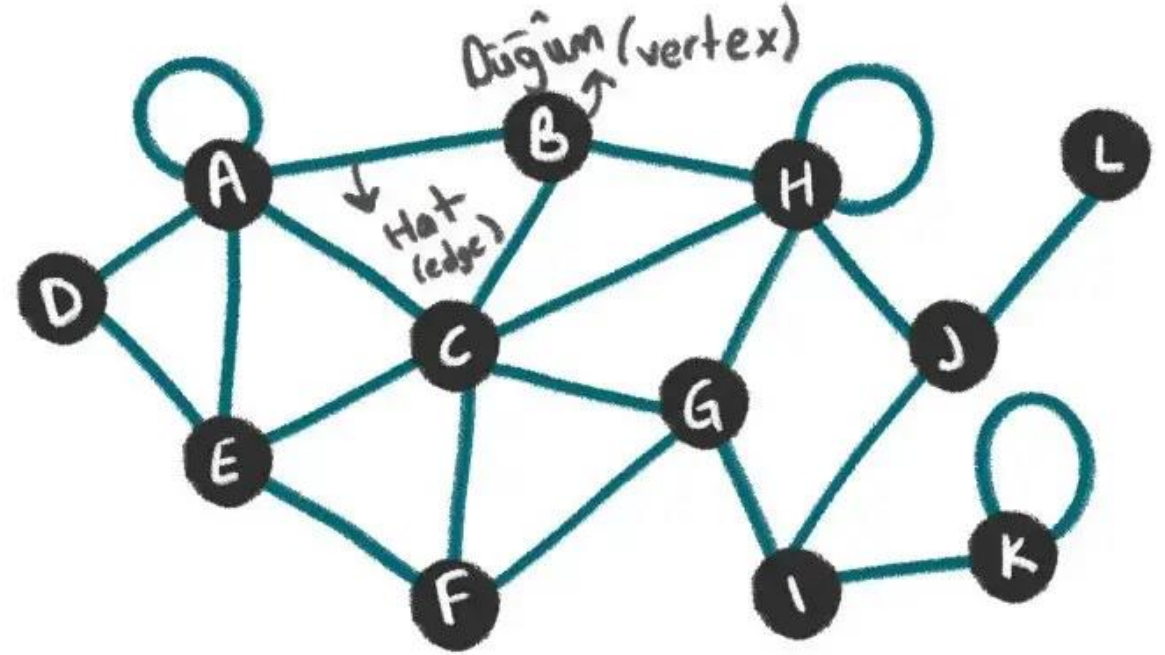


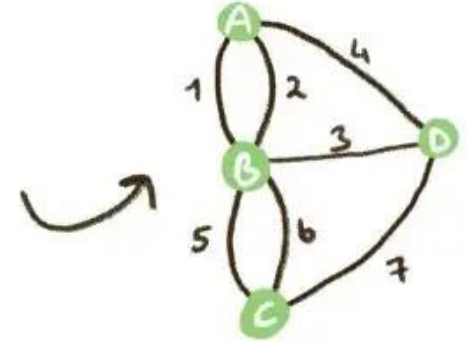
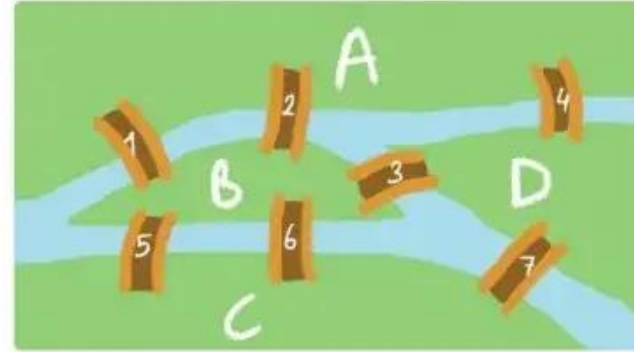
Graph Neural Networks

- Graf nedir'in kısaca tanımını yapacak olursak buna düğümler ve kenarlardan oluşur diyebiliriz.
- Yuvarlak yapılar node (düğüm), düz çizgiler ise edge (kenar) anlamına gelmektedir.
- Gerçek hayatta hemen hemen her şeyi graflar ile modellemek mümkündür.
- Birden fazla varlık ve bu varlıklar arasında ilişki varsa orada bir graf vardır.
- Kenarlar ilişkiyi, düğümler varlıkları gösterir
- Örnek vermek gerekirse bir arkadaş çevresindeki bireyler düğüm olarak belirtilirken arkadaş çevresi içindeki güven ilişkisi kenarlar ile belirtilebilir (kim kime güveniyor).



Graf Teorisi

- Euler'in Köprüler Problemi - Königsberg şehri birbirine 7 köprüyle bağlanan 4 bölgeden oluşmaktadır. Dönemin insanları bu köprülerden yalnızca 1 defa olmak üzere hepsinden geçerek tüm bölgeler ziyaret edilebilir mi diye düşünmüşler.
- Euler, harita üzerinde çalışmak yerine köprüleri ve bölgeleri graf üzerinde göstererek problemi çözmüştür.

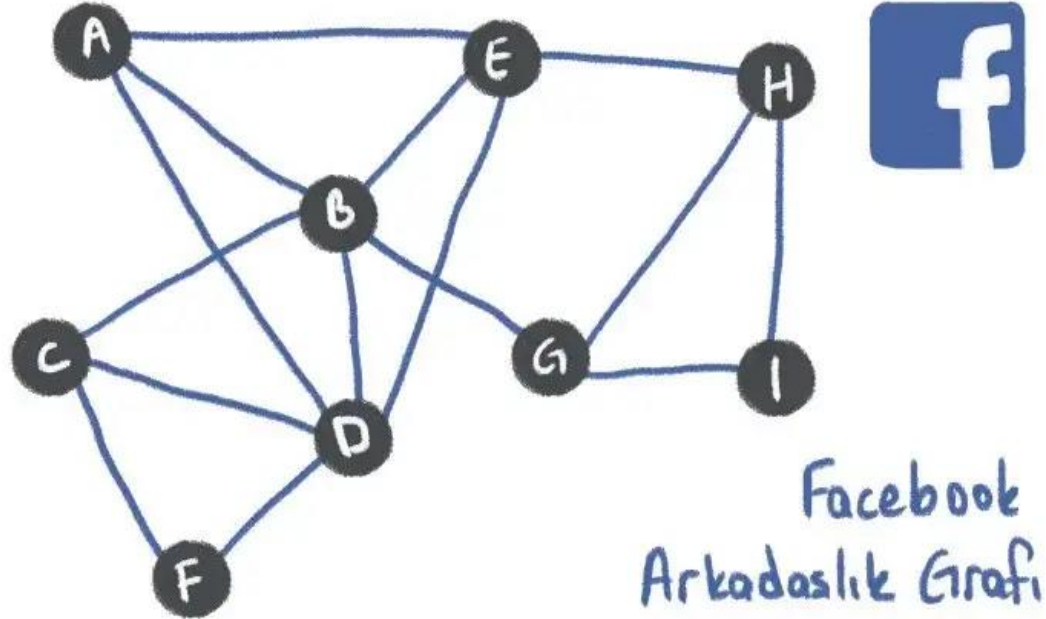


Graf Teorisinin Kullanıldığı Örnekler

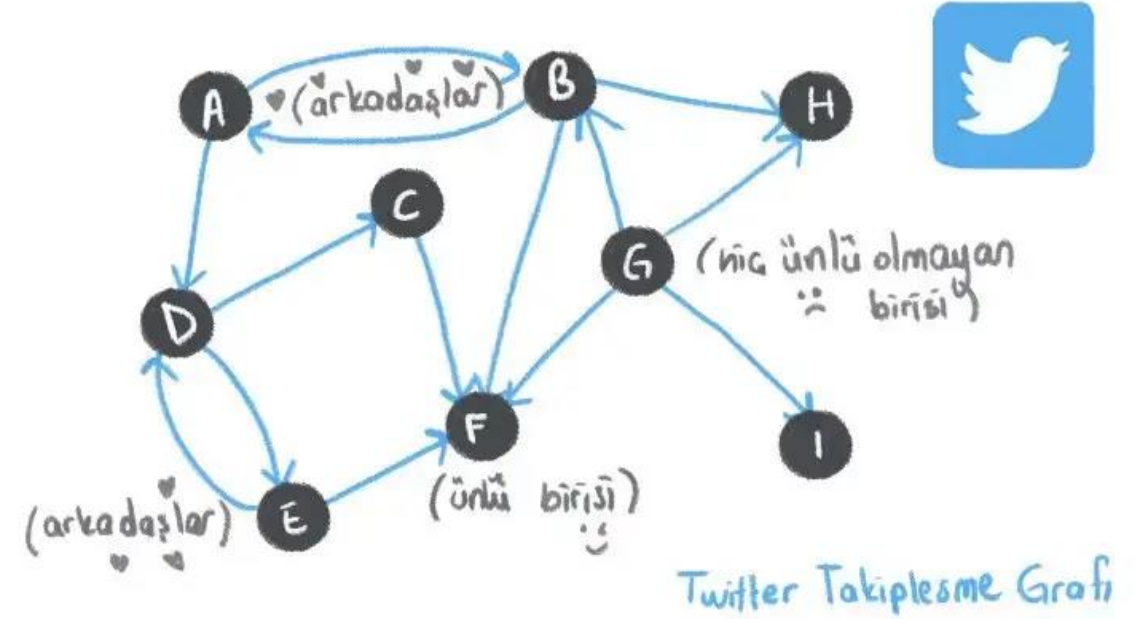
- Seven bridges of königsberg
- Kircoff teorisi
- Harita boyama işlemi
- Taşımacılıkta en kısa yol



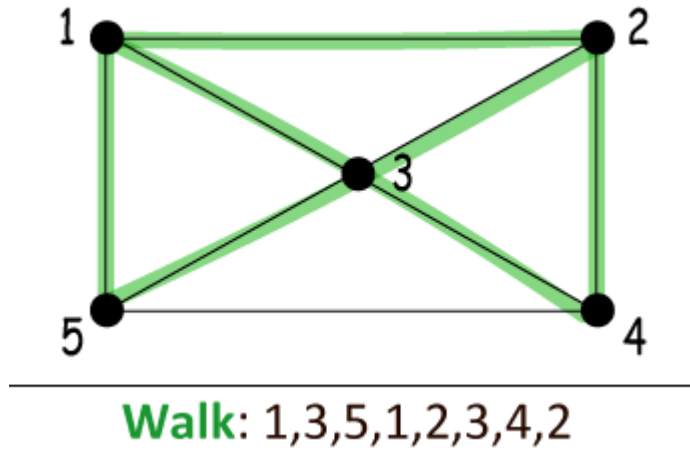
Graf çeşitleri



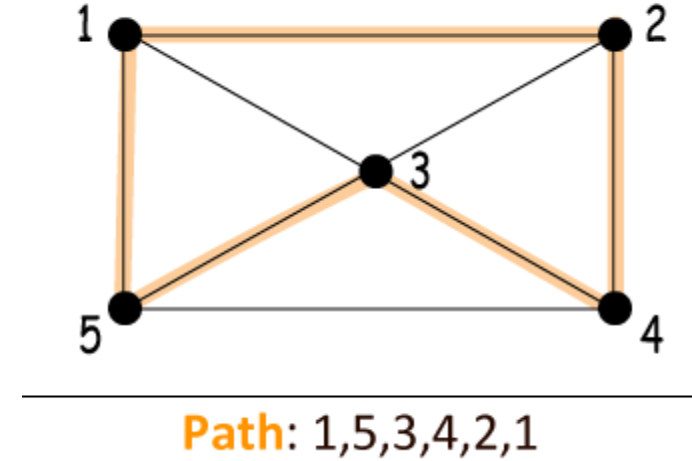
- **Yönsüz Graf (Undirected Graph)**
- Bu graf çeşidinde düğümler arasındaki hatların yönü yoktur. Yönsüz ilişkiye Facebook'daki arkadaşlık ilişkilerini örnek verebiliriz. A kişisi B kişisi ile arkadaş olduğunda B kişisi de A kişisi ile arkadaş olmuş olur.



- **Yönlü Graf (Directed Graf)**
- Bu graf çeşidinde düğümler arası yöne dayalı bir ilişki vardır. A noktası B'ye bağlı diye B de ona bağlıdır anlamını çıkaramayız. Buna Twitter'daki takipleşme ilişkisini örnek gösterebiliriz



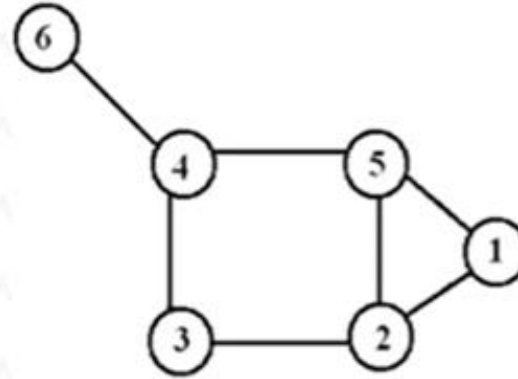
- Yürüyüşte tekrar olabilir.
 - Aynı düğüm ve aynı kenarlardan geçebilirsiniz



- Tekrarı olmayan yürüyüştür
- Farklı kenarlardan geçilmesi gerekiyor
- Aynı düğümlerden geçebilirsiniz fakat aynı kenarlardan geçemezsiniz

Grafları Matris Şeklinde Gösterebiliriz

- İnsidans matrisi: kenar ve düğümleri birlikte tutar
- Komşuluk matrisi: sadece kenarları tutar



	1,2	1,5	2,3	2,5	3,4	4,5	4,6
1	0	0	0	0	0	0	0
2	0	0	1	1	0	0	0
3	0	0	1	0	1	0	0
4	0	0	0	0	1	1	1
5	0	1	0	1	0	1	0
6	0	0	0	0	0	0	1

	1	2	3	4	5	6
1	0	1	0	0	1	0
2	1	0	1	0	1	0
3	0	1	0	1	0	0
4	0	0	1	0	1	1
5	1	1	0	1	0	0
6	0	0	0	1	0	0

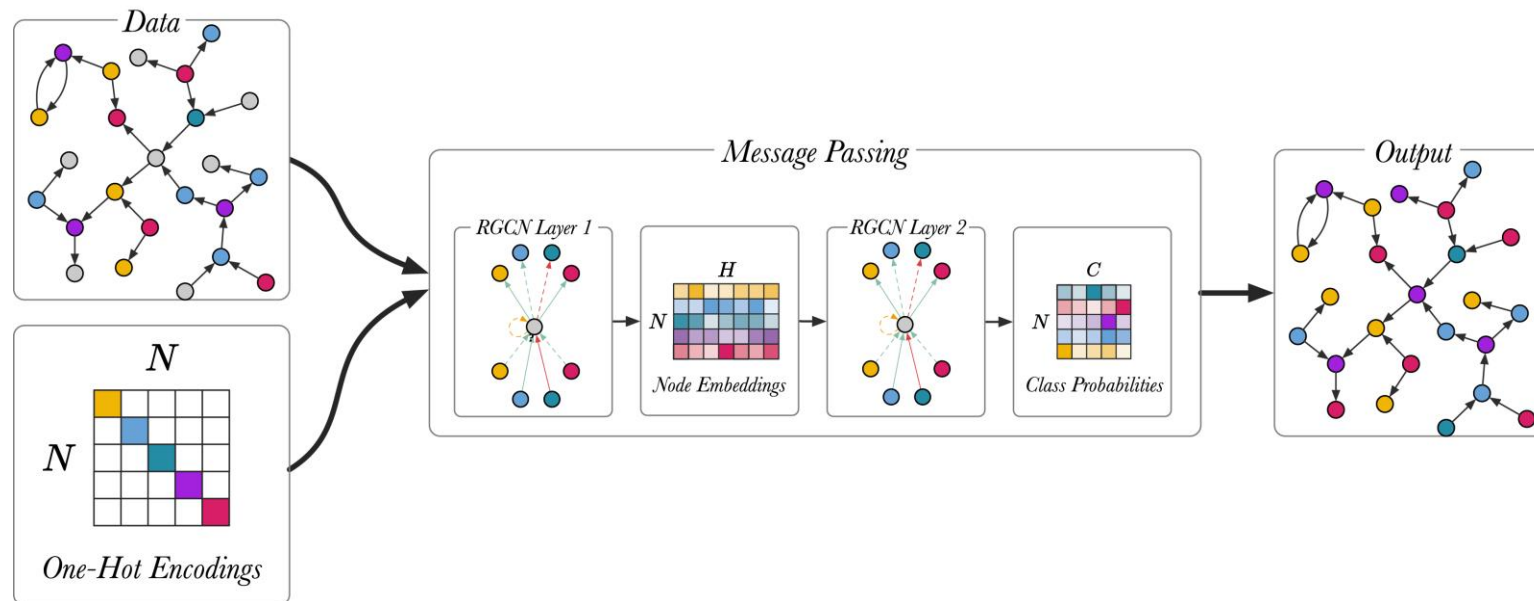
Graph Neural Networks'e Genel Bakış

GNN kavramları:

Message passing

Node Embedding

Forward / Back Propagation

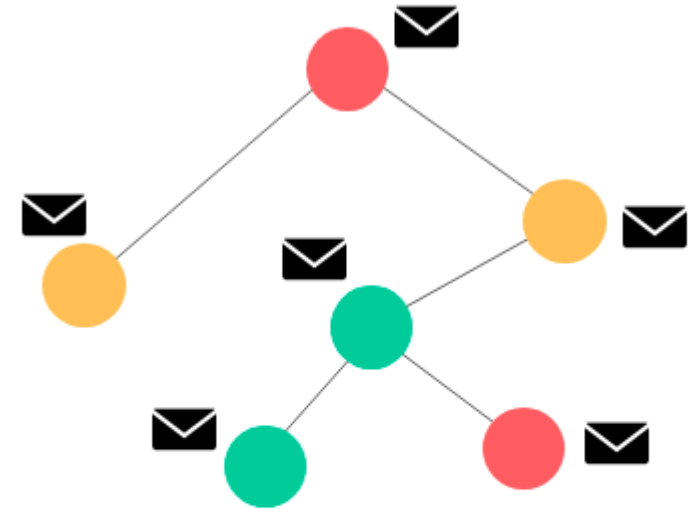


Message Passing

Message passing GNN'in en önemli yapıtaşıdır.

Her message passing yapıldığında node'lar ve edge'ler hakkında bilgi edinilir.

Node'ların durumları, komşu node'ların özellikleri ve aralarındaki edge'ler kullanılarak hesaplanır. Yeni node durumları, aslında yeni bir graph oluşturur.

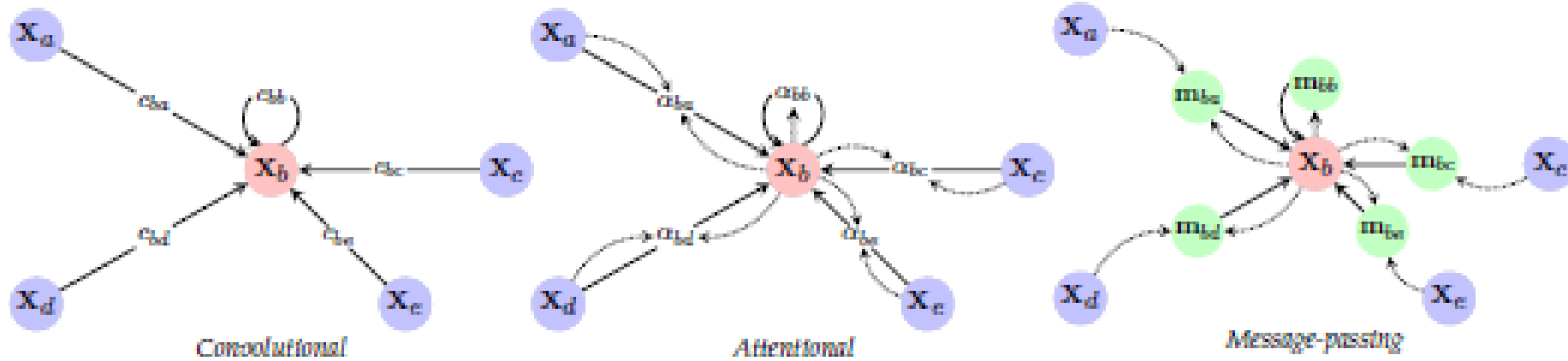


$$H = \sum \left(\begin{matrix} \text{envelope} & \text{envelope} \\ \text{envelope} & \text{envelope} \\ \text{envelope} & \text{envelope} \end{matrix} \right)$$

Message Passing

Message passing her layer'da iteratif bir şekilde yapılır ve böylece önce en yakın komşular hakkında bilgi alınırken sonraki iterasyonda komşuların komşuları hakkında bilgi alınır.

Bir GNN'de ne kadar layer olduğu böylece en uzak hangi komşular hakkında bilgi alabileceğimizi söyler.



Message Passing

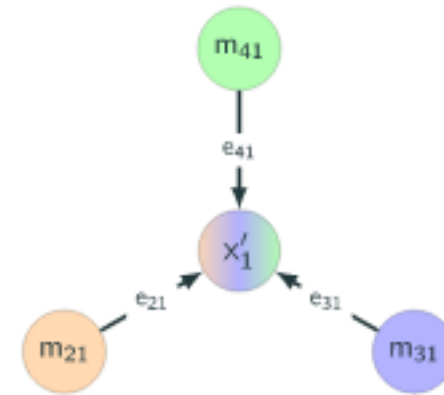
Message passing aslında her Neural Network işlevi gibi bir fonksiyon. Her node'un komşu node'ları ile olan ilişkisini belirten bir fonksiyon.

Message passing işlevi de Node Embedding'i sağlayan kısım aslında.

A general scheme for message-passing networks:

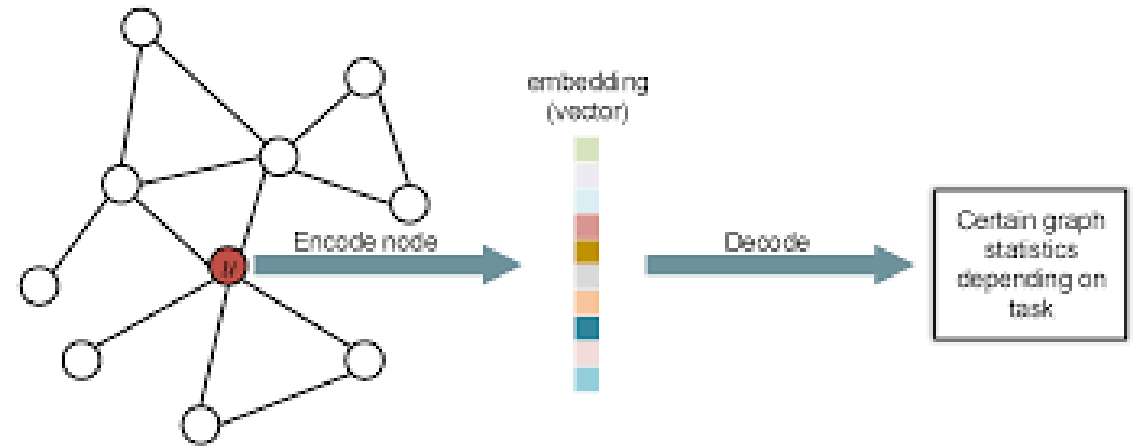
$$x'_i = \gamma \left(x_i, \square_{j \in \mathcal{N}(i)} \phi(x_i, x_j, e_{ji}) \right),$$

- ϕ : **message function**, depends on x_i , x_j and possibly the edge attribute e_{ji} (we call messages m_{ji});
- $\square_{j \in \mathcal{N}(i)}$: **aggregation function** (sum, average, max, or something else...);
- γ : **update function**, final transformation to obtain new attributes after aggregating messages.



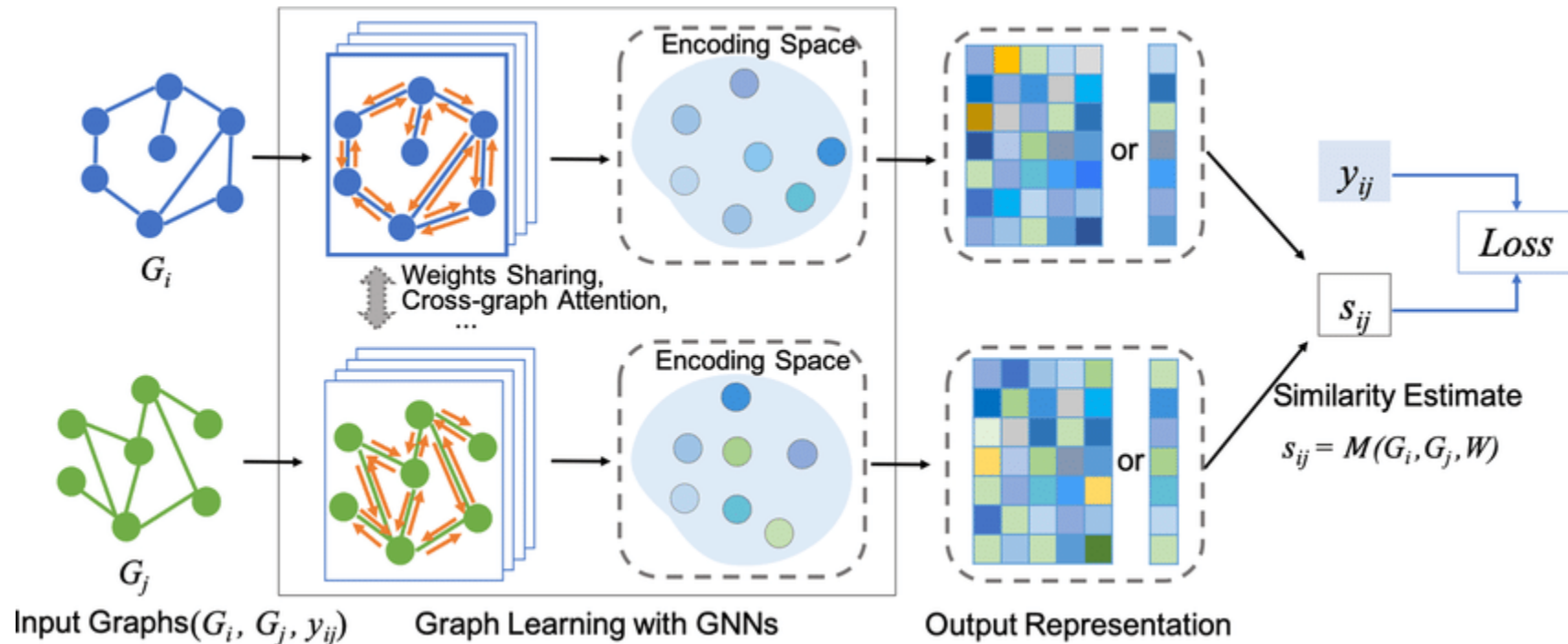
Node Embedding

- Graph'ın numerik veri haline getirilmesi.
- Node Embeddings sayesinde her node aslında diğer node'lar hakkında bilgi ediniyor. Burada node'ların özellikleri (features), aralarındaki ilişkiler (edges) kullanılarak graph Embedding Space'e taşınır.

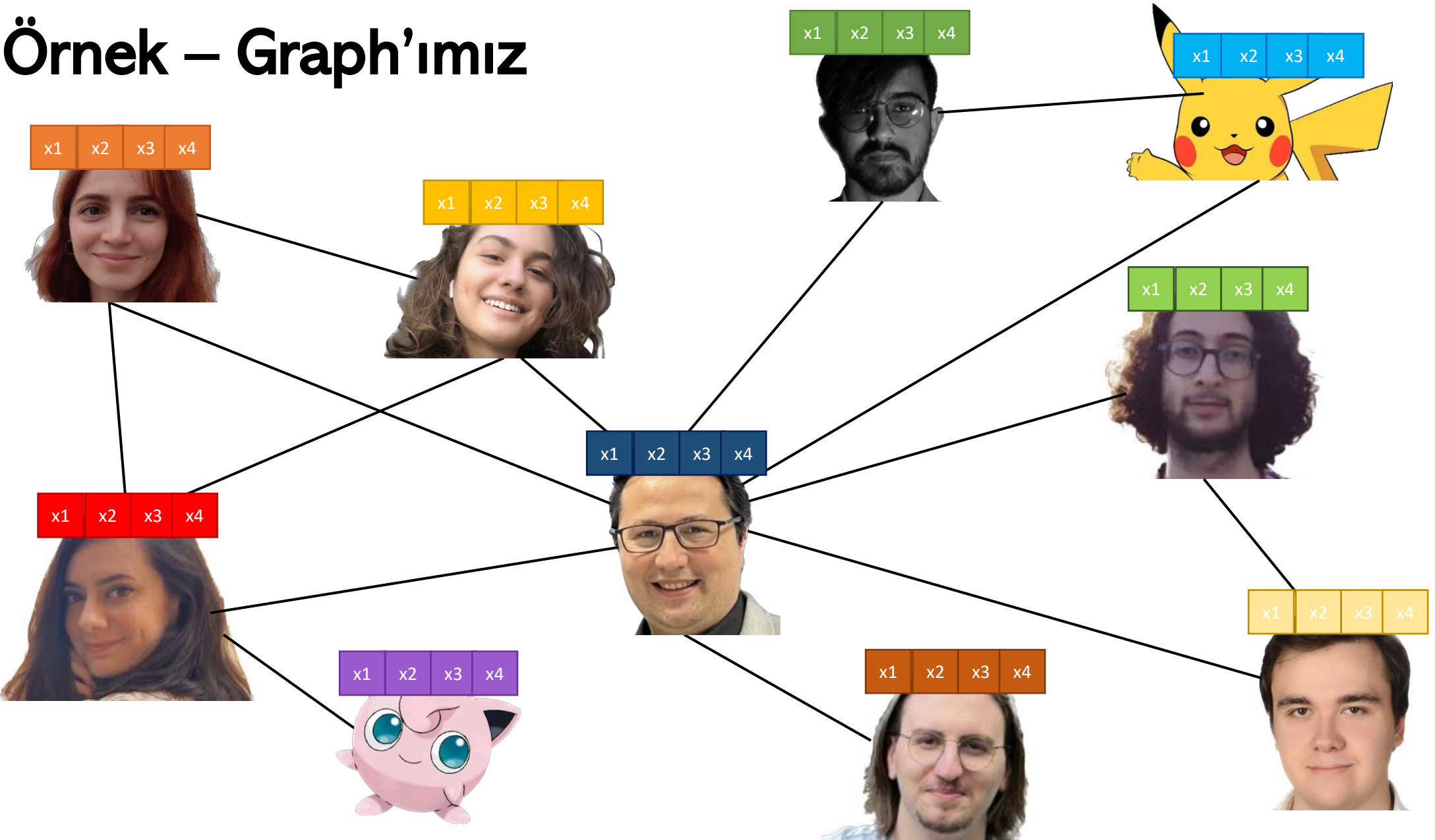


Node Embedding

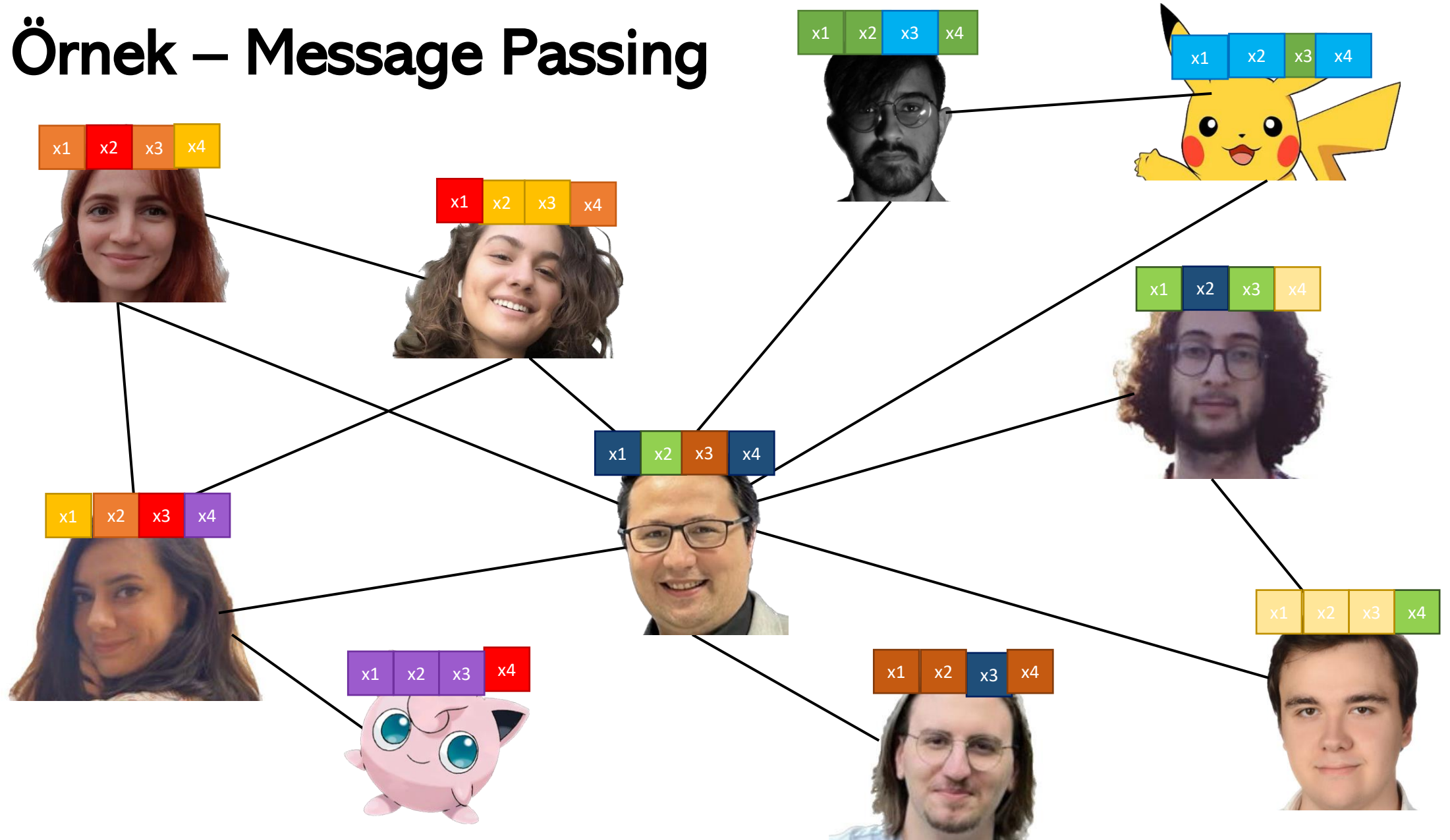
- Node Embedding yapıldıktan ve graph Embedding Space'e taşındıktan sonra diğer Neural Network yapılarında olduğu gibi back propagation ile message passing parametreleri loss function'a göre düzenlenir.



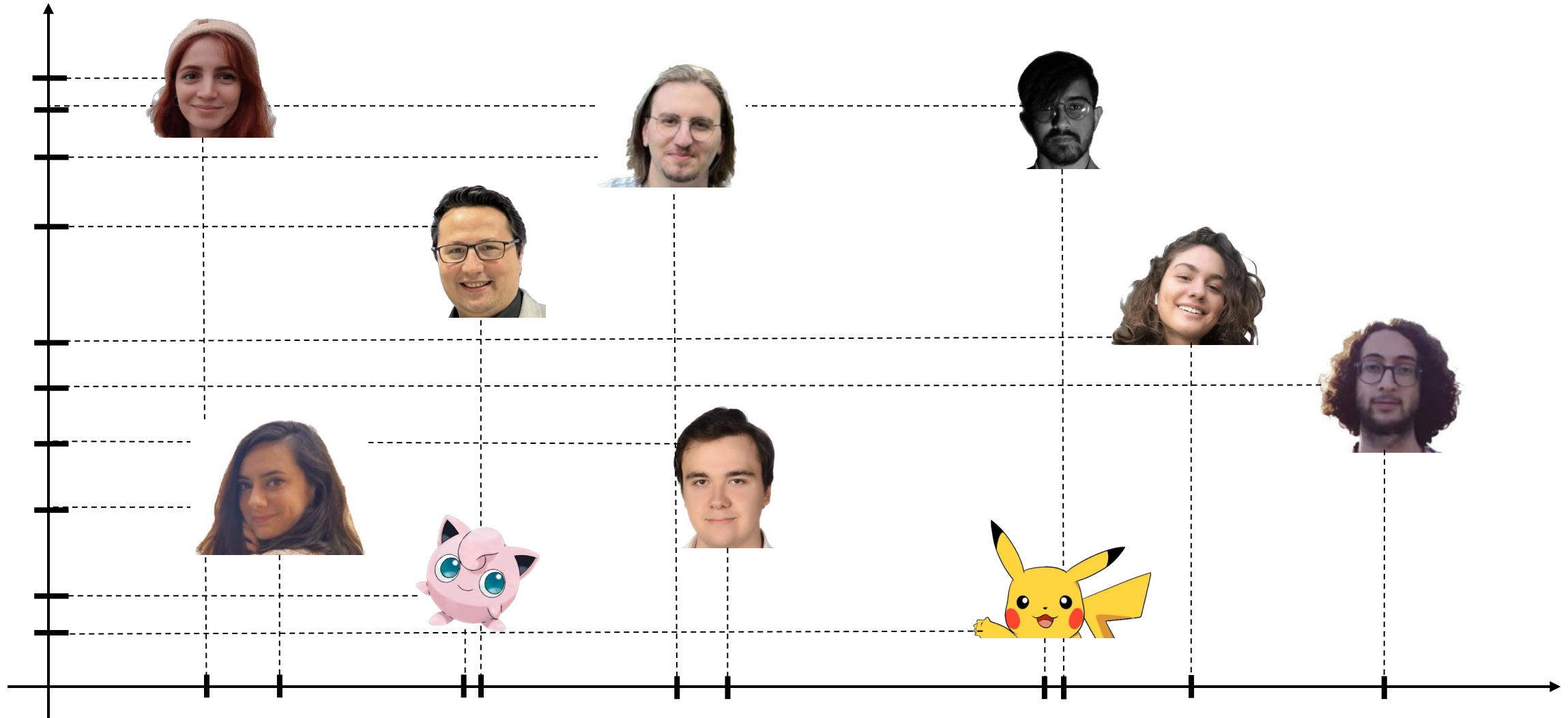
Örnek – Graph'ımız



Örnek – Message Passing



Örnek – Node Embeddings

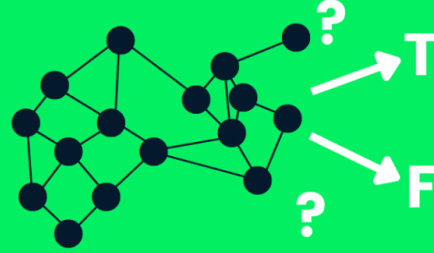


GNN Türleri

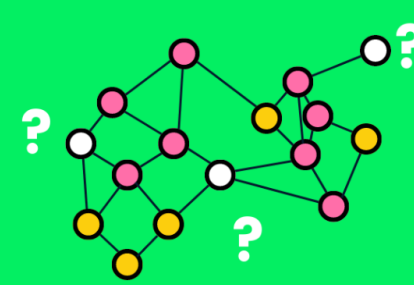
GNN türleri de graph'ları kullanarak hangi bilgiyi elde etmek istediğimize bağlı olarak şekilleniyor:

- Link Prediction
- Node Classification
- Clustering
- Graph Classification

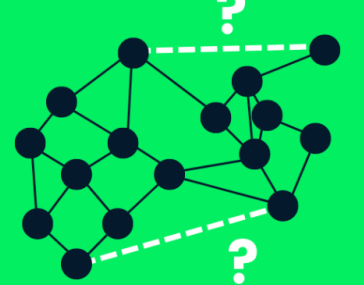
Graph Classification



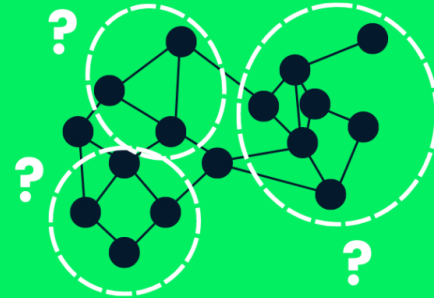
Node Classification



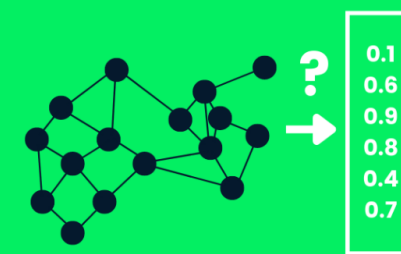
Link Prediction



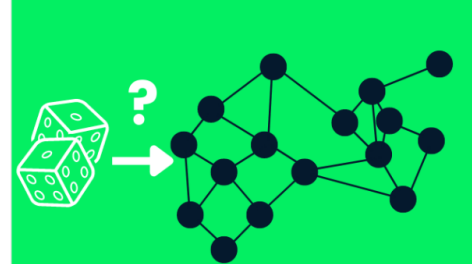
Community Detection



Graph Embedding



Graph Generation



GNN'in Farkı Nedir?

GNN'in en önemli farkı graphları kullandığı için kompleks ilişkileri modelleyebilmesi. Bu geleneksel Neural Network'lerin zorlandığı bir durum.

Geleneksel Neural Network'ler sayılar, metinler ve resimler gibi yapılandırılmış verileri kullanırken graph'larla gösterilen veriler yapılandırılmamıştır, boyutları önceden belli değildir.

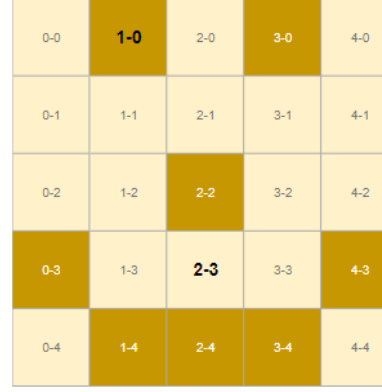
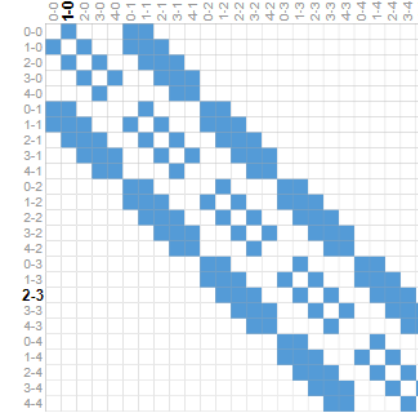
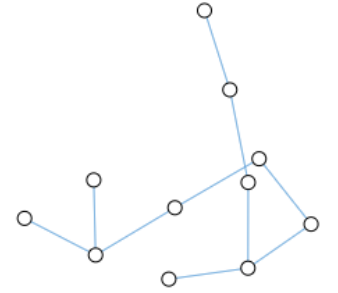
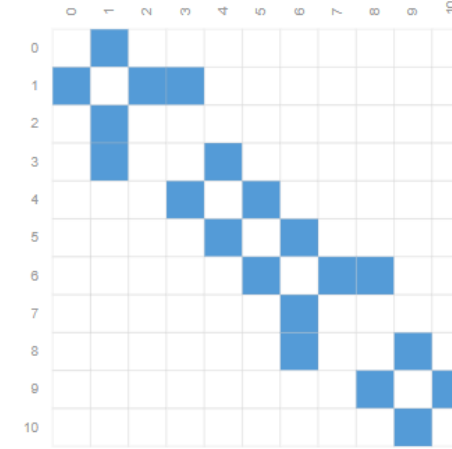
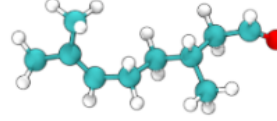
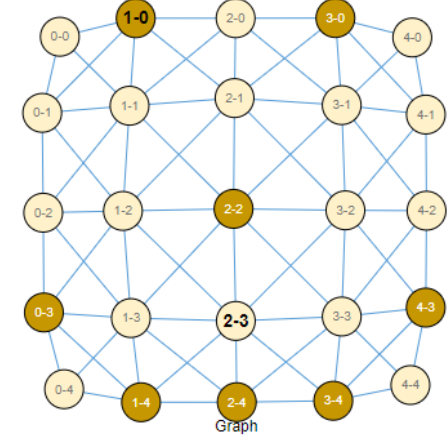


Image Pixels



Adjacency Matrix



GNN Nerelerde Kullanılıyor?

- Improve drug discovery



- Recommendation systems

NETFLIX



- Social networks



- Fraud/Anomaly detection

amazon

Amazon 2017'de fraud detection için GNN kullanacağını açıklamış. 2020 de fraud detection için herkese açık bir GNN servisi koymuşlar.

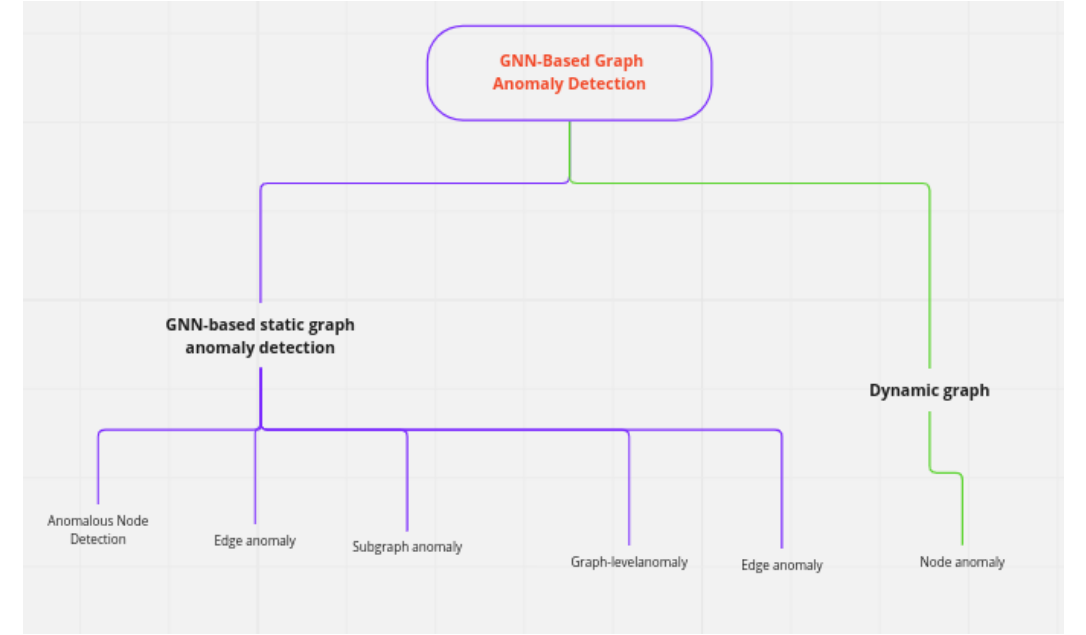
Anomaly Detection için GNN

Anomaly Detection (Anomali Tespiti), en basit anlamıyla bir veride beklenmedik durumların veya kalıpların bulunmasını sağlayan bir tekniktir.

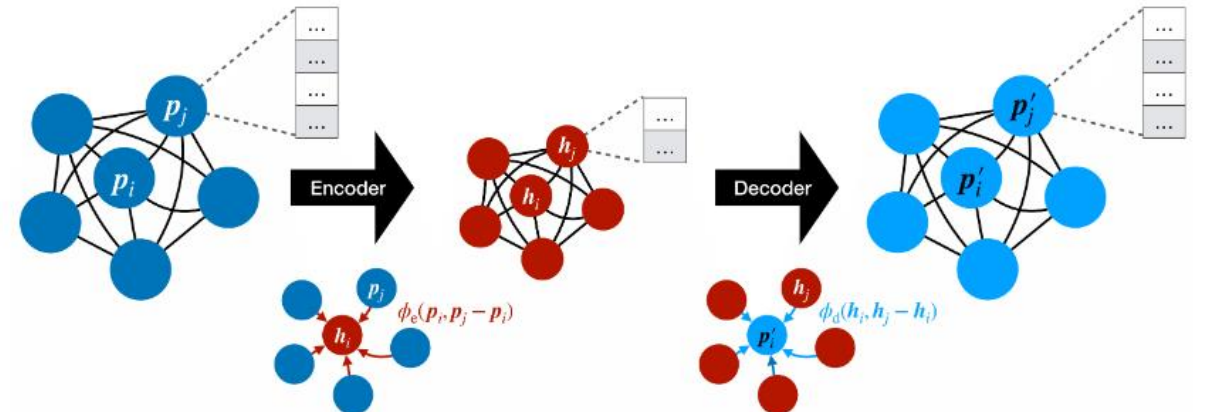
Bu beklenmedik durumlara literatürde outliers (aykırı değerler), exceptions (istisnai durumlar) veya anomaliler denilmektedir.

Anomali Tespitinin Bazı Uygulama Alanları

- Cyber-Intrusion Detection
- Fraud Detection
- Medical Anomaly Detection
- Industrial Damage Detection
- Image Processing
- Textual Anomaly Detection
- Sensor Networks



Particle Graph Autoencoders for Anomaly Detection



Tutorial

colab