**Part1-a**

Bipartite(İki parçalı), ağırlıklı, yönsüz bir user(kullanıcı)-item(ilan) graphı kurdum. User ve item setleri ayrık olduğu için Graph Bipartite.

Düğümler 🡪 user U = client\_id, item I = item\_id

Kenarlar 🡪 Farklı parçalardan düğümler arasında kuruldu.(user-item)

User u bir item i’ye tıklamışsa veya başvurmuşsa, (u,i) arasında bir kenar oluşturdum. Her satır bir etkileşimi temsil ediyor ve aynı (u,i) çiftibirden çok kez geçtiğinde tek kenar olarak tuttup sayıları topladım.

click🡪1, purchase🡪3 değeri verdim. Sebebi purchase daha ağırlıklı bir durum olması.

**Part1-b**

=== Özet Metrikler ===

42k user ve 21k item arasında toplam 96k etkileşim gerçekleşmiş.

kullanıcı sayısı: 42085

ilan sayısı: 21017

kenar sayısı: 96081

Ortalama user yaklaşık 2.3 item bakmış. Bu da çoğu user ın az etkileşim gösterdiğini gösteriyor. Yani (Seyrek bir bağlantı var)

ortalama kullanıcı derecesi: 2.283

Ortalama item is yaklaşık 4.5 user’dan etkileşim almış. Item’ların çoğu da az user tarafından click yada purchase almış.

Yukarıdaki sonuçlar seyrek bir graph olduğunu gösteriyor.

ortalama ilan derecesi: 4.572

bileşen sayısı: 1675

weight\_min: 1.074877

weight\_mean: 1.543639 🡪 Çoğu bağlantının yalnızca tek click veya az sayıda etkileşime girdiğini gösteriyor.

weight\_max: 34.005643 🡪 Bazı user-item ilişkilerinin güçlü olduğunu gösteriyor.

Min ve max arasında bu kadar fark olması verinin dağılımında sıkıntı olduğunu gösteriyor. Veri dengesizdir. “Cold star”t sorununa neden olabilir. Sisteme yeterince veri geldiğinde düzelebilir. Popüler ilanlar öne çıkarken diğer ilanlar kalabilir.

=== Ağırlık İstatistikleri ===

weight\_median: 1.0949833229619041

weight\_p90: 4.312241406584362 🡪 Bağlantıların %90 ı zayıf, %10 luk kısmının yoğun etkileşim olduğunu gösteriyor.

weight\_p99: 4.399618073250005

weight\_std: 1.2068320838826758

**Part2-a**

LightGCN metodunu seçtim. User-Item gibi bipartite(iki parçalı) ve click/purchase gibi verilerin olduğu verilerde öneri sistemi için tasarlanmış hafif bir model. Problem ile uyumlu bir model. Part1-b de gördüğümüz gibi dengesiz verilerle çalışmaya uygun. Eğitimi hızlı, az parametreye sahip olduğu için overfitting riski az. Seyrek Graphlar için uygun.

**Part3**

Yöntem🡪 Cümle düzeyinde semantik gömme (Sentece Embedding)

Model 🡪 sentence-transformers/paraphrase-multilingual-mpnet-base-v2 (SBERT tabanlı, çok dilli)

Semantik benzerlik yapabiliyor. Aynı ilanı anlatan farklı ilanları gömmede birbirine yakınlaştırıyor. Çok dilli Türkçe veri setlerine uygundur. Hızlıdır.

**Part4**

LightGCN ile gerçek kullanıcı davranışlarına dayalı benzerlik, SBERT ile semantik benzerlik kullanılıyor. Bu ikisini birleştirince hem davranışsal hem de anlamsal yakınlık elde ediliyor.

LightGCN popüler ilanları öne çekerken SBERT bunu dengeliyor.