



|  |
| --- |
| PRATİK ÇALIŞMA DALI : Bilgisayar Görü |
| SOYADI, ADI ve NO :Türkmen, Mehmet Vasfi 210229032 |
| BÖLÜM / SINIF : Yazılım Mühendisliği / 3.sınıf |

**KURUMA AİT BİLGİLER**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **KURUMUN ADI** | **KISIM** | | ÇALIŞMA SÜRESİ | | | **HAFTA** |
| Tarihinden | | Tarihine |
| **Diginova** | **Bilgisayar Görü** | | **01.07** | | **05.07** | **1** |
| **Diginova** | **Bilgisayar Görü** | | **08.07** | | **12.07** | **2** |
| **Diginova** | **Bilgisayar Görü** | | **16.07** | | **19.07** | **3** |
| **Diginova** | **Bilgisayar Görü** | | **22.07** | | **26.07** | **4** |
| **Diginova** | **Bilgisayar Görü** | | **29.07** | | **02.08** | **5** |
| **Diginova** | **Bilgisayar Görü** | | **05.08** | | **09.08** | **6** |
|  |  | |  | |  |  |
| **ÖĞRENCİNİN DURUMU** | | | | | | |
| **TAVIR VE HAREKETLER,** | | **ÇALIŞKANLIK,** | | **YETENEK,** | | |
| **1/** KONTROL EDEN AMİR, ÖĞRENCİNİN PRATİK SIRASINDAKİ DURUMUNU BİLDİRİR.  **2/** YUKARIDA YAZILI TAVIR ve HAREKETLER, ÇALIŞKANLIK, YETENEK DURUMLARININ TASNİFİ  **A/** PEKİYİ  **B/** İYİ  **C/** ORTA  **D/** GEÇER  **F/** KÖTÜ | | | | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **01.07.2024 TARİHİNDEN, 09.08.2024 TARİHİNE KADAR 6 HAFTALIK ÇALIŞMA** | | | |
| **GÜN** | **YAPILAN İŞLER** | **SAYFA NO:** | **ÇALIŞILAN SAAT:** |
| **PAZARTESİ** | Tanışma ve Şirket Sisteminin Anlaşılması | 4 | 09.00-19.00 |
| **SALI** | Pydantic Kütüphanesini Öğrenme | 5 | 09.00-19.00 |
| **ÇARŞAMBA** |  |  |  |
| **PERŞEMBE** |  |  |  |
| **CUMA** | Docker Kursu Tamamlama Örnek Projeleri Ayağa Kaldırma | 6 | 09.00-19.00 |
| **CUMARTESİ** |  |  |  |
| **KISIM: Bilgisayarlı Görü** | | **TOPLAM SAAT: 27** | |
| **ÇALIŞMTIĞI İŞYERİ VE KISIM: Diginova - Bilgisayarlı Görü**  **KONTROL EDENİN ÜNVANI:**  **SOYADI, ADI VE İMZASI:** | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **GÜN** | **YAPILAN İŞLER** | **SAYFA NO:** | **ÇALIŞILAN SAAT:** |
| **PAZARTESİ** | Linux Kursunu Başlama ve CLI Komutlarını Kullanma | 7 | 09.00-19.00 |
| **SALI** | Linux kursunun tamamlanması ve admin komutlarının denenmesi | 8 | 09.00-19.00 |
| **ÇARŞAMBA** |  |  |  |
| **PERŞEMBE** |  |  |  |
| **CUMA** | Model Eğitme Öncesi Araştırma | 9 | 09.00-19.00 |
| **CUMARTESİ** |  |  |  |
| **KISIM: Bilgisayarlı Görü** | | **TOPLAM SAAT: 27** | |
| **ÇALIŞMTIĞI İŞYERİ VE KISIM: Diginova - Bilgisayarlı Görü**  **KONTROL EDENİN ÜNVANI:**  **SOYADI, ADI VE İMZASI:** | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **GÜN** | **YAPILAN İŞLER** | **SAYFA NO:** | **ÇALIŞILAN SAAT:** |
| **PAZARTESİ** |  |  |  |
| **SALI** | Model Eğitimi İçin Temel Bilgilerin Edinilmesi | 10 | 09.00-19.00 |
| **ÇARŞAMBA** |  |  |  |
| **PERŞEMBE** |  |  |  |
| **CUMA** | Alternatif Araç ve Yöntemlerin Araştırılması | 11 | 09.00-19.00 |
| **CUMARTESİ** |  |  |  |
| **KISIM: Bilgisayarlı Görü** | | **TOPLAM SAAT: 18** | |
| **ÇALIŞMTIĞI İŞYERİ VE KISIM: Diginova - Bilgisayarlı Görü**  **KONTROL EDENİN ÜNVANI:**  **SOYADI, ADI VE İMZASI:** | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **GÜN** | **YAPILAN İŞLER** | **SAYFA NO:** | **ÇALIŞILAN SAAT:** |
| **PAZARTESİ** | Örnek Projelerin İncelenmesi | 12 | 09.00-19.00 |
| **SALI** | Parametreleri Değiştirerek Doğruluk Oranını Yükseltmek | 13 | 09.00-19.00 |
| **ÇARŞAMBA** |  |  |  |
| **PERŞEMBE** | Modele Uygun Image’ı Build Etme | 14 | 09.00-19.00 |
| **CUMA** |  |  |  |
| **CUMARTESİ** |  |  |  |
| **KISIM: Bilgisayarlı Görü** | | **TOPLAM SAAT: 27** | |
| **ÇALIŞMTIĞI İŞYERİ VE KISIM: Diginova - Bilgisayarlı Görü**  **KONTROL EDENİN ÜNVANI:**  **SOYADI, ADI VE İMZASI:** | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **GÜN** | **YAPILAN İŞLER** | **SAYFA NO:** | **ÇALIŞILAN SAAT:** |
| **PAZARTESİ** | Konfigürasyon İçin Gerekli Dökümanların İncelenemsi | 15 | 09.00-19.00 |
| **SALI** | Hata Düzeltmeleri ve Konfigürasyonların Değiştirilmesi | 16 | 09.00-19.00 |
| **ÇARŞAMBA** | Konfigürasyon Dosyasının Yeniden Yazılması | 17 | 09.00-19.00 |
| **PERŞEMBE** | Konfigürasyon Dosyasındaki Hatanın Düzeltilmesi | 18 | 09.00-19.00 |
| **CUMA** |  |  |  |
| **CUMARTESİ** |  |  |  |
| **KISIM: Bilgisayarlı Görü** | | **TOPLAM SAAT: 36** | |
| **ÇALIŞMTIĞI İŞYERİ VE KISIM: Diginova - Bilgisayarlı Görü**  **KONTROL EDENİN ÜNVANI:**  **SOYADI, ADI VE İMZASI:** | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **GÜN** | **YAPILAN İŞLER** | **SAYFA NO:** | **ÇALIŞILAN SAAT:** |
| **PAZARTESİ** | Export İçin Gerekli Dosyanın Yazılması | 19 | 09.00-19.00 |
| **SALI** | Hata Düzeltme | 20 | 09.00-19.00 |
| **ÇARŞAMBA** | Paketteki Eksik ve Hatalı Dosyaların Düzeltilmesi | 21 | 09.00-19.00 |
| **PERŞEMBE** | Modelin Sisteme Eklenmesi | 22 | 09.00-19.00 |
| **CUMA** | Paketin Tamamlanması | 23 | 09.00-19.00 |
| **CUMARTESİ** |  |  |  |
| **KISIM:** | | **TOPLAM SAAT: 45** | |
| **ÇALIŞMTIĞI İŞYERİ VE KISIM: Diginova - Bilgisayarlı Görü**  **KONTROL EDENİN ÜNVANI:**  **SOYADI, ADI VE İMZASI:** | | | |

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 01.07 |
| YAPILAN İŞ: Tanışma ve Şirket Sisteminin Anlaşılması | |
| Stajın ilk gününe şirketin yöneticilerinin, mühendislerinin ve stajyerlerin yer aldığı bir toplantıyla başladık. Tanışmanın ardından şirket sistemi tanıtıldı, şirkete entegre olabilmemiz için gerekli dökümanlar paylaşıldı ve görevlendirmeler özetlendi.  Toplantınn ardından verilen dökümanlara bakıp eksik olduğum alanları tespit ettim ve bunlar üzerine çalıştım. Staj öncesi verilen görevlendirmeyi tekrar inceledim. Daha önceki stajyerlere verilen eğitimlerin kayıtlarını izledim. Bu videolardan Docker’ın şirket sisteminde ne için ve nasıl kullanıldığını gördüm. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 02.07 |
| YAPILAN İŞ: Pydantic Kütüphanesini Öğrenme | |
| Bugün görev olarak şirket sisteminde kullanılan Pydantic kütüphanesini öğrenmemi ve basit alıştırmalar yapmamı ardından github üzerinden alıştırmalarımı paylaşmamı istediler. İşe öncelikle bu kütüphanenin ne amaçla kullanıldığını araştırarak başladım. Ardından kütüphanenin dökümantasyonunu inceledim.  Pydantic, Python'da veri doğrulama ve veri yönetimi için kullanılan bir kütüphanedir. Özellikle veri modelleri oluşturmada ve bu modellerin doğrulanmasında büyük kolaylık sağlar. Kütüphaneyi incelemeye dökümantasyon üzerinden başladım ve ilk olarak veri modellerini nasıl oluşturabileceğimi öğrendim. Anladığım kadarıyla Pydantic, Python'un veri türlerini kullanarak şema tanımlamayı ve bu şemaların otomatik olarak doğrulanmasını sağlayan bir kütüphaneymiş. Bunun yanında hatalı veri girişlerinde, açıklayıcı hata mesajları üretmesi, veri doğrulama sürecini oldukça kolaylaştırmaktadır.  İlk adım olarak, basit bir model oluşturup veri doğrulamasını test ettim. Örneğin, bir kullanıcı modelini tanımlayıp, kullanıcı bilgilerini bu modele göre doğruladım. Pydantic'in veri doğrulama ve dönüştürme özelliklerini kullanarak, eksik veya hatalı verileri tespit etme ve gerekli düzeltmeleri yapma konusunda deneyim kazandım.  Bu alıştırmaların ardından oluşturduğum örnek projeyi GitHub üzerinde paylaştım. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 05.07 |
| YAPILAN İŞ: Docker Kursu Tamamlama Örnek Projeleri Ayağa Kaldırma | |
| Bugün görev olarak verilen Docker kursunun temel bölümlerini tamamlamamı ve alıştırma yapmamı istediler. Kurs hem İngilizce hem de Urduca olduğu için anlamam biraz zor oldu ama videodaki şemalar ve örnek kodlar üzerinden gördüklerimi inceleyerek ilerledim. Docker’ı özetlemek gerekirse Docker, uygulamaların bir container içinde çalışmasını sağlayan bir platformdur. Bu containerlar, uygulamanın tüm bağımlılıklarını ve ortam yapılandırmalarını içerir, böylece uygulamalar her ortamda tutarlı bir şekilde çalışabilir. Docker, yazılımın “bir kez yaz, her yerde çalıştır” prensibini gerçekleştiren bir araçtır. Geliştiricilere, uygulamaları ve servisleri izole edip bağımsız çalıştırma imkanı sunar, bu da özellikle geliştirme, test ve dağıtım süreçlerinde büyük kolaylık sağlar.  Kapsayıcıların temel avantajı, sanal makinelerden daha hafif olmasıdır. Çünkü Docker, işletim sisteminin tamamını sanallaştırmak yerine sadece uygulama için gerekli bileşenleri izole eder ve çalıştırır. Bu sayede, sistem kaynaklarını daha verimli kullanmak ve farklı platformlarda uyumluluk sorunlarını en aza indirmek mümkün olur.  Dockerfile, bir uygulamanın nasıl containerize edileceğini tanımlayan bir betik dosyasıdır. Docker image ise uygulamanın çalıştırılabilir bir sürümüdür ve bu image’den türeyen container’lar sayesinde uygulama çalıştırılır. Volume'ler, uygulamanın veri depolamasını sağlar ve container silinse bile verilerin kaybolmamasına olanak tanır.  Kurs sırasında öğrendiklerimi pekiştirmek için çeşitli alıştırmalar yaptım. Örneğin, basit bir Python uygulamasını Docker container içerisinde çalıştırmayı denedim ve başarılı bir şekilde çalıştırdım. Ardından github’tan herhangi bir projeyi alıp docker üzerinden ayağa kaldırmaya çalıştım. Bu kısımda tam istediğimi başaramadım ama bir image’ın nasıl çalıştığını nasıl kullanıldığını öğrendim diyebilirim. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 08.07 |
| YAPILAN İŞ: Linux Kursunu Başlama ve CLI Komutlarını Kullanma | |
| Bugün görevim Linux kursunun belirli kısımlarını tamamlamak ve temel CLI komutlarını öğrenmek. Komutları denemek için WSL üzerinden Ubuntu kullandım. Linux kursunda, temel komut satırı arayüzü (CLI) komutlarını öğrenmeye odaklandım ve pratik yapmak için Windows Subsystem for Linux (WSL) üzerinden Ubuntu kullandım. CLI, Linux sistemlerinde sıkça kullanılan güçlü bir araçtır ve sistem yönetiminden dosya yönetimine kadar birçok işlemi komutlarla gerçekleştirmeye olanak tanır. Öğrendiğim temel komutlar bazıları şunlar:  ls: Dizin içeriğini listelemek için kullanılır. Örneğin, ls -l komutu, dosya ve dizinlerin detaylı bilgilerini gösterir.  cd: Dizin değiştirme komutudur. Örneğin, cd /home komutuyla /home dizinine geçiş yaparım.  pwd: Bulunduğunuz dizinin yolunu gösterir.  mkdir: Yeni bir dizin oluşturur. Örneğin, mkdir test komutuyla test adlı bir dizin yaratılır.  rm: Dosya veya dizin silmek için kullanılır. rm -r komutuyla bir dizini ve içeriğini silmek mümkündür.  cp: Dosya ya da dizin kopyalamak için kullanılır. cp file1.txt file2.txt komutuyla file1.txt dosyasını file2.txt olarak kopyalarım.  mv: Dosya veya dizin taşımak ya da yeniden adlandırmak için kullanılır.  Bu komutları WSL üzerinde deneyerek pratiğe döktüm ve Linux sistemini daha iyi anlamamı sağladı. CLI kullanımı sayesinde grafik arayüze bağımlı olmadan, hızlı ve verimli bir şekilde işlemlerimi gerçekleştirebildim. Bu temel komutlar, ileri düzey Linux işlemleri için güçlü bir temel oluşturdu. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 09.07 |
| YAPILAN İŞ: Linux Kursunu Tamamlama ve Admin Yetkilerini Kullanma | |
| Bugün önceki gün aldığım notları tekrar ettim ve ardından daha özel amaçlı komutların yer aldığı bölümü izledim. Bu kısımda genelde detay işlemler yapıldığı için ilerlemem biraz daha yavaş oldu. Admin yetkileriyle ilgili alıştırmalar yaptım.  Admin yetkileri gerektiren işlemler üzerinde çalışmak, sistem yönetimi ve güvenlik açısından önemli bir adım oldu. Bu komutlarla dosya izinlerini ayarlayarak, sistemdeki kullanıcılar arasında veri güvenliğini sağlama konusunda daha fazla bilgi edindim. Aldığım notları uygulamaya dökerek yetkili işlemler ve sistem yönetimi konularında daha yetkin hale gelmeyi hedefledim. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 12.07 |
| YAPILAN İŞ: Model Eğitme Öncesi Araştırma | |
| Yeni görevim web sisteminde kullanılmak üzere bir model eğitmek. Bu model yüz ifadelerinden duygu çıkarımı yapacak. Daha önce YOLO kullandığım için aklıma CNN kullanmak geldi ama alternatif yöntemlere de bakıp karşılaştırmam istendi.  Daha önce benzer modeller eğitildiği için öncelikle onları inceledim. Genelde CNN kullandıkları için benim de seçtiğim yöntem bu oldu. Örnek veri setleri, modelleri ve bunlarla beraber kullanılan kütüphaneleri içeren bir rapor hazırladım. Ardından CNN ile ilgili videolar izledim. İşin bu tarafı kod yazmaktan çok araştırmakla geçiyor diyebilirim. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 16.07 |
| YAPILAN İŞ: Model Eğitimi İçin Temel Bilgilerin Edinilmesi | |
| Modeli eğitmeden önce daha önce eğitilen modellerden birkaçı karşılaştırdığım bir tablo hazırladım. Ayrıca bazı üniversitelerin hazırladığı özel veri setlerine ulaşmak için mail göndererek izin istedim. Olumsuz dönüş alsam da iyi bir deneme olduğunu düşünüyorum.  Bugün makine öğrenmesi ve derin öğrenme hakkında edindiğim bilgileri özetlemek üzerine çalıştım. Makine öğrenmesi, bilgisayarların veriler üzerinden öğrenerek belirli görevleri otomatik olarak gerçekleştirmesine imkan tanıyan bir yapay zeka dalı olarak öne çıkıyor. ML modelleri, deneyim yoluyla öğrenip genelleme yaparak yeni durumlara uyum sağlayabiliyor. Öğrendiğim kadarıyla, makine öğrenmesi üç ana kategoriye ayrılıyor: denetimli öğrenme, denetimsiz öğrenme ve pekiştirmeli öğrenme. Örneğin, denetimli öğrenmede, bir fotoğrafı kediler ve köpekler olarak etiketlediğimizde, model bu etiketleri kullanarak daha önce görmediği fotoğrafların kedimi yoksa köpek mi olduğunu tahmin etmeye çalışıyor. Denetimsiz öğrenmede ise elimizde etiket yok ve model, elimizdeki fotoğraflarda benzerlikleri bularak bunları sınıflara ayırmaya çalışıyor. Pekiştirmeli öğrenme, daha çok oyunlarda veya robotikte kullanılıyor; örneğin bir robot, doğru hareketleri yaparak ödüller kazandıkça doğru stratejiyi öğreniyor.  Makine öğrenmesinin kapsadığı ve yaygın kullanılan bir öğrenme çeşidi olan, derin öğrenme dikkatimi çekti. Yapay sinir ağları kullanarak çok katmanlı yapılar oluşturuyor ve her katman bir önceki katmanın verisini daha karmaşık bir şekilde işleyip, örüntüler keşfediyor. Bunu görevimi tamamlamak için kullanabileceğimi düşündüm. Yine araştırma yapmaya devam ettim. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 19.07 |
| YAPILAN İŞ: Alternatif Araç ve Yöntemlerin Araştırılması | |
| Bugün ilk olarak, Vision Transformer (ViT) modelini araştırdım ve bu modelin yüz ifadelerinden duygu çıkarımı üzerindeki etkisini değerlendirmek amacıyla FER (Facial Expression Recognition) datasetini denedim. ViT elimdeki veri seti için daha düşük bir doğruluk sonucu verdi ama mantığı ve ilerde başka alanlarda kullanabilmem açısından öğrenme sürecime önemli bir katkı sağladı.  Ayrıca, bugünkü çalışmalarım sırasında bir Excel tablosu oluşturarak elde ettiğim sonuçları güncelledim. Bu tablo, modelin performansını daha iyi anlamama ve analiz etmeme yardımcı oldu. Ancak, duygulara dayalı veri setlerini araştırmaya devam ederken, 7 duygudan fazla duygu barındıran bir veri seti bulmakta zorlandım. Bu, modelimi daha kapsamlı bir şekilde test edebilmem için önemli bir eksiklik oluşturdu.  Ayrıca, daha yüksek doğruluk oranları elde etmek için birkaç makale buldum. Ancak, bu makalelerde yer alan yöntemlerin detaylarını tam olarak anlayamadığım için uygulama aşamasına geçemedim. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 22.07 |
| YAPILAN İŞ: Örnek Projelerin İncelenmesi | |
| Bugün, birçok projeyi inceledim ve bunların yöntemlerini anlamaya çalıştım. Yüksek doğruluk oranı elde eden projeleri örnek alarak, seçtiğim projeler üzerinden ilerledim. Ancak, Google Colab üzerinde başarılı bir şekilde birleştirme işlemi yapamadım. Henüz yerel ortamda denemedim ama gerekli kurulumları yaparak en kısa zamanda bunu denemeyi planlıyorum.  Ayrıca, GitHub üzerindeki repoları inceledim ve uygulanacak yöntemleri seçtim. Bu süreçte örnek projeleri detaylı bir şekilde analiz ettim. Bu çalışmalar, farklı yaklaşımlar hakkında daha fazla bilgi edinmeme ve kendi projemde uygulayabileceğim stratejileri belirlememe yardımcı oldu. Gelecek günlerde bu projelerin yöntemlerini uygulamaya geçirme hedefim var. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 23.07 |
| YAPILAN İŞ: Parametreleri Değiştirerek Doğruluk Oranını Yükseltmek | |
| Bugün, farklı parametreler kullanarak modelimi eğitme sürecine odaklandım. Eğitim aşamasında epoch saysı, batch boyutu ve optimizasyon algoritması gibi kritik parametreleri değiştirerek çeşitli deneyler yaptım. Bu değişikliklerin modelin performansına olan etkilerini gözlemlemek, benim için son derece öğretici bir deneyim oldu.  Yüksek epoch kullanmanın modelin daha yüksek doğruluk yüzdesi verebildiğini ancak aşırı öğrenmenin (overfitting) riskini de beraberinde getirdiğini fark ettim. Bunun yanı sıra, farklı batch boyutları ile denemeler yaparak, modelin nasıl tepki verdiğini gözlemledim. Küçük batch boyutlarıyla çalışmanın, modelin daha fazla gürültüye maruz kalmasına yol açtığını ve büyük batch boyutlarının daha kararlı sonuçlar verdiğini keşfettim.  Tüm bu süreç, yalnızca teknik bilgi birikimimi artırmakla kalmadı, aynı zamanda farklı stratejiler denemenin ve bu stratejilerin sonuçlarını analiz etmenin önemini anlamamı sağladı. Yeni şeyler öğrenmek, modelin nasıl çalıştığını daha derinlemesine kavramama yardımcı oldu | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 25.07 |
| YAPILAN İŞ: Modele Uygun Image’ı Build Etme | |
| Görevimin bir sonraki adımı modelin web ortamında kullanılabilmesini sağlamak. Bunun için bugün bu konuyla ilgili temel adımları halletmem gerekecek. Öncelikli adımım uygun çalışma ortamını bilgisayarıma kurmak oldu.  Şirketin github reposunda model için hangi kütüphaneyi kullanacaksak onun için hazırlanmış bir docker image’ı bulunuyor. Kullanmayı seçtiğim modeli tensorflow kullanarak eğittiğim için tensorflow image’ını seçtim. Bu image’ın güncel olmadığı söylendi bu biraz beni zorlayacak da olsa ilerleyebileceğimi düşünüyorum.  Image’ı localde ayağa kaldırdım. Bir sonraki aşama modelin web ortamında kullanılabilecek hale getirilmesi. Bunun eğitimini farklı bir gün alacağım için günü geri kalanını araştırma yapmakla ve diğer stajyerlerle fikir alışverişi yaparak geçirdim. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 29.07 |
| YAPILAN İŞ: Konfigürasyon İçin Gerekli Dökümanların İncelenemsi | |
| Görevim eğittiğim modelin şirket sisteminde bir uygulumaya şeklinde kullanılabilmesini sağlamak. Bunun için önceden hazırlanmış olan dökümantasyonlar ve ayrıca videolu eğitim verildi.  Süreç konfigurasyon ayarlarının yapılması, web sitesinde uygulamaya çevrilmesi ve ardından bu uygulamanın yerel bilgisayarda denemesi şeklinde olacak diye tahmin ediyorum. Bugünü ilgili dökümanları inceleyerek ve basit konfigürasyonlar yaparak geçirdim.  Stajın ilk haftasında öğrendiğim Pydantic kütüphanesinin kullanıldığını görmem sürecin güzel ilerlediğini ve kendime bilgiler kattığımı düşünmeme neden oldu. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 30.07 |
| YAPILAN İŞ: Hata Düzeltmeleri ve Konfigürasyonların Değiştirilmesi | |
| Bugün, sistemdeki PackageModel.py dosyasına istek (request) işlemi yapan inference.py adlı dosyayı düzenledim. Bu düzenlemeyi yaparken, karşılaştığım hataları tespit etmek ve çözüm bulmak için debugging aracını kullandım. Debugging süreci, kodun hangi aşamalarda hata verdiğini net bir şekilde görmemi sağladı ve oldukça faydalı oldu.  Özellikle, isteklerin doğru şekilde gönderilip gönderilmediğini takip etmek için adım adım ilerleyip hataları hızlıca bulabildim. Hataları çözmek için debug modunda kodu çalıştırmak, işlem sürecini daha net anlamamı sağladı ve sorunları daha hızlı çözdüm.  Bu süreç, hem kodun düzgün çalışmasını sağlamak hem de debugging aracını kullanma becerilerimi geliştirmek açısından çok verimli oldu. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 31.07 |
| YAPILAN İŞ: Konfigürasyon Dosyasının Yeniden Yazılması | |
| Konfigürasyonları PackageModel.py adındaki bir Python dosyası üzerinden yazmam gerekiyor ama dün bunu yazarken bir hata yapmış olmalıyım ki istek atmaya çalıştığımda istediğim sonucu alamadım. Bunun nedenle bugün bu konuyla ilgilendim. Burada eksiğimin Nesneye Yönelik Programlama olduğunu fark ettim. Hemen bu konuyla ilgili videolar izledim.  Konfigürasyonlar için kullandığımız diğer dosyalara göz attım dökümantasyonları yeniden inceleyip PackageModel.py dosyasını yeniden yazdım. Deneme fırsatım olmadı çünkü bu sefer web sistemine yükleyemedim. İlgili mühendis sorunun benim yazdığım koddan kaynaklandığını tekrar düzenlemem gerektiğini söyledi. Tekrar incelemem gerekecek gibi görünüyor. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 01.08 |
| YAPILAN İŞ: Konfigürasyon Dosyasındaki Hatanın Düzeltilmesi | |
| Bugün PackageModel.py dosyasındaki yani konfigürasyonların yer aldığı dosyadaki hatalarımı bulup düzeltmeye çalıştım. Temel sorun kalıtım aldığım sınıflarla uyuşmayan özellikler belirlemem olduğunu buldum. Bunun yanı sıra web sistemine yükleme yönetimimiz değişti. Daha önce developer olarak doğrudan yükleme yapıp çıktıyı json formatında alabiliyordum. Yeni sistemde uygulamayı denemek için kullanıcı gibi girip uygulama oluşturmamız gerekiyor. Aslında böyle olması benim açımdan daha iyi oldu çünkü burada Docker Compose kullandığımız için kendimi geliştirdiğimi düşünüyorum. Ayrıca kullanıcının nasıl bir çıktı aldığını daha iyi görebilmiş oldum.  Son değişikliklerin ardından uygulamayı denedim henüz istek atamıyorum ama konfigürasyon tarafındaki sorunların çözüldüğünü düşünüyorum. Request atabilmek için inference.py adında bir dosyayı yeniden düzenlemem gerekecek. Konfigürasyon dosyasından ders çıkardığımı düşünüyorum ve bunu daha kısa sürede bitirmeyi hedefliyorum. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 05.08 |
| YAPILAN İŞ: Export İçin Gerekli Dosyanın Yazılması | |
| Birkaç gündür aynı noktada takıldım kaldım. Süreci ilerletmek amacıyla konfigürasyonlarımın doğru olduğunu varsayıyorum ve bir sonraki adıma geçiyorum.  Bu adımda bir önceki adımda belirlediğim seçenekleri seçerek kendi istediğim özellikte bir uygulama hazırlamak istiyorum. Girdi olarak fotoğraf çıktı olarak ise hem duygu tespiti hem de fotoğraftaki yüzün çerçeveletilmiş halini istedim. Bu uygulamayı bilgisayarımda denediğimde çalışmadığını fark ettim. Ekip arkadaşımdan yardım istedim. Beraber sorunu inceledik. Bu aşamaya kadar daha önce debug kullanmıştım ama ne kadar kritik bir araç olduğunu bugün fark ettim.  Bugün sorunu çözemedik ama daha başka küçük hataları tespit ettik ve onları düzeltmemi istedi. Bu hatalar genelde modele verilen girdiler beklenen çıktılarla ilgili olduğu için kapı yine konfigürasyon dosyama çıkıyordu. Bir sonraki hedefim hatayı bulup düzeltmek olacak. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 06.08 |
| YAPILAN İŞ: Hata Düzeltme | |
| Neredeyse bir haftadır ilerleyememek canımı çok sıktı ve çalışma isteğimi büyük oranda düşürdü. İstediğim özellikleri ekleyemiyor bazen bunu başarsam bile web tarafına yüklediğimde beklediğim sonuçla karşılaşmıyordum. Bu nedenle pek mantıklı bir çözüm olmasa da sadece hataya sebep olamayan bir iki konfigürasyon hariç çoğu konfigürasyonu sildim. En azından temel isterleri karşılayan bir paket hazırlarsam ilerde özellik ekleyip çıkarmanın daha mantıklı olabileceğini düşündüm.  Ardından şirkette bir sonraki adımla ilgili olan dökümanları inceledim videoları izledim. Videolarda adımları uygulamaya başladım. Takıldığım noktalarda ekip arkadaşlarımdan yardım almaya başladım. Stajın bitimine yaklaştığım için verilen görevi bitirmek istiyordum bu nedenel her hatayla boğuşup zamanımı kötü yönetmek istemedim.  Ardından modelin sisteme yüklenmesi için gerekli olan utils.py adındaki dosyayı yazmaya çalıştım. Örnek olarak verilen önceki proje kapsamlı ve karmaşık olduğu için dosyayı istediğim şekilde yazamadım. Yarınn bu ufak sorunları aşmayı hedefliyorum. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 07.08 |
| YAPILAN İŞ: Paketteki Eksik ve Hatalı Dosyaların Düzeltilmesi | |
| Son düzenlemenin ardından özellikleri deneyip bir hata olup olmadığını kontrol ettim. Dün model yükleme için gerekli olan dosyada kalmıştım. Temel kısımları halletmiş bile olsam dosya doğru çalışmıyor ve modeli indiremiyordum. Bu nedenle bu dosyayı yazmış olan mühendisten yardım aldık ve bizimle sadeleştirilmiş halini paylaştı.  Ne yazık ki problemler burda da bitmedi. Model indiriliyor ama bu sefer web tarafında çalışmıyordu. Kendim çözmeye çalıştığımda bir sonuca varamadım  ekip arkadaşım modeli hangi sürümde eğittiğimi sorunca sorunu bulmuş olduk. Eğitmek için kullandığım Tensorflow sürümü ile image’da bulunan tensorflow sürümü aynı değildi.  Bu sorunu çözmek için Tensorflow 2.13.7 sürümünde yeni bir model eğittim. Bunu yarın sisteme ekleyip modeli sistemde test etmeyi hedefliyorum. Genel olarak bugün bunların dışında hata düzeltmeleriyle ve dosyaları düzenlemekle geçti. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 08.08 |
| YAPILAN İŞ: Modelin Sisteme Eklenmesi | |
| Staj başında eğittiğim model versiyon uyuşmazlığı nedeniyle sistemde çalışmamıştı bu nedenle sistemdeki tensorflow versiyonuna uygun bir model eğitip sisteme ekledim.  Yeni model önceden düzenlediğim utils dosyasıyla model sorunsuz bir şekilde indirilip çalıştırılıyordu. Modelden alınacak çıktıyı test etmek için inference.py dosyasını yazmıştım. Bugün hedefim bu dosyayı hatasız şekilde çalıştırılabilri hale getirmek ve bütün senaryoları request atarak test etmek.  Gün sonunda adım adım paketi bitimeye yaklaştığımı fark ettim.Çıktı olarak 7 duygunun her birinin yüzdesini içeren bir dizi alıyorumdum. Burdan sonrası sadece alınan çıktıyı biraz düzenlemek. Yarınki hedefim buradan en yüksek yüzdeyi alıp çıktı olarak sunmak. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**

|  |  |
| --- | --- |
| KISIM: Bilgisayarlı Görü | TARİH: 09.08 |
| YAPILAN İŞ: Paketin Tamamlanması | |
| Bugün modelinn girdilerini ve çıktılarını düzenleyeceğim ve böylelikle paket bitmiş olacak. Geriye sadece sürecin ve projenin dökümantasyonunun hazırlanması kalıyor.  Model girdi olarak yüz ifadesini almalı, yüz tespitini kendisi yapamıyor. Web sisteminde yüz tepiti yapan başka bir paketin hazır olması nedeniyle girdi olarak o modelin çıktısını vermeyi hedefliyordum ama geçici bir çözüm olarak doğruluk oranı daha düşük de olsa Haar Cascade kullandım. Çıktı kısmında da en yüksek yüzdeli duyguyu string tipinde yazdırdım. Paket genel olarak bitti denilebilir ama Tensorflow image’ı güncellenmesi gerekiyor.  Bugün son staj günümdü görevimi kısmen de olsa bitirdiğimi düşünüyorum. Süreç içeirsind ebirçok hata yaptım özellikle hataları kendim çözme konusunda yetersiz olduğumu gördüm bu yönümü geliştrimek için uğraşmam gerekecek. Ayrıca yazılım geliştirme sürecinde geri bildirim yapmayarak süreci aksattığımı düşünüyorum. Özellikle modelin girdi olarak sadece yüz ifadesi alması gerektiğini geç fark edip kendimce bulduğum çözümün yetersiz olabileceğini düşünmem gerekirdi.  Her şeye rağmen kendime bir şeyler kattığımı düşünüyorum. Özellikle daha tecrübeli ekip arkadaşlarımın destekleri ve soruna yaklaşma yöntemleri bana çok şey kattı. | |
| KONTROL SONUCU: | |

**Kocaeli Üniversitesi Mühendislik Fakültesi**