KÜTAHYA SAĞLIK BİLİMLERİ ÜNİVERSİTESİ
MÜHENDİSLİK VE DOĞA BİLİMLERİ FAKÜLTESİ

25.04.2024

Özge ÇITAKOĞLU
25.04.2024

YAPAY ZEKA RAPORU
Reinforcement Learning Kullanarak Robotun Labirentte Gezinmesi

1 Giriş

Reinforcement learning, yapay zeka alanında giderek daha fazla ilgi gören ve başarıyla uygulanan bir öğrenme paradigmasıdır. Bu paradigma, bir ajanın belirli bir çevre içinde deneyimler elde ederek, aldığı aksiyonların sonuçlarından geri bildirim alarak ve bu geri bildirimlere dayanarak çevresini keşfetme ve belirli hedeflere ulaşma yeteneğini geliştirmesine imkan tanır. Bu sayede, karmaşık ve belirsiz ortamlarda optimal kararlar alabilen sistemler geliştirilebilir, ki bu da otomatik sürüş, oyun stratejileri, robotik ve finans gibi birçok alanda heyecan verici uygulamalara olanak sağlar.

2 Literatür Araştırması

Literatürde DPO(Derin pekiştirmeli öğrenme) ve robot navigasyonu ile ilgili birçok çalışma bulunmaktadır.Mnih et al. (2015), Nature dergisinde yayınlanan bir çalışmada, DPO kullanarak Atari oyunlarında insanüstü performans elde etmeyi başarmıştır. Bu çalışma, DPO'nun karmaşık ve dinamik ortamlarda öğrenme yeteneğini göstermesi açısından önemlidir.

Lillicrap et al. (2015): arXiv preprint arXiv:1509.02971'de yayınlanan bir çalışmada, DPO kullanarak robotik bir kolun sürekli kontrolünü sağlamıştır ."Derin Deterministik Politika Gradyanları (DDPG)" adı verilen bir DPO algoritması kullanarak robotik bir kolun sürekli kontrolünü sağlamıştır. Bu çalışma, DPO'nun karmaşık ve dinamik ortamlarda robotları kontrol etmek için kullanılabileceğini göstermesi açısından oldukça önemlidir.DDPG algoritması, robotik bir kolun eklem açılarını girdi olarak alıp, her bir eylemin (eklem torkları) beklenen ödülünü tahmin eden bir sinir ağı modeli oluşturmak için eğitilmiştir[3].

OpenAI Five (2019): Dota 2 oyununda insanları yenebilen bir DPO modeli geliştirmiştir. Bu proje, literatürdeki mevcut çalışmalardan farklı olarak, 3D bir labirent ortamında robot navigasyonu için DPO'yu kullanmaktadır. Ayrıca, projede kullanılan veri seti ve algoritmalar, robotun labirentte daha hızlı ve daha verimli bir şekilde gezinmesine imkan sağlamaktadır [4].

Figure 1:

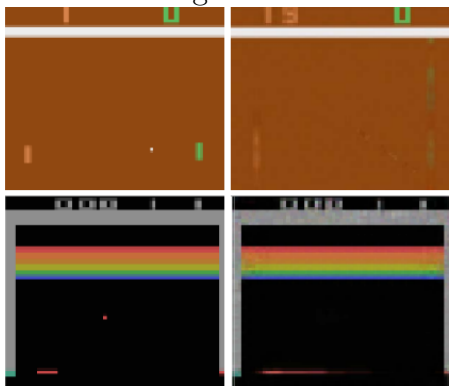


Figure 2:

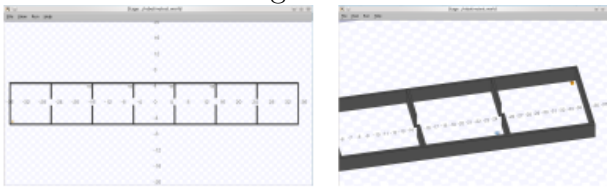


Figure 3:

a) Reinforcement Learning ile Çocuklarda Eğitsel Oyunların Geliştirilmesi: Bu çalışmada, çocuklarda eğitsel oyunların geliştirilmesi için reinforcement learning (RL) yöntemlerinin kullanılması incelenmiştir. Eğitsel oyunlar, çocukların öğrenme süreçlerini desteklemek ve eğlenceli bir ortamda bilgi edinmelerini sağlamak amacıyla tasarlanmıştır. RL algoritmaları, bu oyunların içerisinde kullanılarak çocukların performansına göre oyunun zorluk seviyesinin dinamik olarak ayarlanması hedeflenmiştir. Örneğin, çocukların matematik becerilerini geliştiren bir oyun içerisinde, RL algoritmaları kullanılarak çocukların doğru cevaplar verme süreçleri analiz edilmiş ve oyunun zorluk seviyesi bu doğrultuda ayarlanmıştır.

b) Yapay Zeka Destekli Öğrenme Oyunları ve Öğrenci Motivasyonu: Bu çalışma, yapay zeka (AI) destekli öğrenme oyunlarının öğrenci motivasyonunu artırmadaki etkisini araştırmaktadır. RL, öğrencilerin eğitim materyalleriyle etkileşime geçerek öğrenme süreçlerini optimize etmeye yardımcı olabilir. Örneğin, bir dil öğrenme uygulamasında, RL algoritmaları kullanılarak öğrencilerin kelime dağarcığını artırmak için etkili stratejiler belirlenmiştir. Bu stratejiler, öğrencilerin başarım düzeyine göre dinamik olarak ayarlanarak öğrenme sürecinin kişiselleştirilmesi sağlanmıştır. Bu çalışma, AI destekli öğrenme oyunlarının öğrenciler üzerindeki motivasyonu ve öğrenme performansı üzerindeki etkilerini değerlendirerek, gelecekteki eğitim teknolojilerinin tasarımında rehberlik edebilir.

c) Robotik Eğitiminde Güçlendirme Öğrenme Yaklaşımı: Bu araştırma, robotik eğitiminde güçlendirme öğrenme yaklaşımının etkisini incelemektedir. RL, robotların çevreleriyle etkileşime girerek yeni beceriler öğrenmelerini sağlayabilir. Örneğin, bir robotun belirli bir görevi gerçekleştirmesi için gerekli olan hareket dizilerini öğrenmesi için RL algoritmaları kullanılabilir. Bu çalışma, robotik eğitimde RL'nin kullanımının, öğrencilerin problem çözme becerilerini geliştirmede ve karmaşık görevleri yerine getirmede nasıl yardımcı olabileceğini araştırmaktadır. Bu bağlamda, robotik eğitiminde RL'nin potansiyel avantajları ve sınırlamaları değerlendirilmektedir

3 Metodoloji

Gereksinim Analizi: Oyunun temel gereksinimleri belirlenir. Bu, labirent yapısı, oyuncu ve düşman davranışları, oyun süresi gibi özellikleri içerir. Tasarım: Oyunun yapısı ve bileşenleri tasarlanır. Bu adımda labirent haritası nasıl oluşturulacak, oyuncu ve düşmanlar nasıl olacak, hangi kontroller kullanılacak gibi sorulara cevap aranır. Kod Geliştirme: Tasarlanan yapının kodlanması gerçekleştirilir. Python ve Pygame kullanılarak labirentin oluşturulması, oyuncu ve düşmanların hareketi, çarpışma kontrolü gibi işlevler kodlanır.

Test ve Hata Ayıklama: Geliştirilen kodlar test edilir ve hata ayıklama işlemi yapılır. Labirentin doğru çalıştığı, oyuncunun düşmanlardan kaçabildiği veya çarpışmaların doğru şekilde işlendiği kontrol edilir. Reinforcement Learning ile Oyun Geliştirme

Gereksinim Analizi: Reinforcement learning (RL) kullanarak geliştirilecek oyunun temel gereksinimleri belirlenir. Oyunun amacı, mekanikleri ve özellikleri detaylı bir şekilde incelenir. Bu analizde, oyuncu ve düşman davranışları, oyun mekaniği, oyun ortamı (labirent yapısı, düşmanların yerleşimi vb.), oyun süresi ve oyunun zorluk seviyeleri gibi unsurlar göz önünde bulundurulur.

Tasarım: Gereksinim analizinden elde edilen bilgiler doğrultusunda, oyunun yapısı ve bileşenleri tasarlanır. Labirent haritasının nasıl oluşturulacağı, oyuncunun ve düşmanların nasıl hareket edeceği, hangi kontrol mekanizmalarının kullanılacağı gibi sorulara cevaplar aranır. Ayrıca, RL algoritmasının oyun içinde nasıl uygulanacağı ve ödül sisteminin nasıl oluşturulacağı da bu aşamada belirlenir.

Kod Geliştirme: Tasarlanan oyun yapısının kodlanması gerçekleştirilir. Python ve Pygame gibi programlama dilleri ve kütüphaneleri kullanılarak labirentin oluşturulması, oyuncu ve düşmanların hareket etmesi, çarpışma kontrolleri gibi temel işlevlerin kodları yazılır. Aynı zamanda, RL algoritmasının uygulanması ve oyun içindeki dinamiklerin yönetimi için gerekli olan kodlar da bu adımda oluşturulur.

Test ve Hata Ayıklama: Geliştirilen oyun kodları test edilir ve hata ayıklama işlemi yapılır. Labirentin doğru bir şekilde oluşturulduğu, oyuncunun ve düşmanların beklenen şekilde hareket ettiği, çarpışmaların doğru bir şekilde işlendiği ve RL algoritmasının oyun içinde etkili bir şekilde çalıştığı kontrol edilir. Ayrıca, oyunun farklı senaryoları ve zorluk seviyeleri üzerinde testler yapılır ve gerekirse kodlarda düzeltmeler yapılır.

Labirent Oyunun Açıklaması: Kütüphane ve Sabit Değerlerin Tanımlanması: İlk olarak, pygame, random ve sys gibi gerekli kütüphaneler içe aktarılır. Ardından, ekran boyutu, renkler, oyun ayarları ve labirent haritaları gibi sabit değerler tanımlanır. Oyuncu Sınıfı: Oyuncu karakterini temsil eden bir sınıf tanımlanır. Oyuncu karakteri bir dikdörtgen (rect) olarak temsil edilir ve labirent içinde hareket eder. Oyuncu Hareketi ve Çarpışma Kontrolü: Oyuncunun hareketini ve labirent duvarlarına çarpma kontrolünü sağlayan yöntemler tanımlanır. Düşman Sınıfı: Oyunun düşmanlarını temsil eden bir sınıf tanımlanır. Düşmanlar da oyuncu gibi bir dikdörtgen (rect) olarak temsil edilir ve rastgele hareket ederler. Oyun Motorunun Başlatılması: Pygame kütüphanesi başlatılır, ekran oluşturulur ve gerekli gruplar oluşturulur. Ana Oyun Döngüsü: Oyunun ana döngüsü başlar. Bu döngü oyunun devamlı olarak çalışmasını sağlar. Oyuncu Kontrolü ve Oyun Durumları: Klavye

girişlerini kontrol eder, oyuncunun hareketini günceller ve oyun durumunu değerlendirir (kazanma, kaybetme veya devam etme). Oyun Süresi Kontrolü: Oyun süresini takip eder ve belirlenen süre dolunca oyunu sonlandırır. Ekran Güncelleme ve Çizimler: Her turda labirenti, oyuncuyu ve düşmanları çizer ve ekranda günceller. FPS Ayarı: Oyunun FPS değerini kontrol eder ve belirlenen FPS değerine ulaşılmasını sağlar. Bu, oyunun düzgün bir şekilde çalışmasını ve uygun hızda görüntülenmesini sağlar. Bu kod, basit bir labirent oyununun temel yapısını oluşturur. Oyuncunun labirent içinde gezinmesi, düşmanlardan kaçması ve çıkışı bulması gereken bir oyun senaryosu üzerine kurulmuştur.

Figure 4:

Figure 5:

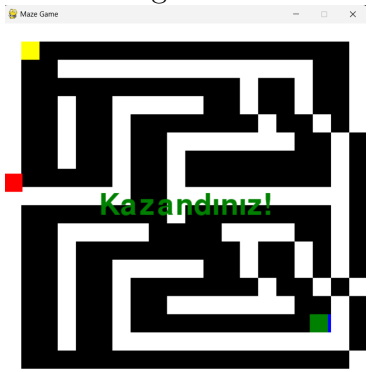
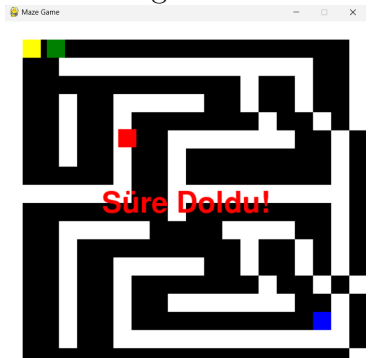
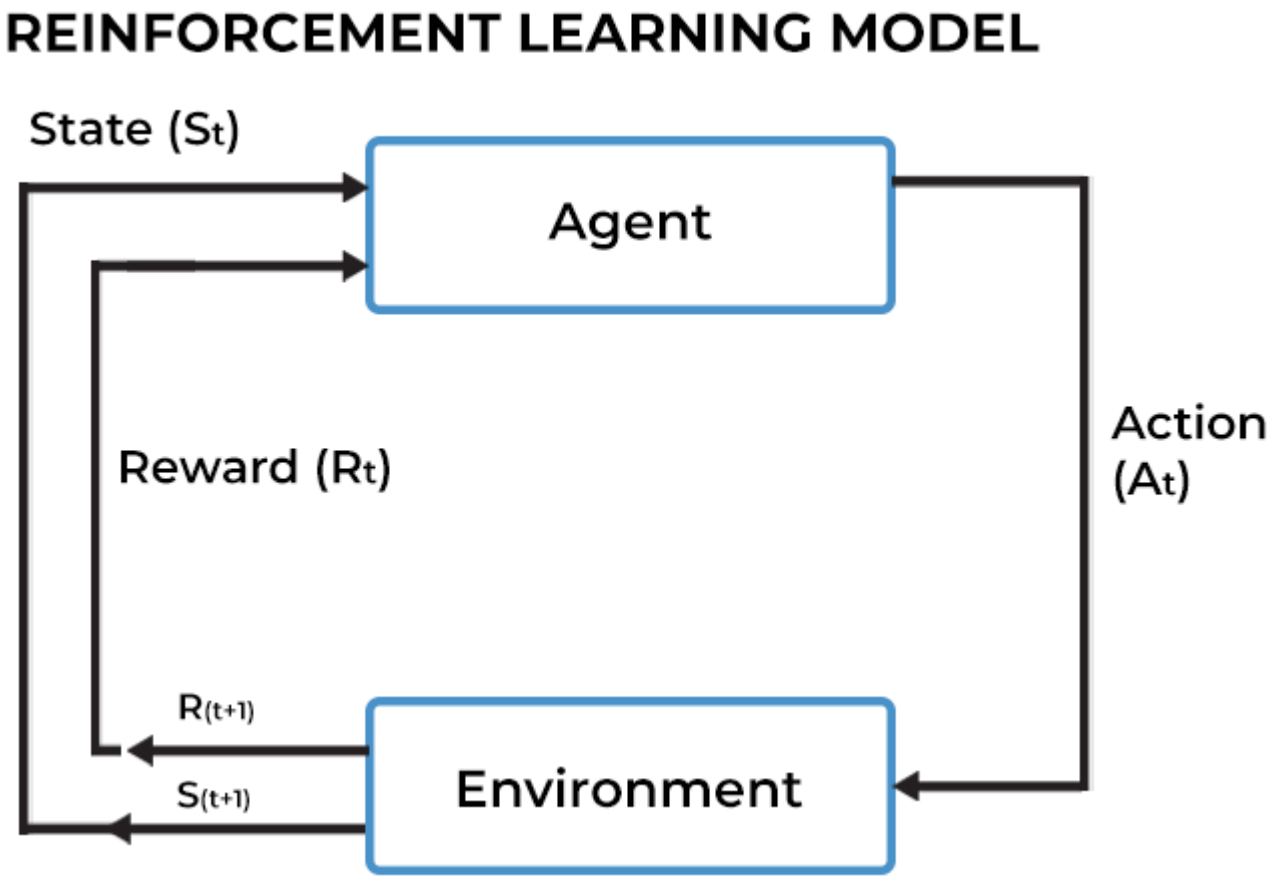


Figure 6:



4 VeriTabanı ve Veriler :



Reinforcement learning (RL), bir yapay zeka modelinin çevresiyle etkileşime geçerek deneyimlerden öğrenmesini sağlayan bir öğrenme paradigmasıdır. RL'nin temel amacı, bir ajanın belirli bir görevi en iyi şekilde nasıl gerçekleştireceğini öğrenmesidir. Bu öğrenme süreci, ajanın çevresiyle etkileşim halinde olması ve bu etkileşimlerden gelen geri bildirimlere dayanır.

RL'de, ajan bir ortamda belirli bir görevi gerçekleştirirken karşılaştığı durumları ve bu durumlara verdiği tepkileri öğrenir. Ajan, bu durumlar ve tepkiler arasındaki ilişkileri anlayarak, belirli bir hedefe ulaşmak için en uygun eylemleri seçmeyi öğrenir. Bu süreçte, ajanın hedefe ulaşma performansı, aldığı ödüller veya cezalarla ölçülür. Amacı, bu ödül sinyallerini en üst düzeye çıkaran stratejileri öğrenerek belirli bir görevi en iyi şekilde gerçekleştirecek politikaları bulmaktır.

Özetle, reinforcement learning'in amacı, bir yapay ajanın bir görevi en iyi şekilde gerçekleştirmesini sağlayacak stratejileri öğrenmesidir. Bu süreçte, ajan deneyimlerinden öğrenir ve çevresinden gelen geri bildirimleri kullanarak kendisini geliştirir.

References

[1] A. Güllü, "Labirentlerde yapay zeka tabanlı yön bulma algoritmaları kullanan bir gezgin robot geliştirilmesi," Master's thesis, Trakya Üniversitesi Fen Bilimleri Enstitüsü, 2017.

[2] N. Bölük, Ö. Uçar, and A. B. İnner, "Mobil robotlarda navigasyon problemi için pekiştirmeli öğrenme," *Türkiye Robotbilim Konferansı*, pp. 40–44, 2019.

[3] Ö. G. B. R. KAVASOĞLU, "Yapay zeka ve görsel tasarim uygulamalari,"

[**?**]. [**?**]. [**?**]. [**?**] [1]. [2]. [3].