

# Grafik Sinir Ağlarına Genel Bir Bakış

## An Overview of Graph Neural Networks

 Hamza Talha Gümüş<sup>1</sup>, Can Eyüpoğlu<sup>2</sup> 

<sup>1</sup>Bilgisayar Mühendisliği Ana Bilim Dalı, Atatürk Stratejik Araştırmalar ve Lisansüstü Eğitim Enstitüsü,  
Milli Savunma Üniversitesi, İstanbul, Türkiye  
hamzatalhagumus@gmail.com

<sup>2</sup>Bilgisayar Mühendisliği Bölümü, Hava Harp Okulu, Milli Savunma Üniversitesi, İstanbul, Türkiye  
ceyupoglu@hho.msu.edu.tr

### Öz

Grafik sinir ağları (GNN), yapay sinir ağı (ANN) ailesine mensup ve grafikler üzerinden bilgi çıkarımı işlemi gerçekleştiren bir derin öğrenme yöntemidir. İlk kullanımı 2008 yılında gerçekleşmiş, gelişimi ise 2014 yılı ve sonrasında olmuştur. Evrişimli sinir ağlarının (CNN) görseller üzerinde beklenen performansı karşılamamasına karşılık olarak geliştirilen GNN'ler; fizik, kimya, biyoloji, siber güvenlik gibi birçok alanda kullanılmaktadır. Bu çalışmada GNN'ler ve modelleri temel olarak anlatılmış ve kapsamlı bir literatür taraması gerçekleştirilmiştir. Çalışma içerisinde bir GNN modelinin tasarım adımlarına değinilerek geliştirilen modeller incelenmiş, GNN modellerinin CNN ve yinelemeli sinir ağları (RNN) etkili modellere karşı güçlü ve zayıf yönleri gösterilmiştir.

**Anahtar Kelimeler:** Grafik Sinir Ağları, Yapay Sinir Ağları, Derin Öğrenme, Evrişimli Sinir Ağları, Yinelemeli Sinir Ağları

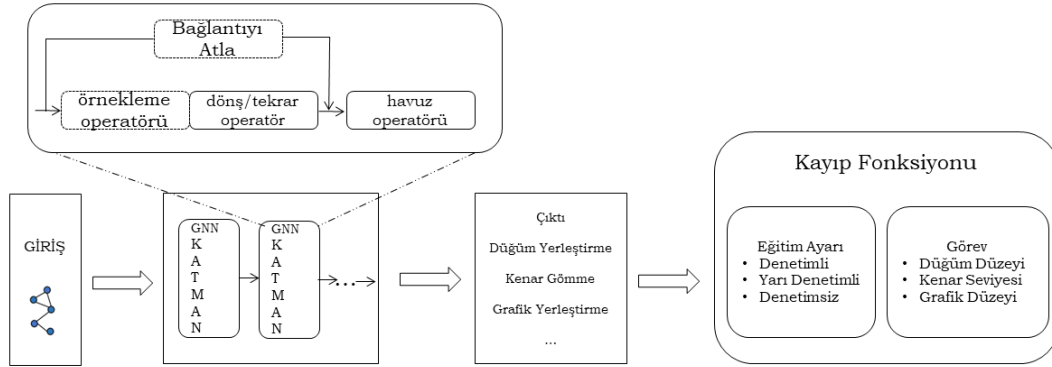
### Abstract

Graph neural networks (GNN) are a deep learning method that belongs to the artificial neural network (ANN) family and performs information extraction from graphs. It was first used in 2008 and its development started in 2014 and onwards. GNNs, which were developed in response to the failure of convolutional neural networks (CNN) to meet the expected performance on visuals, are used in many fields such as physics, chemistry, biology and cyber security. In this study, GNNs and their models are basically explained and a comprehensive literature review is carried out. In the study, the developed models are examined by referring to the design steps of a GNN model, and the strengths and weaknesses of GNN models against CNN and recurrent neural networks (RNN) effective models are shown.

**Keywords:** Graph Neural Networks, Artificial Neural Networks, Deep Learning, Convolutional Neural Networks, Recurrent Neural Networks

### 1. Giriş

Grafik Sinir Ağları (Graph Neural Networks-GNN), grafik üzerinde çalışan sinir ağı ailesine mensup bir kavramdır. Derin öğrenmenin alt dalı olan Evrişimli Sinir Ağlarının (Convolutional Neural Networks-CNN) yetersiz kaldığı Öklid dışı uzayda kullanılmak için geliştirilmiş ve büyük başarılar yakalamıştır. Evrişimli sinir ağlarına bakıldığında ise görsel bilgi analizi üzerinde başarılı olmuş ve yaygın olarak kullanılmaktadır. Genel kullanım alanları resim tanıma, resim sınıflandırma ve tıpta görüntü analizi olsa da doğal dil işlemede dâhil olmak üzere birçok alanda kullanılmaktadır. Tarihsel gelişimi ilk olarak 1962 yılında Hubel ve Wiesel [1] tarafından kullanılmıştır. Bu çalışma sonrasında sinir ağlarına yeni bir bakış gelerek gelişimi hızla devam etmiştir. CNN'ler 1D ve 2D'de büyük başarılar sağlamış ve kendini kanıtlamıştır. Tarihsel süreçte geliştirilen sinir ağlarına fikir oluşturmuştur. GNN ise günümüzde çok yeni bir çalışma alanı olup her geçen gün gelişmektedir. Tarihsel gelişimini incelediğimizde Yinelemeli Sinir Ağlarının (Recurrent Neural Networks-RNN) grafikler üzerinde kullanılması ile başlamıştır. 1997 yılında Sperduti ve Starita [2] tarafından ortaya atılan bu düşünce 2009 yılında Scarselli [3] tarafından İleri Beslemeli Sinir Ağları (Feed Forward Neural Networks-FFNN) ile kullanılarak GNN'ler için bir algoritma geliştirilmiştir. İlk GNN modeli grafik etki alanlarını öğrenmek için 2005 yılında Gori ve ark. [4] tarafından literatüre dahil edilmiştir. Geleneksel yöntemlerin yetersiz olması ve veri kaybına sebep olması ile sonuçta ön adımlarda takılı kalılabileceği sorunu çözmek için çalışma gerçekleştirilmiştir. GNN modelinin önerilmesi ve farklı grafik türleri üzerinde başarı sağlandığı belirtilerek, model üzerinde deneyler yapılmış ve bilim dünyasına sunulmuştur [4]. Gori ve ark. bu modeli geçmişte yaptıkları çalışmaların üzerine ekleyerek bulmuştur. Daha önceki yıllardan örnek vermek gerekirse yönlendirilmiş grafikler için RNN kullanarak 2003 yılında bir sinir ağı modeli önerilmiştir [5]. GNN ile ilgili kapsamlı bir çok araştırma ve karşılaştırma [6-10] yapılmıştır. GNN'ler ilk olarak 2005 yılında ortaya atılsa da gelişimi 2014 yılı ve sonrasında olmuştur. Temel olarak bir GNN modeli Şekil 1'de [7] görülmektedir.



Şekil 1: Grafik sinir ağı modeli temel tasarım hattı

## 2. Grafik Sinir Ağları

GNN model tasarımı temel olarak dört işlem adımından oluşmaktadır. İlk adım grafiğin yapısını bulmak, ikinci adım grafiğin tipi ve ölçeğini belirlemek, üçüncü adım tasarım kayıp fonksiyonunu hesaplamak ve son adımda hesaplama modülleri kullanılarak tasarımı tamamlamak şeklindedir [7]. İlk adım olan grafiğin yapısı bulunurken iki farklı durum gözlemlenmektedir. Birinci durum grafikler üzerinde bulunan bilgilerin açık olduğu yapısal senaryolardır. İkinci durum ise bilgilerin açık olmadığı yapısal olmayan senaryolar olarak adlandırılmaktadır. Bu işlem gerçekleştirildikten sonra grafik türü belirlenmektedir. Grafikler türlerine göre yönlü/yönsüz, homojen/heterojen, statik/dinamik gibi farklı türleri bulunmaktadır. Aynı şekilde bu grafiklerin birleşiminden de bir grafik oluşturulabilmektedir. Grafikler karmaşık yapılar olduğu için bu adım oldukça önemlidir [7]. Şekil 2'de [7] grafik türü ve ölçeğine göre değişkenler şeması görülmektedir.

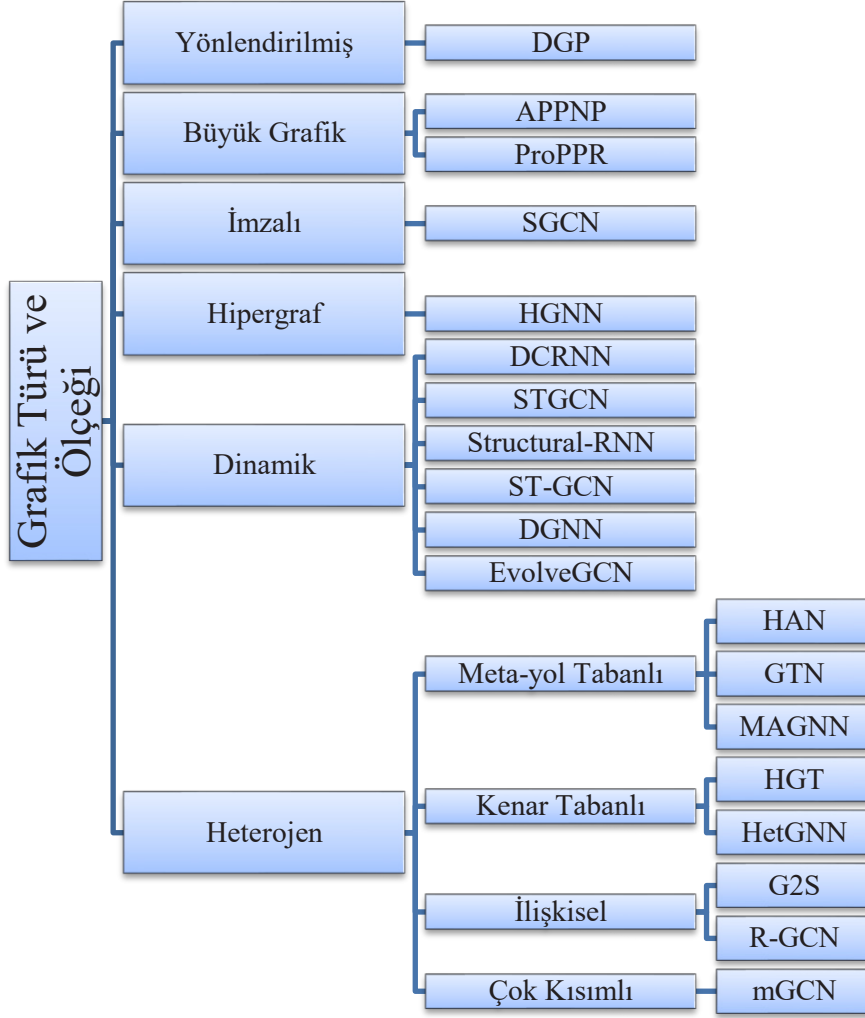
### 2.1. Grafik Türü ve Ölçeği

Grafiklerde kenarlar bilgiler içermektedir. Yönlendirilmiş grafiklerde bu kenarlar kullanılarak modelleme yapılmaktadır. Bu tür durumlar için Yoğun Grafik Yayılımı (Dense Graph Propagation-DGP) geliştirilmiştir [11]. Heterojen grafikler ise farklı tiplerden oluşmuş düğüm ve kenarlardan meydana gelmektedir. İlk tür olan meta yol tabanlarında, tüm konularda bulunan düğümlerin türleri belirlenmektedir. Kenar tabanlı yöntemler ise genellikle örnekleme, toplama ve özellik kodlama için kullanılmaktadır. İlk olarak web ölçeği kapsamında Heterojen Grafik Dönüştürücü (Heterogeneous Graph Transformer-HGT) [12] modeli önerilmiştir. Bir diğer model olan Heterojen Grafik Sinir Ağı (Heterogeneous Graph Neural Network-HetGNN) [13] ise grafik madenciliği ve grafik model görevleri olan, bağlantı tahmini ve düğüm sınıflandırmayı kolaylaştırmak için geliştirilmiş bir modeldir. İlişkisel grafikler çok fazla bilgi içeren grafikler olarak tanımlanabilmektedir. Bilgi grafiklerinden sağlanan faydayı arttırmak için İlişkisel Evrişimli Sinir Ağları (Relational Graph Convolutional Network-R-GCN) [14] modeli önerilmiştir. Bir diğer model olan G2S doğal dil işlemede genellikle Geçitli Grafik Sinir Ağları (Gated Graph Sequence Neural Networks-GGNN) [15], üzerinde bulunan tüm bilgileri kodlayan bir modeldir. Grafikten dizi oluşturmada ortaya çıkan bir problem olan dilbilgisi ve veri kaybına karşı geliştirilmiştir. Çok katmanlı

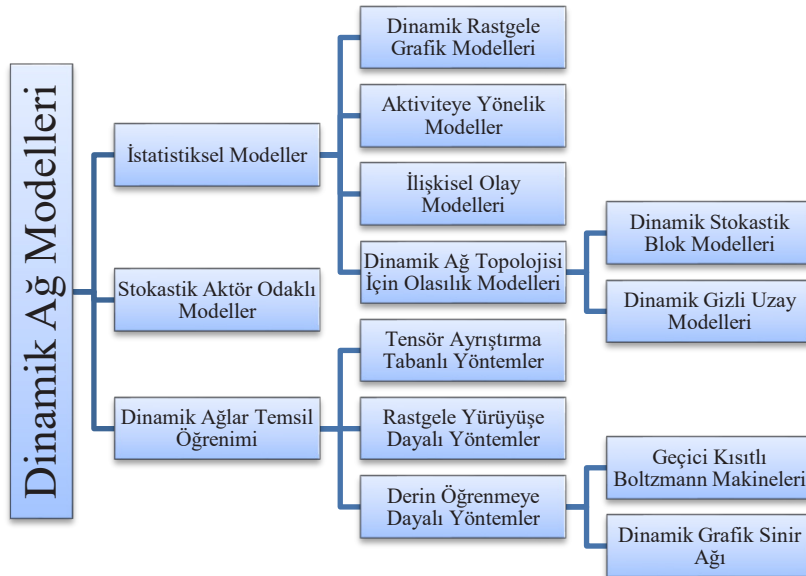
grafikler, düğümlerin birden fazla kenarla ilişkili olması durumunda ortaya çıkmaktadır. Bu tür grafikler çok boyutlu olduğu için tek boyutlu grafiklere önerilen modeller yetersiz kalmıştır. Bu sebeple çok boyutlu grafik sinir ağı olarak bilinen mGCN [16] modeli önerilmiştir. Dinamik grafikler ise zamana bağlı olarak değişen grafiklerdir. İlk örnek modelimiz olan Difüzyon Evrişimli Tekrarlayan Sinir Ağı (Diffusion Convolutional Recurrent Neural Network-DCRNN) [17], trafik modelleri gibi karmaşık, zamana göre değişen ve uzun süreli tahmine dayalı etkiler göz önüne alınarak geliştirilmiş bir modeldir. Mekansal-Zamansal Grafik Evrişimli Ağlar (Spatiotemporal Graph Neural Network-STGNN) [18], trafik analizini zamansal ve mekansal olarak ele almış, tahmin etme sorununa farklı bir bakış açısıyla yaklaşarak yeni bir model önermiştir. Yapısal-RNN [19], gelişmiş uzay zaman grafiklerinde RNN yardımıyla gerçek dünyada işe yarayan bir form oluşturma amacıyla geliştirilmiştir. Bilinen çalışma ise insan hareketlerinin modellenmesidir. Aynı şekilde benzer bir durum Mekansal-Zamansal Grafik Evrişim Ağları'nda da (Spatial Temporal Graph Convolutional Networks-ST-GCN) [20] bulunmaktadır. İnsan eylemlerini tanımlamak amacıyla iskelet modellemesi gerçekleştirilmek istenmiş ve dinamik iskelet modeli önerilmiştir. Dinamik Grafik Evrişimli Ağlar (Dynamic Graph Neural Networks-DGNN) [21], Uzun-Kısa Süreli Bellek Ağları (Long Short-Term Memory-LSTM) ve Grafik Evrişim Ağlarını (Graph Convolutional Networks-GCN) içeren zamansal değişime göre en fazla fayda sağlamaya çalışan yeni bir model olarak önerilmiştir. Zaman yapısını kodlayan LSTM ile grafik yapısını kodlayan grafik evrişim katmanları kullanılırken Birleştirilmiş Dinamik Grafik Evrişim Ağı (Concatenate Dynamic Graph Convolutional Networks-CD-GCN) ve köşe kod çözümü yapısına sahip Şelale Dinamik Grafik Evrişim Ağı (Waterfall Dynamic Graph Convolutional Networks-WD-GCN) adında iki farklı model tekniği de sunulmuştur. Yakın geçmişte Evrim Geçiren Grafik Evrişim Ağları (Evolving Graph Convolutional Networks-EvolveGCN) [22] modeli, zamansal grafiklerde düğüm farklılığından doğan sorunlara çözüm olması sebebiyle geliştirilmiştir. RNN'lerin düğüm bilgisi eksikliği ve düğüm kümesi değişiminde karşılaştığı zorluğa çözüm amacıyla çalışmaya başlanmış ve yapılan deneylerde daha iyi bir performans sergilediği görülmüştür. Dinamik ağlar, GNN'de yeni kullanılmaya başlanmıştır. Dinamik Sinir Ağı (Dynamic Neural Networks-DNN) modelleri ile ilgili kapsamlı bir araştırma [23] mevcuttur. DNN modelleri üç

farklı tabanda incelenmiştir. Şekil 3'te DNN'lerde grafik sinir

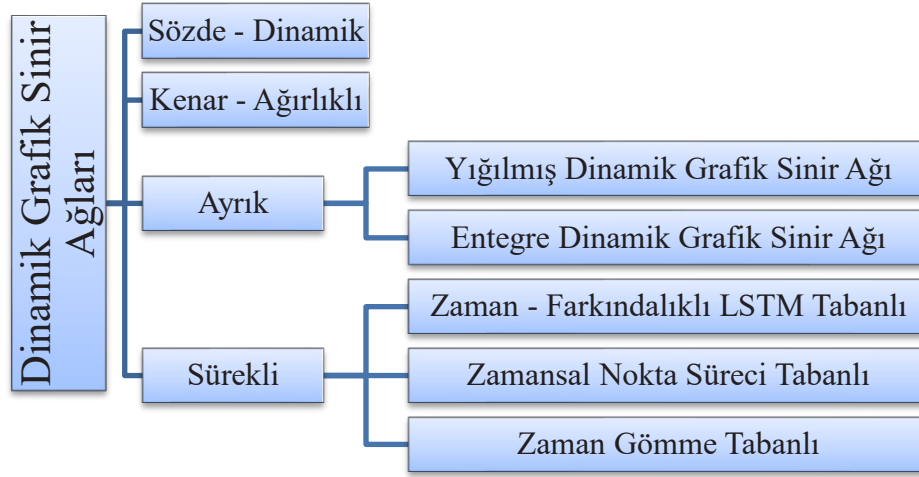
model genellemesi görülmektedir.



Şekil 2: Grafik türü ve ölçeğine göre değişkenler



Şekil 3: Dinamik sinir ağlarında grafik sinir model genellemesi



Şekil 4: Dinamik sinir ağ türleri

Şekil 3'ten yola çıkarak istatistiksel modellerin çıkarımlara ve istatistiksel verilere dayalı olduğu görülmektedir. Rastgele modeller, Rastgele Grafik Modeller (Random Graph Models-RGM) ve Üstel Rastgele Grafik Modelleri (Exponential Random Graph Models-ERGM) üzerinde çalışmaktadır. Aktiviteye yönelik modeller, düğümlerin davranışlarını modellemektedir. İlişkisel olay modelleri ise zamansal modeller olarak tanımlanabilmektedir. Gelecekte gerçekleşme olasılığını hesaplamaktadır ve bunu düğümler arasındaki ilişkilerle yapmaktadır. Üretici olasılık modeller olarak bilinen dinamik modellerden gizli uzay modeli, Markov Zinciri Monte Carlo (Markov Chain Monte Carlo-MCMC) yöntemini kullanarak düğümleri ölçeklemektedir. Stokastik blok modeli ise gizli uzay modelinden farklı olarak daha fazla düğümü ölçekleyebilmektedir. Stokastik Aktör Odaklı Modeller (Stochastic Actor Oriented Models-SAOM), model içerisinde düğümlerin konum ve davranışlarını temsil etmektedir. Dinamik ağlar, temsil öğrenme gizli grafiklerde kullanılmaktadır. Tensör ayrıştırma tabanlı yöntemler, zaman özneline matris genelleştirmeleridir. Rastgele yürüyüşe dayalı yöntemler gömme görevlerine yardımcı olmaktadır. Derin öğrenmeye dayalı modeller ise adından da anlaşılacağı gibi derin öğrenme yaklaşımları içermektedir. Boltzman makineleri olasılıksal çıkarımlarda üretkenlikte kullanılmaktadır. DNN'lere baktığımızda ise GNN'lerde olduğu gibi grafiği kodlayan fakat isminden de anlaşılacağı gibi dinamik grafikleri kodlayan ağlardır. Şekil 4'te DNN türleri görülmektedir [23].

Sözde dinamik modeller ağın birden fazla anlık görüntüsünü modellemektedir. Ayrıca bağlantı eklemede de kullanılmaktadır [24]. Üretken Grafik Evrişim Ağı (Generative Graph Convolutional Networks-G-GCN) [25], büyüyen grafiklerde süreç modellenirken soğuk başlatmanın oluşturduğu sorunlara karşılık bu modelden tamamlanmıştır. Model üst seviye performans göstermiştir. Düğümler için bağlantı tahmini yapabilmektedir. Aynı durum Varyasyonel Grafik Otomatik Kodlayıcı (Variational Graph Auto-Encoders-VGAE) [26] için de geçerlidir. VGAE denetimsiz öğrenmeyle eğitilmektedir. Yönlendirilmiş grafiklerde başarılı ve gizli bilgileri öğrenme yeteneğine sahiptir. Aynı durum yarı denetimli öğrenme kullanılarak da denenmiş ve başarılı olduğu belirtilmiştir [27]. Kenar ağırlıklı modeller, dinamik ağ modelinin basitleştirilmiş halidir. Dinamik ağ

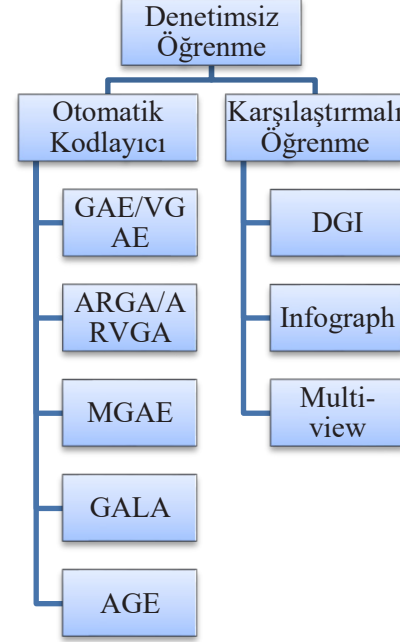
basitleştirilerek kenar ağırlıklı bir ağa dönüştürülür ve statik GNN modelleri ile kullanılır. En bilinen model ise Zamansal Bağımlı Grafik Sinir Ağı (Tree Decomposed Graph Neural Network-TDGNN) [28] adıyla bilinmektedir. Bu yöntem ile hedef düğüme komşu olan düğüm bilgileri ve özellikleri, zamansal toplayıcı (Temporal Aggregator-TDAgg) ile toplanmaktadır. Model bağlantı tahmini için kullanılmaktadır. Ayrık modelleme kullanılarak statik grafiklerde anlık görüntü alma başarılıdır. Statik dinamik modellerde Birleştirilmiş Dinamik Grafik Evrişim Ağı (Concatenate Dynamic Graph Convolutional Networks-CD-GCN) ve Şelale Dinamik Grafik Evrişim Ağı (Waterfall Dynamic Graph Convolutional Networks-WD-GCN) kullanılmaktadır. Entegre Dinamik Sinir Ağları ise birçok dinamik sinir ağı modelinde olduğu gibi GNN ve RNN'leri birleştirerek modellemeyi, bu birleşim ile kodlayan modellerdir [23]. Konvolüsyonlu Uzun Kısa Süreli Bellek (Convolutional Long Short-Term Memory-ConvLSTM) [29] modelinden ilham alınarak geliştirilmiştir. Bu model tahmin için Tekrarlayan Sinir Ağlarından olup girdiden duruma ve duruma geçiş için evrişimsel yapıyı bulunmaktadır. Tahmin için kullanılan tahminini geçmiş durumlardan çıkarmaktadır. Sonraki yıllarda bu modeller genişletilerek yeni modeller geliştirilmiştir. Yağış tahminleri ve uçuş güvenliği için Evrişimsel LSTM, değişken konumlarda yapı öğrenmesi için Yörünge GRU (Trajectory GRU-TrajGRU) [30], geçmişten çıkarım yaparak geleceği tahmin eden Üreten Tekrarlayan Sinir Ağı (Recurrent Neural Networks for Predictive Learning-PredRNN) [31], Gradient Otoyol Birimi önerisiyle geliştirilen tekrarlayan ağ ve PredRNN'in gelişmiş versiyonu olan ve zaman ikilemini çözmek için önerilen PredRNN++ [32], video tahmini için etkin bir model [33] ve biyomedikal sektörde görüntü segmentasyonu için U-Net [34] modeli geliştirilmiştir [35]. Evrişimsel LSTM'den esinlenerek geliştirilen bir diğer model ise girdi olarak düğümlerde ConvLSTM'nin kullandığı 3B tensörlerin aksine özellik vektörü kullanan GCRN-M2 [36] modelidir. ImageNet [37], Moment in Time [38] ve Places [39] veri setleri entegre ağlarda çoğunlukla kullanılmaktadır. Dinamik otomatik kodlayıcılar ve üretken modellerde ayrık bölümde işlenmektedir [23]. Burada en bilinen model Dinamik Grafik Gömme Modelidir (Deep Embedding Method for Dynamic Graphs-DynGEM) [40]. Dinamik grafiklerde hesaplamayı iyileştirmek için geliştirilen model

anlık görüntüleri kullanmaktadır. Sürekli ağlar ise üç farklı yaklaşım üzerinde incelenmektedir [23]. İlk yaklaşım RNN tabanlı modellerdir. Yenilemeden kaynaklı olarak ağ sürekli güncellenmektedir. Bu tabanda iki farklı model geliştirilmiştir. Bunlarda ilki özel olarak tasarlanmış ve yönlendirilmiş grafikler üzerinde çalışan Akış Grafik Sinir Ağı (Streaming Graph Neural Networks-SGNN) [41], diğeri ise gelecekteki yerini modelleyebilen ve gömme yörüngeleri üzerinde öğrenim sağlayan JODIE [42] modelidir. Zamansal nokta süreci tabanında en bilinen yöntem know-evolve [43] yöntemi olmuştur. Doğrusal olmayan bir biçimde gelişen varlık temsilini öğrenebilmektedir. Etkileşim ağlarına dönüşüm içinde Geçici Nokta Süreci (Temporary Point Process-TTP) parametreleri yardımı ile gerçekleştirilmektedir. Yapılan deneyler sonucu başarı göstermiştir. Geliştirilen bir başka model olan DyRep [43], düşük boyutlu grafiklerde düğüm yerleştirme işlemi için kullanılmaktadır. Bilgi grafiklerinde başarılı olmuş, yapısal ve düğümler arası iletişimi ifade edebilmesi yeteneği ile modelleme gerçekleştirebilmektedir [23]. Son taban modeli olan zaman gömmeli modeller, zaman temsilinde konumsal kodlama kullanmaktadır [44]. Zamansal Grafik Dikkat (Temporal Graph Attention-TGAT) [45] katmanı ile zaman özellik öğrenmesi gerçekleştirirken aynı zamanda komşuluk özelliklerini toplamasında sağladığı verim ile öne çıkmaktadır. Diğer grafiklere baktığımızda ise hipergraflar ve imza grafikleri görülmektedir. Hipergraflar üst düzey veri kümelerini kodlayabilen modellerdir. Bilinen model olan Hipergraf Sinir Ağları (Hypergraph Neural Networks-HGNN) [46], karmaşık verilerin modellenmesi için geliştirilmiştir. Yapılan deneyler sonucu model iyi bir performans göstermiştir. İmzalı grafikler ise negatif ya da pozitif gibi farklı ve zıt olabilen işaretli grafiklere denilmektedir [7]. Bu modeller sadece pozitif grafiklere odaklanıldığı için gelişim sürecinde ortaya çıkan sorunlar nedeniyle doğmuştur. Model ayrıca düğüm temsilini belirlemek için geliştirilmiş ve İmzalı Grafik Evrişimli Ağ (Sparse Graph Convolution Network-SGCN) [47] olarak adlandırılmıştır.

## 2.2. Tasarım Kayıp Fonksiyonu

Grafik türü ve ölçeği belirlendiğinde ise sırada tasarım fonksiyonunu anlamak ve oluşturmak vardır. Tasarım kaybı fonksiyonu Şekil 1'de de görüldüğü gibi eğitim ayarı ve görev kısımlarından oluşmaktadır. Görevler; düğüm düzeyi, kenar seviyesi ve grafik düzeyi olarak ayrılabilir. Düğüm düzeyi görevleri, düğümlere odaklanmaktadır. Düğüm ile ilgili en bilinen görevler ise düğüm sınıflandırması ve düğüm kümesi görevleridir. Kenar seviyesi, kenarlarla ilgilenmekte ve kenar türünün sınıflandırılmasına odaklanmaktadır. Bağlantı tahmini ve kenar sınıflandırma, örnek görev olarak verilebilmektedir. Grafik düzeyi ise grafik temsil amaçlı öğrenmeye odaklanmaktadır. Grafik sınıflandırma bu görevlerde en bilinen örnektir [7]. Eğitim ayarına bakıldığında ise derin öğrenmede olduğu gibi denetimli, yarı denetimli ve denetimsiz türler görülmektedir. Denetimli eğitim, etiketlenmiş veriler sayesinde gerçekleşmektedir. Yarı denetimli eğitim ise az sayıda etiketlenmiş düğüm ve belirsiz sayıda etiketlenmemiş düğüm ile gerçekleşmektedir. Bu iki eğitim ayarı ile tasarım kolayca gerçekleşmektedir. Denetimsiz eğitimler ise sadece etiketlenmemiş veriler ile gerçekleştirilmekte ve grafik özellikleri bu ayarda grafik tarafından sağlanan bilgiler ile oluştuğu için oldukça zordur.

Bu sebeple otomatik kodlayıcılar ve karşılaştırmalı öğrenme yöntemleri ile yapılmaktadır. Şekil 5'te denetimsiz ayar modelleri görülmektedir.



Şekil 5: Denetimsiz ayar modelleri

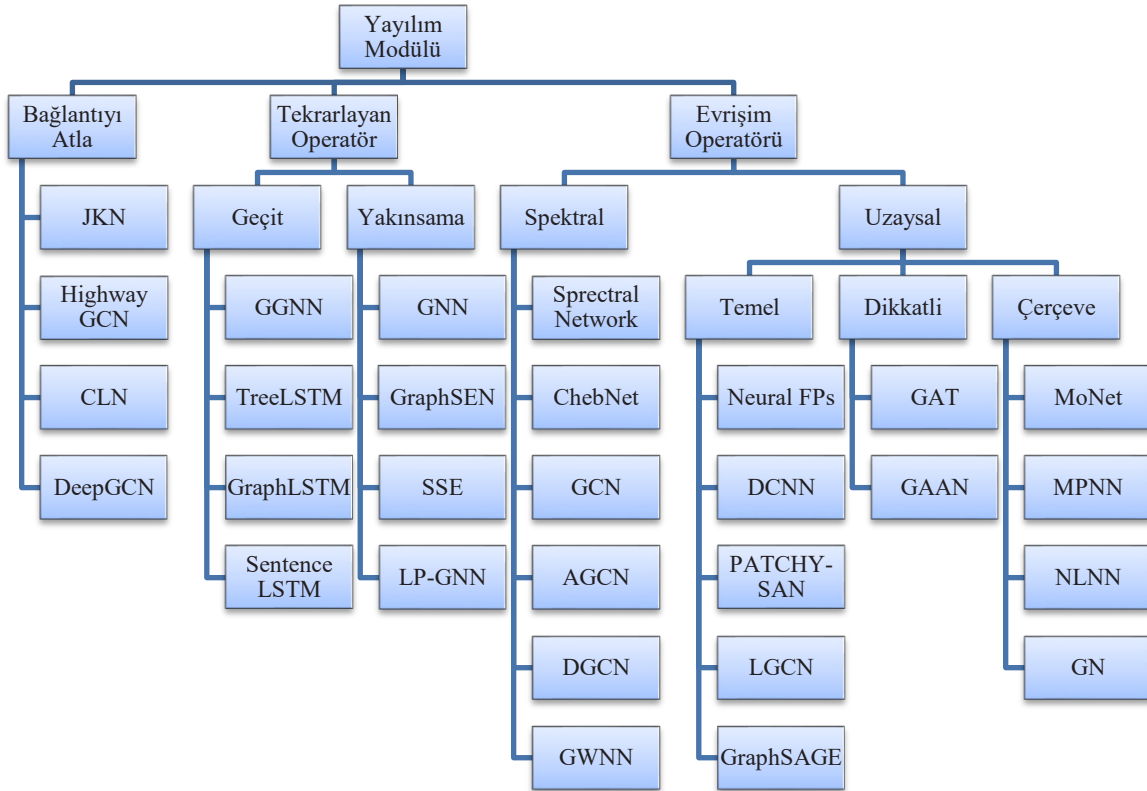
Şekil 5'te [7] görüleceği üzere denetimsiz öğrenme için farklı modeller geliştirilmiştir. Öncelikle otomatik kodlayıcıları incelendiğinde beş farklı model görülmektedir. İlk olarak Değişken Otomatik Kodlayıcı (Graph Autoencoders-GAE) yapısına sahip grafikler üzerinde çalışabilen Varvasyonel Grafik Otomatik Kodlayıcı (Variational Graph Autoencoders-VGAE), ilk geliştirilen modellerden birisi olmuştur. Bu model için grafik evrişimli ağlar kullanılmaktadır [26]. Grafik gömme görevi, düşük boyutlarda etkin bir yöntem olmuştur fakat eşitlemelerde geçmiş dağıtımda zorlanmaktadır. Bu soruna çözüm oluşturmak için çekişmeli olarak Düzenlenmiş Grafik Otomatik Kodlayıcı (Adversarially Regularized Graph Autoencoder-ARGA) [48] ve düşmanca Düzenlenmiş Değişken Grafik Otomatik Kodlayıcı (Adversarially Regularized Variational Graph Autoencoder-ARVGA) [48] modelleri geliştirilmiştir. Geliştirilen modeller grafik sınıflandırma ve bağlantı tahmini görevlerinde iyi bir performans elde etmiştir. Geleneksel yöntemler modernleşirken, ağlarda bulunan bir başka soruna da çözüm bulmuştur. Bu sorun grafik ağlarını keşfetme temelli, grafik kümelemedir. Marjinalleştirilmiş Grafik Otomatik Kodlayıcı (Marginalized Graph Autoencoder-MGAE) [49] modeli, temsili öğrenmede sadece denetimsiz olarak çalışmamaktadır ve bozuk özellikler marjinalleştirilerek işlem yapılmaktadır. Aynı zamanda en önemli özelliği de otomatik kodlayıcıyı grafik alanına iletmesi olarak belirtilmiştir. Deneyler sonucu gösterdiği performans başarılı sayılmış düğüm temsili görevinde kullanılmıştır. Bir başka model olan simetrik otomatik kodlayıcı GALA [50], Laplacian yardımıyla işaretli grafikler üzerinde kararlı bir model olmuştur. Bağlantı tahmini ve kümelemede iyi bir performans elde etmiş olan GALA, görüntü kümeleme görevlerinde performansını arttırmıştır. Bir başka model olan Uyarlamalı

Grafik Kodlayıcıyı (Adaptive Graph Encoder-AGE) [51] ise GCN'lerin dezavantajlarından yola çıkarak geliştirilmiş bir modeldir. Modelin sağladığı faydalar performans ve sağlamlık amacı, gerçek dünyada tutarlı olmayan matris kurtarımına önem vermesi olmuştur. İki modülden oluşan AGE ilk modülünde Laplacian yumuşatması kullanırken ikinci modülünde ise özellikleri güçlendiren bir kodlayıcı kullanmaktadır. Deneylerden elde edilen sonuçlara göre model performansı son derece başarılıdır. İkinci taban olan karşılaştırmalı öğrenme içinde farklı modeller geliştirilmiştir. Derin Grafik Infomax (Deep Graph Infomax-DGI) [52], iyi temsiller öğrenmek için görüntü temsili model önermektedir. Görsel temeli bulunan çok modellenli bir öğrenme yaklaşımı sunarken, sınırlı verilerle ilgilenmektedir. Düşük boyutlu veriler üzerinde bilgiyi arttırmaktadır. InfoGraph [53], karşılık bilgiyi maksimum seviyeye getirmeye çalışan bir modeldir, asıl amacı ise grafik temsil öğrenmedir. Deneyler

sonucu etiketlenmemiş düğümler üzerinde etkili olduğu görülmüştür. Farklı ölçeklerde olan grafiklerde düğüm, kenar ve üçgen özneline başarılı olmuştur. Son model olan çoklu öğrenme ise [54], kendi kendini denetleyen ve grafik difüzyonları ile elde edilmiş bir modeldir. Deneyler sonucunda %50 başarı gösterilmiş, görüntü sayısı artışı ve çoklu katmanda performansın iyileşmediği belirtilmiştir.

### 2.3. Hesaplama Modülleri

GNN'lerin temel yapısının incelendiği çalışmamızda son incelenecek olan kısım hesaplamalı modüllerin bulunduğu kısımdır. Bu kısımda yayılım, örnekleme ve havuzlama modülü incelenecektir. Yayılım modülü, düğümler arası bilgi aktarımı için, örnekleme modülü, büyük grafiklerde yayılım yapmak amacıyla, havuzlama modülü ise düğümlerden bilgi çıkarma amacıyla kullanılmaktadır [7]. Şekil 6'da [7] yayılım modülü görülmektedir.



Şekil 6: Yayılım modülü

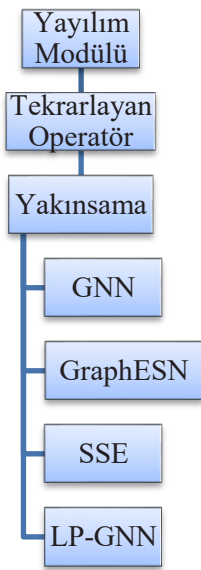
#### 2.3.1. Yayılım Modülü

Yayılım modülü üç farklı grup içerisinde incelenmektedir. İlk grup bağlantıyı atla operatörleridir. Bağlantıyı atla operatörleri derin GNN'ler elde etme amacıyla modellenmektedir. İlk incelediğimiz model olan Sıçrayan Bilgi Ağları (Jumping Knowledge Networks-JKN) [55], komşu toplama prosedürüyle özellik analizi yapmaktadır. Komşu düğüm aralığında olan mesafeyi rastgele yürüyüş benzetimli komşu aralıklı öğrenme ile amaçlayan bir modeldir. Sosyal ve biyoinformatik ağlarda başarılı olmuştur. JKN'ler evrişimli ağlar ve dikkat ağlarından daha iyi yararlanmak içinde kullanılmaktadır. Otoyal ağları, LSTM tabanında derine inebilmek için

geliştirilmiş bir modeldir [56]. Dil modelleme deneylerinde başarı göstermiştir. Ayrıca geliştirilerek evrişimli ağlarda coğrafi konum belirlemede de kullanılması için çalışmalar yapılmıştır [57]. Yapılan çalışma performans açısından başarılı olmuştur. Sütun Ağı (Column Network-CLN) [58], topluluk sınıflandırma amacıyla kullanılmaktadır. Toplu sınıflandırma genel olarak zor bir görev olduğu için geliştirilen Sütun Ağı, çok sayıda ilişkili alanlar özneline toplu sınıflandırmayı gerçekleştirmektedir. Model dünya tabanlı uygulamalar olan yazılımsal gecikme tahmini, medikal diyabet sınıflandırması ve film-tür sınıflandırmasında başarılı olmuştur. En önemli özelliği ise öğrenme ve çıkarımı ağ üzerindeki ilişkilerin sayısıyla doğrusaldır. Derin Grafik Evrişimli Ağlar (Deep Graph

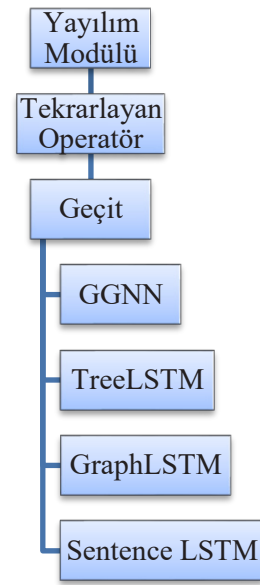
Convolutional Networks-DeepGCN) [59], model tabanlı optimizasyon yöntemleri ve ayırt edici öğrenme yöntemlerinin sorunlarına el atmaktadır. DeepGCN'lerin amacı ise gürültü gidericiler ve değişken bölme yöntemleri yardımıyla hızlı ve etkili bir CNN elde etmektir. Deneyler sonucu Gauss gürültü giderici ve düşük seviyeli görüş uygulamalarında başarı yakalamıştır. DeepGCN'ler farklı modeller olan ResGCN, ResNet ve DenseGCN'den ilham almıştır. ResNet [60], derin ağların eğitimi için kullanılmaktadır. Farklı veri kümelerinde yapılan deneyler sonucunda başarı elde etmiştir. Ayrıca ResNet [61], kimlik eşlemeleri için de kullanılmaktadır. Bu kullanımda ileri ve geri sinyalleri bloklar arası atlama ile gerçekleştirmektedir. Bu durum eğitimi kolaylaştırmak ve genelleme yapımını geliştirmek için kullanılmaktadır. Yoğun Bağlantılı Evrimsel Ağlar olarak bilinen DenseNet [62], giriş ve çıkış arasında bağlantı kısılalığından yararlanarak derin ve verimli bir eğitim içeren modellemedir. Her katman birbirinden beslenmekte ve bu sayede doğruluk yüksek tutulmaktadır. Bu ağlarda özellikler tekrar kullanabilmekte ve parametre sayısının azalmasıyla özellik yayılımını kuvvetlendirmektedir.

Artık Grafik Evrişim Ağı olarak tanımlanan ResGCN [63] ise anormal düğüm tespiti ile izinsiz giriş tespiti yapabilmektedir. Tespit işleminde karşılaşılan durumlar olan artık modelleme ve ağ yumuşatma, ResGCN ile çözülebilmektedir. Model sayesinde anormal düğümlerin oluşturduğu etkiler azaltılmaktadır. Derin ağlar sadece grafik ağlarında değil CNN'ler üzerinde de bulunmaktadır. GCN ve CNN'ler birbirlerini evrensellemektedir. Derin CNN'lere bakıldığında [64] görüntü sınıflandırma, belge analizi ve konuşma tanıma alanlarında uygulanmaktadır. Aynı şekilde Derin GAN yani Derin Üretken Düşman Ağı görüntüden görüntüye çeviri ve yüz yaşlanması gibi alanlarda kullanılmaktadır. Tekrarlayan operatörler, yayılım modülünde bulunan bir diğer operatör grubudur. Bu operatörlerde katmanların ağırlıkları eşittir [23]. Tekrarlayan operatörler, yakınsama tabanlı yöntemler ve geçit tabanlı yöntemler olarak iki farklı grupta incelenmektedir. Şekil 7'de [7] tekrarlayan operatörler görülmektedir.



Şekil 7: Tekrarlayan operatörler-yakınsama

İlk incelenen grup olan yakınsama tabanlı yöntemler, dört farklı modelden oluşmaktadır. Temel olarak GraphESN, SSE ve LP-GNN modelleri ele alınmıştır. GraphESN yani Grafik Yankı Durum Ağlarını anlayabilmek için öncelikle ESN'lerin yani Yankı Durum Ağlarının anlaşılması gerekmektedir. ESN'ler RNN yankı durumlarının oluşması ile adlandırılmaktadır. İlk bakış sinir ağlarında gerçek zamanlı durumlarda işlenebileceğinin düşünülmesiyle temeli oluşturulmuştur. İlk çalışma sinirsel nöron okuması, nörofizyoloji ve nöroteknoloji'de kullanılmıştır. Kullanım Turing makinelerine dayanmaktadır ve titiz bir şekilde matematiksel işlemler üzerinde çalışmaktadır [65]. ESN'ler, RNN'lerin yankı durumlarında davranışları ve onlara karşı geliştirilen bir model olarak karşımıza çıkar [66]. GraphESN'ler ise ESN'lerin grafikler üzerinde geliştirilmesiyle oluşturulmaktadır. GraphESN'ler farklı grafik türlerinde (yönlendirilmiş/yönlendirilmemiş, döngüsel, etiketli) Özyinemeli Sinir Ağlarının yani RecNN modeline karşı oluşturulmuştur. ESN'ler ile bu yönü benzerdir. Performans için eğitimsiz kodlama kullanılmaktadır. Bir avantajı ise sinir ağları ve çekirdek tabanında yaklaşımları karşılaştırabilmektedir [67]. SSE modeli stokastik kararlı durum gömme ağ modeli olarak bilinmektedir. Grafiklerde yinelemeli algoritma kullanımında kararlı durum öğrenimi için kullanılmaktadır. Model ayrıca algoritmanın kullanımında 10.000.000'dan fazla düğüm için grafik işleyebilmektedir. Ayrıca model karşımıza çıkan sabit durumlarda kısıtlamalar arasında geçiş yapabilmektedir [68]. Lagrange Yayılımı Grafik Sinir Ağı (Lagrangian Propagation Graph Neural Network-LP-GNN) modelinde grafiklerde bilgi yayılımında Lagrange çerçevesi kullanılmaktadır. LP-GNN, GNN mimarisi üzerinde inşa edilmiş GNN yapısını genişleten bir modeldir. Komşu düğüm arasındaki durum incelemelerinde difüzyon mekanizmasından yararlanmaktadır. Difüzyon mekanizması, nöral ağlarda karşılaşılabilen problemlerden olan derinlik ve yeniden sarma gibi problemlerin üstesinden gelmek için kullanılan Grafik Makine Öğrenmesi yöntemidir. LP-GNN'de GNN hesaplamaları kolaylaştırılmaya çalışılmıştır. Grafiklerin işlenmesi için başarılı olacağı belirtilmiştir [69].



Şekil 8: Tekrarlayan operatörler-geçit

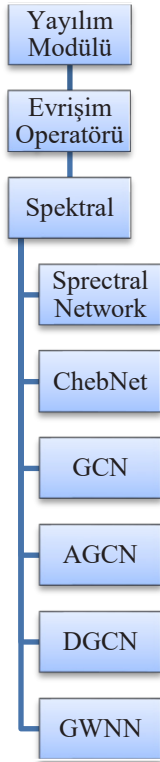
Tekrarlayan operatörler içerisinde incelenen bir diğer model ise Şekil 8’de [7] de görüldüğü gibi geçit ya da kapı tabanlı yöntemlerdir. Bu yöntemler dört farklı model ile anlatılmaktadır. Modellerin anlaşılabilirliği için öncelikle Kapılı Tekrarlayan Birim (Gated Repetitive Unit-GRU) ve Geleneksel Uzun Kısa Süreli Bellek (LSTM)’in anlaşılması gerekmektedir. Bu yöntemler kapı kullanmayı amaçlamaktadır. Tekrarlayan sinir ağlarında kodlayıcı ve kod çözücü işlemlerinde yardımcı olmaktadır [70]. GRU’lar derinleştirilerek bilgi akışını kontrol etmektedir. Derin Basit Kapılı Birim (Deep Simple Gated Unit-DSGU) ve Basit Kapılı Birim (Simple Gated Unit-SGU), uzun süreli bağımlılıkları öğrenmek için kullanılmaktadır. GRU bilgi akışını kontrol etmek için birden fazla kapı kullanırken SGU ve DSGU ise sadece çarpımsal kapıları kullanmaktadır. Bu sayede öğrenme hızı arttırılabilmektedir. Bu kapı birimleri RNN’lerde ilişki analizi için kullanılan RNN Geleneksel Grafiği (RNN Conventional Graph-RCG) için de kullanılmaktadır [71]. Geleneksel Uzun Kısa Süreli Bellek (LSTM) ise geniş zaman aralıklarında kullanılmaktadır. Asıl amacı bilgi depolamayı öğrenmektir. LSTM gelişmişlik düzeyiyle hesaplama karmaşıklığını sıfıra indirmektedir. Tekrarlayan sinir ağları tarafından çözülemeyen zorluk olan uzun süre gerektiren gecikmeli görevleri de çözebilmektedir. LSTM başarılı olmuştur ve öğrenme seviyesi yüksek olmakla birlikte hızlıdır [72]. GGNN yani Kapılı Grafik Sinir Ağı, özellik öğrenmeye dayalı geliştirilmiş bir metottur. GNN’ler yaygın olarak bilgi tabanları, doğal dil bilimleri ve sosyal bilimler alanlarında kullanılmaktadır. Referans olarak kapılı tekrarlayan modüller alınmış ve optimizasyon kullanılmıştır. Geliştirilen model sıra-geçit tabanlı modellere göre daha olumlu sonuçlar vermiş yapay zekâ ve grafik öğrenme algoritmalarında kullanılmıştır. Model temel olarak yüksek performans sergilemiştir [73]. Model sonraki yıllarda grafik – sıralı öğrenme için revize edilerek geliştirilmiştir. GNN’lerde yaygın bir biçimde kullanılan NLP öğrenme aşamasında farklı sorunlar yaratabilmektedir. Geliştirilen model dilbilgisi tabanlı öğrenme yaklaşımlarında iyi bir performans elde etmiştir [74]. GGNN’ler, GNN’lerin yetersizliklerini ortadan kaldırmak için geliştirilmiş ve bu amaçla Kapı Tekrarlayan Birimlerini (GRU) ve hesaplama için zaman içinde geri yayılımı (Backpropagation Through Time-BPTT) kullanmaktadır [7]. Sinir ağlarının gelişiminde farklı algoritmalar kullanılmıştır. Bu algoritmaların biri de Geri Yayılım (Backpropagation-BP) algoritmasıdır. Algoritma makine öğrenme algoritması olan gradient descent’i kullanmaktadır. Gradient descent algoritması GNN’lerde grafik üzerinde bulunan noktaların ya da belirlenecek bir işlevin yerel ekstremum noktalarını bulmak için kullanılan yinelemeli bir optimizasyon algoritmasıdır. Makine öğrenme ve derin öğrenme de sıklıkla kullanılmasıyla birlikte en çok kullanım amacı en az kayıp oluşturmaktır. BP algoritması bu algoritma özelinde türetilmiştir. Bu sayede geliştirilen model ile önyargı ve nöron ağırlığı hataları tespit edilebilmektedir. BPTT ise BP algoritmasının yinelenen sinir ağlarına uygulanması için geliştirilmiş ve sıralı modeller için verim sağlamıştır. Bunun başlıca nedeni RNN üzerinde yapılan deneylerde RN’lerin tekrarlama yapması ve geçmiş bilgisini tutmasıdır [75-77]. TreeLSTM, ağaç yapısı kullanılarak genişletilen LSTM türevi bir metottur. Metot yapılan deneyler sayesinde ve düşünce amacı ile iki cümle anlam tahmini ile duygu sınıflandırmasında kullanılmıştır. Kompozisyonel Dağılım Semantik Modellerinin değerlendirilmesi ile anlamsal ilişki ve yargı tahmini-iki

cümle anlam tahminini oluşturmuştur. Öte yandan uzun cümlelerin anlam ifadelerinde bulunan anlamsızlıkları ortadan kaldırmak için zengin küme ve denetimli eğitim kullanılması gerekmektedir. Bu durum sonucu duygu ağacı bankası oluşturulmuştur. Ağaç kümesinde eğitilen Özyinelemeli Tensör Sinir Ağı ile duygu tahmini yapılmış ve yüksek performans göstermiştir. Ayrıca bu model olumlu ve olumsuz ifade kavramı ile alanında tek model olmuştur. TreeLSTM, LSTM mimarisi üzerinde geliştirilerek Child-Sum Tree-LSTM ve N-ary Tree-LSTM modellerini de oluşturmuştur. Child-Sum, yüksek dallanma ve çocuk sırasız için uygunken, N-ary, N sayılı dallanma ve sıralı ağaç yapılarında kullanılmaktadır [7, 78-80]. Grafik LSTM, TreeLSTM’in geliştirilmiş versiyonudur. TreeLSTM modellerine uygulanabilmektedir. Grafik yapılı LSTM ilişki ayıklama [81], nesne ayrıştırma [82] ve konuşma modellemesi [83] gibi farklı alanlarda kullanılmıştır. İlişki ayıklamada, yapılan çalışmalar genellikle ikili ilişkilerde ayıklama ile yapılmaktadır. Daha fazla ilişki için n-ary kullanılarak, grafik uzun vadeli bellek ağları yardımı ile daha fazla ilişki ayıklanması denenmiştir. Bu sayede çok görevli öğrenme sağlanacak ve çapraz bilgi üretme gerçekleşecektir. Bu ilişkilerde keyfi aritmetik kullanılmaktadır. Geliştirilen model başarılı olmuştur [81]. Anlamsal nesne ayrıştırması ile LSTM kullanılarak geliştirilen süper pikseller yardımıyla yeni bir grafik elde edilmektedir. Bu sayede Graph LSTM görüntülere daha iyi odaklanabilmekte ve daha fazla başarı sağlamaktadır [82]. Konuşma modellemesinde ise diğer kullanım amaçlarında kullanılan uzun vadeli bellek ağları yerine uzun kısa süreli bellek kullanılmıştır. Bu sayede çift yönlü ağaç kullanılmış, modelleme ve yorumlamada tanımlama işlemi kolaylaşmıştır [82]. Tekrarlayan operatörlerde incelenen son model olan S-LSTM, metin kodlaması amacıyla geliştirilmiştir. Geliştirilen model, dizi etiketleme için kullanılan BiLSTM modeliyle rekabet edebilmektedir [84-85].

Evrişim operatörü, yayılım modülü altında incelenen bir diğer operatördür. Evrişim operatörleri en çok kullanılan yayılım modülü operatörüdür. Evrişim operatörü uzaysal yaklaşım ve spektral yaklaşım olmak üzere iki başlık altında incelenmektedir. Şekil 9’da [7] evrişim operatörü spektral yaklaşımlar görülmektedir.

Spektral yaklaşımlar, grafiklerin görüntü yönüne odaklanmaktadır. Bu durum grafiklerin sinyal işleme alanına dayandığını göstermektedir. Büyük veriler grafiklerde ağırlık dengelemek için uç köşelerde bulunmaktadır. Bu sebeple grafiklerde sinyal işleme yapılırken sayısal ve görüntü yönüne dikkat edilmektedir. Bu düzensiz yapılar grafik görüntü işleminde düzene sokularak temel işlemler genelleştirilmektedir. Bunun sebebi verimli ve güvenli bir şekilde bilgi çıkarımı yapmaktır [7, 86]. Evrişimin temelinde Fourier dönüşümü bulunmaktadır. Doğrusal sinyal işleme için frekans filtreleme kullanılmaktadır [87]. Spektral yaklaşımlar altı farklı alt başlıkta incelenmektedir.

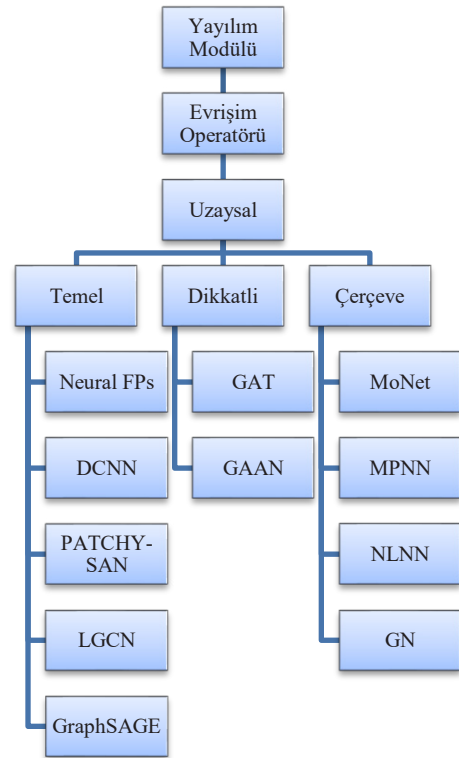




Şekil 9: Evrişim operatörü–spektral

Spektral ağ, geometrik derin öğrenmenin geliştirilmesiyle elde edilmektedir. Bilgisayarlı görme, doğal dil işleme ve ses analizi konularında derin sinir ağları kullanılmaktadır. Geometrik derin öğrenme bu alanlardaki sorunlara çözüm olma amacıyla oluşturulmuştur [88]. Aynı şekilde Öklid olmayan uzayda derin öğrenme gerçekleşmesi için büyük verilerde test edilen model başarılı olmuştur [89]. ChebNet, Chebyshev polinom yaklaşık algoritması ile dönüşüm hesabı yapmak için geliştirilmiştir. Amaç grafik köşelerinde dalgacık dönüşümü gerçekleştirmektir. Çözüm için grafik Laplacian spektral ayrışmasını kullanmaktadır [90]. Laplacian, grafik öğrenmede sıklıkla kullanılmaktadır. Karışık ve büyük verilerde başarı göstermektedir [91]. Aynı şekilde CNN üzerindeki uygulamalarda geliştirilerek grafik spektral teorisi ile sorun çözülmektedir [92]. Yakın geçmişte Chebnet gelişerek yüksek boyutlu fonksiyonlarda da verimli olmuştur. Düzeltilmiş doğrusal birimleri (Rectified Linear Unit-ReLU), düzeltilmiş güç üniteleri (Rectified Power Units-RePU) ile karşılaştırılarak verimleri ölçülmüştür. Model sayısal olarak kararlı ve verimlidir. RePu daha güzel sonuçlar vererek başarısını kanıtlamıştır [93]. Bu çalışma sonrasında Chebnet'e olan ilgi artmış ve pratik olarak karşılaştırma tekrar yapılmıştır. Chebyshev polinom yaklaşık hiyerarşi kuramı üzerine yeni bir model geliştirilmiş, geliştirilen modelde RePu kullanılmıştır. Yapılan karşılaştırma bir sonraki deneylerde ChebNet ile yapılarak ChebNet'in daha kararlı ve verimli olduğu ortaya konulmuştur [94]. AGCN, yani Uyarlanabilir Grafik Evrişimli Sinir Ağları CNN'lerin evrimleşmesi ile gerçekleşmektedir. CNN'ler alt modellerine ayrılırken grafik uygulamaları için Graph CNN'ler geliştirilmiştir [95]. Graph CNN, görüntü analizi ve grafiklerde kullanılmaktadır. Başarısı kanıtlanmış ve nokta bulutlarında sıklıkla kullanılmaktadır [96]. AGCN, temel

girişlerde karmaşa olsa da CNN'in geliştirilmesi ile görev odaklı eğitim sonucu uygulanabilir GNN'yi oluşturmaktadır. Öğrenme için metrik öğrenme tavsiye edilmiş, deneyler sonucunda ise üstün başarı göstermiştir [97]. DGCN, yani ikili grafik evrişim ağı grafik analizi ile veri ayıklamayı amaçlamaktadır. Yarı denetimli öğrenme ile küresel tutarlılık ve yerel tutarlılık sağlamaktadır. Deneyler sonucu başarı gösteren DGCN, iki evrişimsel ağ kullanarak çalışmakta ve bu çalışma sonucunda denetimsiz bir kayıp çıkmaktadır [98]. GWNN, yani Grafik Dalgacık Sinir Ağı grafik dalga dönüşümü kullanarak Spektral Grafik CNN'lerin eksikliklerini ortadan kaldırmak için geliştirilmiş bir metottur. Metot matris ayrışmasına ihtiyaç duymadan elde edilebilmekte ve seyrek–yerel grafik dalgaları sayesinde verimlidir. Yarı denetimli sınıflandırma ile Pubmed, cora ve citiseer ile test edilmiş CNN'lere göre yüksek başarı göstermiştir [99]. Evrişim operatörlerinde inceleyeceğimiz bir diğer başlık ise Uzaysal yaklaşımlardır. Uzaysal yaklaşımlar Şekil 10'da [7] görülmektedir.



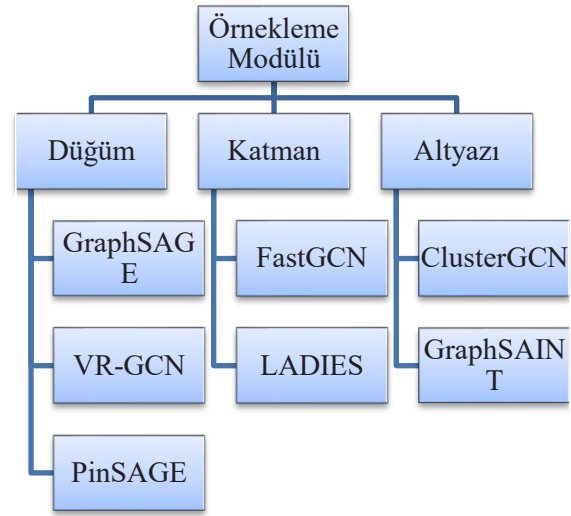
Şekil 10: Evrişim operatörü–uzaysal

Evrişim operatörü uzaysal yaklaşımlar üç farklı başlık altında incelenmektedir. Bu başlıklar, temel uzaysal yaklaşımlar, dikkat temelli uzaysal yaklaşımlar ve çerçeve genellemesinde uzaysal yaklaşımlar şeklindedir [7]. Temel uzaysal yaklaşımlar, örtüşme miktarı grafiklerin değişmeyen özellikleri üzerinde zaman eksenini boyunca gerçekleşen yaklaşımlardır. Bu yaklaşımların karşılaştığı zorluk ise farklı boyutlardaki kümelerin evrişim operatörü tanımlanması ve CNN tabanında değişmezliklerin kontrol edilmesidir. Yaklaşımlar beş farklı model ile incelenmektedir [7]. Neural FPs, sinirsel parmak izi olarak tanımlamakta ve temeli dairesel parmak izlerine dayanmaktadır. Model düşüncesi sinirsel parmak izlerinin sabit parmak izlerine kıyasla yeni avantajlar sunacağı araştırmasıyla doğmuştur. Tahmin

performansı, yorumlama ve parsimoni avantajları ile öne çıkan sinirsel parmak izleri, doğrusal yönden grafikler üzerinde çalışmaktadır. Veri odaklı özellikler makine görüşü, konuşma tanıma ve doğal dil işleme gibi alanlarda manuel giriş ve özelliklerin yerini almakta olduğu bilinmektedir. Geliştirilen model bu alanlarda başarı sağlamak ayrıca sanal tarama, malzeme tasarımı ve ilaç tasarımı çalışmaları için öncü olduğu düşünülmüştür [100]. DCNN, en sık kullanılan grafikli sinir ağlarından olup difüzyon-konvolüsyonel sinir ağları olarak adlandırılmaktadır. Düğüm sınıflandırılmada kullanılan DCNN, grafiklerin yapılandırılan verilerden öğrenimini difüzyonel gösterimler ile sağlamaktadır [101]. PATCHY-SAN, grafik verilerinde öğrenme sorunu üzerinde durmaktadır. Sorun temele indiğinde iki ana sorun halinde düşünülmüş ve çözüm üretilmeye çalışılmıştır. Komşuluk sorunları üzerinde duran model, düğüm dizilerinin anlaşılabilirliği ve normalleştirme ile eşleştirilmesi durumlarını rastgele grafiklerde yapmak için geliştirilmiştir. Rastgele grafikten bahsedilen ise grafiklerin yönlü, yönsüz, homojen, heterojen, ayrık ve sürekli olma durumlarıdır. CNN temelinde yapılan deneyler eşliğinde model üstün performans göstermiş, gelecek çalışmalar için ise RNN tarzı sinir ağlarının kullanılmasıyla ön eğitim ve ilişkisel grafik modellerin oluşturulabileceğini belirtmiştir [102]. LGCN, öğrenilebilir grafik evrimsel ağlar olarak tanımlanmaktadır. CNN'lerin genel sorunu olan ızgara verilerinde elde edilen başarının grafiklerde de elde edilmesi istemiyle geliştirilmiştir. Genel anlamda eşli işlem kullanılması gerekli olan durumlar için yani komşu sayılarının değişken olduğu durumlarda ele alarak öğrenilebilir grafik eşlemsel katmanı (Learnable Graph Convolutional Layer-LGCL) altında bir öneri ile sorunu ele almıştır. LGCL verileri bir boyutlu ızgara verilerine çevirirken, komşu sayısını sabit seçerek eş işlem yapılmasını mümkün kılmaktadır. Transdüktif ve endüktif öğrenme yöntemleri ile büyük boyutlu grafiklerde de işlem alt grafik eğitimi ile gerçekleştirilmiştir. Önerilen yöntem farklı veri setlerinde deneyerek yüksek performanslı sonuçlar elde etmiştir. Alt grafikler için önerilen LGCL tabanlı algoritma büyük boyutlu grafiklerin eğitiminde başarı sağlamıştır. Gelecek çalışmalarda ise düğüm sınıflandırma konusunda geliştirilebileceği gözlemlenmiştir [103]. GrapSAGE ise büyük grafiklerde endüktif temsili öğrenmeyi amaçlamaktadır. Temel de düğüm gömmelerinin verimli bir şekilde sağlanmasıyla model kendini göstermektedir. Model farklı veri setlerinde deneylerle gözlemlenmiş ve başarı göstermiştir [104].

### 2.3.2. Örneklem Modülü

Örneklem modülü, GNN temel tasarım evreleri olan üç modülden birisidir. İkinci modül olarak incelenen örneklem modülü, komşu patlaması sorununu çözmek için oluşturulmuştur. Komşu patlaması, komşu katmanlardan toplanan mesajların birikmesi ve birden fazla GNN katmanı içerisinde takipleşmesiyle boyutun derinliklere inmesi olarak tanımlanmaktadır. Bir başka önemli özelliği de her düğüm için komşuluk bilgileri yüksek boyutlara ulaşması sebebi ile biriktirilerek depolanması sonucu işlenememektir. Örneklem modülü, burada yayılımı gerçekleştirir ve işlenmesinin önündeki engeli ortadan kaldırmaktadır. Örneklem modülü temel olarak düğüm örneklem, katman örneklem ve alt grafik örneklem olarak üç başlık altında incelenebilmektedir [7]. Şekil 11'de [7] örneklem modülü ve alt başlıklar halinde örneklem modülünü kapsayan modeller görülmektedir.



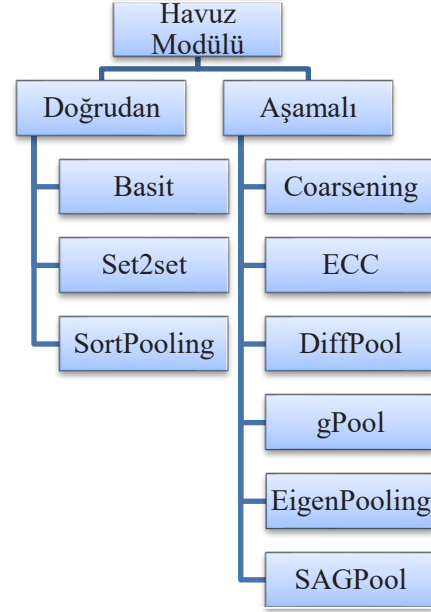
Şekil 11: Örneklem modülü

Düğüm örnekleme, düğümlerin boyutlarının küçültülmesi ile gerçekleşmektedir. Bu işlem için en basit yöntem komşu düğüm alt kümelerinden birisinin seçilmesidir. Düğüm örnekleme üç farklı model ile işlenmektedir [7]. GraphSAGE, bu işlemi her bir düğüm için 2-50 düğüm komşuluğunda sabit ve az bir sayı ile sağlamaktadır [7]. VR-GCN, Varyans azaltmalı grafik evrişim ağları olarak tanımlanmaktadır. GCN'de bulunan temel sorunlardan biri olan hesaplanan boyutun özyinelemeli olması sebebiyle sürekli artması ve azaltmaya yönelik çalışmaların başarısız olma noktasını referans almaktadır. Bu duruma karşı olarak komşu boyut örnekleme yapabilmeyen kontrol değişken tabanlı bir algoritma geliştirilmiştir. Geliştirilen algoritma GCN'de azaltma çalışmalarının başarısını garanti etmektedir [105]. Pingsage, önem öncelik temelinde geliştirilmiş bir örneklem modülü modelidir. Sistem, hedef düğümü başlangıç kabul ederek en üstteki normalleştirilmiş ziyaret düğümlerini seçmektedir. Bu sistem bir nevi kıyaslama sistemi olarak tanımlanmaktadır. Teknolojinin gelişmesi yüksek performanslı yol açsa da kullanıcı sayısı çok fazla olan web tabanlı sistemlerde bu büyük bir sorun olarak algılanmaktadır. Bu soruna çözüm oluşturmak için geliştirilen model, grafik yapısı ile birlikte düğüm özelliklerini de içeren düğümlerin gömmelerini oluşturmaktadır. Geçmiş çalışmalara kıyasla daha verimli bir model geliştirmek için yüksek verimli rastgele yürüyüşlere dayanan bir model önerilmiştir. Gömme oluşturmaya yardımcı olması için de MapReduce çıkarım algoritması geliştirilmiştir. Model Pinterest üzerinde bulunan farklı ortamlarda denenmiş ve başarılı olmuştur. Model günümüz itibarı ile en başarılı grafik gömme modeli olmuştur. Ayrıca web tasarım ölçeğinin yeni nesil versiyonlarının önünü açarak öncü olma konumuna geçmektedir [106]. Bir diğer örneklem yöntemi olan katman örnekleme, her katmanda küçük bir düğüm tutulmasıyla gerçekleşmektedir. Bunun sebebi genişlemeyi engellemektir. Katman örnekleme iki farklı model altında incelenmektedir [7]. FastGCN, GCN'lerin temel sorunu olan bellek boyutlama konusunda yeni bir yaklaşım sunmaktadır. Tasarlanan model

eksiklerin eğitimini tamamlarken ayrıca model sonucu çıkarımları da göstermektedir. Yapılan deneyler sonucu, tahminler belirli bir doğrultuda kalırken eğitimin verimli bir başarı gösterdiği gözlemlenmiştir. FastGCN, örnekleme alıcı alanı örnekleme yapmaktadır [107]. Sabit örnekleme aksine, katman bazında örnekleme olması için eğitilebilir ve uygulanabilir bir model geliştirilmiştir. Model öznelinde hızlı grafik gösteriminde kullanımı düşünülmüştür. Model üst katmandan alt katmana doğru örnekleme gerçekleştirirken, farklı üst düğüm paylaşımı ve sabit boyut örnekleme olması sebebi ile alt katmanları örnekleme yapmaktadır. Bu açık varyans azaltmayı sağlayabildiği için eğitim de gelişmektedir. Farklı deneylerde hız konusunda performans gösterilmiş olsa da doğruluğu konusunda karara varılamamıştır [108]. LADIES, katman bazında örnekleme oluşan seyreklik sorunlarına azaltmak için önerilmiş bir modeldir. Kelime anlamı olarak katmana bağlı önem örnekleme olarak tanımlanabilmektedir. Üst katmanda bulunan örneklenmiş düğümleri referans alan model, düğümleri seçerek olasılık hesaplamaktadır. Olasılığı hesaplarırken bipartite grafiğini kullanmaktadır. Bu grafiğin en temel özelliği, köşelerini ayrık ve bağımsız kümeler ayırabilmesidir. Özyinelemeli olarak sabit sayıda katman başına düğüm örnekleme yapmaktadır. Yapılan deneyler sonucu verimli sonuçlar elde edilmiştir [109]. Örnekleme modülünde incelenen son yöntem ise alt grafik örnekleme modülüdür. Alt grafik örnekleme modülünde iki farklı model başlığı altında incelenmektedir. Temel yöntem alt grafiğe örnekleyerek, komşuluk aramasında alt grafikleri kısıtlamaktır [7]. ClusterGCN, GCN'lerde katmanların yükselmesiyle artan maliyetler ve geniş alan gereksinimi sorunlarına çözüm oluşturmak için önerilen bir küme algoritmasıdır. Önerilen model GCN'lerde SGD tabanlı algoritmalar için de gelişim göstermiştir. SGD algoritmaları, stokastik gradyan iniş algoritmaları olarak tanımlanmaktadır. Bir işlevin pürüzsüzlüğü yani sahip olduğu alanlar öznelinde sürekli türevlerin sayısı ile ölçülen özelliği optimize etmek için kullanılan yinelemeli alan bir yöntemdir. Küme-GCN'in çalışma mantığı, adımlarında grafik kümeleme algoritması kullanarak oluşturulan alt grafiklerin ilişkilendirilmesi sonucu düğüm bloklaşması ve mahalle aramasını kısıtlamasıyla oluşmaktadır. Bu sayede belleklerde ve hesaplamalarda verim oluşturmaktadır. Yapılan deneyler sonucu başarı elde etmiş ve yüksek performans göstermiştir [110]. GraphSAINT'te ise alt grafik oluşturma amacıyla düğümler ve kenarlar örnekleme yapmaktadır. Bu yönüyle kümeGCN'den ayrılmaktadır. Yinelemeli grafik tabanlı endüktif öğrenme yöntemidir. Her yinelemede düzgün bir alt grafik oluşturulmaktadır. Yapılan deneyler sonucunda üstün performans göstermiş ve başarısını kanıtlamıştır [111].

### 2.3.3. Havuzlama Modülü

Havuzlama modülü, GNN temel tasarımı için geliştirilen üç modül arasından incelenen sonuncu modüldür. Genel özellikler elde etme amacıyla çoğunlukla evrişim modüllerinden sonra gelmektedir. Bilgisayarlı görüde sıklıkla kullanılmaktadır. Büyük yapı ve karışık düzenli grafiklerin görevleri gerçekleştirebilmesi için havuzlama katmanında tasarlanmaktadır. Havuzlama katmanı doğrudan ve aşamalı olmak üzere iki başlık altında incelenmektedir. Şekil 12'de havuzlama katmanı modülleri ve modelleri görülmektedir [7].



Şekil 12: Havuz modülü

Doğrudan havuzlama modülleri, okuma işlevleri olarak variant öznelinde isimlendirilmektedir. Farklı düğüm seçme stratejisine sahip olan düğümler, temsili öğrenmeyi doğrudan grafik düzeyinde sağlamaktadır. Üç farklı model üzerinde incelenmektedir. Basit havuzlama yöntemi, farklı modeller tarafından kullanılmaktadır. Model temel olarak maksimum, toplam, ortalama ve dikkat işlevleri üzerinde durmaktadır. Bu işlemler ile düğüm bazında yardımcı olarak grafik gösterimi elde edilmektedir [7]. Set2Set, MPNN modellerinde grafik öğrenme için kullanılmaktadır. Set2set, zincir kuralı kullanarak karmaşık yapılarda ortak olasılık öğrenimini temsil etmek için önerilmektedir. Yapılan deneyler sonucu set2set'in sıralama ve bilinmeyen grafik modellerinin ortak olan olasılıklarını hazırlamada verimli bir tutum sergileyerek yüksek performans göstermiştir. Temsil için ise LSTM tabanlı yöntemler kullanılmaktadır [112]. SortPooling, sıralı havuzlama olarak tanımlanabilmektedir. Tensör formlarında bulunan genel iki zorluğun çözüme ulaştırılması düşünülmüştür. Bu zorluklar, grafik içerisinde kodlanmış bilginin analiz edilerek faydalı şekilde nasıl çıkarılacağı ve bir grafiğin anlamlı ve verimli bir şekilde nasıl sıralanacağı sorunlarıdır. Analiz zorluğu için GCN modeli tasarlanmış ve çekirdek bağlantıları sağlanmıştır. Sıralama zorluğunda ise sıralı havuzlama modeli devreye girerek katman oluşturulmuştur. Karşılaştırmalı grafikler ile yapılan deneylerde başarı sağlamıştır. En önemli husus ise bu model sayesinde gradyan tabanlı öğrenmeyi desteklemesidir [113].

Hiyerarşik havuzlama modülünde ise karmaşık grafiklerin hiyerarşik durumlarının da incelendiği modellerdir. Altı farklı modelden oluşmaktadır. Genel adıyla graph coarsening yani grafik kabalaştırma, ilk sinir ağlarının dayandığı kümeleme yöntemidir. İçerisinde bulunan öz bileşim adımı modeli verimsizleştirmektedir. Model geliştirilerek Graclus adında yeni bir model oluşturulmuştur. Bu model, k-means yöntemini kullanmaktadır. Model, hedefini doğrudan optimize edebilmekte ve yüksek kalitelidir. Deneysel sonuçlarda yüksek başarı elde etmiştir. Bu modelin en büyük başarısı ise daha hızlı olması ve

havuzlama sağlamasıdır. Model, ChebNet ve MoNet'te düğüm çiftlerini eşlemek ve birleştirmek için kullanılmaktadır [114]. ECC yani kenar koşullu evreşim, özyineleme kullanarak altörnekleme ile havuzlama modülünü tasarlamaktadır. Altörnekleme yöntemi Laplacian'ın öz vektörünün (en büyük ve işaret dâhilinde) ikiye bölünmesi ile oluşturulmaktadır [7]. Model temel olarak normal ızgaralardan geliştirilerek rastgele grafikler elde etmektedir. Yapılan deneyler sonucu üstün performans göstermiştir [115]. DiffPool yani fark havuzu, grafiklerde hiyerarşik yapıların temsillerini oluştururken, GNN mimarilerini uç uca birleştirebilmektedir. İşlem gerçekleşirken fark havuzu öncelikle küme ataması öğrenmektedir, düğümler küme ile eşlenerek GNN temel tasarımı için girişi kaba bir şekilde oluşturmaktadır. Yapılan deneyler sonucu üstün performans göstermiş ve son teknolojiye ulaşmıştır [116]. gPool, proje vektörü kullanarak düğümlerde puan öğrenme ve en yüksek puanı seçmektedir. Diffpoll ile olan farkı her katmanda vektör kullanmasıdır. gPool, havuz oluşturma için önerilirken ayrıca biriktirme içinde gUnpool modeli önerilmiştir. GUnpoll modeli, gPool modelinin tersine çalışması olarak da tanımlanabilmektedir. gPool, grafik sınıflandırma ve düğüm sınıflandırma için kullanılmaktadır. Deneyler sonucu yüksek başarı göstermiştir [117]. Eigenpooling, GNN'lerde görev dönüşümü ile başarı sağlamayı amaçlamaktadır. Düğüm özellikleri belirli işlemler dâhilinde işlenerek birçok görevin performansını yükseltmektedir. En bilinen yaygın kullanım ise düğümlerin gösterimlerinin birleştirilmesidir. Model temel olarak havuzlama işlemi gerçekleşirken düğümlerin özelliklerini ve yapılarını Fourier prensibine dayanarak gerçekleştirebilmesidir. Teorik analizde yüksek başarı göstermiş fakat uygulama deneylerinde başarı göstermesine rağmen teorik analiz başarı seviyesine yükselmemiştir. Kısaca bu model, düğüm özellikleriyle birlikte yerel yapının kullanılması amacıyla tasarlanmış ve alt grafik bilgi çıkarımı için kullanılmaktadır [118]. Havuzlama modülünde ve GNN temel tasarımı için incelenen son model olan SAGPool, grafik temsil öğrenmesinde özellik ve topoloji kullanımı sağlayarak öğrenme gerçekleştirme amacıyla önerilmiştir. Evreşim işlemlerinde geliştirilme yapılması ve havuzlama modülleri kullanımının yaygınlaşmasıyla performanslarda olan artıştan etkilenen SAGPool, grafiklerde alt örnekleme işleminde bulunan zorluklara odaklanmaktadır. Yapılan deneyler diğer modeller ile karşılaştırılarak üstün performansa ulaştığı gözlemlenmiştir. SAGPool'ün en dikkat çeken özelliği ise zaman ve mekân karmaşıklığı ile dikkat temelli bir yöntem oluşturarak kullanılması olmuştur [119].

### 3. Grafik Sinir Ağlarının Dezavantajları

GNN'ler temelde Yapay Sinir Ağı (Artificial Neural Network-ANN) ailesine mensup CNN'lerden türetilmiştir. Geliştirilen modeller kendine özgü olmakla beraber hedefe odaklı geliştirilmektedir. Genel olarak GNN'lerin dezavantajlarına bakacak olursak; ölçeklenebilirlik, görülmemiş grafiklerde sınırlı genelleme, dinamik grafiklerin işleme zorluğu, yorumlanabilirlik, standart mimari eksikliği, veri seyrekliği ve gürültüsü ile teoride sınırlılık şeklinde gösterilebilir.

GNN modelleri her ne kadar başarı sağlamış olsa da dezavantajları bulunmaktadır. Bu dezavantajlar model

geliştirmeyi güçlendirmekte ve anlam karmaşasına yol açmaktadır. Gerçek dünya uygulamalarında, topolojiler esnek ve karmaşıktır. Bu durum heterojen ve dinamik grafiklerde zorluk oluşturmaktadır. Zamansal grafikler, grafik tahmini, çizge model geliştirilmesi ile kenar sinyallerinin öznitelikleri bu zorluklara örnek olmaktadır. Dinamik grafiklerde sinyallerin zaman içinde değişmesi, gelişen grafiklerde ise yapının değişmesine bağlı sinyal değişikliği sebebiyle bu durum oluşmaktadır. Dinamik grafiklere GNN modellerin etkin bir çalışma alanı olan trafik ağları örnek olabilirken, gelişen grafiklere sosyal ve bilgisayar ağları örnek verilebilmektedir. Farklı grafik türleri için de bu durum geçerlidir. Geliştirilen model kendine özgü grafikte kolaylıkla işlem yapabilirken farklı bir grafikte zorlanmaktadır. Özel havuzlama operatörü oluşturularak bu zorluk yok edilebilir. Mevcut havuzlama yöntemleri grafik düzeyindeki görevlere odaklanmıştır. Bu durum GNN modellerinin farklı görev ve çoklu görevler üzerindeki etkisini azaltmaktadır. Geliştirilen modeller bu sebeple bir ya da iki görev için geliştirilmiştir. Benzer görevlerde ise bu durum avantaja çevrilebilir. Model derinliği, sadece sinir ağlarının değil derin sinir mimarilerinin de zorlukları arasındadır. Karşılık olarak evreşim katmanı kullanılmasıyla performansa beklenti karşılanamamaktadır. Bu durum derin modellerde etkili olmakla birlikte, günümüzde devam etmektedir. Aynı şekilde büyük ölçekli verilerde performansa etki eden durumlar arasındadır. Havuzlama modellerinin küçük kıyaslama veri kümelerini seçmesi karşılaştırma durumunu azaltmaktadır. Kullanılan veri setleri yaygın olsa da yeterlilik bakımından tartışılmaktadır. Karmaşık ağlar da geliştirilen modellerin önem gösterdiği konulardan olmuştur. GNN'lerin çoğunlukla kenar ve düğüm ile sınırlı olması karmaşık modeller yüksek dereceli yapı olarak adlandırılmaktadır. Bu durum da GNN modellerinin zorluklarındandır. Verilerde bulunan gürültü, GNN modellerinde gürültü giderici katmanlara yönlendirmiştir. Fakat bu durum yine de sorunu tam anlamıyla çözememiştir. İlaç tasarımı ve hastalık teşhisi gibi durumlarda gerçek verilerin kullanılması topolojiyi bozması sebebiyle performansı düşürmektedir. Aynı durum yorumlanabilirlikte de karşımıza çıkmakta ve her model için havuzlama modeli geliştirilmesi önerilmektedir. Gerçek yaşam üzerinde işlenmesi, modelin ne kadar yorumlanabildiğine bağlı olmaktadır. Mevcut havuzlama yöntemlerinin sezgisel tasarımı da modelin ifade gücünü azaltması sebebiyle yorumlanabilirlik ve ifade gücü zorluklarını oluşturmaktadır. Bir diğer sorun olan ölçeklenebilirlik ise, büyük grafiklerle modellenen endüstri uygulamaları için karşılaşılan bir zorluktur. Twitter örneğinde, milyarlarca düğüm bulunmaktadır ve modeller bu denli büyük bağlamlar için uygunsuzdur. Bu durum göz ardı edilerek çözülmüş olsa da ölçekleme sorunu GNN modellerinin en sık karşılaştığı zorluklardan biri olmaktadır. GNN modelleri, her ne kadar bir amaç uğruna konu kapsamında geliştirilse de bu zorluklarla karşılaşabilmektedir. Bu durum GNN'lerin başarısını azaltsa da diğer sinir ağlarına göre performans-verim bakımından en yetenekli sinir ağı olma durumunu sürdürmektedir. Tablo 1'de sinir ağı modellerinin karşılaştırması görülmektedir [120-122].

Tablo 1: Sinir Ağlarının Karşılaştırması

Özellik	ANN	GNN	CNN	RNN
<b>Giriş Verileri</b>	Metin, görüntü, zaman serisi verileri, sayısal değerli veriler ve kategorik veriler.	Düğümün ve kenarların sırasıyla varlıkları ve ilişkilerini temsil ettiği, sosyal ağlar, moleküler grafikler vb. gibi grafik yapıları veriler.	Görüntüler, videolar ve konuşma sinyalleri gibi ızgara yapıları veriler.	Zaman serileri, metin dizileri, konuşma sinyalleri vb. gibi sıralı veriler.
<b>Veri Gösterimi</b>	Satır ve sütunlardaki yapılandırılmış veriler veya özellik vektörleri gibi tabular veya vektörleştirilmiş veriler.	Düğümün ve kenarların sırasıyla varlıkları ve ilişkilerini temsil ettiği, sosyal ağlar, moleküler grafikler vb. gibi grafik yapıları veriler.	Piksel yoğunluklarına veya ızgaraya özgü diğer özelliklere sahip sabit boyutlu ızgaralar.	Kelime dizileri veya zaman adımları gibi zamansal bir düzene sahip veri dizileri.
<b>Topoloji İşleme</b>	Tipik olarak tam bağlı katmanlar olarak tasarlanan sabit giriş-çıkış bağlantısı varsayar.	Dinamik grafik topolojilerini modelleyebilir ve grafik yapısındaki değişikliklere uyum sağlayabilir.	Yerel konvolüsyon işlemleri ile sabit boyutlu alıcı alanlar kullanır.	Zamansal dinamikleri yakalayan giriş verileri içindeki sıralı bağımlılıkları modeller.
<b>Komşuluk Toplama</b>	Sabit boyutlu giriş özellikleri üzerinde, komşuluk toplamayı açık bir şekilde ele almadan çalışır.	Mesaj geçişi veya grafik konvolüsyon işlemleri yoluyla komşu düğümlerden/kenarlardan özellikleri toplar.	Tipik olarak sabit ızgara topolojisi varsayar ve ızgara yapısındaki değişiklikleri açıkça modellemez.	Sıralı kalıpları yakalayan tekrarlayan bağlantılar aracılığıyla zamansal bağımlılıkları modeller.
<b>Özellik Öğrenme</b>	Katman aktivasyonlarına ve ağırlık parametrelerine dayalı olarak girdi özelliklerinin hiyerarşik temsillerini öğrenir.	Düğümün ve kenarların temsillerini grafik yapısına ve düğüm/kenar özniteliklerine dayalı özellikler olarak öğrenir.	Verilerdeki yerel örüntülerin hiyerarşik temsillerini öğrenerek uzamsal hiyerarşileri yakalar.	Tekrarlayan bağlantılara ve gizli durumlara dayalı olarak sıralı veri temsillerini öğrenir.
<b>Ölçeklenebilirlik</b>	Büyük ölçekli veri kümelerini işleyebilir ve özellikle optimize edilmiş uygulamalarla tipik olarak ölçeklenebilir.	Dinamik çizge yapıları ve mesaj geçirme işlemleriyle başa çıkma ihtiyacı nedeniyle ölçeklenebilirlik sorunlarından muzdarip olabilir.	Sabit boyutlu ızgaralar üzerinde çalıştıkları ve optimize edilmiş konvolüsyon işlemlerinden yararlandıkları için genellikle ölçeklenebilirdir.	Değişken uzunluktaki dizilerle başa çıkabilir ancak eğitim sırasında kaybolan veya patlayan gradyanlardan muzdarip olabilir.
<b>Yorumlanabilirlik</b>	Karmaşık öğrenilmiş özellikler ve veri yapısının açık bir şekilde temsil edilmemesi nedeniyle yorumlanabilirlikten yoksun olabilir.	Çizge yapısına dayalı olarak düğüm ve kenar özelliklerinin yorumlanabilir gösterimlerini sağlayabilir.	Karmaşık öğrenilmiş özellikler ve açık uzamsal hiyerarşi eksikliği nedeniyle yorumlanabilirlikten yoksun olabilir.	Karmaşık öğrenilmiş özellikler ve gizli durumlar nedeniyle yorumlanabilirlikten yoksun olabilir.
<b>Uygulama</b>	Sınıflandırma, regresyon ve örüntü tanıma gibi tablo halindeki veya vektörleştirilmiş veriler üzerindeki görevler için olarak kullanılmaktadır.	Sosyal ağ analizi, ilaç keşfi, tavsiye sistemleri gibi çizge yapıları üzerindeki görevler için çok uygundur.	Görüntü sınıflandırma, nesne algılama, konuşma tanıma vb. gibi ızgara yapıları üzerindeki görevler için yaygın olarak kullanılmaktadır.	Dil modelleme, konuşma tanıma, zaman serisi tahmini gibi sıralı veriler üzerindeki görevler için yaygın olarak kullanılmaktadır.

#### 4. Sonuç

GNN, her geçen gün kullanım alanı genişleyen ve popülerliği artan bir kavram olmaktadır. Modellerin geliştirilmesi, alana özgü olmaması ve belirli bir alan için geliştirilen modelin farklı alanlarda kullanılması GNN'lerin yaygınlaşmasını sağlamaktadır. Bu çalışmada GNN'ler kapsamlı bir şekilde ele alınmıştır. Öklid dışı uzayda yüksek performans sergileyen GNN'ler hakkında farklı alanlar özelinde uluslararası birçok çalışma yapıldığı görülmektedir. Ancak Türkiye'de gerçekleştirilen çalışmaların az olduğu ve ülkemizde bu ağ türünün popülerlik elde edemediği değerlendirilmektedir. Çalışmada GNN modeli temel alınarak geliştirilen yöntemler ve uygulamalar anlatılmıştır. Ayrıca GNN modellerinin genel eksik yönlerine değinilmiş ve sınır ağları karşılaştırılmıştır. Bu çalışmanın ilk Türkçe kaynaklardan biri olmakla birlikte ardılı olacak çalışmalara katkı sağlayacağı düşünülmektedir.

#### 5. Teşekkür

Bu çalışma Milli Savunma Üniversitesi Atatürk Stratejik Araştırmalar ve Lisansüstü Eğitim Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı Siber Güvenlik Tezsiz Yüksek Lisans Programına bağlı olarak yürütülen "Siber Güvenlikte Grafik Sinir Ağları" adlı dönem projesinin bir bölümüdür.

#### 6. Kaynakça

- [1] D.H. Hubel, T.N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex." *J Physiol.* 1962 Jan; vol. 160, no. 1, pp. 106–154.2, 1962.
- [2] A. Sperduti, A. Starita, "Supervised neural networks for the classification of structures." in *IEEE Transactions on Neural Networks*, vol. 8, no. 3, pp. 714-735, May 1997
- [3] F. Scarselli, M. Gori, A.C Tsoi, M. Hagenbuchner, G. Monfardini, "The Graph Neural Network Model." *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61-80. 2019.
- [4] M. Gori, G. Monfardini, F.A. Scarsell, "New Model for Learning in Graph Domains." *Proceedings. 2005 IEEE International Joint Conference on Neural Networks*, vol. 2, pp. 729-734, 2005.
- [5] M. Gori, M. Maggini, L. Sarti, "A recursive neural network model for processing directed acyclic graphs with labeled edges." in *Proceedings of the International Joint Conference on Neural Networks*, Portland (USA), pp. 1351–1355, July 2003.
- [6] Z. Chen, J. Xu, C. Alippi, S.X. Ding, Y. Shardt, T. Peng, C. Yang, C. "Graph neural network-based fault diagnosis: a Review." *ArXiv.* 2021
- [7] J. Zhou, et. Al., "Graph neural networks: A review of methods and applications." *AI Open.* Vol. 1, p.57-81, 2020.
- [8] N.A. Asif, Y. Sarker, R.K. Chakraborty, M.J. Ryan, H. Ahamed, D.K., Saha, F.R. Badal, S.K. Das, F. Ali, S.I Moyeen, R. Islam, Z. Tasneemi "Graph Neural Network: A Comprehensive Review on Non-Euclidean Space." April 27, 2021.
- [9] M.M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, P. Vandergheynst, "Geometric deep learning: going beyond Euclidean data." *IEEE SIG PROC MAG*, 2017
- [10] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, P.S. Yu, "A Comprehensive Survey on Graph Neural Networks." *Journal Of Latex Class Files*, Vol. 32, No. 1, pp. 4-24, August 2019
- [11] M. Kampffmeyer, Y. Chen, X. Liang, H. Wang, Y. Zhang, E.R. Xing, "Rethinking Knowledge Graph Propagation for Zero-Shot Learning." *CVPR* 2019
- [12] Z. Hu, Y. Dong, K. Wang, Y. Sun, Y. "Heterogeneous graph transformer." In: *Proceedings of WWW*, 2020, pp. 2704–2710
- [13] C. Zhang, D. Song, C. Huang, A. Swami, N.V. Chawla, "Heterogeneous graph neural network." In: *Proceedings of KDD.* 2019, pp. 793–803.
- [14] M. Schlichtkrull, T.N. Kipf, P. Bloem, R. van den Berg, I. Titov, M. Welling, "Modeling relational data with graph convolutional networks." In: *Proceedings of ESWC.* Springer, 2018, pp. 593–607.
- [15] D. Beck, G. Haffari, T. Cohn, "Graph-to-sequence learning using gated graph neural network." *Proceedings of ACL*, 2018, pp. 273–283.
- [16] Y. Ma, S. Wang, C.C. Aggarwal, D. Yin, J. Tang, J., "Multi-dimensional graph convolutional networks." In: *Proceedings of SDM*, 2019, pp. 657–665.
- [17] Y. Li, R. Yu, C. Shahabi, Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting." in *Proceedings of the 7th International Conference on Learning Representations*, 2018.
- [18] B. Yu, H. Yin, Z. Zhu, "Spatiotemporal Graph Convolutional Networks: A Deep Learning Framework for Traffic Forecasting. *Proceedings of IJCAI.* 2018, pp. 3634–3640.
- [19] A. Jain, A.R. Zamir, S. Savarese, A. Saxena, "Structural-rnn: deep learning on spatio-temporal graphs." In: *Proceedings of CVPR*, 2016, pp. 5308–5317.
- [20] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition." in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 3634–3640
- [21] F. Manessi, A. Rozza, M. Manzo, "Dynamic graph convolutional networks. *Pattern Recogn.*" *Pattern Recognition Vol. 97* Is. C, Jan 2020.
- [22] A. Pareja, G. Domeniconi, J. Chen, T. Ma, T. Suzumura, H. Kanezashi, T. Kaler, T. Schardl, "Evolvegcn: evolving graph convolutional networks for dynamic graphs." *Proceedings of AAAI* 34, 2020, pp. 5363–5370.
- [23] J. Skarding, B. Gabrys, K. Musial, "Foundations and modelling of dynamic networks using Dynamic Graph Neural Networks: A survey." in *IEEE Access*, vol. 9, pp. 79143-79168, 2021.
- [24] A. Goldenberg, A.X. Zheng, S.E. Fienberg, E.M. Airoldi, "A Survey of Statistical Network Models." *Foundations and Trends® in Machine Learning*, Vol. 2, no. 2, pp. 129–233, February 2010.
- [25] D. Xu, C. Ruan, K. Motwani, E. Korpeoglu, S. Kumar, K. Achan, "Generative Graph Convolutional Network for Growing Graphs." *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2019, pp. 3167–3171,
- [26] T.N. Kipf, M. Welling, "Variational graph auto-encoders." In: *NIPS Bayesian Deep Learning Workshop.CoRR*, 2016.

- [27] T.N. Kipf, M. Welling, “Semi-supervised classification with graph convolutional networks.” In 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings. OpenReview.net, 2017.
- [28] M. Ou, P. Cui, J. Pei, Z. Zhang, W. Zhu, “Asymmetric transitivity preserving graph embedding.” in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 1105–1114.
- [29] X. Shi, Z. Chen, H. Wang, D.Y. Yeung, W.K. Wong, W.C. Woo, “Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting.” *NeurIPS* 2015.
- [30] X. Shi, Z. Gao, L. Lausen, H. Wang, D.Y. Yeung, E.K. Wong, W.C. Woo, “Deep learning for precipitation nowcasting: A benchmark and A new model.” In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing, 2017*, pp. 5617–5627.
- [31] Y. Wang, M. Long, J. Wang, Z. Gao, P.S. Yu, “PredRNN: Recurrent neural networks for predictive learning using spatiotemporal LSTMs.” *Advances in Neural Information Processing Systems. 2017*, pp. 880–889.
- [32] Y. Wang, M. Long, J. Wang, Z. Gao, P.S. Yu, “Predrnn++: Towards A resolution of the deep-in-time dilemma in spatiotemporal predictive learning.” In Jennifer G. Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning. ICML 2018, Stockholm, Sweden, July 10-15, 2018*, vol. 80 of *Proceedings of Machine Learning Research*, pages 5110–5119. PMLR, 2018.
- [33] J. Xu, H. Xu, B. Ni, X. Yang, T. Darrell, “Video prediction via example guidance.” In Hal Daume III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, PMLR, Jul 2020, pp. 10628–10637.
- [34] O. Ronneberger, P. Fischer, T. Brox, “U-net: Convolutional networks for biomedical image segmentation.” *CoRR*, 2015.
- [35] Y. Aoyagi, N. Murata, H. Sakaino, “Spatio-temporal Predictive Network For Videos With Physical Properties.” *CVPR2021, 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2021, pp. 2268-2278.
- [36] Y. Seo, M. Defferrard, P. Vandergheynst, X. Bresson, “Structured Sequence Modeling with Graph Convolutional Recurrent Networks.” In *Neural Information Processing, Lecture Notes in Computer Science*, 2018, pp. 362–373.
- [37] J. Deng, W. Dong, R. Socher, L. Li, K. Li, F. Li, “Imagenet: A large-scale hierarchical image database.” In 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248–255.
- [38] M. Monfort, B. Zhou, S.A. Bargal, A. Andonian, T. Yan, K. Ramakrishnan, L.M. Brown, Q. Fan, D. Gutfreund, C. Vondrick, A. Oliva, “Moments in time dataset: one million videos for event understanding.” *CoRR*, 2018.
- [39] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, A. Torralba, “Places: A 10 million image database for scene recognition.” *IEEE Transactions on Pattern Analysis and Machine Intelligence*. Vol. 40 no. 6, pp. 1452–1464, 2018.
- [40] P. Goyal, S.R. Chhetri, N.S. Mehrabi, E. Ferrara, A. Canedo, “DynamicGEM: A library for dynamic graph embedding methods.” *arXiv preprint arXiv:1811.10734*, 2018.
- [41] Y. Ma, Z. Guo, Z. Ren, E. Zhao, J. Tang, D. Yin, “Streaming Graph Neural Networks.” *Arxiv*, October 2018.
- [42] S. Kumar, X. Zhang, J. Leskovec, “Predicting dynamic embedding trajectory in temporal interaction networks.” In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019*, pp. 1269–1278.
- [43] R. Trivedi, H. Dai, Y. Wang, L. Song, “Know-evolve: Deep temporal reasoning for dynamic knowledge graphs.” In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research, PLMR*, 2017, pp. 3462–3471.
- [44] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, “Attention is All you Need.” In *Advances in Neural Information Processing Systems vol. 30*, pp. 5998–6008, 2017.
- [45] D. Xu, C. Ruan, E. Korpeoglu, S. Kumar, K. Achan, “Inductive representation learning on temporal graphs.” *ArXiv*, 2020.
- [46] Y. Feng, H. You, Z. Zhang, R. Ji, Y. Gao, “Hypergraph neural networks.” In: *Proceedings of AAAI*, vol. 33, pp. 3558–3565. 2018.
- [47] T. Derr Y. Ma J. Tang “Signed graph convolutional networks.” in 2018 IEEE International Conference on Data Mining (ICDM), IEEE, 2018, pp. 929–934.
- [48] S. Pan, R. Hu, G. Long, J. Jiang, L. Yao, C. Zhang, “Adversarially regularized graph autoencoder for graph embedding.” in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, 2018. Proceedings of IJCAI, 2018*. Pp. 2609–2615.
- [49] C. Wang, S. Pan, G. Long, X. Zhu, J. Jiang, “Mgae: marginalized graph autoencoder for graph clustering.” In: *Proceedings of CIKM. 2017*. pp. 889–898.
- [50] J. Park, M. Lee, H.J. Chang, K. Lee, J.Y. Choi, “Symmetric graph convolutional autoencoder for unsupervised graph representation learning.” In: *Proceedings of ICCV, 2019*. pp. 6519–6528.
- [51] G. Cui, J. Zhou, C. Yang, Z. Liu, “Adaptive graph encoder for attributed graph embedding.” In: *Proceedings of KDD, 2020*. pp. 976–985.
- [52] P. Velickovic, W. Fedus, W.L. Hamilton, P. Lio, Y. Bengio, R.D. Hjelm, “Deep Graph Infomax.” *ICLR* 2019.
- [53] F.Y. Sun, J. Hoffmann, V. Verma, J. Tang, “Infograph: unsupervised and semisupervised graph-level representation learning via mutual information maximization.” *Proceedings of ICLR. 2020*.

- [54] K. Hassani, A.H. Khasahmadi, "Contrastive multi-view representation learning on graphs." In: Proceedings of ICML, pp. 4116–4126. 2020.
- [55] K. Xu, C. Li, Y. Tian, T. Sonobe, K. Kawarabayashi, S. Gegelka, "Representation Learning on Graphs with Jumping Knowledge Networks." In: Proceeding of ICML. 2018. pp. 5449–5458.
- [56] J.G. Zilly, R.K. Srivastava, J. Koutnik, J. Schmidhuber, "Recurrent highway networks." In: Proceedings of ICML. 2016, pp. 4189–4198.
- [57] A. Rahimi, T. Cohn, and T. Baldwin, "Semi-supervised User Geolocation via Graph Convolutional Networks." In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, vol 1, pp. 2009–2019, 2018.
- [58] T. Pham, T. Tran, D. Phung, S. Venkatesh, "Column networks for collective classification." In: Proceedings of AAAI, 2017. pp. 2485–2491.
- [59] G. Li, M. Muller, A. Thabet, B. Ghanem, "DeepGCNs: Can GCNs Go as Deep as CNNs?." ICCV 2019 In: Proceedings of ICCV, 2019, pp. 9267–9276.
- [60] K. He, X. Zhang, S. Ren, J. Sun, "Deep Residual Learning for Image Recognition." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.
- [61] K. He, X. Zhang, S. Ren, J. Sun, "Identity Mappings in Deep Residual Networks." European Conference on Computer Vision, ECCV 2016: Computer Vision – ECCV 2016, pp 630–645.
- [62] G. Huang, Z. Liu, L. Van Der Maaten, K.Q. Weinberger, "Densely connected convolutional networks." In: Proceedings of CVPR, 2016, pp. 4700–4708.
- [63] Y. Pei, T. Huang, W.V. Ipenburg, M. Pechenizkiy, "ResGCN: Attention-based Deep Residual Modeling for Anomaly Detection on Attributed Networks." Mach Learn vol. 111, pp. 519–541, 2022.
- [64] R.N. Abirami, P.M.D.R. Vincent, K. Srinivasan, U. Tariq, C.Y. Chang, "Deep CNN and Deep GAN in Computational Visual Perception-Driven Image Analysis." Complexity, cilt 2021, Article ID 5541134, 30 sayfa, 2021.
- [65] W. Maass, T. Natschlager, H. Markram, "Real-time computing without stable states: A new framework for neural computation based on perturbations." Neural Computation, vol. 14, no. 11, pp. 2531–2560, 2002.
- [66] H. Jaeger, The "Echo State." Approach to Analysing and Training Recurrent Neural Networks-With an Erratum Note, German National Research Center for Information Technology GMD Technical Report, vol. 148. p. 13. 2001.
- [67] C. Gallicchio, A. Micheli, "Graph Echo State Networks." The 2010 International Joint Conference on Neural Networks (IJCNN), 2010, pp. 1-8.
- [68] H. Dai, Z. Kozareva, B. Dai, A. Smola, L. Song, "Learning Steady-States of Iterative Algorithms Over Graphs." 35th International Conference on Machine Learning, PMLR 80, 2018, pp. 1106–1114.
- [69] M. Tiezzi, G.M. Dimitri, M. Bianchini, M. Maggini, F. Scarselli, "Inductive–transductive learning with graph neural networks." In: IAPR Workshop on Artificial Neural Networks in Pattern Recognition. Springer. pp. 201–212. 2018.
- [70] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, "Learning phrase representations using rnn encoder–decoder for statistical machine translation." Proceedings of EMNLP, 2014, pp. 1724–1734.
- [71] Y. Gao, D. Glowacka, "Deep Gate Recurrent Neural Network." JMLR: Workshop and Conference Proceedings, 2016, pp.350–365.
- [72] J. Cheng, L. Dong, M. Lapata, "Long short-term memory networks for machine reading." In: Proceedings of EMNLP, 2016, pp. 551–561.
- [73] Y. Li, D. Tarlow, M. Brockschmidt, R. Zemel, "Gated graph sequence neural networks." in Proceedings of the 5th International Conference on Learning Representations, 2016, pp. 273–283.
- [74] D. Beck, G. Haffari, T. Cohn, "Graph-to-sequence learning using gated graph neural network." Proceedings of ACL, 2018, 273–283.
- [75] J Guo, "BackPropagation Through Time." 2013.
- [76] V. Ranganathan, S. Natarajan, "A New Backpropagation Algorithm without Gradient Descent." Arxiv. 2018.
- [77] S. Ruder, "An overview of gradient descent optimization algorithms." Arxiv. 2016.
- [78] K.S. Tai, R. Socher, C.D. Manning, "Improved semantic representations from treestructured long short-term memory networks." In: Proceeding of IJCNLP, 2015, pp. 1556–1566.
- [79] R. Socher, A. Perelygin, J.Y. Wu, J. Chuang, C.D. Manning, A.Y. Ng, C. Potts, "Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank." Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, 2103, pp. 1631–1642.
- [80] M. Marelli, L. Bentivogli, M. Baroni, R. Bernardi, S. Menini, R. Zamparelli, "SemEval-2014 Task 1: Evaluation of Compositional Distributional Semantic Models on Full Sentences through Semantic Relatedness and Textual Entailment." Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), 2014, pp. 1–8.
- [81] N. Peng, H. Poon, C. Quirk, K. Toutanova, W.T. Yih, "Crosssentence N-ary relation extraction with graph LSTMs." Trans. Assoc. Comput. Linguistics, vol. 5, pp. 101–115, Dec. 2017.
- [82] X. Liang, X. Shen, J. Feng, L. Lin, S. Yan, "Semantic object parsing with graph lstm." In: Proceedings of ECCV, 2016, pp. 125–143.
- [83] V. Zayats, M. Ostendorf, "Conversation modeling on reddit using a graph-structured LSTM." Trans. Assoc. Comput. Linguistics, TACL. vol. 6, pp. 121–132, 2018.
- [84] Y. Zhang, Q. Liu, L. Song, "Sentence-state LSTM for text representation." In: Proceedings of ACL. Vol. 1, pp. 317–327, 2018.
- [85] Z. Huang, X. Xu, K. Yu, "Bidirectional LSTM-CRF Models for Sequence Tagging." ArXiv, 2015.
- [86] Y. Shulman, "SimPool: Towards Topology Based Graph Pooling with Structural Similarity Features."



- ArXiv. 2020.
- [87] S. Mallat, “A Wavelet Tour of Signal Processing.” Elsevier. 1999.
- [88] J. Bruna, W. Zaremba, A. Szlam, Y. LeCun, “Spectral networks and locally connected networks on graphs.” in 2nd Int. Conf. Learn. Repres., Banff, AB, Canada 2014.
- [89] M. Henaff, J. Bruna, Y. Lecun, “Deep Convolutional Networks on Graph-Structured Data.” ArXiv. 2015.
- [90] D.K. Hammond, P. Vandergheynst, R. Gribonval, “Wavelets on graphs via spectral graph theory.” Appl. Comput. Harmon. Anal. Vol. 30, pp. 129–150, 2011.
- [91] H.P. Matic, P. Frossard, “Graph Laplacian mixture model.” Arxiv, 2020.
- [92] M. Defferrard, X. Bresson, P. Vandergheynst, “Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering.” NeurIPS, 2016, pp. 3844–3852.
- [93] B. Li, S. Tang, H. Yu, “Better Approximations of High Dimensional Smooth Functions by Deep Neural Networks with Rectified Power Units.” Communications in Computational Physics vol. 27, no. 2, pp. 379–411, 2020.
- [94] S. Tang, B. Li, H. Yu, “ChebNet: Efficient and Stable Constructions of Deep Neural Networks with Rectified Power Units using Chebyshev Approximations.” ArXiv, 2019.
- [95] R. Li S. Wang F. Zhu J. Huang “Adaptive graph convolutional neural networks.” In Thirty-Second AAAI Conference on Artificial Intelligence, 2018, pp. 2546-3553.
- [96] Y. Wang, Y. Sun, Z. Liu, E. Sarma, M.M. Bronstein, J.M. Solomon, “Dynamic Graph CNN for Learning on Point Clouds.” ACM Transactions on Graphics, Vol. 38, no. 146, pp 1–12, 2019.
- [97] R. Li S. Wang F. Zhu J. Huang, “Adaptive graph convolutional neural networks.” In Thirty-Second AAAI Conference on Artificial Intelligence, 2018, pp. 2546-3553.
- [98] C. Zhuang Q. Ma, “Dual graph convolutional networks for graphbased semi-supervised classification.” in Proc. Web Conf., 2018, pp. 499–508.
- [99] B. Xu, H. Shen, Q. Cao, Y. Qiu, X. Cheng, “Graph wavelet neural network.” In: Proceedings of ICLR. 2019.
- [100] D.K. Duvenaud, D. Maclaurin, J. Aguileraiparraguirre, R. Gomezbombarelli, T.D. Hirzel, A. Aspurguzik, R.P. Adams, “Convolutional networks on graphs for learning molecular fingerprints.” In: Proceedings of NIPS, Neural Inf. Process. Syst., 2015, pp. 2224–2232.
- [101] J. Atwood, D. Towsley, “Diffusion-Convolutional Neural Networks.” 29th Conference on Neural Information Processing Systems (NIPS 2016), 2016, pp. 1993–2001.
- [102] M. Niepert, M. Ahmed, K. Kutzkov, “Learning convolutional neural networks for graphs.” in Proc. Int. Conf. Mach. Learn., 2016, pp. 2014– 2023.
- [103] H. Gao, Z. Wang S. Ji “Large-scale learnable graph convolutional networks.” in Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, ACM, 2018, 1416–1424.
- [104] W.L. Hamilton, R. Ying, L. Leskovec, “Inductive representation learning on large graphs.” In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS’17). 2017, pp. 1025–1035.
- [105] J. Chen, J. Zhu, L. Song, “Stochastic Training of Graph Convolutional Networks with Variance Reduction” Proceedings of the 35 th International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018.
- [106] R. Ying, R. He, K. Chen, P. Eksombatchai, W.L. Hamilton, J. Leskovec, “Graph convolutional neural networks for web-scale recommender systems.” in Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining. 2018. Pp. 974–983.
- [107] J. Chen, T. Ma, C. Xiao, “FastGCN: Fast Learning with Graph Convolutional Networks via Importance Sampling.” ICLR, 2018.
- [108] W. Huang, T. Zhang, Y. Rong, J. Huang, “Adaptive sampling towards fast graph representation learning.” Proceedings of NeurIPS, 2018, pp. 4558–4567.
- [109] D. Zou, Z. Hu, Y. Wang, S. Jiang, Y. Sun, Q. Gu, “Layer-dependent importance sampling for training deep and large graph convolutional networks.” In: Proceedings of NeurIPS, 2019, pp. 11249–11259.
- [110] W.L. Chiang, X. Liu, S. Si, Y. Li, S. Bengio, C.J. Hsieh, “Cluster-GCN: An Efficient Algorithm for Training Deep and Large Graph Convolutional Networks.” In The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD ’19), 2019, pp. 257–266.
- [111] H. Zeng, H. Zhou, A. Srivastava, R. Kannan, V.K. Prasanna, “Graphsaint: graph sampling based inductive learning method” In: Proceedings of ICLR. 2020.
- [112] O. Vinyals, S. Bengio, M. Kudlur, “Order Matters: Sequence to Sequence for Sets.” ArXiv. 2015.
- [113] M. Zhang, Z. Cui, M. Neumann, Y. Chen, A<sup>n</sup> End-to-End Deep Learning Architecture for Graph Classification” AAAI’18/IAAI’18/ EAAI’18: Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence, 2018, no. 544, pp. 4438–4445.
- [114] I.S. Dhillon, Y. Guan, B. Kulis, “Weighted graph cuts without eigenvectors amultilevel approach.” IEEE TPAMI vol. 29, pp. 1944–1957, 2007.
- [115] M. Simonovsky, N. Komodakis, “Dynamic edge-conditioned filters in convolutional neural networks on graphs.” In: Proceedings of CVPR, 2017. pp. 3693–3702.
- [116] Z. Ying, J. You, C. Morris, X. Ren, W. Hamilton, J. Leskovec, “Hierarchical graph representation learning with differentiable pooling.” Advances in neural information processing systems, NeurIPS, 2018, pp. 4805-4815.
- [117] H. Gao, S. Ji, “Graph u-nets.” In: Proceedings of ICML, 2019, pp. 2083–2092.
- [118] Y. Ma, S. Wang, C.C. Aggarwal, J. Tang, “Graph

- Convolutional Networks with EigenPooling.” In: Proceedings of KDD, 2019, pp. 723–731.
- [119] J. Lee, I. Lee, J. Kang, “Self-attention graph pooling.” ICML, 2019, pp. 3734–3743.
- [120] L., Waikhom, R., Patgiri, “Graph Neural Networks: Methods, Applications, and Opportunities” Arxiv, 2021.
- [121] C., Liu, Y., Zhan, C., Li, B., Du, J., Wu, W., Hu, T., Liu, & D., Tao, “Graph Pooling for Graph Neural Networks: Progress, Challenges, and Opportunities,” ArXiv, 2022, abs/2204.07321.
- [122] S. Georgousis, M. P. Kenning and X. Xie, "Graph Deep Learning: State of the Art and Challenges," in IEEE Access, vol. 9, pp. 22106-22140, 2021, doi: 10.1109/ACCESS.2021.3055280.

## Özgeçmişler



**Hamza Talha Gümüş**, lisans eğitimini Kırıkkale Üniversitesi Elektrik Elektronik Mühendisliği bölümünde tamamladı. Milli Savunma Üniversitesi Siber Güvenlik (Tezsiz) Yüksek Lisans programından Şubat 2023 tarihinde mezun oldu. Adalet ön lisans, Web Tasarım ve Kodlama ön lisans ve Bilgisayar Programcılığı ön lisans programlarından mezuniyeti bulunmaktadır. Ayrıca Coğrafi Bilgi Sistemleri ön lisans, Yönetim Bilişim Sistemleri Lisans ve Siyaset Bilimi ve Uluslararası İlişkiler Lisans programlarına devam etmektedir. Türkiye Hayat ve Emeklilik A.Ş.'de Bilgi Güvenliği Uzman Yardımcısı olarak çalışmaktadır.



**Can Eyüpoğlu**, lisans eğitimini 2012 yılında İstanbul Kültür Üniversitesi Bilgisayar Mühendisliği Bölümü'nde tamamlamıştır. Yüksek lisans ve doktora derecelerini 2014 ve 2018 yıllarında İstanbul Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı'nda almıştır. 2021 yılında Bilgisayar Bilimleri ve Mühendisliği Bilim alanında Doçent unvanı almıştır. Halen Milli Savunma Üniversitesi Hava Harp Okulu Bilgisayar Mühendisliği Bölümü'nde Doçent Bölüm Başkanı olarak görev yapmaktadır. Araştırma alanları arasında yapay zeka, makine öğrenmesi, veri mahremiyeti ve görüntü işleme bulunmaktadır.