

İSTANBUL TEKNİK ÜNİVERSİTESİ BİLGİSAYAR VE BİLİŞİM FAKÜLTESİ

YouTube için Regresyon Analizi Yaparak Müzik Öneren Chrome Uzantısı

Bitirme Projesi Son Raporu

**Alptekin Aktaş
150150005**

**Departman: Bilgisayar Mühendisliği
Bölüm: Bilgisayar Mühendisliği**

Danışman: Asst. Prof. Dr. Mehmet Tahir Sandıkkaya

Ağustos 2022

Statement of Authenticity

I/we hereby declare that in this study

1. all the content influenced from external references are cited clearly and in detail,
2. and all the remaining sections, especially the theoretical studies and implemented software/hardware that constitute the fundamental essence of this study is originated by my/our individual authenticity.

İstanbul, August 2022

Alptekin Aktaş

YouTube için Regresyon Analizi Yaparak Müzik Öneren Chrome Uzantısı

(ÖZET)

Müzik önerme algoritmaları, internet ile hızla büyüyen müzik sektörünün vazgeçilmez bir parçası haline geldi. Herkesin müzik platformlarında kolaylıkla şarkı paylaşabilir hale gelmesiyle erişebileceğimiz şarkılar büyük bir boyuta erişti. Bu durum beraberinde de birçok meta veriyi getirdi ve dinleyicilerin şarkıları meta verilerine göre sınıflandırıp dinlemesine olanak sağladı. Müzik platformları ise müzik önerme sistemleri geliştirerek, dinleyicilere uygun şarkıları önermeye başladı.

Bu çalışmada, müziğin meta verilerinden birisi olan duyguyu baz alarak YouTube ile çalışabilecek bir Chrome uzantısı geliştirildi. Öncelikle, yaklaşık 2000 adet şarkının duygu değişkenlerinin verilerinden ve ses dosyalarından oluşan DEAM/MediaEval veri seti eğitim veri seti olarak belirlendi. Bu veri setinde bulunan ses dosyalarının ses analizleri yapıldı ve bu analizlerin sonucu yeni bir veri setinde tutuldu. MSD (Million Song Dataset)'nin sağladığı bir müzik meta veri seti test veri seti olarak belirlendi. Test veri setindeki şarkıların YouTube URL'leri YouTube Data API yardımı ile YouTube'dan alındı ve ses analizleri yapıldı. Rastgele orman (Random Forest) regresyonu kullanılarak şarkıların duygu değişkenlerinin değerleri bulundu. Elde edilen bütün veriler test veri setine eklendi. Bu işlemleri Chrome uzantısı ile çalışabilecek hale getirmek için bir API yapıldı ve Chrome uzantısına entegre edildi.

Chrome Extension for Music Recommendation on YouTube Based on Regression Analysis

(SUMMARY)

Music recommendation algorithms have become an indispensable part of the rapidly growing music industry with the internet. With everyone being able to share music on the music platforms, the amount of accessible songs has reached great sizes. This environment came with a good deal of metadata, and allowed listeners to categorize and listen to the songs according to their metadata. As for music platforms, they started to develop music recommendation systems and recommend suitable songs to listeners.

In this study, a Chrome extension that is compatible with YouTube was developed on the basis of emotions which is one of the metadata of the songs. Firstly, the DEAM/MediaEval data set consisting of the data of the emotion variables of 2000 songs and audio files was determined as the training data set. Audio analyses of the audio files in this dataset were performed and the results of these analyses were stored in a new dataset. A music meta dataset provided by MSD (Million Song Dataset) was chosen as the test dataset. The YouTube URLs of the songs in the test dataset were obtained from YouTube with the help of YouTube Data API and audio analyses were made on these songs. The values of the emotion variables of the songs were found with the random forest regression model. All obtained data through these processes were added to the test dataset. To make these processes work with the Chrome extension, an API was made and integrated into the Chrome extension.

İçindekiler

1	Giriş ve Proje Özeti	1
2	Karşılaştırmalı Literatür Taraması.....	2
2.1	Duygu Modelleri ve Müzik Duygu Veri Setleri	2
2.1.1	Duygu Modelleri.....	2
2.1.2	Müzik Duygu Veri Setleri	3
2.2	Müzik Meta Veri Setleri ve Ses Dosyaları	4
2.2.1	Müzik Meta Veri Setlerinin Elde Edilmesi	4
2.3	Ses Özellikleri ve Analizi	5
2.3.1	Spektrogram, Mel spektrogram ve MFCC	5
2.3.2	Tempo ve Tempogram.....	6
2.3.3	Kroma Özelliği ve Tonnetz	7
2.3.4	Sıfır Geçiş Oranı	7
2.3.5	Spektral Özellikler	7
2.4	Regresyon Modelleri	8
2.4.1	Küçük Kareler Doğrusal Regresyonu	8
2.4.2	Ridge Regresyonu.....	8
2.4.3	Lasso Regresyonu	9
2.4.4	Bayesian Doğrusal Regresyonu	9
2.4.5	SVR.....	10
2.4.6	k-NN Regresyonu	10
2.4.7	Karar Ağacı Regresyonu.....	11
2.4.8	Rastgele Orman Regresyonu	12
2.4.9	MLP Regresyonu	12
2.5	Chrome Uzantısı	13
3	Geliştirilmiş Yaklaşım ve Sistem Modeli	14
3.1	Veri Modeli.....	14
3.2	Yapısal Model.....	15
3.3	Dinamik Model.....	18
4	Deney Ortamı ve Deney Dizaynı.....	19
5	Karşılaştırmalı Değerlendirme ve Tartışma.....	20
6	Sonuç ve Olasılıklar.....	21
7	Referanslar	22

1 Giriş ve Proje Özeti

Gelişen teknoloji ile müzik de artık dijital raflarda yer almaktadır. Bu sayede insanlar çok fazla müziğe, çok kolay bir şekilde erişim sağlayabilmektedirler. İnsanların müzik dinleme alışkanlıkları değişmiş; müzik, meta veriler ile sınıflandırılarak dinlenilmeye, sunulmaya başlanmıştır. Asıl amaçlarından biri duyguyu aktarmak, yaşatmak olan müziğin en önemli meta verilerinden birisi de duygusudur.

Dijital müzik platformlarında sanatçılar kendi şarkılarının çoğu meta verisini kendileri sağlasa da şarkıdaki duygu bunlardan birisi değildir. Bu noktada müziğin duygusunun teşhisi önemli bir konu haline gelmektedir. Müziğin duygusunun teşhisi; müzik enstitüsünde sıkça kullanılan müzik önerme algoritmalarının, gün geçtikçe sayısı artan müzik besteleyen yapay zekaların ve benzeri konuların temelini oluşturmaktadır.

Müziğin duygusunu öğrenebilen algoritmalar çoğu dijital müzik platformunda kullanılmaktadır. Geçtiğimiz yıllarda The Echo Nest'i bünyesine katan Spotify, bunun en büyük örneklerinden birisidir [1]. Dinleyicinin dinlediği müziğin duygusuna göre dinleyiciye özgü çalma listesi oluşturma, benzer müzikleri önerme gibi noktalarda müziğin duygusunun teşhisine başvurulmaktadır. Bir video platformu olan YouTube, bünyesinde her ne kadar şarkı videoları barındırsa da böyle bir sisteme sahip değildir. YouTube, müzik meta verisinden ziyade video meta verisini tutmaktadır [2].

YouTube, video önerirken kullanıcıların izlenme geçmişine, izledikleri kanallara ve videoların başlıklarına bakmaktadır [3]. Bu bilgilerin maalesef müzikal anlamda bir değeri yoktur ve bu sebeple önerilen videoların müzik videosu olmama ihtimali ile birlikte, önerebileceği şarkılar da dinleyicilerin beklentisini karşılayamamaktadır. Fakat herhangi bir müzik meta verisini kullanmadığı için, önerilen şarkılar bir o kadar da birbirinden bağımsız olabilmektedir. Bu proje ile amaçlanan, YouTube'un bu bağımsızlığını bozmadan, sadece şarkıların duygusuna dayalı olarak geliştirilmiş bir müzik önerme sistemi tasarlamaktır.

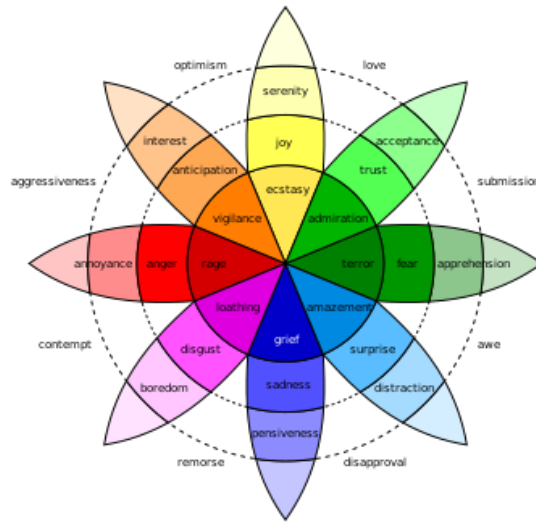
YouTube müzik önerme sistemi olarak en uygun ara yüzün Chrome uzantısı olacağına karar verildi. YouTube platformundan ayrılmadan, video akışını durdurmadan ve kolay bir şekilde kullanılabilecek olması bu ara yüzün seçilmesinde rol oynayan özellikler olmuştur. Fakat Chrome uzantısı ile ses analizi ve veri tabanı yönetimi kolay olamayacağı için, bu işlemlerin arka planda bir API yardımı ile gerçekleştirilip Chrome uzantısı ile çağırılması ve kullanıcıya sonucun yansıtılmasının en efektif tercih olacağına karar kılınmıştır. Bu yüzden ara kapı olarak Python ile bir API tasarlanmıştır. Tasarlanan API ile, öncelikle müzik videosuna ait ses dosyası elde edilir. Daha sonra ses dosyasının öznitelikleri çıkartılıp önceden oluşturulmuş bir veri seti ile eğitilir. Bu işlemin sonucunda duygu değişkenleri elde edilen şarkı ile veri setinde en yakın değişkenlere sahip şarkılar Chrome uzantısı ile kullanıcıya aktarılır.

2 Karşılaştırmalı Literatür Taraması

2.1 Duygu Modelleri ve Müzik Duygu Veri Setleri

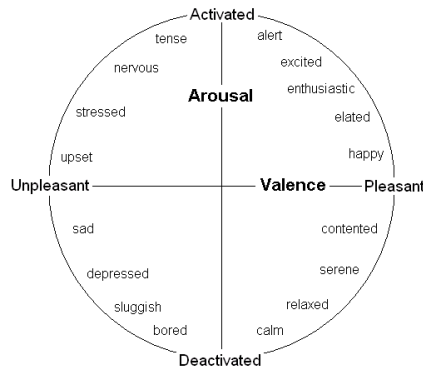
2.1.1 Duygu Modelleri

Projede kullanılacak müzik duygu veri setini seçmeden önce duyguların nasıl modellendiğini öğrenmek gerekmektedir. Müzik ve duygu konusunda yapılan araştırmalarda öne çıkan iki farklı model vardır. Bunlar ayırık ve boyutsal duygu modelleridir [4]. En temel ayırık duygu modelinde duygu; mutluluk, üzgünlük, kızgınlık, korku, iğrenme ve şaşkınlık kavramları ile sınıflandırılmaktadır ve diğer duyguların bu temel kavramlardan türetilbileceği ileri sürülmektedir [5].



Şekil 2.1: Ayırık duygu modeli

Boyutsal modellerde ise en az iki boyut üzerinde her bir duygunun kendine ait bir noktası bulunmaktadır [6]. Posner, Russell ve Peterson'a göre bütün duygular iki bağımsız değerden anlaşılabilir. Bu değerler değerlik ve uyarılma değerleridir [7]. Ayrıca ayırık ve boyutsal modeller arasında yapılan doğrusal eşleştirme ile valans(valence) ve uyarılma(arousal) düzlemleri arasında yüksek bir örtüşme olduğu ortaya çıkmıştır [8]. Bu sayede valans-uyarılma bazlı yeni duygu modelleri oluşturulmuştur [9].



Şekil 2.2: Boyutsal duygu modeli

2.1.2 Müzik Duygu Veri Setleri

Müzik duygu veri setleri, şarkıların duygusu üstünde çalışmalar yapılmasını sağlayan veri setleridir. Bu veri setleri, müzik küratörleri tarafından manuel olarak hazırlanmıştır. Küratörler, şarkıları dinledikten sonra bu şarkıları duygu modellerine göre puanlarlar veya sınıflandırır. Elde edilen verilerin her şarkı için ortalaması, standart sapması gibi değerleri bulunarak bir müzik duygu veri seti oluşturulur.

Müzik duygu veri setlerinde ayırık ve boyutsal duygu modelleri kullanılmaktadır. Ayrıca her veri setinde değerlendirilmiş duygu çeşidi de algılanan ve indüklenen olarak ayrılmaktadır. Bu alanlarda çeşitlilik sunan müzik duygu veri setleri, müziğin duygusu ile alakalı farklı alanlarda çalışmalar yapılmasına olanak sağlamaktadır. Projede kullanılacak olan müzik duygu veri setinin doğru bir şekilde ve özveri ile seçilmesi gerekmektedir.

Bir şarkının nitelik olarak duygusunu bulmakta ayırık duygu modeli yeterli olacaktır, ancak bir şarkı için mutlu kelimesini kullanmak, duygu bazlı önerme sistemini kısıtlayacaktır. Duyguya dayalı müzik önermesi, ayırık duygu modeli ile değil boyutsal duygu modeli ile yapılmalıdır. Önemli olan konu şarkının hangi duygu sınıfına girdiği değildir. Şarkının duygusunun bir düzlemde hangi noktada olduğunu bulmaktır. Bu sayede bu noktaya daha yakın şarkılar bulunabilir. Mutlu sınıfına giren şarkılar, büyük veri setlerinde çok sayıda olacaktır ve bu veri setinden önerilecek olan şarkılardan duygusal anlamda daha yakın şarkılar boyutsal duygu modeli ile elde edilebilir. Yapılan araştırmalar ışığında, valans uyarılma bazlı iki boyutlu duygu modelleri, yapılacak olan çalışma için yeterli olacaktır [8].

Boyutsal duygu modelinde karar kılındıktan sonra, müzik duygu veri setindeki duygunun algılanmış veya indüklenmiş olması göz önünde bulundurulmalıdır. Gabrielson, indüklenen duyguyu dinleyici tarafından hissedilen duygu olarak, algılanan duyguyu ise müzikte anlatılan duygu olarak tanımlamaktadır [10]. Algılanan duygu verileri ile çalışmak, projenin amacı doğrultusunda tercih edilmesi gereken duygu çeşidi olacaktır. Bu elemeler sonucunda geriye 2 adet müzik duygu veri seti kalmaktadır: DEAM/MediaEval ve MoodSwings. Bu iki veri setinin birbirinden farkı, kullanılan şarkıların sayısıdır. Ses analizi yaparken yaklaşık olarak 50 adet ses özelliği elde edilmesi planlanmıştır. İçinde 240 ses dosyası barındıran MoodSwings duygu veri setinin, makine öğrenmesi için yeterli olmayacağı düşünülmüştür. Bu yüzden 2058 ses dosyasından oluşan DEAM duygu veri seti bu projede eğitim veri seti olarak tercih edilmiştir [11].

AMG1608	2015	1608 excerpts (30s long)	WAV	4.3GB	"Discrete" Dimensional	Perceived
Emotify	2016	400 excerpts (60s long)	MP3	363MB	Categorical (GEMS)	Induced
CH818	2017	818 excerpts (30s long)	MP3	393MB	Dimensional	Perceived
4Q-emotion	2018	900 excerpts (30s long)	MP3	291MB	Categorical (Quadrants)	Perceived
DEAM/Mediaeval	2018	2058 excerpts (45s long)	MP3	1.4GB	Dimensional (Time-continuous A-V)	Perceived
PMemo	2018	794 full songs	MP3	1.3GB	Dimensional (Time-continuous A-V)	Induced

Şekil 2.3: Müzik duygu veri setleri

2.2 Müzik Meta Veri Setleri ve Ses Dosyaları

Meta veri, bir kaynağın özelliklerini anlatan veri şeklidir. Birçok sektörde kullanılan meta veriler, müzik sektöründe de önemli bir yere sahiptir. Temel bir müzik meta verisi, sanatçının adı, şarkının adı, şarkının türü, dili, yayınlanma tarihi gibi veriler barındırmaktadır. Önceden bahsedildiği gibi duygu da bir meta veridir. Daha gelişmiş meta veri setlerinde duygu değerleri de bulunmaktadır. Ayırık veri modelleri ile ses analizi yaparken üzgün, mutlu gibi kavramları barındıran meta setleri kullanılmaktadır. Boyutsal modelde ise valans ve uyarılma değerleri bu meta veriye dahil olmalıdır.

Dijital müzik platformları gün geçtikçe artmaktadır ve dolayısıyla meta veri setlerinin sayısı ve çeşitliliği de azımsanamayacak bir boyuta erişmiştir. Deezer, Spotify, Apple Music gibi büyük platformların kendilerine ait meta verileri bulunmaktadır ve herkes tarafından erişilebilmektedir. MusicBrainz, last.fm gibi açık kaynaklı, kullanıcıların girdileri ile oluşturulan meta veri kaynakları da bulunmaktadır [12].

YouTube için duygu modelleri ile kullanılabilir bir meta veri seti bulunmamaktadır. YouTube'a müzik önerme sistemi geliştirirken seçilecek olan test meta veri setinde bu sebepten ötürü ayırık veya boyutsal veriler aranmamalıdır. Bu meta veri setlerinde duygu değerlerinin nasıl bulunduğu dair bir bilgi müzik platformları tarafından paylaşılmamaktadır. Ses analizi yaparken bu değerleri kullanmak sonuçları olumsuz yönde etkileyecektir. Bu yüzden YouTube ile kullanılacak meta veri seti temel verileri barındırmalıdır. Bu temel veriler, bahsi geçen meta veri sağlayıcıların hepsinde bulunmaktadır. Ancak YouTube'da müzik önermesi yapabilmek için, şarkıların YouTube URL'lerinin bilinmesi gerekmektedir ve böyle bir meta veri seti bulunmamaktadır. Temel meta verileri MSD (Million Song Dataset) veri tabanından alınmıştır [13]. Şarkıların temel meta verilerini elde ettikten sonra, YouTube URL'lerinin bulunması ve yeni oluşturulacak bir meta setine eklenmesi gerekmektedir. Bunu elde edebilmek için kullanılacak tek kaynak YouTube Data API'sidir.

2.2.1 Müzik Meta Veri Setlerinin Elde Edilmesi

YouTube Data API, YouTube veri tabanında bulunan kaynakların meta verilerini elde etmek için kullanılan bir API'dir. Bu API yardımı ile videoların, kanalların, çalma listelerinin ve benzeri elementlerin URL'leri yardımıyla meta verileri bulunabilmektedir. Ayrıca bu API ile anahtar kelimeler kullanılarak da meta veri araması yapılabilmektedir [14]. Fakat API üzerinde yapılan istekler kota bazında limitlendirilmiştir ve anahtar kelimeler ile yapılan aramalar URL ile yapılan aramalara göre daha fazla kota harcamaktadır [15]. Herhangi bir müzik meta veri setinde YouTube URL'leri olmadığı için, anahtar kelimeler kullanılarak şarkıların YouTube URL'leri bulunmalı ve meta veri setine eklenmelidir.

Youtube Data API, maalesef videoların indirilmesi için bir kaynak sağlamamaktadır. Bu noktada ses dosyalarını elde etmek için başka kaynaklara başvurulması gerekmektedir. PyTube ve youtube-dl, ses dosyalarını elde etmek için kullanılan Python kütüphaneleridir.

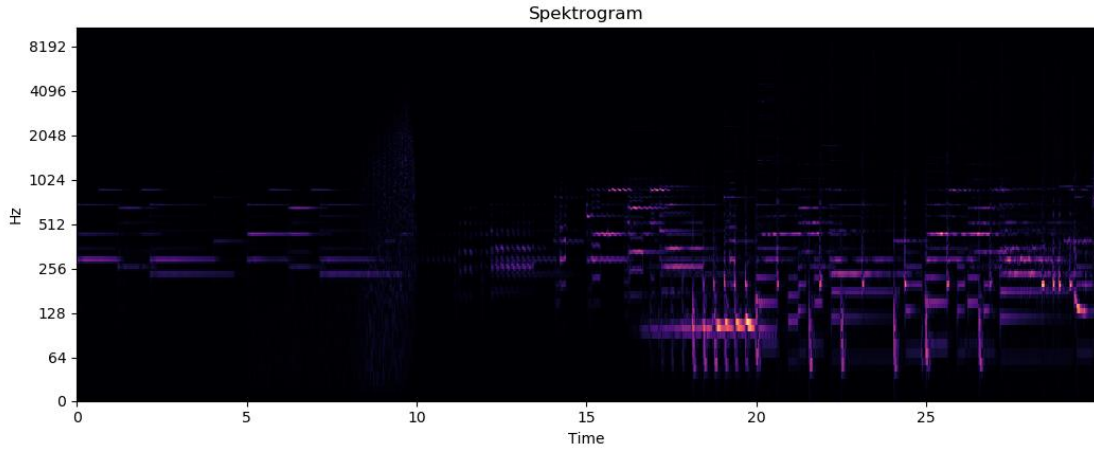
youtube-dl, Youtube ve başka video platformlarından video indirmeyi sağlayan açık kaynaklı bir kütüphanedir. youtube-dl ile videolarda bulunan ses dosyası elde edilebilmektedir. [16]. PyTube ise sadece YouTube için tasarlanmış, daha hafif bir kütüphanedir [17]. Çalışma sırasında iki kütüphane de test edilmiş. youtube-dl'in sağladığı indirme hızının çok az olması sebebi ile PyTube tercih edilmiştir.

2.3 Ses Özellikleri ve Analizi

Ses, maddelerin titreşmesi sonucunda oluşan bir enerji çeşididir ve dalgalar halinde yayılmaktadır. Ses sinyali ise ses dalgasının elektronik gösterim şeklidir. Ses sinyali, elektronik ortamda sesin saklanması, iletilmesini, ses hakkında bilgiler edinilmesini ve ses üstünde oynamalar yapılmasını sağlar. Müziğin dijital ortama taşınması ile şarkılar ses sinyalleri ile saklanmaya başlanmıştır. Bu da bir şarkının ses sinyalini analiz ederek o şarkı hakkında karakteristik özelliklerin bulunmasına olanak sağlamaktadır. Bir şarkının ses özelliklerini bulmadan önce, bu özelliklerin ne anlama geldiğini bilmek gerekmektedir.

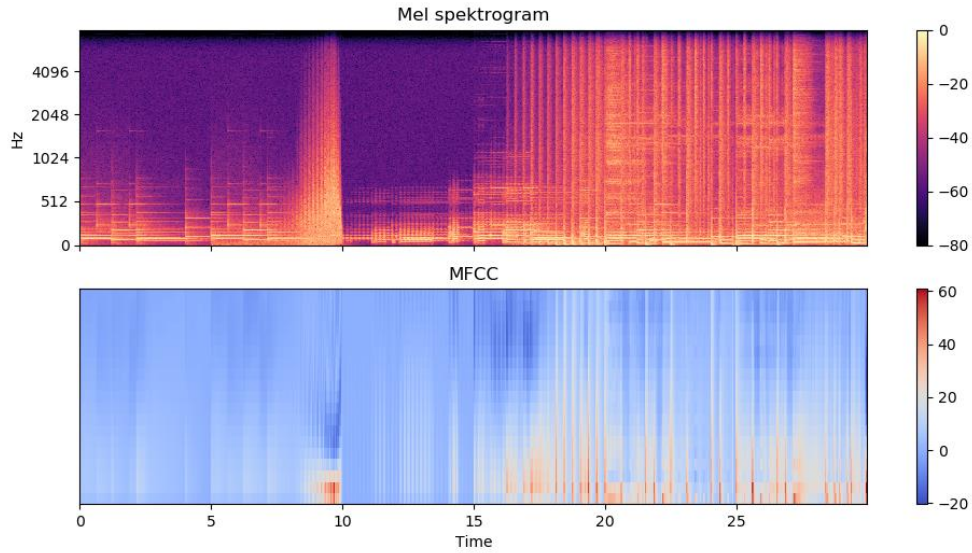
2.3.1 Spektrogram, Mel spektrogram ve MFCC

Spektrogram, bir ses sinyalinin frekans spektrumudur. Spektrogramda x eksenini zaman, y eksenini ise frekansı belirtmektedir. Ayrıca spektrogramda grafik göstergesi olarak genlik (şiddet veya yükseklik) bulunmaktadır. Bu sayede bir ses sinyalinin belirli bir zamanda belirli bir frekansa ait genliği gözlemlenebilmektedir.



Şekil 2.4: Spektrogram

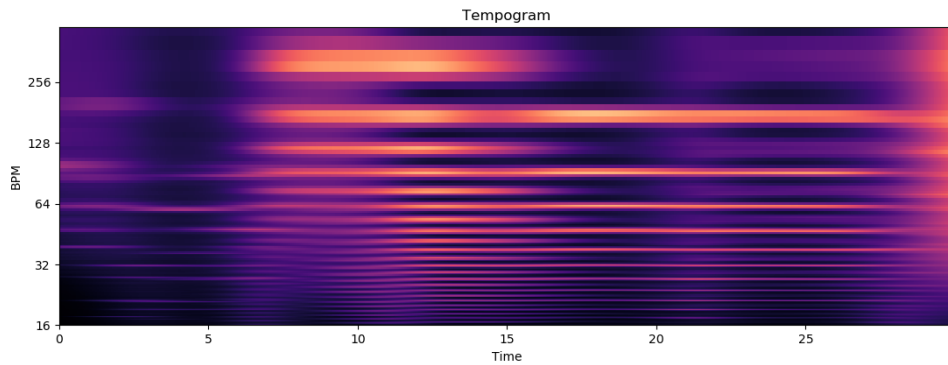
Mel spektrogram ise frekansları Mel ölçeğine ve genliği desibel değerlerine dönüştürerek elde edilen spektrogram çeşididir. Mel ölçeği, sesin frekans değişiminin insanlar tarafından nasıl algılandığını baz alır. Mel spektrogram bu yüzden insan algısına yakın bir yapıya sahiptir. MFCC (Mel Frekans Kepstral Katsayıları), spektrum yoğunluğunun zarf eğrisinin genel şeklini tanımlayan bir özellik seti olarak düşünülebilir. MFCC, bir ses sinyalinin tınısını bulmakta kullanılır [18]. Her enstrümanın kendine ait bir tınısı vardır. Tını ile şarkıda kullanılan enstrümanlar bulunabilir.



Şekil 2.5: Mel spektrogram ve MFCC

2.3.2 Tempo ve Tempogram

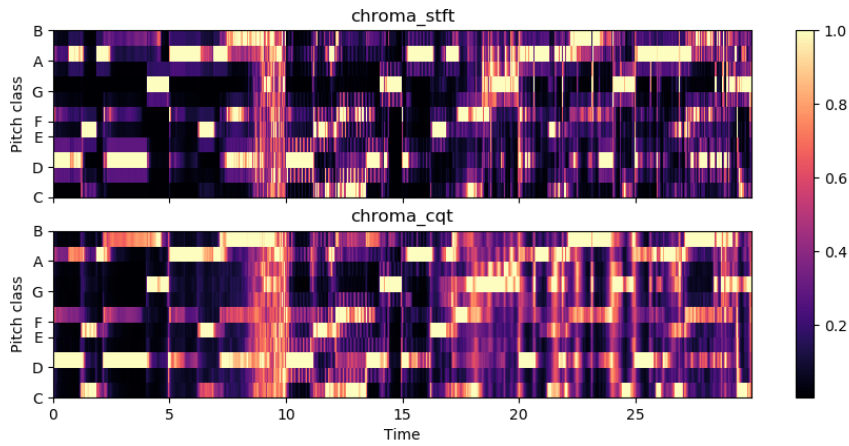
Tempo, bir şarkının ne kadar hızlı veya yavaş olduğunu anlamamızı sağlayan değerdir. Dakika başına düşen vuruş sayısı (BPM) olarak da adlandırılmaktadır. Bir şarkının enerjisini bulurken kullanılan önemli bir değerdir. Tempogram ise ses sinyalinde temponun zamana göre dağılımının gösterilmesidir.



Şekil 2.6: Tempogram

2.3.3 Kroma Özelliği ve Tonnetz

Kroma özelliği veya basit bir ifadeyle nota-perde özelliği, bir şarkının kromatik gamını bulmada kullanılmaktadır. Kromatik gam ise frekansların notalar ile ifade edildiği biçimdir. Bir gam 12 noktadan oluşmaktadır ve bu gamlar farklı duyguları aktarmakta kullanılmaktadır. En çok bilinen gam çeşitleri majör ve minör gamlardır. Majör gamlar müzik eğitiminde mutlu gamlar olarak tanımlanırken, minör gamlar mutsuz olarak tanımlanır. Bunun sebebi bir gamda bulunan notaların arasındaki mesafelerdir, yani frekansların birbiri ile nasıl bir uyum içinde olduğudur. Tonnetz (Tonal Centeroid Features) ise 12 farklı notayı 6 farklı nota şeklinde sınıflandırarak ses dosyasının kromatik özelliklerini elde etmektedir.



Şekil 2.7: Kroma özelliği

2.3.4 Sıfır Geçiş Oranı

Sıfır Geçiş Oranı (Zero-Crossing Rate), bir ses sinyalinin pozitif ve negatif değerler arasındaki değişim hızıdır. Vurmalı enstrümanların çokça kullandığı müzik türlerinde bu oran daha fazladır.

2.3.5 Spektral Özellikler

Spektral Kütle Merkezi, ses spektrumunun kütle merkezinin nerede olduğunu öğrenmek için kullanılır. Sesin parlaklığını tahmin etmekte iyi olduğu için ses analizinde yaygın bir şekilde kullanılmaktadır [19]. Spektral etek, belirlenen bir spektral enerjinin altında kalan frekansları temsil etmektedir. Spektral etek ile müzikte konuşma seslerin ötümlü veya ötümsüz olduğunu bulmada kullanılır.

Ses özellik çıkarımı için kullanılan birçok araç bulunmaktadır. Projenin geliştirme sürecinde yazılım dili olarak Python programlama dili kullanılacağı için Python kütüphanesine sahip bir aracın seçilmesi planlanmıştır. librosa, bu konuda çokça kullanılan bir kütüphanedir [20]. Hesaplama süresi diğer araçlara göre uzun olsa da yapılacak olan çalışma için yeterli olacaktır. Çok iyi bir dokümantasyona sahiptir ve librosa ile elde edilebilecek ses özellikleri gerekli olan ses özelliklerini karşılamaktadır [21].

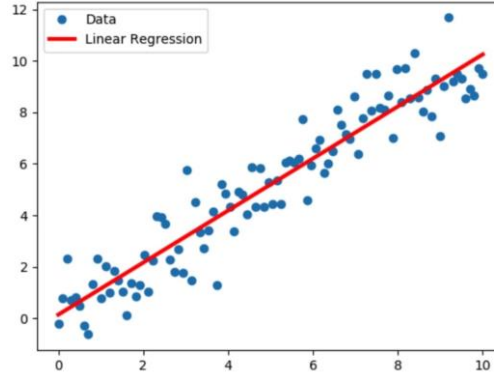
2.4 Regresyon Modelleri

Müzik duygu teşhisi için makina öğrenmesinde sınıflandırma ve regresyon modelleri kullanılmaktadır. Sınıflandırma modellerinde nitelik, regresyon modellerinde ise nicelik değişkenleri kullanılır. Sınıflandırma öğrenme modelleri daha çok şarkıların türünü bulmada, dilini bulmada ve ayırık duygu modeline dayalı duygu öğrenme sistemlerinde kullanılmaktadır [22]. Boyutsal duygu modellerinde makine öğrenmesi yapabilmek için tercih edilen öğrenme modeli regresyon öğrenme modelleridir.

Regresyon modellerindeki amaç, bağımlı değişken ile bağımsız değişkenler arasındaki ilişkiyi hesaplamaktır. Bu sayede bilinmeyen bağımlı değişkenler tahmin edilir veya bağımsız değişkenlerin bağımlı değişken üzerindeki etkileri araştırılabilir [23]. Günümüzde birçok regresyon modeli bulunmaktadır. Bu modellerin kullanım alanları farklılık gösterse de her biri incelenmeli ve test edilmelidir.

2.4.1 Küçük Kareler Doğrusal Regresyonu

Küçük kareler doğrusal regresyonu, bağımlı değişken(y) ve bağımsız değişkenler(x) arasında bir doğrusal bağlantı hesaplar. Bu doğrusal bağlantıyı regresyon katsayılarını(w) bularak yapmaktadır. Regresyon katsayılarını bulmak için küçük kareler yöntemini kullanmaktadır. Bağımsız değişkenler ve bağımlı değişken arasında $\min ||xw - y||_2^2$ işlemini yaparak en iyi uyum doğrusu olarak da bilinen bir regresyon doğrusu tanımlar.



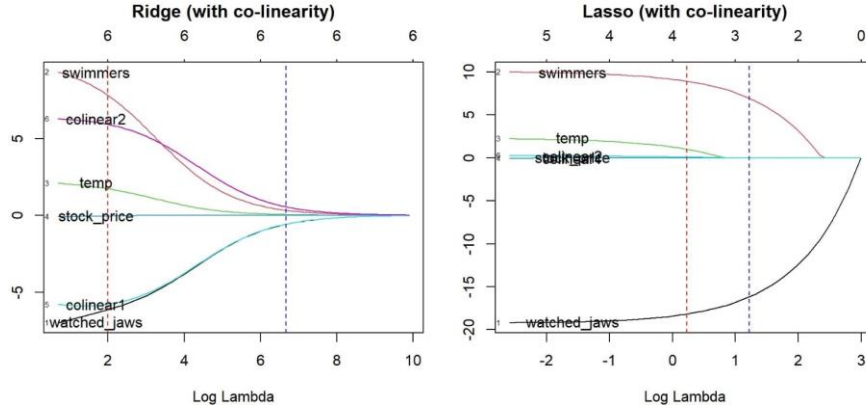
Şekil 2.8: Küçük kareler doğrusal regresyonu

2.4.2 Ridge Regresyonu

Bağımsız değişkenlerin arasında yüksek korelasyon olduğu durumlarda ortaya çıkabilecek bir hata payı bulunmaktadır. Bu hata payı küçük kareler doğrusal regresyonunda hesaba katılmamaktadır. Ridge regresyonunda ise bu hata payı da hesaplama yaparken kullanılmaktadır. Bu sayede aralarında yüksek korelasyon olan bağımsız değişkenlerin bulunduğu veri setlerinde Ridge regresyonu tercih edilmektedir. Matematiksel denklemi $\min ||xw - y||_2^2 + a||w||$ olan Ridge regresyonunda yüksek korelasyon, a değişkeni ile çözülmektedir.

2.4.3 Lasso Regresyonu

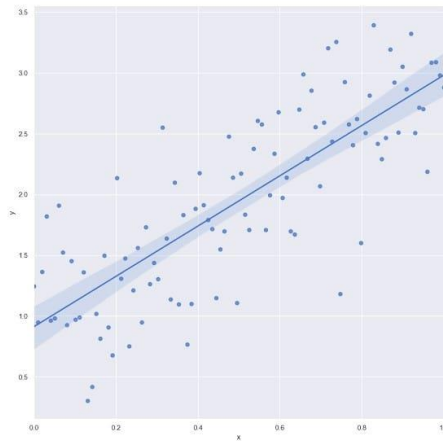
Doğrusal bir regresyon olan Lasso regresyonu, regresyon katsayısının mutlak büyüklüğünü azaltarak bir doğrusal bağlantı elde eder. Bu sayede seyrek regresyon katsayıları problemine çözüm getirmektedir. Bunu elde etmek için $\frac{1}{2n} \text{Min} ||xw - y||_2^2$ fonksiyonunu kullanır. Ayrıca $\alpha ||w||$ ile bu sonucu toplayarak hata payını en aza indirmektedir.



Şekil 2.9: Ridge ve Lasso regresyonları

2.4.4 Bayesian Doğrusal Regresyonu

Bayesian doğrusal regresyonunda regresyon katsayıları küçük kareler yöntemi yerine Bayes teoremi ile bulunmaktadır. Bu sayede değişkenlerin sonsal olasılığı elde edilir. Bayesian regresyonu ile elde edilecek verilerin, küçük kareler doğrusal regresyona göre gerçek verilere daha yakın değerlerde olacağı öngörülmektedir.

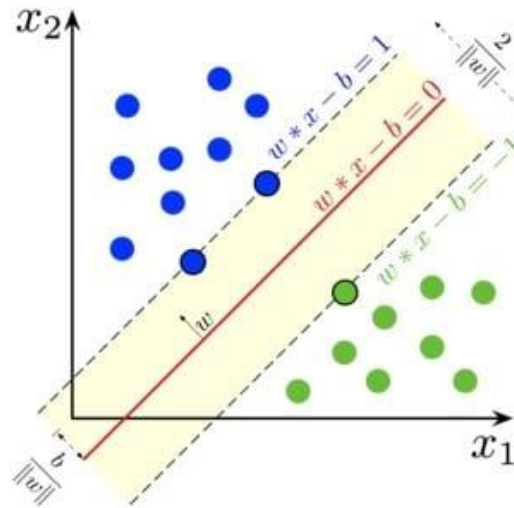


Şekil 2.10: Bayesian doğrusal regresyonu

2.4.5 SVR

SVM (Destek Vektör Makinesi), denetimli makine öğrenme modelidir ve hem sınıflandırma hem de regresyon analizi için kullanılabilir [24]. SVR (Destek Vektör Regresyonu)'deki amaç, verileri sınıflandıracak bir hiperdüzlem oluşturmaktır. En çok verinin bulunduğu hiperdüzlem bulunmaya çalışılır ve en iyi uygun doğrusu olarak kabul edilir. Yapılacak olan tahminler, bu hiperdüzlem baz alınarak yapılmaktadır. Bütün veriler karar sınırları arasında olmalıdır. Hiperdüzlemin denklemini $y = wx + b$ şeklinde düşünecek olursak, $w x + b = +a$ ve $w x + b = -a$ karar sınırlarının denklemleri olacaktır. Bu durumda hiperdüzlem $-a < y - wx + b < +a$ koşulunu sağlamalıdır. a değeri seçilirken, hiperdüzleme en yakın olan verilerin, bir diğer adıyla destek vektörlerinin karar sınırları içinde olması gerekmektedir.

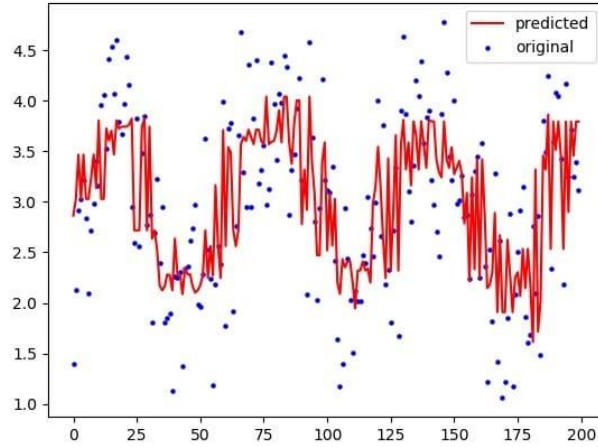
SVR modelinde farklı kernel (çekirdek) çeşitleri kullanılmaktadır. Kernel, matematik fonksiyonlardan oluşan set olarak tanımlanabilir. Kernel ile veriler alınarak istenilen formata dönüştürülür ve hiperdüzlem oluşturmada kullanılır. Doğrusal, doğrusal olmayan ve polinom bazı kernel çeşitleridir. SVR modelinde kullanılacak kernel önemli bir bileşendir ve veri setine en uygun olacak Kernel şeklini seçmek gerekmektedir.



Şekil 2.11: SVR

2.4.6 k-NN Regresyonu

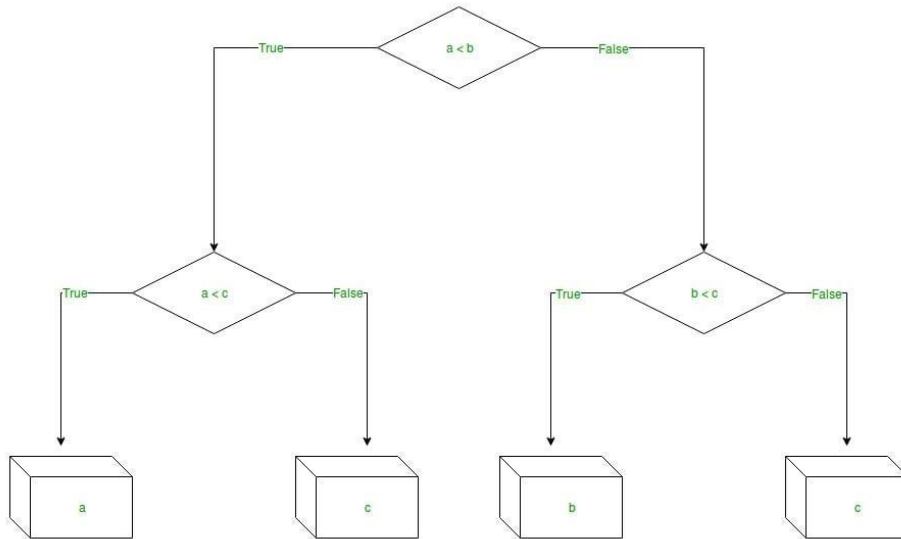
Denetimli öğrenme metodu olan k-NN (k-En Yakın Komşu) algoritması, bağımsız değişkenler(x) ile bağımlı değişkenler(y) arasındaki bağlantıyı, en yakın k komşunun ortalamasını alarak bulmaktadır. $y=1$ alınacak olursa, 1 değerini sağlayan veya bu değere yakın olan x değerleri bir komşuluk ilişkisi içindedir. Bu komşuluk ilişkisi içinde olan x değerlerinin ortalaması alınır ve 1 değerine denk gelen bilinmeyen x değeri bulunmuş olur. k-NN regresyonunda dikkat edilmesi gereken önemli unsur; k değerini, yani komşu sayısını belirlemektir. Ayrıca bağımsız değişkenlerin, yani boyutun sayısı arttıkça, k-NN regresyonu ile elde edilen sonucun güvenilirliği azalmaktadır [25].



Şekil 2.12: k-NN regresyonu

2.4.7 Karar Ağacı Regresyonu

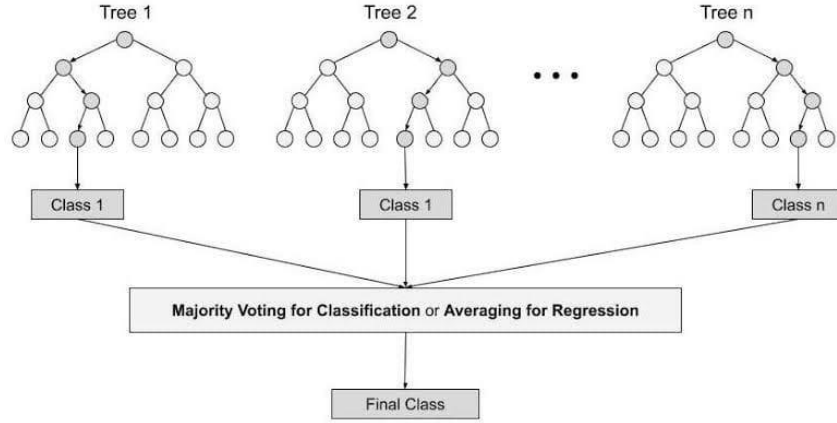
Karar ağacı (Decision Tree) bir karar verme aracıdır ve ağaç yapısını kullanmaktadır. Karar ağacında kök düğüm ve iç düğümler bir denklem temsil etmektedir. Yaprak düğümlerinde ise sonuçlar bulunmaktadır. Kök düğümden başlayan karar verme işlemi, doğru/yanlış metodu ile gerçekleşir ve yaprak düğüme ulaşınca son bulur. Bir yaprak düğümünde bulunan bağımlı değişkenlerin ortalaması, bu değişkenin tahmin edilen sonucu olur.



Şekil 2.13: Karar ağacı regresyonu

2.4.8 Rastgele Orman Regresyonu

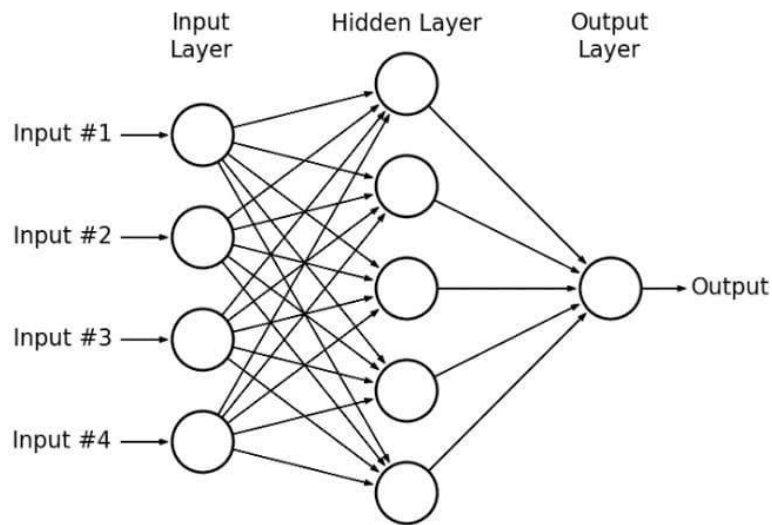
Rastgele orman (Random Forest) regresyonu, karar ağacına dayalı bir modeldir. Öğrenme aşamasında birden fazla karar ağacı oluşturularak tahmin yapmaktadır. Bu tahmin, bütün karar ağaçlarından elde edilen sonuçların ortalaması alınarak elde edilir.



Şekil 2.14: Rastgele orman regresyonu

2.4.9 MLP Regresyonu

İleri beslemeli yapay bir sinir ağı olan MLP (Çok Katmanlı Algılayıcı), bir veri setinde eğitilerek bağımsız veriler ve bağımlı veriler arasında ilişkisel bir fonksiyon elde eder. Bu fonksiyonu kullanarak test veri setindeki bağımlı değerleri tahmin eder. MLP 3 farklı katmandan oluşmaktadır. Bu katmanlar girdi katmanı, gizli katmanlar ve çıktı katmanıdır. Girdi ve çıktı katmanları tek bir katmandan oluşurken, gizli katmanlar bir veya daha çok katman barındırabilir. Gizli katmanlar sebebiyle MLP dış bükey olmayan kayıp fonksiyonuna sahiptir.



Şekil 2.15: MLP regresyonu

Ses özelliklerini çıkarma ile alakalı yapılan çalışmalarda birçok farklı regresyon modeli kullanılmıştır. Çalışmaların duygu veri setine göre ve kullanılan bağımsız değişkenlere göre en iyi sonucu veren regresyon modeli değişmektedir. Yapılan bir çalışmada MLR (Çoklu Doğrusal Regresyon) modeli daha iyi sonuç verirken [26], başka bir çalışmada SVR daha iyi sonuç vermektedir [27]. Bu yüzden regresyon modeli seçerken kullanılan duygu veri setinin büyüklüğü ve bağımsız değişkenlerin sayısı göz önünde bulundurulmalıdır. Bu standartları karşılayan regresyon modelleri öğrenme veri setinde test edilmeli ve sonuçları en uygun olanı seçilmelidir.

2.5 Chrome Uzantısı

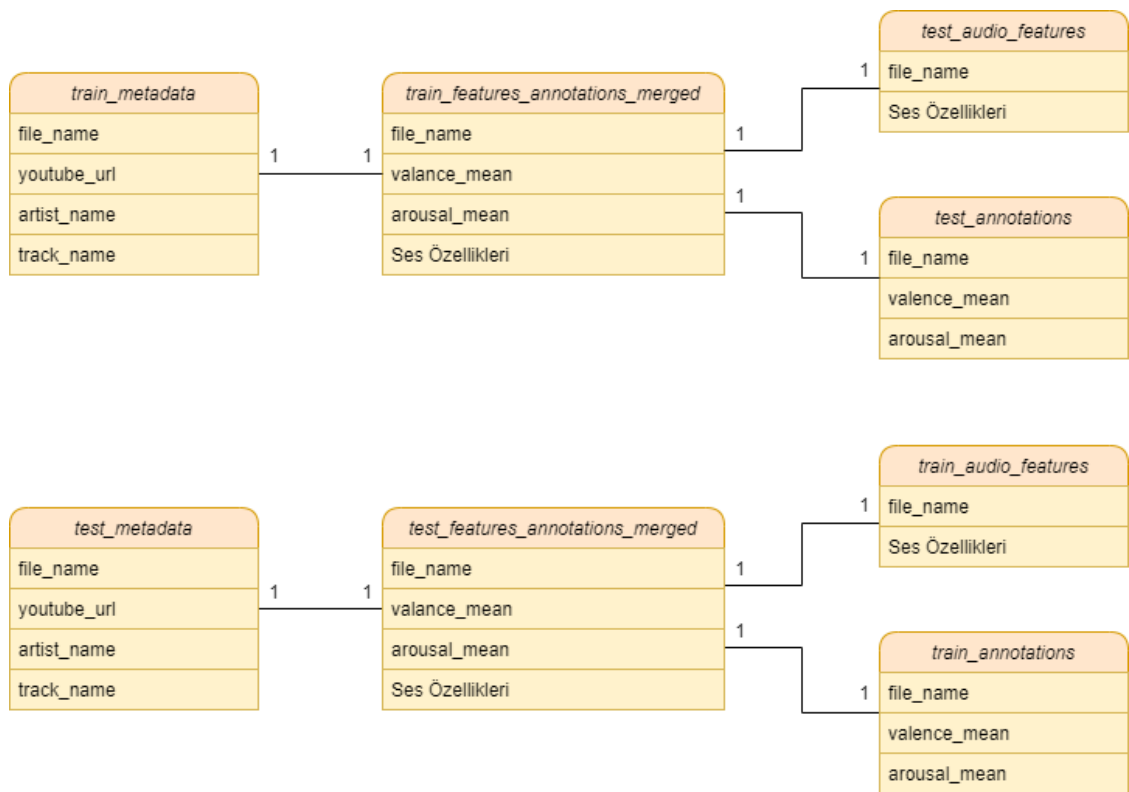
Chrome, günümüzde en çok kullanılmakta olan web tarayıcısıdır [28]. Bir uygulama ne kadar çok kişi tarafından kullanılırsa, uygulamanın sağladığı özelliklerin de geliştirilmesi, arttırılması beklenir. Chrome, bu noktada uzantılar ile geliştiricilerin katkıda bulunmasına olanak sağlamaktadır. Chrome uzantıları aslında bir web sayfasıdır. Bu web sayfaları Chrome API'si ile eklenti şeklinde Chrome'a entegre edilir [29]. Temelde bu eklentiler HTML, CSS ve JavaScript ile yapılmaktadır. JavaScript üzerinden Chrome'a bağlanmaktadırlar.

Araştırmalar sonucunda YouTube üzerinden müzik önermesi yapan çalışmalara çok nadir rastlanılmıştır. Bu çalışmalar, arayüz tercihi olarak web sayfası veya direk bir çerçeveye üstünden müzik videosu sunmaktadır. Bu projede kullanıcıdan bir girdi beklenmektedir ve kullanıcı müzik videosunu izlerken arka planda çalışması hedeflenmektedir. Bu sebepten ötürü kullanıcı arayüzü olarak Chrome uzantısı seçmek, YouTube arayüzünden ayrılmadan, videoyu duraklatmadan, kullanıcının URL ile uğraşmasına gerek kalmadan şarkı önerisinde bulunacak basit ve hızlı bir çözüm olacaktır.

3 Geliştirilmiş Yaklaşım ve Sistem Modeli

3.1 Veri Modeli

Çalışmada kullanılan veriler, test ve eğitim verileri için 2 farklı veri setinden oluşmaktadır. Her bir veri setinin dizaynı aynıdır. Her bir veri setinde 4 farklı csv dosyası bulunmaktadır. Verilerin manuel olarak daha kolay bir şekilde anlaşılması için bu format tercih edilmiştir. Bu dosyalar, birbiri ile dosya adı üzerinden birbirine bağlanmaktadır. Ayrıca geliştirme sürecinde ses özelliklerini elde edebilmek için gerekli olan şarkıların ses dosyaları da MP3 şeklinde veri setlerine dahildi. Son versiyonda bu ses dosyaları bulunmamaktadır. Kullanıcı arayüzü ile elde edilen veriler de bulunmaktadır, fakat bu veriler gerçek zamanlı işlenmektedir ve veri tabanında saklanmamaktadır.



Şekil 3.1: Statik veri yapısı

3.2 Yapısal Model

Test data setini oluşturmak için YouTube URL'lerini bulmak gerekmektedir. Şarkıların YouTube URL'lerini elde edebilmek için de YouTube Data API'sinin sağladığı search() fonksiyonu kullanılmıştır. Bir anahtar kelime kullanılarak videolar arasında şarkıya ait olabilecek en olası sonuç elde edilmeye çalışılmıştır.

```
with open(input_file, mode='r') as csv_file, \
    open(output_file, 'a') as output_csv:
    csv_reader = csv.reader(csv_file)
    csv_writer = csv.writer(output_csv)
    line_count = 0
    for row in csv_reader:
        if line_count == 0:
            line_count += 1
            row.append("youtube_url")
        else:
            artist_n = row[1]
            song_n = row[2]
            request = youtube.search().list(
                part='id, snippet',
                q("{} - {} official music video".format(artist_n, song_n),
                #order='viewCount',
                type='video',
                maxResults=1)
            response = request.execute()
            for video in response['items']:
                youtube_url = video['id']['videoId']
                youtube_title = video['snippet']['title']
                row.append(youtube_url)
                csv_writer.writerow(row)
                print("{} - {}".format(youtube_url, youtube_title))
            df = df.iloc[1:]
            df.to_csv(input_file, index=False)
```

Şekil 3.2: YouTube Data API ile şarkı URL'lerinin elde edilmesi

Youtube üzerinden şarkı analizi yapabilmek için öncelikle Chrome uzantısı aracılığı ile URL'yi almak gerekir. URL'ye erişim için “*tabs*” yetkisinin API'ye verilmesi gerekir. Bu yetki verildikten sonra Chrome API'si yardımı ile URL elde edilir. Elde edilen bu URL, fonksiyonları çağırmak için geliştirilen API'ye gönderilir. Bu noktada uzantı tarafında dikkat edilmesi gereken önemli noktalar vardır. Dizayn olarak background ve popup adında iki farklı JavaScript programcığı kullanılmaktadır. Background arka plan işçisi olarak çalışmaktadır, yani boş durumda diğer popup'tan mesaj beklemektedir. Öncelikle popup ve background arasında bir bağlantı kurulur. popup programcığı Chrome uzantısına tıklanıldığında çalışır ve background programcığına mesaj gönderir.



Şekil 3.3: Programcıklar arası haberleşme

API, Chrome uzantılarında `fetch()` komutu ile çağrılmaktadır. API'den elde edilen 5 adet benzer şarkının URL'si, Chrome API'sinin sağladığı ve bütün JavaScript kodlarının erişebileceği ortak bir depolama alanına kaydedilir. popup programcığı ile depolama alanından alınıp HTML dosyasına enjekte edilir.

```
# background
fetch(API_URL + YouTube_URL){
    Yanıtı JSON formatına çevir

    for i in Yanıt:
        i anahtarını ve verisini depolama alanına kaydet

    popup koduna 'bitti' mesajı gönder
}

-----
# popup
if mesaj 'bitti':
    HTML dosyasındaki elementleri depolama alanındakilerle değiştir
```

Şekil 3.4: API'nin çağırılması ve elde edilen verilerin kullanıcıya gösterilmesi

Ses dosyasından özellik çıkartmak için kullanılan kod parçacığı Şekil 3.3'de yer almaktadır. Elde edilen ses özellikleri bir dizin şeklindedir. Her bir dizinin ortalaması, standart sapması ve varyansı alınarak her bir ses dosyası için 54 farklı veri elde edilmiştir.

```

# Spectral Features
S = np.abs(librosa.stft(y))
chroma_stft = librosa.feature.chroma_stft(y=y, sr=sampling_rate) # y = audio_file
chroma_cqt = librosa.feature.chroma_cqt(y=y, sr=sampling_rate)
chroma_cens = librosa.feature.chroma_cens(y=y, sr=sampling_rate)
melspectrogram = librosa.feature.melspectrogram(S=S)
mfcc = librosa.feature.mfcc(S=S)
delta_mfcc = librosa.feature.delta(mfcc)
rms = librosa.feature.rms(S=S)
spectral_centroid = librosa.feature.spectral_centroid(S=S)
spectral_bandwidth = librosa.feature.spectral_bandwidth(S=S)
spectral_contrast = librosa.feature.spectral_contrast(S=S)
spectral_rolloff = librosa.feature.spectral_rolloff(S=S)
poly_features = librosa.feature.poly_features(S=S)
tonnetz = librosa.feature.tonnetz(y=y, sr=sampling_rate)
zero_crossing_rate = librosa.feature.zero_crossing_rate(y=y)
# Rhythm Features
tempo, beats = librosa.beat.beat_track(y=y, sr=sampling_rate) # BPM
# Onset Detection
onset_frames = librosa.onset.onset_detect(y=y, sr=sampling_rate)
frames_to_time = librosa.frames_to_time(onset_frames[:20], sr=sampling_rate)
# Effects
harmonic = librosa.effects.harmonic(y)
percussive = librosa.effects.percussive(y)

```

Şekil 3.5: librosa ile ses özelliklerinin elde edilmesi

Valans ve uyarılma değerlerini bulmak için öncelikle ses özellikleri -1 ve 1 değerleri arasında standartlaştırılmıştır. Bunun için scikit-learn kütüphanesinin sağladığı StandardScaler fonksiyonu kullanılmıştır. Bu fonksiyon, değerleri standartlaştırırken eğitim veri setindeki bağımsız değişkenlerin ortalamalarını ve standart sapmalarını kullanılır. Ses özellikleri standartlaştırıldıktan sonra, rastgele orman modeli kullanılarak test setindeki valans ve uyarılma değerleri tahmin edilmiştir.

```

# scaler transformation
scaler = StandardScaler()
scaler.fit(x_train)
x_train = scaler.transform(x_train)
x_test = scaler.transform(x_test)

# RFR
model = RandomForestRegressor(random_state=0)
model.fit(x_train, y_train_1.ravel())
y_pred_1 = model.predict(x_test)

model.fit(x_train, y_train_2.ravel())
y_pred_2 = model.predict(x_test)

```

Şekil 3.6: Standartlaştırma ve duygu değerlerinin tahmini

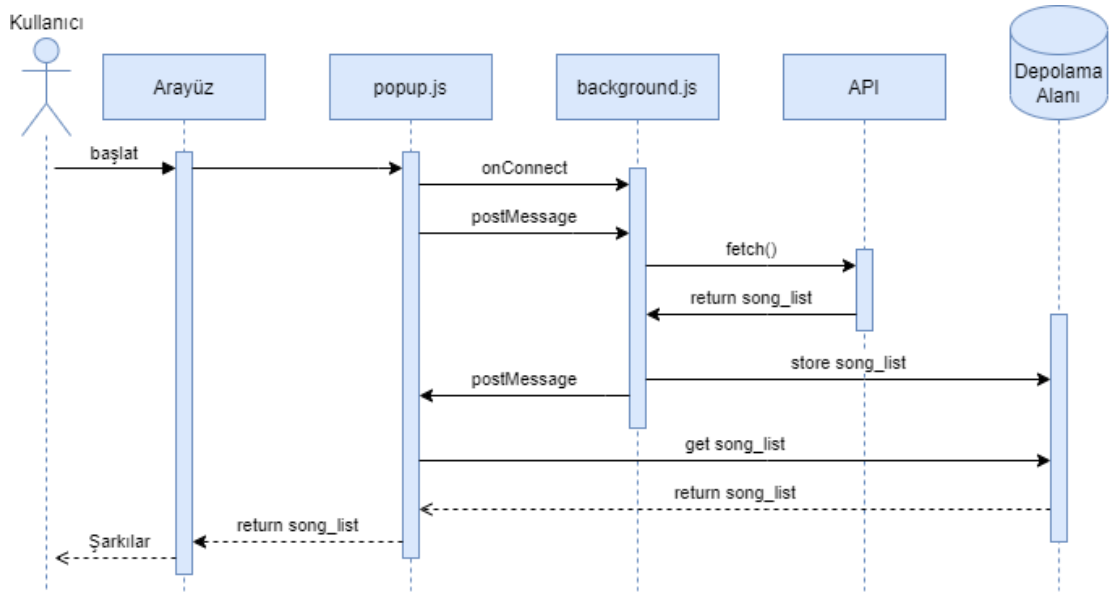
Veri setinden en yakın valans ve uyarılma değerlerine sahip 5 adet şarkıyı bulmak için Öklid uzaklığı kullanılmıştır. Öklid Uzaklığı, $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$ formülü ile bulunmaktadır. Şekil 3.7’de bu formülün kodu görülebilmektedir.

```
# euclidian distance
i = (((df['valence_mean'].values - test_merged['valence_mean'].values) ** 2 +
      (df['arousal_mean'].values - test_merged['arousal_mean'].values) ** 2) ** 0.5).argsort()[:5]
```

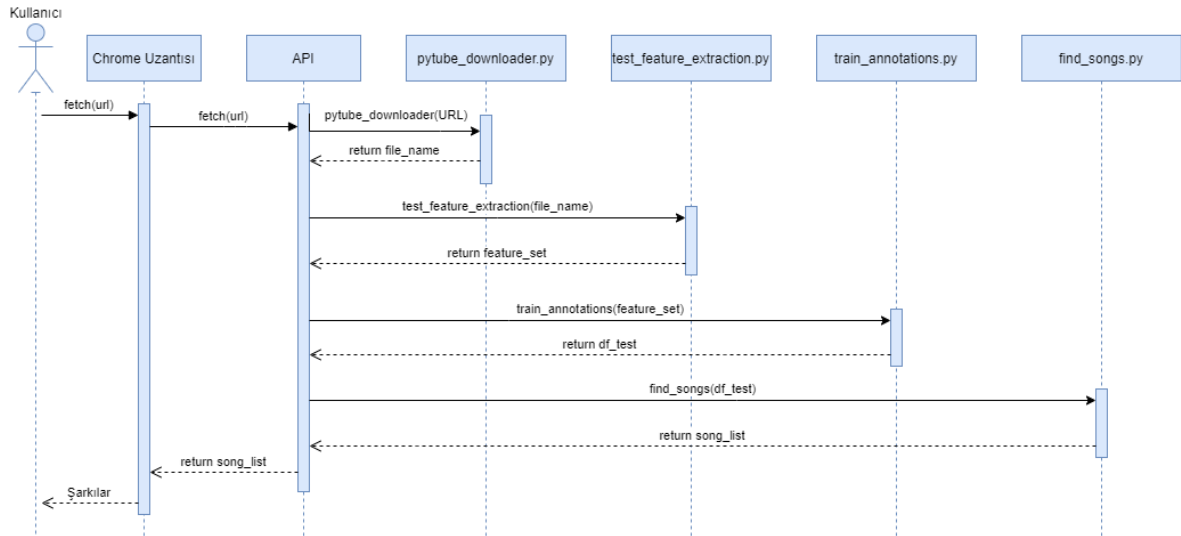
Şekil 3.7: Öklid Uzaklığı ile en yakın değerlere sahip şarkıların bulunması

3.3 Dinamik Model

Sistemin çalışma şekli Chrome uzantısı ve API olmak üzere incelenebilir.



Şekil 3.8: Chrome uzantısı dinamik modeli



Şekil 3.9: API dinamik modeli

4 Deney Ortamı ve Deney Dizaynı

Çalışmayı yaparken uzun süren deneyler yapılmıştır. Bu deneyler doğru regresyon modelini bulma, duygu değerlerini öğrenebilmek için gerekli olan ses özelliklerini anlama ve uygulamanın çalışmasını hızlandırma üstünedir. Çoğu test lokal bir Linux sisteminde gerçekleştirilmiştir. Ek olarak Google Collab ve Chrome arayüzü kullanılmıştır.

Regresyon modeli seçmek için yapılan deneyler eğitim veri seti üstünde yapılmıştır. Denenmiş olan çoğu regresyon modeli için farklı koşullar sağlanmış, özellik seçme metotları kullanılmış ve farklı parametreler ile bu modeller çağrılmıştır. Elde edilen sonuçlar doğrultusunda en iyi sonuçlar rastgele orman regresyon modeli ile elde edilmiştir. İkinci olarak SVR modeli ile de iyi sonuçlar alınmıştır. SVR 0.53 r2 skoruna, rastgele orman regresyonu ise 0.56 r2 skoruna kadar çıkabilmiştir. Yapılan testlerden elde edilen bir uyarılma skoru sonucu Şekil 4.1'de görülebilmektedir. R2 skorunda rastgele orman regresyonu daha iyi sonuçlar verse de SVR sonuca biraz daha hızlı ulaşmaktadır. Bu iki model daha sonra Chrome üzerinde çalıştırılmıştır ve gerçek zamanlı olarak test edilmiştir. Yapılan gözlemler sonucu uygulama için rastgele orman regresyonu seçilmiştir.

```
LinearRegression score: 0.4774332528655979
Ridge Regression score: 0.47757336868156486
Lasso Regression score: 0.14920343163214045
Bayesian Regression score: 0.4674656324735038
SGD Regression score: 0.4674945563473605
Kernel Ridge Regression score: -17.436184410778942
linear SVR score: 0.4688604074182988
poly SVR score: 0.10472414209730929
rbf SVR score: 0.5192116247998887
sigmoid SVR score: -236.72358014251978
{'n_neighbors': 15}
knn score: 0.4583910201875472
PLS score: 0.4168677656741929
tree score: 0.05473773641111668
poisson forest score: 0.5414113079136482
squared_error forest score: 0.5532894459423892
absolute_error forest score: 0.5471753742853973
mlp score: 0.44042212667582425
XGBRegressor score: 0.4511172350203806
```

Şekil 4.1: Uyarılma değeri için r2 skoru

5 Karşılaştırmalı Değerlendirme ve Tartışma

Bu konuda yapılan çok az sayıda çalışma bulunmaktadır. Yapılan çalışmalarda genellikle ayrık duygu modeli ve sınıflandırmaya dayalı makine öğrenme metotları kullanılmaktadır. Lakin ses analizinde kullanılan regresyon modellerini kıyaslayan çalışmalar bulunmaktadır. SVR, BoostR ve MLR modellerinin karşılaştırılması sonucu SVR modeli ile %28,1 valans %58,3 uyarılma uygunluk değerleri elde edilmiştir [27]. Bu projede elde edilen sonuçlar ise %50,8 valans ve %55,8 uyarılma uygunluk değerleridir. Fakat kullanılan veri setinin ve ses özelliği çıkartma aracının sonuçlarda etkisi olmaktadır. Bu yüzden bu konuda bir karşılaştırma doğru olmayacaktır.

6 Sonuç ve Olasılıklar

Müzik evrensel bir dildir. İnsanlar anlatmak isteyip de anlatamadıklarını, hissettiklerini, duygularını müzik aracılığıyla birbirleriyle paylaşırlar. Her geçen gün müziğe erişim kolaylaşmaktadır. Ancak hızla artan şarkı sayısı ile aynı zamanda zorlaşmaktadır. Müzik önerme sistemleri buna bir çözüm sağlamaya çalışmaktadır. Yapılan çalışmada YouTube için bir müzik önerme sistemi geliştirilmiştir. Bu sistem geliştirilirken birçok regresyon modeli denenmiştir ve en iyi sonuç rastgele orman regresyonu ile elde edilmiştir. Sadece duygular üzerinden şarkılar öneren sistem, şarkı paletini geniş tutmaktadır ve bu sistem %50,8 valans ve %55,8 uyarılma uygunluk değerlerine sahiptir. Bu değer farklı makine öğrenme dizaynları ile, derin öğrenme ile arttırılabilir. Sistemde kullanılan veri seti statik bir veri setidir. Bunun sebebi YouTube'da aynı şarkıya ait birden fazla video olmasıdır. Eğer bunu kontrol edebilen bir sistem tasarlanabilirse veri seti dinamik hale getirilip veriler gerçek zamanlı olarak kaydedilebilir. Ayrıca YouTube'dan bir video kaldırılınca video URL'si yeni yüklenen bir URL'ye verilebilmektedir. Bu da veri setinin sürekli olarak kontrol edilmesini gerektirmektedir. Aslında proje aşamasında bu fonksiyonun eklenmesi planlanmıştır fakat YouTube Data API kota sistemini kullandığı için bu fonksiyon entegre edilememiştir. Daha fazla kota sağlandığında veri seti kontrolü de belirli aralıklarla sağlanabilir. Geliştirilen sistem API üzerinden çalıştığı için kolaylıkla başka sistemlere entegre edilip kullanılabilir.

7 Referanslar

- [1] M. R. Dickey, “Spotify Just Bought A Company That Will Deliver A Huge Blow To Its Rivals,” *Business Insider*. <https://www.businessinsider.com/spotify-buys-the-echo-nest-2014-3>.
- [2] “Videos | YouTube Data API,” *Google Developers*. <https://developers.google.com/youtube/v3/docs/videos>.
- [3] “On YouTube’s recommendation system,” *blog.youtube*. <https://blog.youtube/inside-youtube/on-youtubes-recommendation-system/>.
- [4] P. N. Juslin, *Handbook of Music and Emotion: Theory, Research, Applications*. 2010. doi: 10.1093/acprof:oso/9780199230143.001.0001.
- [5] P. Ekman, “Facial Expressions of Emotion: New Findings, New Questions,” *Psychol Sci*, vol. 3, no. 1, pp. 34–38, Jan. 1992, doi: 10.1111/j.1467-9280.1992.tb00253.x.
- [6] W. M. Wundt and C. H. Judd, *Outlines of Psychology*. Leipzig; New York: W. Engelmann; G.E. Stockert, 1897.
- [7] J. Posner, J. A. Russell, and B. S. Peterson, “The circumplex model of affect: an integrative approach to affective neuroscience, cognitive development, and psychopathology,” *Dev Psychopathol*, vol. 17, no. 3, pp. 715–734, 2005, doi: 10.1017/S0954579405050340.
- [8] T. Eerola and J. K. Vuoskoski, “A comparison of the discrete and dimensional models of emotion in music,” *Psychology of Music*, vol. 39, no. 1, pp. 18–49, Jan. 2011, doi: 10.1177/0305735610362821.
- [9] P. Gomez and B. Danuser, “Affective and physiological responses to environmental noises and music,” *Int J Psychophysiol*, vol. 53, no. 2, pp. 91–103, Jul. 2004, doi: 10.1016/j.ijpsycho.2004.02.002.
- [10] A. Gabrielsson, “Emotion perceived and emotion felt: Same or different?,” *Musicae Scientiae*, vol. Spec Issue, 2001-2002, pp. 123–147, 2001, doi: 10.1177/10298649020050S105.
- [11] M. Soleymani, A. Aljanaki, and Y.-H. Yang, “DEAM: MediaEval Database for Emotional Analysis in Music,” p. 3.
- [12] “List of online music databases,” *Wikipedia*. Aug. 04, 2022. Accessed: Aug. 11, 2022. [Online]. Available: https://en.wikipedia.org/w/index.php?title=List_of_online_music_databases&oldid=1102263206

- [13] T. Bertin-Mahieux, D. P. W. Ellis, B. Whitman, and P. Lamere, “The Million Song Dataset,” 2011.
- [14] “YouTube Data API Overview,” *Google Developers*.
<https://developers.google.com/youtube/v3/getting-started>.
- [15] “YouTube Data API (v3) - Quota Calculator | Google Developers.”
https://developers.google.com/youtube/v3/determine_quota_cost.
- [16] “Debian -- Details of package youtube-dl in sid.”
<https://packages.debian.org/sid/youtube-dl>.
- [17] “pytube — pytube 12.1.0 documentation.” <https://pytube.io/en/latest/>.
- [18] S. S. Stevens, J. Volkman, and E. B. Newman, “A Scale for the Measurement of the Psychological Magnitude Pitch,” *The Journal of the Acoustical Society of America*, vol. 8, no. 3, pp. 185–190, Jan. 1937, doi: 10.1121/1.1915893.
- [19] J. M. Grey and J. W. Gordon, “Perceptual effects of spectral modifications on musical timbres,” *The Journal of the Acoustical Society of America*, vol. 63, no. 5, pp. 1493–1500, May 1978, doi: 10.1121/1.381843.
- [20] B. McFee *et al.*, “librosa/librosa: 0.9.2.” Zenodo, Jun. 27, 2022. doi: 10.5281/zenodo.6759664.
- [21] D. Moffat, D. Ronan, and J. D. Reiss, “An Evaluation of Audio Feature Extraction Toolboxes,” p. 7, 2015.
- [22] D. Imran, “Music Emotion Recognition.” Aug. 12, 2022. Accessed: Aug. 13, 2022. [Online]. Available: <https://github.com/danz1ka19/Music-Emotion-Recognition>
- [23] “Different Types of Regression Models,” *Analytics Vidhya*, Jan. 19, 2022.
<https://www.analyticsvidhya.com/blog/2022/01/different-types-of-regression-models/>.
- [24] C. Cortes and V. Vapnik, “Support-vector networks,” *Mach Learn*, vol. 20, no. 3, pp. 273–297, Sep. 1995, doi: 10.1007/BF00994018.
- [25] E. Fix and J. Hodges, “Discriminatory Analysis - Nonparametric Discrimination: Consistency Properties,” CALIFORNIA UNIV BERKELEY, Feb. 1951. Accessed: Aug. 13, 2022. [Online]. Available: <https://apps.dtic.mil/sti/citations/ADA800276>
- [26] T. Eerola, O. Lartillot, and P. Toivainen, “Prediction of Multidimensional Emotional Ratings in Music from Audio Using Multivariate Regression Models,” Jan. 2009, pp. 621–626.
- [27] Y.-H. Yang, Y.-C. Lin, Y.-F. Su, and H. H. Chen, “A Regression Approach to Music Emotion Recognition,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 2, pp. 448–457, Feb. 2008, doi: 10.1109/TASL.2007.911513.

[28] “Browser Market Share Worldwide,” *StatCounter Global Stats*.
<https://gs.statcounter.com/browser-market-share>.

[29] “API Reference,” *Chrome Developers*.
<https://developer.chrome.com/docs/extensions/reference/>.