

Yinelemeli ve Koşullu Derin Yapay Sinir Ağları ile Sistem Dinamiklerinin Öğrenilmesi

Learning System Dynamics via Deep Recurrent and Conditional Neural Systems

Mehmet Pekmezci

Bilgisayar Mühendisliği Bölümü

Özyeğin Üniversitesi

İstanbul, Türkiye

mehmet.pekmezci@ozu.edu.tr

Emre Ugur

Bilgisayar Mühendisliği Bölümü

Boğaziçi Üniversitesi

İstanbul, Türkiye

emre.ugur@boun.edu.tr

Erhan Oztop^{1,2}

Bilgisayar Mühendisliği Bölümü

¹Özyeğin Üniversitesi, İstanbul, Türkiye

²OTRI, SISReC, Osaka University, Osaka, Japon

erhan.oztop@{otri.osaka-u.ac.jp, ozyegin.edu.tr}

Özetçe —Sistem dinamiklerinin modellenmesi için çeşitli matematiksel yöntemler mevcut olsa da, verilerden yola çıkarak derin öğrenme ile daha genel çözümler bulunabilir. Yapay sinir ağlarındaki gelişmelere paralel olarak alternatif derin öğrenme yöntemleri sunulmaktadır. Bu çalışmada, zaman serisi verileri kullanarak seçilen dinamik sistem modelinin öğrenilmesi için hem LSTM tabanlı yinelemeli derin öğrenme yöntemi, hem de CNMP tabanlı koşullu derin öğrenme yöntemi kullanılarak uygulamalar yapılmıştır. Öğrenme için ihtiyaç duyulan zaman serisi veri miktarının ve öğrenilen sistem modeli kullanılarak yapılan tahminlerde ihtiyaç duyulan başlangıç girdi uzunluğunun her iki yönteme etkileri analiz edilmiştir.

Anahtar Kelimeler—LSTM, CNMP, sistem dinamikleri, derin öğrenme

Abstract—Although there are various mathematical methods for modeling system dynamics, more general solutions can be achieved using deep learning based on data. Alternative deep learning methods are presented in parallel with the improvements in artificial neural networks. In this study, both LSTM-based recurrent deep learning method and CNMP-based conditional deep learning method were used to learn the system dynamics of the selected system using time series data. The effects of the amount of time series data needed for training and the initial input length needed for predictions made using the learned system model on both methods were analyzed.

Keywords—LSTM, CNMP, system dynamics, deep learning

I. GİRİŞ

Bir sistemin kontrol edilmesi ya da o sistem içerisindeki bir aktörün davranışlarının öğrenilebilmesi gibi işlemlerin çoğunda ilgilendiğimiz sistemin gerçeğe olabildiğince yakın yapay bir modelinin olması tasarlanan sistemin başarımı için büyük önem taşır. Özellikle öğrenme adımıdaki veya kontrolör tasarımıdaki parametrelerin optimizasyonu için farklı seçimlerin çeşitli testlere tabi tutulması gerekir. Çoğu zaman gerçek sistemler üzerinde bu testlerin gerçekleştirilmesi çok maliyetli olabilir ya da teknik olarak mümkün olmayabilir.

Karmaşık sistemleri tanımlamak ve davranışlarını anlamak için 1961 yılında Forrester tarafından sistem dinamiklerinin formülize edilmesi fikri sunulmuştur [1]. Sonrasında karmaşık sistemlerin dinamiklerinin modellenebilmesi için yapılan çalışmalar da yapay sinir ağları alanındaki gelişmelerden etkilenmiş ve sistem dinamiklerini yapay sinir ağları ile öğrenme üzerine birçok çalışma yapılmıştır [5], [6], [10], [12]. Daha önceki çalışmalarda basit yapay sinir ağlarından [3], Convolutional Neural Network'lere [7] ve LSTM'e [8] kadar çeşitli farklı yapılarla dinamik sistem modelleme çalışmaları yapılmıştır.

Bu çalışmada OpenAI Gym platformunda modellenmiş olan sarkaç sisteminin [13] dinamiklerinin LSTM [4] ve CNMP [11] tabanlı iki farklı derin öğrenme yöntemi ile öğrenilmesi üzerinde analizler yapılmıştır. Sistem dinamiklerinin öğrenilmesi sürecinde kullanılan veri miktarının ve öğrenilen sistem dinamiklerinden yola çıkarak tahmin yapma aşamasında ihtiyaç duyulan girdi uzunluğunun her iki sistemin öğrenme başarımına etkisi incelenmiştir.

Yapılan deneylerle, göreceli olarak LSTM tabanlı yöntemin eğitim verisi miktarına hassasiyetinin daha düşük olduğu, ancak yeterli miktarda eğitim verisi sunulduğunda CNMP'nin sistem dinamiklerini daha başarılı öğrenebildiği gözlemlenmiştir. Öğrenilen sistem dinamiklerini kullanarak yapılan tahminlerde ise CNMP'nin girdi uzunluğuna olan hassasiyeti LSTM'e göre daha düşük olduğu, çok küçük girdi uzunlukları ile dahi başarılı tahminler yapabildiği gözlemlenmiştir.

II. İLİŞKİLİ ÇALIŞMALAR

A. LSTM (Long Short Term Memory)

1997 yılında Hochreiter ve Schmidhuber tarafından LSTM hücreleri önerilmiştir [4]. LSTM hücreleri, üzerinde sonradan yapılan çalışmalarla birlikte, içerisinde bir durum bilgisini barındıran, bu bilgiyi zaman içerisinde belli kurallara göre yeni gelen verilerle güncelleyen, eski bilgileri yine benzer kurallarla unutan ve çıktısı da bu kurallarla durum bilgisine göre güncelleyen bir yapıdır. LSTM hücrelerini bir araya getirerek oluşturulan LSTM katmanlarının standart yapay sinir ağları ile uygun şekilde birleştirilmesi vasıtasıyla oluşturulan LSTM yapay sinir ağları ile derin öğrenme gerçekleştirilmektedir. LSTM

yapay sinir ağlarının girdi boyutu kadar ardışık zaman serisi veri girdi olarak iletilir. Böylece her bir LSTM hücresi hem kendi girdilerini hem de kendisinden önceki LSTM hücresinin durumunu ve çıktısını kullanarak kendi durumunu ve çıktısını oluşturur.

B. CNMP (Conditional Neural Movement Primitives)

2018 yılında sunulan CNP (Conditional Neural Processes) yöntemi ile Bayesçi methodların [2] sunduğu önceki bilgiyi kullanarak öğrenme/tahminleme imkanı derin öğrenme için de sunulmuştur [9]. 2019 yılında sunulan CNMP bu fonksiyonalliteyi kullanan bir gösterim ile öğrenme yöntemidir [11]. Bu yöntemi zaman serisi verileri işlemek ve tahminlemek için kullanabilmek istediğimizde her bir öğrenme adımı aşağıdaki gibi olmaktadır.

- Her bir öğrenme adımında gösterimlerden bir tanesi seçilir.
- Seçilen gösterim içerisinde gözlem noktaları seçilir, işlenerek gizli bir katmana kodlanır.
- Seçilen gösterim içerisinde gözlem noktaları dışında koşul noktaları seçilerek yapay sinir ağı bu noktalardan geçecek bir çıktı üretmek üzere koşullandırılır.
- Seçilen gösterim içerisinde gözlem ve koşul noktaları dışında kalan noktalardan rastgele seçilen bir ya da birden çok nokta öğrenme için kullanılır.

Öğrenme sonrasında tahminleme yapabilmek için gözlem noktaları girdi olarak verilir, tahminleme yapılması istenen nokta ya da noktalar için koşullama yapılarak ilgili noktalar için tahminleme yapılması sağlanabilir.

III. METOD

Bu araştırma kapsamında sistem dinamiklerinin tespiti için zaman serisi veriler işlenmektedir. Open AI Gym [13] platformu üzerinde yapılan deneysel çalışmalarla zaman serisi veriler oluşturulmakta (durum yörüngesi), ve bunlar CNMP ya da LSTM derin öğrenme yapıları ile öğrenilmektedir. Bu işlemler sırasında yörüngeler örnekleme ile vektör haline dönüştürülerek işleme alınmaktadır. Öğrenme tamamlandıktan sonra herhangi bir yörünge vektörünün ilk n başlangıç değeri CNMP veya LSTM sistemine girdi olarak verilerek yörüngesinin sonraki bilinmeyen m durum bilgisini tahmin etmesi istenmektedir. Çalışmamızda tahmin etme performansının öğrenmede kullanılan yörünge sayısına ve n değerine bağlı olarak değişimi analiz edilmiştir.

A. Yapay Sinir Ağ Modelleri

1) *CNMP*: CNMP mimarisinde, kodlayıcı bir yapay sinir ağı ile gözlem noktaları gizli bir katmanda temsil edilecek şekilde kodlanmakta, sonrasında kod çözücü bir yapay sinir ağı ile kodlanmış temsili değerleri kullanarak koşullanan noktalar için durum bilgisi üretilmektedir. Empirik deneyler sonucunda kod çözen yapı 128 nöronlu 3 katmanlı, kod çözen yapı ise benzer şekilde 128'e nöronlu 2 adet gizli katmanlı ve 1 adet çıktı üreten katmanlı şekilde kurgulanmıştır.

Tahminleme aşamasında n adet girdi noktası ile tahminleme yapılması isteneceğinden, sisteme öğrenme aşamasında

verilecek maksimum sayıda koşul nokta sayısı n (pencere boyutu) ile sınırlandırılmıştır. Öğrenme süreci farklı girdiler için farklı olmakla birlikte, eşit koşullarda gözlemleyebilmek adına yapılan denemelerle, tüm deneyler için uygun bir tekrar sayısı olarak 1 milyon adım sayısı belirlenmiştir. Bu seçimin öğrenmenin tamamlanması için yeterli olduğunu kontrol etmek için yapılan 2 milyon adımlı öğrenme denemelerinde performansta önemli bir ilerleme görülmemiştir. CNMP sistemi verilen koşul noktalar altında istenen tüm zaman aralığı için tahminleme yapma yeteneğine sahiptir [11]. Bu sebeple CNMP ile tahminleme yapılacak yörüngenin ilk n noktası koşul olarak verilerek yörüngenin tamamının tek seferde tahminlenmesinin yapılması beklenmektedir.

2) *LSTM*: Bu çalışmada kullanılan LSTM modeli çeşitli testlerde farklı girdi boyutları (zaman penceresi) olan LSTM yapay sinir ağıdır. LSTM yapay sinir ağlarının çıktısı verilen zaman penceresinden bir sonraki adımdaki durum bilgisi olacak şekilde tasarlanmıştır. Böylece n uzunluğundaki zaman penceresindeki n adet ardışık durum bilgisi sisteme girdi olarak verildiğinde $n + 1$ 'inci durumun tahminlenmesi sağlanabilmektedir.

CNMP ve LSTM modellerini benzer koşullarda kıyaslayabilmek adına, seçilen CNMP modelinin parametre sayısına uygun bir LSTM modeli seçilmiştir. Buna göre her birinde 58'er adet LSTM hücresi bulunan 3 gizli katmanlı bulunan LSTM yapay sinir ağı tasarlanmıştır. Buna göre LSTM ve CNMP modelleri toplam yaklaşık 68000 adet parametre ile tanımlanmış hale gelmiştir. LSTM ile eğitim tüm veri seti 500 kez ağı sunulacak şekilde yapılmıştır. Empirik gözlemlerimiz bu miktar bir öğrenmenin ağı parametrelerinin yakınsamasına yettiğine işaret etmektedir.

LSTM yapay sinir ağına girdi olarak n adet zaman sıralı başlangıç durumu verilir. Her tahminden sonra oluşan çıktı bir sonraki tahmine girdi olarak kullanılan zaman penceresinin son adımı olarak tanımlanıp, zaman penceresinden bir önceki tahminde kullanılan ilk adım çıkarılır. Bu işlem m kez tekrarlanarak sonraki zaman sıralı m adet durumun tahmini gerçekleştirilebilir. Böylece bir yörüngenin ilk n değeri ile yörüngenin tümünün tahmini gerçekleştirilmiş olmaktadır.

B. Deney Düzenliği ve Ölçütler

Dinamiği öğrenilecek sistem olarak OpenAI Gym platformundaki sarkaç sistemi seçilmiştir [13]. Sarkaç düzenğinde hareketli bir eksene bağlı bir çubuk sistemin dinamiğine göre hareket etmektedir. Sistem iki serbestlik dereceli olup, durum vektörü çubuğun açısı ve açısal hızdan oluşmaktadır.

OpenAI Gym platformunda ise sarkacın durumu sarkacın açısal hızı, açısal konumunun sin ve cos değerleri ile ifade edilmektedir [13]. Açının kendisi yerine sin ve cos değerlerinin kullanılması devamlı bir durum uzayı oluşturulmasını sağlamaktadır. Sarkaç başlangıç hızı verilmeksizin bırakıldığı farklı konumlardan sistemin dinamiğine göre hareket ederek farklı açısal konumlar ve farklı açısal hızlara ulaşarak farklı durum yörüngeleri oluşturmaktadır.

Veri toplama aşamasında sarkacın başlangıç noktası olarak seçilebilecek $0 - 2\pi$ aralığındaki herhangi bir nokta başlangıç konumu olarak seçilir, başlangıç hızı verilmeksizin sarkaç sistem dinamikleri ile hareket etmek üzere serbest bırakılır.

Sistem dinamikleri etkisi ile yapılan t adımlık hareket esnasında içerisinde bulunan tüm durumların ardışık listesinden bir adet yörünge oluşur.

Sistem dinamiği başarılı şekilde öğrenilebildiği durumda, verilen başlangıç noktası ya da noktaları sonrasında sarkaç düzeneğinin durum yörüngesinin kalan kısmını başarılı şekilde tahminlemesi beklenmektedir. Yapılan tahminin genel doğruluğu, açısal konum ve açısal hız tahminlerinin başarımlarının bileşkesi olarak hesaplanmıştır. Bu hesaplamada konum ve hız tahminlerindeki yanılma miktarının doğruluk değerini benzer oranlarda etkilemesi için açısal hızın varyansı ile $\sin$ ve $\cos$ değerlerinin varyansları göz önüne alınarak Denklem 1'deki formül ile konsolide hata (H) hesaplanmış ve başarımlar ölçütü olarak kullanılmıştır.

$$H = 8(|\cos \theta - \cos \hat{\theta}| + |\sin \theta - \sin \hat{\theta}|) + |\dot{\theta} - \hat{\dot{\theta}}| \quad (1)$$

Bir yörünge tahmininin hata miktarı ölçülürken, ilgili yörünge tahmin yapılan her noktası için Denklem 1'deki formül çalıştırılarak hata değerleri hesaplanır. Hesaplanan değerlerin ortalaması yörünge tahmin hata değerini verir.

Yapılan deneyler için $0-2\pi$ aralığını eşit şekilde bölen 100 adet başlangıç açısı belirlenerek 50'şer adımlık 100 adet yörünge oluşturulmuştur. Her bir deney için ilgili yörüngelerden alt kümeler oluşturulmakta, yapılan deney sonucunda ilgili alt küme altındaki tüm yörüngeler için hesaplanan hata değerinin ortalaması alınarak ilgili alt kümeye ait ortalama hata değeri hesaplanmaktadır. Her bir deney için alt kümelerin hatalarının ortalaması ile standart hata değerleri ölçülerek bu bilgilerle analiz yapılmaktadır. 2 farklı tip deney ile inceleme yapılmıştır:

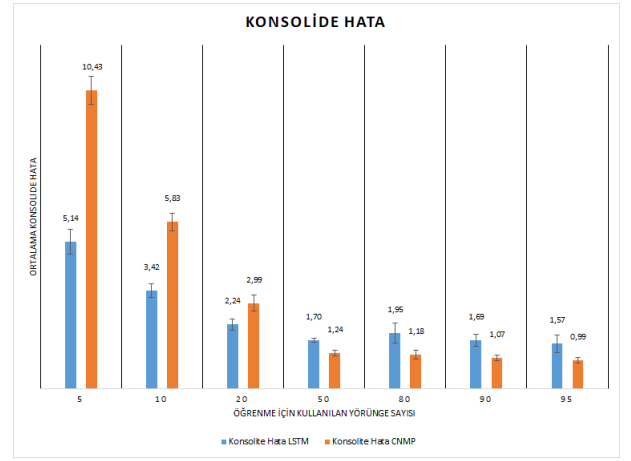
- Sistem dinamiğinin öğrenilmesi için sunulan yörünge sayısının öğrenme başarımına etkisi
- Yörünge tahmininin devamının tahminini yapmak üzere verilen başlangıç adım sayısının öğrenme başarımına etkisi

IV. PERFORMANS KARŞILAŞTIRILMASI

A. Öğrenme Aşamasında Kullanılan Yörünge Sayısı

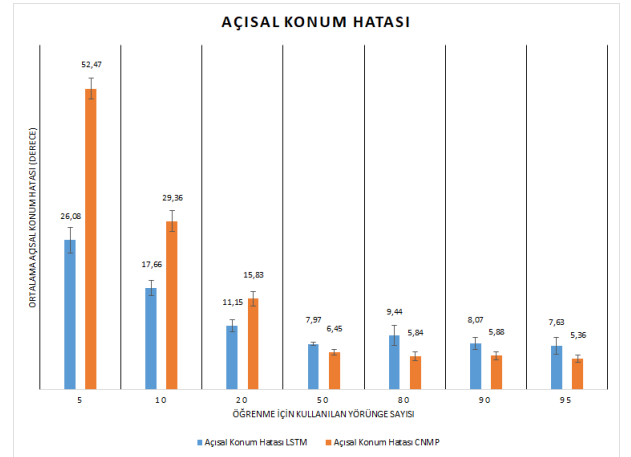
Bu deneyde CNMP ve LSTM yapay sinir ağları ile 50 adımlık yörünge tahmininin ilk 10 adımına bakarak geri kalan 40 adım tahminlemesi yapılması hedeflenmiştir. Bu tahminleme işleminde farklı sayılarda yörüngeyi eğitim için kullanarak tüm 100 adet yörünge tahminlemesinin yapılması sağlanmıştır. Bunun için farklı sayıda alt kümeler içeren deney grupları oluşturulmuştur. Yapılan deneye ait konsolide hataların kıyaslanması sonucu Şekil 1'de gösterildiği gibidir. Tüm şekillerde hata çizimleri standart hata hesabı ile işaretlenmiştir.

Buna göre LSTM tabanlı sistemin sistem dinamiklerini öğrenirken girdi verisine olan hassasiyeti göreceli olarak daha düşük olmuştur. Daha az sayıda yörünge ile yapılan eğitimlerde daha yüksek öğrenme başarımı göstermiştir. Ancak eğitim için kullanılan veri miktarı arttırıldıkça CNMP tabanlı yöntemin performansı tutarlı şekilde hızla artmış ve yeterli miktarda veri sunulduğunda LSTM tabanlı yöntemden daha yüksek doğrulukla sistem modelini öğrenebildiği gözlemlenmiştir.



Şekil 1: CNMP ve LSTM ile farklı sayılarda yörünge ile yapılan derin öğrenme performanslarının karşılaştırılması

Öte yandan LSTM ile yapılan testlerde hata oranı eğitim setindeki yörünge sayısı 50'ye çıkana kadar düşmüş, fakat 50'den sonra artan yörünge sayılarına rağmen hata oranının düşmediği gözlemlenmiştir.



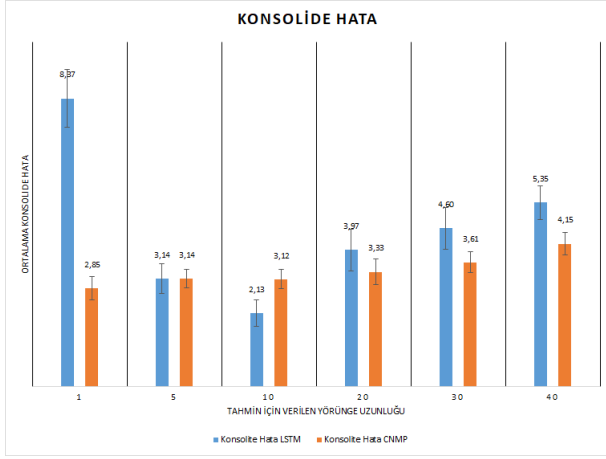
Şekil 2: CNMP ve LSTM ile farklı sayılarda yörünge ile yapılan derin öğrenme performanslarının açısal konum hatası ile kıyaslanması

Açısal konum hatası konsolide hatanın sadece bir bileşkesi olsa da Şekil 2'de görülen grafik sistem dinamiğini ne ölçüde başarılı tahminlediğine dair sezgisel bir anlayış için imkan sunmaktadır. Buna göre CNMP ile yapılan tahminlemelerde öğrenme için kullanılan yörünge sayısı arttırıldıkça tutarlı olarak 10 kata yakın bir performans artışı gözlemlenmiştir. Öte yandan LSTM ile yapılan tahminlerde en düşük hata oranına ortalama bir yörünge sayısında ulaşılmış olup, eğitim için kullanılan yörünge sayısında yapılan daha fazla artışın sistem dinamiğini tahminleme performansına olumlu bir etkisi olmadığı gözlemlenmiştir.

B. Tahminleme Aşamasında Verilen Yörünge Başlangıç Adım Sayısı

Bu deneyde CNMP ve LSTM yapay sinir ağları ile 50 adımlık yörünge tahmininin ilk n adımına bakarak geri kalan $50 - n$

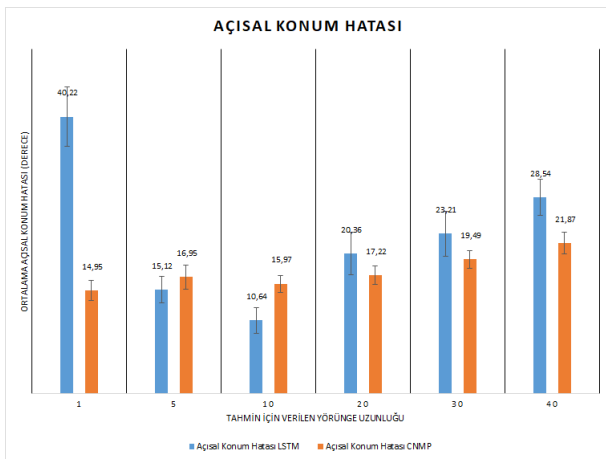
adımın tahminlemesi yapılması hedeflenmiştir. Bu tahminleme işleminde tüm testler için 5 farklı veri seti kullanılmıştır. Her bir veri seti 100 adet yörünge için farklı 20'şer adet eğitim ve 80'er adet test yörüngesine bölünmesi ile oluşur. Her bir deney alt kümesi ilgili veri seti için farklı bir n değeri seçilmesi ile oluşur. Gerçekleştirilen deneye ait konsolide hataların kıyaslanması sonucu Şekil 3'de gösterildiği gibidir.



Şekil 3: CNMP ve LSTM ile farklı girdi uzunlukları ile yapılan derin öğrenme performanslarının kıyaslanması

LSTM yönteminin girdi uzunluğuna hassasiyetinin CNMP'ye göre daha yüksek olduğu gözlemlenmiştir. Verilen girdi uzunluğu 10'a kadar yükseltildiğinde performans artışı görülmekle birlikte, girdi uzunluklarının 10'dan daha fazla olacak şekilde uzatıldığında hata miktarlarında gözle görülür artışlar tespit edilmiştir.

Aynı sonucun sezgisel olarak yorumlanabilmesi adına Şekil 4'de öğrenilen sistem dinamiği kullanılarak yapılan tahminlemelerde verilen girdi uzunluğunun yapılan tahminlerde ne kadarlık bir açısıl hataya neden olduğu bilgisi sunulmuştur.



Şekil 4: CNMP ve LSTM ile farklı girdi uzunlukları ile yapılan derin öğrenme performanslarının açısıl konum hatası ile kıyaslanması

SONUÇ VE İLERİKİ ÇALIŞMALAR

LSTM, zaman serisi veriyi, girdi uzunluğu boyutunda bir pencereyi kaydırarak öğrenmektedir. Yapılan deneylerde de girdi uzunluğunun LSTM'in öğrenme ve tahminleme performansı için önemli bir etmen olduğu gözlemlenmiştir. Ancak CNMP öğrenirken zaman serisi verinin tamamını gizli bir katmana kodladığı ve öğrenme aşamasında rastgele uzunlukta gözlem noktaları kullandığı için tahminleme aşamasında kısa girdi uzunlukları ile de yüksek başarımlar elde edebilmektedir. Öte yandan bir demonstrasyon yapması gereken test noktalarında başarımları düşük olabilmektedir. Yapılan deneylerde, rastgele örneklerle oluşturulan eğitim setinin testlerde ekstrapolasyon gerektirmeyecek bir dağılıma sahip olabilmesi adına, örnek sayısının belli bir değerin üzerinde olmasının CNMP başarımları için önemli bir etmen olduğu gözlemlenmiştir.

Buna göre, yeterli miktar ya da uygun dağılımda eğitim verisi mevcut ise ve/veya küçük girdilerle tahminleme yapılması gerekli ise CNMP'nin sistem dinamiği tahminleme performansı daha yüksek olacaktır. Diğer taraftan, eğitim için kullanılacak veri miktarı daha az fakat tahminleme için gözlemlenen girdi uzunluğu daha yüksek ise, LSTM seçimi daha yüksek performans sağlayacaktır.

İleriki çalışmalarda kıyaslamalara farklı derin öğrenme yöntemleri eklenebilir, sistem karmaşıklığının öğrenmeye etkisi incelenebilir.

KAYNAKLAR

- [1] J.W. Forrester. Industrial Dynamics. M.I.T. Press: Cambridge, MA., 1961.
- [2] G.E.P. Box, and G.C. Tiao "Bayesian Inference in Statistical Analysis," Reading, MA: Addison-Wesley, 1973
- [3] K. Narendra, K. Parthasarathy, "Identification and Control of Dynamic Systems Using Neural Networks," Neural Networks, IEEE Transactions on. 1. 4 - 27. 10.1109/72.80202, 1990
- [4] S. Hochreiter and J. Schmidhuber. "Long short-term memory," Neural computation, 9(8):1735–1780, 1997.
- [5] M. Atencia, G. Joya, and F. Sandoval, "Hopfield Neural Networks for Parametric Identification of Dynamical Systems," Neural Processing Letters, Vol. 21, pp. 143–152, 2005.
- [6] O.P. Ogunmolu, X. Gu, S.B. Jiang and N.R. Gans, "Nonlinear systems identification using deep dynamic neural networks," CoRR, 2016.
- [7] M. Lopez and W. Yu, "Nonlinear system modeling using convolutional neural networks," 2017 14th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE), pp. 1-5, Mexico City, 2017
- [8] W. Yu, "A new concept using LSTM Neural Networks for dynamic system identification," American Control Conference (ACC), 2017
- [9] M. Garnelo, D. Rosenbaum, C. Maddison, T. Ramalho, D. Saxton, M. Shanahan, Y. W. Teh, D. Rezende, and S. M. A. Eslami, "Conditional neural processes," In ICML, 1704-1713 2018.
- [10] W.A. Ramirez, Z.Q. Leong, H. Nguyen, S.G. Jayasinghe, "Non-parametric dynamic system identification of ships using multi-output Gaussian Processes," Ocean Engineering, Volume 166, Pages 26-36, 2018
- [11] M. Y. Seker, M. Imre, J. Piater, and E. Ugur, "Conditional neural movement primitives," in Robotics: Science and Systems (RSS), 2019.
- [12] P. Rajendra, V. Brahmajirao, "Modeling of dynamical systems through deep learning," Biophys Rev 12, 1311–1320, 2020.
- [13] Open AI Gym, Pendulum-v0, [Online]. Available: <https://gym.openai.com/envs/Pendulum-v0/> [Accessed Feb. 27, 2021]