Özdetik Bazı Tahminciler ile Çoklu-Nesne Sembolleri Üyegrenimi

Deep Multi-Object Symbol Learning with Self-Attention Based Predictors

Alper Ahmetoğlu
Boğaziçi Üniversitesi, İstanbul, Türkiye
alper.ahmetoglu@boun.edu.tr

Erhan Öztop
Özyeğin Üniversitesi, İstanbul, Türkiye
erhan.oztop@ozyegin.edu.tr

Emre Uğur
Boğaziçi Üniversitesi, İstanbul, Türkiye
emre.ugur@boun.edu.tr

Özetçe — Bu makalede, bir robotun değişken sayıda sesne ile etkileşime geçerek sûreki gösterimdeki alcı-motor deneyimlerinden sembolik gösterimler öğrenilen bir mimari önerilmiştir. Önceki çalışmalarında farklı olarak bu çalışma öğrenilen semboller üzerinde sabit sayıda etkileşen sesne veya önceden tanımlanmış sembolik yapılar gibi kısıtlamaları kaldırmaya amaçlamaktadır. Önerilen mimari, tekili nesneler için sembollerin ve aralarındaki ilişkilerin birleşik bir şekilde öğrenilebilidir ve gizgöz-gizgözücü sinir ağdır. Mimari, ilkli aktivasyon katmanı ve ikili aktivasyon katmanları bulunmaktadır. Deneyler, değişken sayıda nesnenin bulunduğu bir robotik manipülasyon ortamında gerçekleştirilmiştir. Sonuçlar, robotun kesfedilen semboller kullanarak değişken sayıda nesnenin bulunduğu etkileşimin dinamiklerini başarıyla kodladığını göstermektedir. Ayrıca, kesfedilen semboller ile bir robotik manipülasyon ortamında gerçekleştirebilir ve bu ağın hedef sembolik durumlarla ulaşmak için planlamada kullanılabilirliği gösterilmiştir.

Anahtar Kelimeler — robotik, sembol öğrenme, derin öğrenme

Abstract — This paper proposes an architecture that can learn symbolic representations from the continuous sensorimotor experience of a robot interacting with a varying number of objects. Unlike previous works, this work aims to remove constraints on the learned symbols such as a fixed number of interacted objects or pre-defined symbolic structures. The proposed architecture can learn symbols for single objects and objects in a unified manner. The architecture is an encoder-decoder network with a binary activation layer followed by self-attention layers. Experiments are conducted in a robotic manipulation setup with a varying number of objects. Results showed that the robot successfully encodes the interaction dynamics between a varying number of objects using the discovered symbols. We also showed that the discovered symbols can be used for planning to reach symbolic goal states by training a higher-level neural network.

Keywords — robotics, symbol learning, deep learning

I. Giriş

Bir robotun etkileşime geçtiği ortam sembolik bir formda ifade edilebilir ve bu girişi süreçlerine gereken eylem planı klasik yapay zeka arama tekniklerine başvurularak çözülebilir [1]. Bazi ortamlar uzmanlar tarafından tasarlanmış semboller ile tanımlanabilise de bu zaman alcı ve genel olmayan bir çözümdür. Son yıllarda önerilen bazı yöntemler sembollerini önceden tanımlamak yerine robotun ortamla etkileşimin verisi kullanarak ortamda planlama yapmayı sağlayacak semboller öğrenmektedir [2]–[6]. Bu çalışmalardan DeepSym [6] robotun ortamla etkileşimi sonucu toplanan veri ile tekli ve ikili nesneleri girdi olarak alır ve yapılan eylemin etkisi tahmin etmesini sağlayan semboller öğrenir. Bu nedenle oluşan semboller tek razı ile ilgildir, veya hik nesnelerin ilişkisi ile alakalıdır. Ayrıca tekli nesneler ve ikili nesneler için iki farklı sinir ağ eğitilmesi gerekmedektir.

Bu çalışmada, robotun değişken sayıda nesneyle etkileşime geçmesiyle ilde edilen veriler üzerinden tekn bir bütün sinir ağ ile eylem tahmini için gerekli sembollerin öğrenilmesi hedeflenmiştir. Bu amaçla darboğaz katmanında ilkli aktivasyonlar ve birlikte özükkatan katmanları [7] için bir sinir ağ mimarisi önerilmiştir. Simülasyon ortamında nesnelerin manipülenyonu ile yapılan deneylerde öğrenilen semboller sadece nesnelere ait özellikler içerikle kıyaslayıp aynı zamanda nesneler arasındaki etkileşimin dinamiklerini de göstermektedir. Sinir ağının eğitimi sonrasında öğrenilen semboller artırılır, ve bu sembollerin etkili tahminlerinde etkili oldukları gözlenmektedir. Ayrıca öğrenilen sembollerin çoklu nesnelerle sembolik muhakeme ile istenen hedeflere ulaşmak için planlamada kullanılabilirliği gösterilmiştir.

II. Yöntem

Önerilen mimari, Özükkatlı DeepSym, Şekil 1’dede göstelilmektedir. Bu mimarinin amacı verilen bir durumda (örneğin masası üstündeki nesnelerin konularını) robotun uygulayacağı eylemlerin etkilerini tahmin etmektedir. Denir bir yapay sinir ağı olaybu mimarinin normal yapaydan farklı darboğaz katmanında ilkli aktivasyonlar ve özükkatlı katmanlar içermesidir. Mimari, ilkli darboğaz katmanlarıyla birlikte bir gizgöz-gizgözücü h,f, bir gizgözücü g,h, ve bir özükkat modül a, eyelem) bulunmaktadır. Masası üstünden çıktılımsı görtüntü ortamın durumunu kullanmakla tremendur. Burada görtüntünün bölünebilen olduğu, ve nesnelerin yer değişimini anlamak için her bölüntünün zaman içinde hareketinin tespiti edildiği varsayılmaktır. Bu her ne kadar güçlü bir varsayım olsa da son teknoloji bölüneleme yöntemleri burada kullanılmadan ortamdan daha kârşık ortamlarda
nesneleri bölütleyebilir ve takip edebilir [8], [9]. Yüksek boyutlu sürekli durum için bölütleme modülü kullanılarak önce \((z_1, z_2, \ldots, z_k)\) bölümlerine bölünür. Birkaç nesneye ilişkin bilgileri kodlayabilen ilişkisel semboller elde etmeyi beklemiştir. Üstelik modülü çalışmadan temel eylem öğelerinin (nesneleri tutma, bırakma) var olduğu varsaydık. Gızılayıcı yüksek seviyede parametreli hareket öğeleri olarak kullanılmaktadır. Normal şartlarda bu fonksiyonun türevi her noktada sıfırdır, bu da geri yarım algoritması ile sinir ağının eğitilmesini engeller. Bu sorunu çözmemek için Gumbel-sigmoid fonksiyonu kullanılır [10], [11]. Gızılayıcı başarı için 1.000 örnek ve geriye kalanı 1.000 örnek test seti olarak kullanılır ve ortalama etki tahmini hataları toplamda 36 ile bulunmaktadır. Planlamaya uygun olması açısından eylemler ayrık bir şekilde gösterilmistir. Art arda uygulanan bazı eylemler Şekil 2b’de gösterilmiştir. Ortam, başlangıçta robot gövdesi daha yakın duraklı bir ila üç nesne ile başlar. Robot, masanın üstündeki yer alan bir derinlik kamerasıyla çevrilen algoritmalar 256 x 256 piksel bölümlenmiştir (her nesne farklı bir bölümler olacaktır, bzk. Şekil 1) derinlik görüntüleridir. Kameranın 64 x 64 piksele kütüptür. Nesnelerin konumlarını için karşılık görüntüün her pikselden x ve y konumlari iki ek kanal olarak birleştirilir.

Otomat k \in 1,2,3 nesneleri başlatır ve k rastgele olarak ayarlanır. Herhangi bir eylem yapılanın önce çevrenin başlangıç derinlik görüntüsünde \(s_i\) ve bölütemesi \(b_i\) kaydedilir. Ardından, rastgele bir eylem \(a_i\) gerçekleştirilebilir ve k nesneleri konunun değişimleri \(e_i\) ([(\(\Delta x_1, \Delta y_1, \Delta z_1\), \ldots, \(\Delta x_k, \Delta y_k, \Delta z_k\))], oluşturulan etkiler olarak kaydedilir. Burada, \(x, y\) ve \(z\) nesnelerin ağırlık merkezinin konum koordinatlarını temsil eder, ve \(\Delta\) ile nesneler için elde edilen derinlik değerleri temsil eder. \(e_i\) nin uzunluğu nesne sayısıyla bağlıdır. Toplamda, etkileşim veri seti olarak 12.000 \((s_i, b_i, a_i, e_i)\) çokzı oluşturur. Eğitim için 10.000 örnek, doğrulama için 1.000 örnek ve geriye kalan 1.000 örnek test için kullanılmıştır.

Şekil 1’deki mimari, \(a_i\) eyleminin ürettiği etki \(e_i\)’yi tahmin etmek üzere eğitilmiştir. Kodlayıcı \(f(s), 64, 128, 256\) ve 512 filtreli dört evrimsel katlandan oluşmaktadır. Evrimsel katmanları arasında yükseklik ve genişlik boyutları üzerinde bir ortama alınır ve sonrasında oluşan sahip boyutlu vektör bir tane yoğun katman ile 8-boyutlu vektöre indirgenir. Son olarak, geri yarım için Gumbel-sigmoid fonksiyonu kullanılarak aktivasyonlar ikili bale getirilir [10], [11]. Özdikkat modülü \(a(z, eylem)\), doğru gizayazı dönüşürücü katalanın bir dönüştürücüdür [7]. Dönüştürücü katalanın için PyTorch’un versiyonları çalıştırılabilir [12]. Gizçözücü \(g(h)\), her biri 256 birime sahip üç gizli katman içeren bir çok katmanlı ağlayıcıdır (MLP). Eğitim hızını artırmak için gizayazı ve gizçözücede yoğun normalleştirilmesi [13] kullanılmıştır.

### A. Etki Tahmini

Sistem öğrenilmişdir. Art arda uygulanan eylemler arasında öğrenilmiştir. Özdikkat, [6]‘da olduğu gibi piksel değerlerindeki değişim gibi daha genel bir etki kullanabilir.

### III. DENEYLER

Deney ortamı Şekil 2’de gösterilmektedir. UR10 robot kolumunun nesneleri önceden tanımlanmış altı konumdan alıp bırakıtır bu iki masası ortamıdır. Alt tutma ve alt bıraktırmak konumunu olduğu için toplamda 36 eylem bulunmaktadır.

![Şekil 1: Özdikkatlı DeepSym mimarisi.](image-url)
Karşılaştırma yapmak için etki setindeki ilgili boyutların ortalaması arasında 50 mm, 68 mm ve 10 mm'dir. Sonuç olarak, önerilen sistemin tek veya çoklu nesneler üzerindeki eylemlerin etkilerini tahmin etme yeteneği ve etkili sembolik gösternmeleri keşfettiğini söylenebiliriz.

**TABLO I:** Şekil 2c’deki örnek durumlar için etki tahmini. Birimler milimetre cinsindedir.

| Yeşil Küp | Turuncu Çubuk | Kırmızı Küp |
|----------|---------------|------------|
| **Dürum 1** | (3, 6, -2) | (-2, 192, -2) | (4, -9, -1) |
| **Dürum 2** | (3, 8, 0) | (-17, 196, 1) | (-10, 180, -5) |
| **Dürum 3** | (-3, 210, -45) | (4, 159, 8) | (-30, 131, -20) |

Bir sonraki analizde sistemin eyleme ilişkin nesnelerden otomatik olarak bilgi içeren semboller keşfetme yeteneğini ve bu sembollerin nesnelerin etkileşim dinamiklerini modelleyip modelleymedigi araştırılmıştır. Bunun için şu şekilde bir senaryo oluşturulmuştur: bir nesneyi uygulanan eylem aynı olsa da, ortamdaki diğer nesneler farklı şekillerde düzellenerek farklı etkilerin elde edilmesi ve sistem tarafından doğru bir şekilde tahmin edilmesi beklenmektedir. Eylem, uzun bir çubuğun bir konumdan alp başka bir konuma bırakmaktr. Diğer iki nesne farklı konfigürasyonlarda yerleştirilmiştir (Şekil 2c). Şekil 2c’de, robot tarafından uygulanan üç farklı örnek etkileşim üç sütunda gösterilmiştir. Her etkileşimin başlangıç anındaki görüntüyü üst sırada gösterilmiştir, eylem gerçekleştilirdiken sonraki görüntü alt sırada gösterilmiştir. Ayrıca her durumda en hassas pozisyonu da rapor edilmiştir. Kırmızı ve yeşil blokların farklı büyüklükte konfigürasyonları nedeniyle farklı etkiler gözlemlenmekte.

Sistem tarafından tahmin edilen etkiler ile gerçek etki değerleri Tablo I’de gösterilmiştir. Sonuçlar, sistemin nesneler arasındaki ilişkisel bilgiler modelleyebildiğini ve dolayısıyla doğru tahminlerde bulunduğunu göstermektedir. Örneğin, Durum 1’de sistem sadece çubuğun konumunun değişeceğini doğru bir şekilde tahmin etmektedir. Durum 2’de, kırık süpür konumu da çubukla birlikte değişmektedir. Son olarak, Durum 3’te, her iki kırık konumu da çubukla birlikte (aynı yönde) değişmektedir. Bu, sistemin "bir nesneden üzerindeki nesneler, altındaki nesne ile birlikte hareket eder" gibi çoklu nesneleri içeren yüksek seviyeli akıl ürünlerine yapayalnızca sağlayan semboller öğrencileri görmemektedir. Dönenürütücü katmanlardaki özdicktik gerçekten de doğru etkiyi tahmin etmek için ikili sembollerin birbirlerile etkileşime geçmesine yardımcı olduğu görülmüştür. Daha önceki çalışmadan, DeepSym [6], böyle etkileşimlerin modellenebilmesi için girdi seviyesinde gerekli nesne kesitleri önceden birleştirilmişdir. Ayrıca her farklı nesnenin sesi için gösterim sağlayabilmek için gerekli nesnelerin aktive edilmeleri gerekmektedir. Bu, DeepSym mimarisi ile karşılaştırıldığında açık bir avantajdır.

**B. Öğrenilen semboller**

Darboğaz katmanındaki ikili birim sayısını 8 birime ayarlanmıştır; bu nedenle en fazla 256 farklı sembol olabiliyor. Bu sembollerin aktivasyonu için analiz edildiğinde, en sık aktive edilen 35 sembollen eğitim setinin %95’inin kapsadığı gözlemlemiştir. Bu 35 belirli nesne sembollünü aktive eden örneklerden elde edilen prototipik değerler, yanı ortalama durumlar Şekil 3’te gösterilmiştir. Görüldüğü üzere bu nesne sembollerin nesnelerin konumunu, derinliğini ve üstünde nesne olup olmayacağı bilgisini kodlamaktadır.

Bazi semboller sadece çubuklar için veya sadece küpler için aktive edilirken, bazıları hem küpler hem de çubuklar için aktive edilmiştir. Örneğin, ikinci sıradaki ve ilk sütundaki semboller hem küpler hem de çubuklar için aktive edilmiştir.

**Şekil 2:** Deney ortamı. Şekil 2a’da altı olası kavrama ve bırakma konumları mor rengiyle gösterilmiştir. Robotun ortam ile etkileşime geçmesi Şekil 2b’da gösterilmiştir.

**Şekil 3:** Her sembol için aktivasyon sayısına göre sıralanmış ortalama derinlik görüntüleri. Sembol ve aktivasyon sayısı solda rapor edilmiş olup, ortalama x-ekseni ve y-ekseni piksel konumları her alt şe kilin üstünde rapor edilmiştir. Renk çubuğu, normalleştiirilmiş derinlik değerini göstermektedir.
İlk durum ve bu nedenle konumunu değiştiremez. Bunu bir ‘kavrana-dolayı mantıklıdır; robot üzeri örtülü nesneyi kavrayamaz. Bu, üretilen etkinin her iki durumda da aynı olmasından çıkmaktadır. Örnekteki ortak özellik, nesnenin üstünde başka bir nesne olduğu atmosferden dolayıdır. Model bu iki nesneyi ayırt etmeden üretebilen etkisini (ortalamada 0 mm, 0 mm, 0 mm) tahmin edebilir. Sonuç olarak, sistem tahmin ve akıl yürütme gerçekleştirdiğinde, bu nesneler arasındaki etkileşim verisine dayanarak çıkarılan, ve buna bağlı olarak bu nesnelerle ve onların ilişkileriyle alakali semboller öğrenen bir model oluşturulmuştur. Bu örneklerdeki ortak özellik, nesnenin üstünde başka bir nesne olduğu atmosferden dolayıdır.

Bu örnek, keşfedilen sembollerin, ağaç arama yapısını elde etmek için oluşturulan ve uygulanan bir planı göstermektedir. Bu örnek, keşfedilen sembollerin, ağaç arama yapısını elde etmek için oluşturulan ve uygulanan bir planı göstermektedir.

C. Keşfedilen Sembollerle Planlama

Bu bölümde öğrenilen sembollerin planlananachsenin uygulanabileceği incelemiştir. Öğrenilen semboller ile planlama yapılabilmesi için mevcut sembolik durum ve eylem verildiğinde bir sonraki sembolik durumu tahmin eden bir model eğitilmiştir. Bu aşamada, iki yoğun katmanın ardından tek bir gözlem katmanı ve sonrasında bulunan iki yoğun katmanından oluşturulmuştur. Ağ, her boyut için iki kılıç çapraz entropi hatası ile eğitilmiştir.

Ilk mimarı (Şekil 1) sürekli, düşük seviye, yüksek seviye, düşük boyutlu girdiler ve çiktılarla sahipken bu model çağrıda sembolik, yüksek seviye, düşük boyutlu girdi ve çiktılarla çalışmaktadır. Bu tür bir model, eğitimden sonra, herhangi bir aşamada arama algoritması [1] kullanarak hedef bir sembolkar durumun aranmasına izin verir. Şekil 5, hedef olarak verilen bir bileşik yapıyı elde etmek için oluşturulmuş ve uygulanan bir plan görüştüktedir. Bu örnek, keşfedilen sembollerin, ağaç arama yoluya hedeflere ulaşmaya işlenmiş izin veren bir iki sembol tam olarak modellenin eğitim için kullanılabilme özelliğini göstermiştir. Bu örnek, bazı gerçekleş eylemler bulunmaktadır. Bunun sebebi yeni eğitilen modeldeki belirsizlikten dolayıdır.

IV. SONUÇ

Bu çalışmada değişik sayıyla nesneyi girdi olarak alabilmemiz, bu nesneler arasındaki ilişkileri etkileşim verisine dayanarak öğrenen ve bu bağlı olarak bu nesnelerle ve onların ilişkileriyle alakali semboller öğrenen bir yapı önermiştir. Öğrenilen semboller farklı nesne dizimlilerini için doğru etki tahmini vermektedir. Ayrıca bu semboller kullanılarak bir sonraki sembolik durumu tahmin eden bir model eğitilmiştir. Bu model sayesinde hedef bir sembolkar durumu üretmek eylem dizisi tahmin edilebilmiştir.

BİLGİLENDİRME

Bu çalışma TÜBİTAK ARDEB 1001 programı tarafından desteklenmiştir (Proje No: 120E274).

KAYNAKLAR

[1] S. J. Russell, Artificial intelligence a modern approach. Pearson Education, Inc., 2010.
[2] G. Konidaris, L. P. Kaelbling, and T. Lozano-Pérez, “Constructing symbolic representations for high-level planning,” in 28th AAAI Conf. on AI, 2014.
[3] E. Ugur and J. Piater, “Bottom-up learning of object categories, action effects and logical rules: From continuous manipulative exploration to symbolic planning,” in 2015 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2015, pp. 2627–2633.
[4] M. Asai and A. Fukunaga, “Classical planning in deep latent space: Bridging the subsymbolic-symbolic boundary,” arXiv preprint arXiv:1705.00154, 2017.
[5] T. Taniguchi, E. Ugur, M. Hoffmann, L. Jamone, T. Nagai, B. Rosman, T. Matsuka, N. Iwahashi, E. Ozbop, J. Piater, et al., “Symbol emergence in cognitive developmental systems: a survey,” IEEE Transactions on Cognitive and Developmental Systems, 2019.
[6] A. Ahmedoglu, M. Y. Seker, J. Piater, E. Ozbop, and E. Ugur, “Deepsym: Deep symbol generation and rule learning for planning from unsupervised robot interaction,” Journal of Artificial Intelligence Research, vol. 75, pp. 709–745, 2022.
[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” Advances in neural information processing systems, vol. 30, 2017.
[8] F. Locatello, D. Weissborn, T. Unterthiner, A. Mahendran, G. Higold, J. Uszkoreit, A. Dosovitskiy, and T. Kipf, “Object-centric learning with slot attention,” Advances in Neural Information Processing Systems, vol. 33, pp. 11 525–11 538, 2020.
[9] G. F. Elsayed, A. Mahendran, S. van Steenkiste, K. Greff, M. C. Mozer, and T. Kipf, “Savi++: Towards end-to-end object-centric learning from real-world videos,” arXiv preprint arXiv:2206.07764, 2022.
[10] C. J. Maddison, A. Mnih, and Y. W. Teh, “The concrete distribution: A continuous relaxation of discrete random variables,” arXiv preprint arXiv:1611.00712, 2016.
[11] E. Jang, S. Gu, and B. Poole, “Categorical reparameterization with gumbel-softmax,” arXiv preprint arXiv:1611.01442, 2016.
[12] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, “Automatic differentiation in PyTorch,” 2017.
[13] S. Ioffe and C. Szegedy, “Batch normalization: Accelerating deep network training by reducing internal covariate shift,” arXiv preprint arXiv:1502.03167, 2015.