

Topluluk Tespiti: Klasik Algoritmalar ve Çizge Sinir Ağları Karşılaştırması

Emre YILDIZ

Ege Üniversitesi
Bilgisayar Mühendisliği Bölümü

June 19, 2025

- **Ders:** Cebirsel Çizge Algoritmaları (Yüksek Lisans)
- **Amaç:** Çizge Teorisi'nin teorik temellerini modern Makine Öğrenmesi uygulamaları ile birleştiren kapsamlı bir topluluk tespiti (community detection) çalışmasıdır.
- **Yöntemler:** Neo4j ve PyTorch kullanarak klasik algoritmaları (Louvain, LPA) Graph Neural Networks (GCN, GraphSAGE, GAT) ile karşılaştırmak.
- **Veriseti:** Cora Citation Network.

Kullanılan Kütüphane ve Araçlar

- **Neo4j (GDS):** Yüksek performanslı çizge veritabanı. Louvain ve Label Propagation gibi klasik topluluk tespit algoritmalarını çalıştırmak için kullanıldı.
- **PyTorch (PyG):** Derin öğrenme ve özellikle Çizge Sinir Ağları (GNN) modelleri oluşturmak ve eğitmek için kullanılan esnek bir kütüphane.
- **NetworkX:** Çizge oluşturma, manipülasyonu ve temel çizge metriklerinin hesaplanması için kullanıldı.
- **Scikit-learn:** GNN modellerinin performansını değerlendirmek için Adjusted Rand Index (ARI) ve Normalized Mutual Information (NMI) gibi metriklerin hesaplanmasında kullanıldı.
- **Seaborn & Matplotlib:** Veri ve sonuçların görselleştirilmesi, karşılaştırma grafikleri oluşturulması için kullanıldı.

Neo4j ve Graph Data Science (GDS) Rolü

Projedeki İşlevi ve İş Akışı

- **Veri Depolama ve Modelleme:** Cora veriseti, Neo4j'de çizge yapısına en uygun şekilde depolanmıştır:
 - Dğümler: :Paper etiketi ile makaleler.
 - İlişkiler: [:CITES] ilişkisi ile makaleler arası alıntılar.
- **In-Memory Çizge Projeksiyonu:** Analizden önce, GDS kütüphanesi ile 'cora-graph' adında bir "in-memory" yansıtma oluşturulur. Bu, algoritmaların disk I/O olmadan çok yüksek hızda çalışmasını sağlar.
- **Klasik Algoritmaların Çalıştırılması:**
 - `gds.louvain.write(...)` ve `gds.labelPropagation.write(...)` fonksiyonları ile topluluk tespiti algoritmaları çalıştırılmıştır.
 - `writeProperty` parametresi sayesinde, her bir algoritmanın bulduğu topluluk kimliği (`louvainCommunityId`, `lpaCommunityId`) doğrudan ilgili düğümlerin bir özelliği olarak veritabanına geri yazılmıştır.
- **Entegrasyon Noktası:** Neo4j, hem klasik algoritmaların analiz merkezi hem de PyTorch ile eğitilen GNN modelleri için bir veri kaynağı görevi görmüştür.

Klasik Algoritmalar

Neo4j GDS ile Uygulama

Louvain Algoritması

Modülerlik skorunu maksimize ederek toplulukları bulmaya çalışan hiyerarşik bir kümeleme algoritmasıdır. İki aşamadan oluşur:

- 1 Her düğüm kendi komşularıyla birleştirilerek modülerlik artışı en üst düzeye çıkarılır.
- 2 Oluşturulan topluluklar tek bir düğüm olarak kabul edilir ve ilk aşama tekrarlanır.

Label Propagation Algorithm (LPA)

Düğümün etiketlerini komşularının en sık rastlanan etiketine göre güncellediği iteratif bir süreçtir. Hızlı ve basittir ancak sonuçları deterministik olmayabilir.

Graph Neural Network (GNN) Modelleri

PyTorch Geometric ile Uygulama

Graph Convolutional Network (GCN)

Komşu düğümlerin özelliklerini toplayarak ve bir sinir ağı katmanından geçirerek düğüm temsillerini öğrenir. Spektral çizge teorisine dayanır.

GraphSAGE (Sample and Aggregate)

Büyük çizgeler için tasarlanmıştır. Her düğüm için komşuluktan sabit sayıda örneklem alır ve bu örneklemelerin bilgilerini birleştirerek (aggregate) düğüm temsili günceller.

Graph Attention Network (GAT)

Komşu düğümlere farklı ağırlıklar atamak için dikkat (attention) mekanizmalarını kullanır. Bu sayede model, hangi komşuların daha önemli olduğuna karar verebilir.

Sonuçlar: Veriseti İstatistikleri

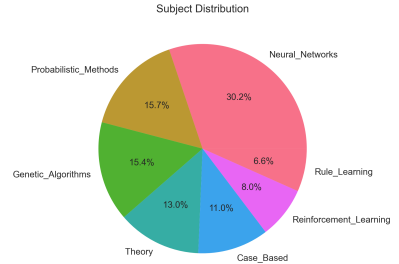
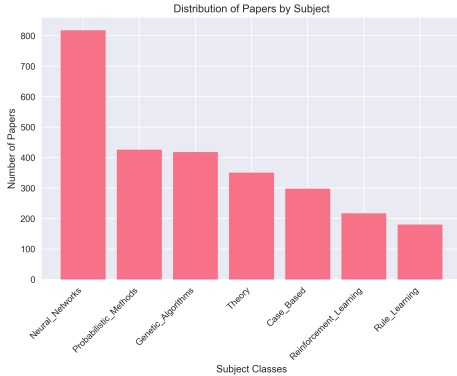
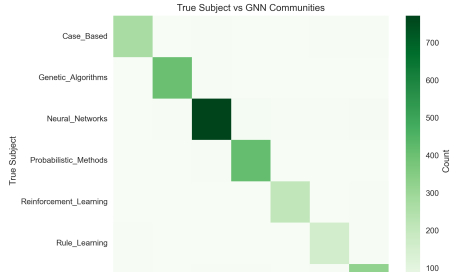
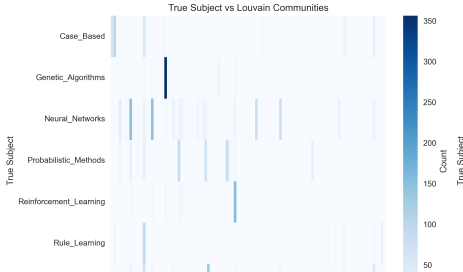
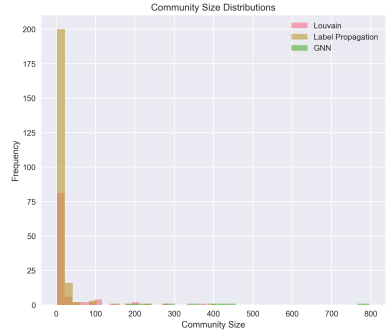
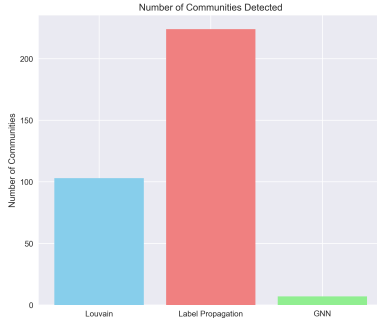


Figure: Cora verisetinin temel istatistikleri ve derece dağılımı.

Sonuçlar: Topluluk Karşılaştırması



Sonuçlar: Değerlendirme Metrikleri

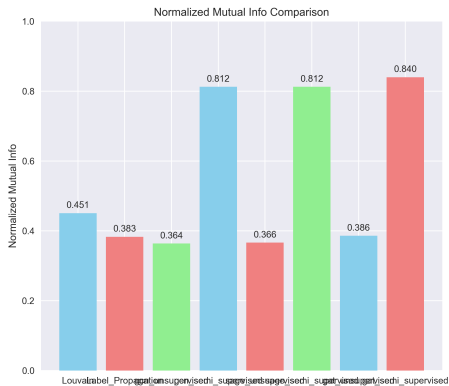
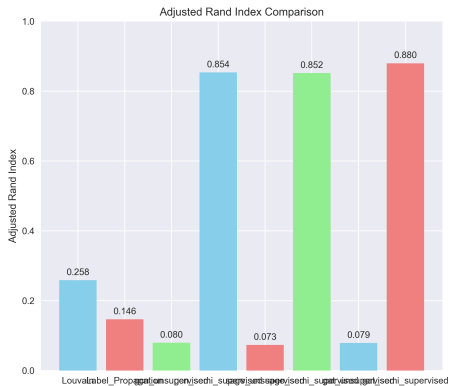


Figure: Algoritmaların ARI, NMI ve Modülerlik metriklerine göre performans karşılaştırması.

Sonuçlar: t-SNE ile Düğüm Gösterimi (Figure 1)

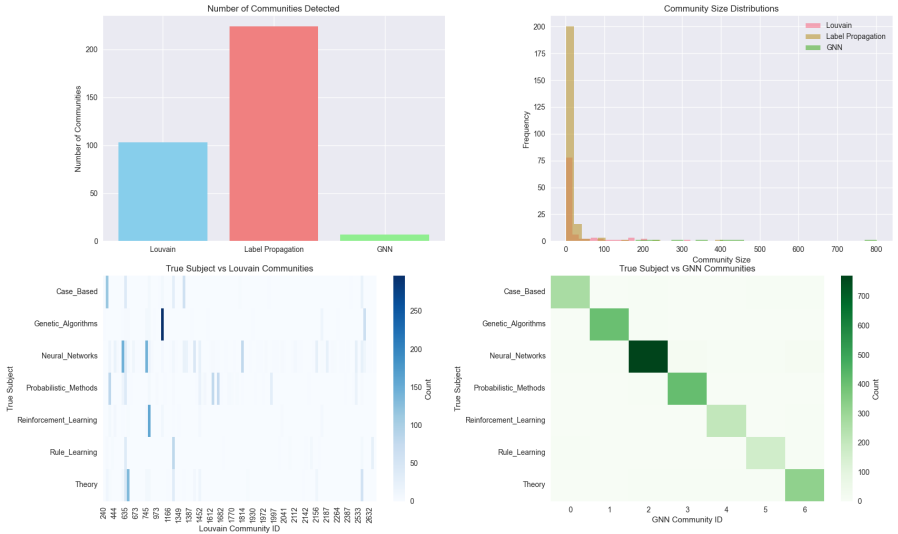


Figure: GCN modeli tarafından öğrenilen düğüm gömülmelerinin (embeddings) t-SNE ile 2 boyuta indirgenmiş hali

Sonuçlar: t-SNE ile Düğüm Gösterimi (Figure 2)

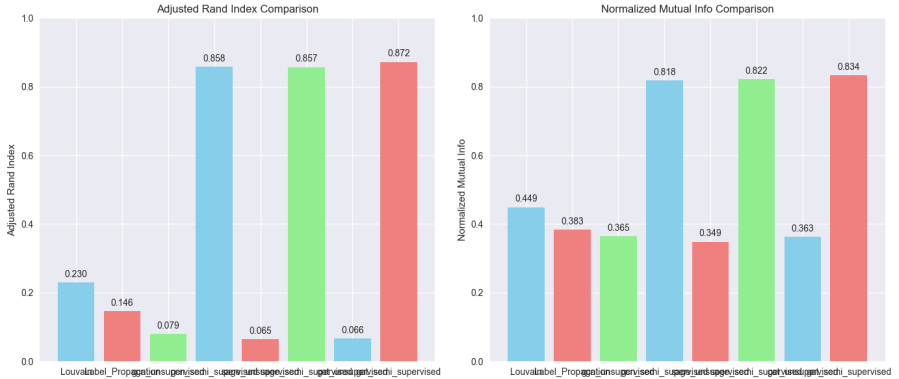


Figure: GraphSAGE modeli tarafından öğrenilen düğüm gömümlerinin (embeddings) t-SNE ile 2 boyuta indirgenmiş hali.

Teşekkürler