.94	%61.1	%25.98	%8.16	%5.94
TreeBagger	**0.76**	**%71.25**	**%23.76**	%4.6	%0.4
Linear Prediction[3]	**0.73**	**%72.73**	**%24.45**	%2.19	%0.31
Multiple Linear Regression[3]	**0.81**	**%67.08**	**%28.21**	%4.08	%0.31
Neural Networks [3]	**0.69**	**%76.8**	**%18.5**	%4.39	%0.31

Tüm algoritmalar 3.40 GHz*2 ve 12 GB RAM'e sahip bir bilgisayarda çalıştırılmıştır. TreeBagger algoritması performans açısından değerlendirildiğinde de en hızlı çalışan algoritmadır ve 75 ağaç sayısı için çalışma süresi 3 dakikadır, daha yüksek ağaç sayıları için bu süre artmıştır. K-NN algoritması, Trestle Tree algoritmasından daha iyi sonuç vermiştir. 7 ağaç sayısı için çalışma süresi yaklaşık 5 dakikadır.

Trestle Tree algoritması ise diğerlerinden daha yüksek hata oranına sahiptir. Performans açısından da daha yavaş çalışmaktadır, çalışma süresi seçilen özniteliklere göre bir 0.5-2 saat aralığında değişmiştir. Ayrıca Trestle Tree'de ortalama hata değerinin son sınıflandırma aralığı olan PAE 3 ve 4 aralığında diğer algoritmalara göre daha yüksek olması, bu sınıflandırıcının başarı oranını düşürmüştür. Çalışma sonuçları, literatürdeki çalışmalar ile kıyaslandığında ise TreeBagger, Multiple Linear Regression yönteminden daha başarılıdır.

Doğruluk oranları kıyaslandığında ise PAE'nin en küçük olma durumu için TreeBagger, Multiple Linear Regression yöntemini geçmiştir. PAE'nin 2'den küçük olma durumunda ise TreeBagger %95.01, Multiple Linear Regression %95.29, Neural Networks algoritması %95.3, Linear prediction algoritması ise %97.18 doğruluk oranı göstermiştir.

Uygulanan 3 yöntemde de en iyi sonucu karma özniteliklerin kullanımı vermiştir. Sonuçlar, bu çalışmanın önerdiği sayısal ve nominal özniteliklerin bir arada kullanılması düşüncesini desteklemektedir.

IV. SONUÇLAR

Bu çalışmada, karma öznitelik kullanımına ve topluluk öğrenmesi (ensemble learning) yöntemine dayalı film derecelendirme puanı tahmin sistemi önerilmiştir. Karma özniteliklerin seçiminde özniteliklerin bilgi kazançları baz alınmıştır. Topluluk öğrenme algoritması olarak TreeBagger yöntemi probleme uygulanmış ve sonuçlar K-NN ve Trestle Tree yöntemlerine ek olarak literatür çalışmaları ile karşılaştırılmıştır. Deneysel sonuçlar, seçilen topluluk öğrenmesi algoritmasının K-NN ve Trestle Tree algoritmalarından daha başarılı olduğunu göstermektedir. Bütün yöntem testlerinde karma öznitelik kullanıldığında tahmin hata oranının düştüğü görülmüştür (Tablo 2). Bu yöntem farklı algoritmalarla birleştirilerek veya özniteliklere farklı ağırlıklar vererek denendiğinde daha başarılı sonuçlar elde edilebilir. Karma veri tiplerinin kullanımının, başka tahmin problemlerinde de hatayı azaltabileceği değerlendirilmektedir.

Başarı oranını artırmak amacıyla, nominal değerlerin sayısal değerlere uyarlanması işlemi için farklı yöntemler denenebilir. Makine öğrenmesi konusunda hala açık bir problem olan nominal değerlerin işlenmesi konusunda denenecek yeni yöntemler, bu çalışmanın başarısını da olumlu yönde etkileyebilir. Gelecek çalışmalar kapsamında, nominal değerlerin işlenmesi konusunda çalışmalar planlanmaktadır.

KAYNAKLAR

[1] Zdravevski E., Lameski P., Kulakov A., Advanced Transformations for nominal and Categorical Data into Numeric Data in Supervised Learning Problems, The 10th Conference for Informatics and Information Technology (CIIT), 2013

[2] Marovi´c M., Mihokovi´c M., Mik˘sa M., Pribil S., and Tus A., Automatic movie ratings prediction using machine learning, MIPRO 2011

[3] Hsu P., Shen Y., and Xie X., Predicting Movies User Ratings with Imdb Attributes, International Conference on Rough Sets and Knowledge Technology, 2014

[4] Debnath S., Ganguly N., Mitra P., Feature Weighting in Content Based Recommendation System Using Social Network Analysis, WWW 2008/Poster Paper, Beijing, China, April 21–25, 2008

[5] Schmit W., Wubben S., Predicting Ratings for New Movie Releases from Twitter Content Proceedings of the 6th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis (WASSA 2015), p. 122–126

[6] Biramane V., Kulkarni H., Bhave A., Kosamkar P., Relationships between Classical Factors, Social Factors and Box Office Collections , 2016 International Conference on Internet of Things and Applications (IOTA), India, 2016

[7] Gao J., Fan W., Han J., On the Power of Ensemble: Supervised and Unsupervised Methods, SDM'2010 Columbus, 2010

[8] https://www.mathworks.com/help/stats/treebagger.html

[9] J. MacLellan J., Harpstead E., Aleven V., Koedinger Kenneth R., "TRESTLE: A Model of Concept Formation in Structured Domains", Advances in Cognitive Systems, 2016

[10] Mitchell T. M., "Machine Learning" , McGra-Hill , 1997