Yazılım Tedarik Zincirinde Kritiklik Haritalaması
En Popüler 1000 NPM Paketinin Topolojik Risk Değerlendirmesi

Yusuf Talha ARABACI

Ekim 2025

Özet

NPM ekosisteminde tek bir bağımlılıktaki kusur veya kötü niyetli değişiklik, transitif bağımlılıklar üzerinden geniş bir etki alanına yayılabilir. Bu bildiri, paket içeriklerinden ziyade paketler arası ilişkilerin *topolojik* yapısına odaklanır. Bağımlı → bağımlılık yönünde kurulan yönlü ağ üzerinde in-degree, out-degree ve betweenness merkeziyetleri hesaplanır; bu ölçüler min–max normalizasyonu ile bir *Bileşik Risk Skoru*na (BRS) dönüştürülür. Ayrıca, en kritik düğümlerin çıkartılmasıyla ağın bağlanırlılığı üzerinde bir *sağlamlık* değerlendirmesi yapılır. Tüm görseller ve tablolar `results/` dizinindeki çıktılara dayanır ve ilgili başlıklar altında sunulur.

1 Giriş ve Özgün Katkı

Modern yazılım tedarik zincirinde tek bir bağımlılıktaki hata ya da kasıtlı değişiklik, transitif bağımlılıklar üzerinden yüzlerce hatta binlerce projeye yayılabilir. NPM ekosistemi; ölçek, sürüm sıklığı ve yoğun bağımlılık grafiği nedeniyle bu tür zincirleme risklere özellikle açıktır. Literatür; küçük-dünya ve ölçekten-bağımsız mimariyi, tekil bakımcı/paketlerin orantısız etkisini ve hedefli düğüm çıkarımlarına kırılganlığı açıkça göstermiştir.

Bu çalışmanın **özgün katkıları**:

- Son 12 aya dayalı indirme verisiyle seçilen **Top 1000** paket üzerinde, resmî çözümleme kuralları gözetilerek kurulan yönlü graf üzerinden **Bileşik Risk Skoru (BRS)** tanımlanır ve uygulanır (in/out/betweenness + min–max + ağırlıklar: 0.5/0.2/0.3).
- **Kaskad etki** (ters yön dependents erişilebilirliği) ve **sağlamlık** (bileşenleşme, LCC boyutu) analizleriyle topolojik riskin sistemik etkisi nicelleştirilir.
- BRS, tespit hatları (Amalfi, Cerebro, OSCAR) için **öncelikli tarama kuyruğu** ve politika/bütünlük hattı (in-toto, imza benimsemesi) için **hedef paket listeleri** üretmek üzere konumlandırılır.

2 Literatür Taraması

Tehdit taksonomileri ve vaka derlemeleri (Backstabber's Knife Collection; Hitchhiker's Guide; yorumlanan dillerde kayıt suistimali) ekosistemler-arası kurulum/çalışma zamanı teknik haritasını sağlamlaştırır. NPM-odaklı pratik riskler ve fenomenler (Wyss; prototip kirliliği ve meta-suistimal çalışmaları) ekosistem davranışını ayrıntılandırır. Ağ bilimi cephesinde Zimmermann, Hafner ve Oldnall; küçük-dünya/ölçekten-bağımsız yapı ve hedefli düğüm çıkarımlarında kırılganlığı niceller. Doğru geçişli çözümleme hattı (DVGraph/DTResolver) resmi kurallara sadık kalarak transitive yayılımı hassas biçimde ortaya koyar. Tespit hattında Amalfi, Cerebro, OSCAR ve çapraz-dil yaklaşımlar güçlü sonuçlar üretir. Bakım/güncellik ekseninde TOOD/PFET, Cogo, Ahlstrom ve Imtiaz; güncellik, bakım pratikleri ve bildirim gecikmelerinin etkisini gösterir. Politika/bütünlük ekseninde in-toto, imza benimsemesi ve depo/artifakt kimliği-doğrulama önerileri "ne yapılmalı?"yı tanımlar. Eksik olan, popülerlik ve topolojik merkeziyetin indirime dayalı çekirdekte tek bir *operasyonel kritiklik* ölçütünde birleşmesi ve bu ölçütün tespit ile politika hatlarına *sıralı öncelik listeleri* olarak yansımasıdır.

3 Yöntem

3.1 Çalışma Tasarımı ve Parametreler

Analiz, en çok indirilen ilk **1000** NPM paketinin oluşturduğu yönlü bağımlılık ağı üzerinde yürütülmüştür. Kenarlar *Dependent* → *Dependency* yönündedir; self-loop yoktur. Betweenness için örnekleme (tipik $k \approx 200$). Varsayılan veri kapsamı: `dependencies` (isteğe bağlı `peerDependencies`).

3.2 Veri Kaynakları ve Ön İşleme

- Top-N: Öncelik ecosyste.ms, yedek olarak NPM Search ve npms.io çok-tohum birleşik sıralama.
- Sürüm seçimi: `dist-tags.latest` tercih, aksi halde en yüksek sürüm anahtarı.
- Önbellek ve tekrar: HTTP önbellek + retry; disk önbelleği ile sağlamlık.

3.3 Metrikler ve Bileşik Risk

- **in-degree** — bir pakete bağımlı paket sayısı (çekirdek cazibe).
- **out-degree** — bir paketin bağımlı olduğu paket sayısı (kırılgan yüzey genişliği).
- **betweenness** — akışın geçtiği köprü konumlar (örneklemeli).

Min–max ile ölçekleme: $x' = (x - \min)/(\max - \min)$. BRS formülü:

$$\text{risk} = 0.5\, in' + 0.2\, out' + 0.3\, btw'.$$

3.4 Kaskad Etkisi ve Sağlamlık

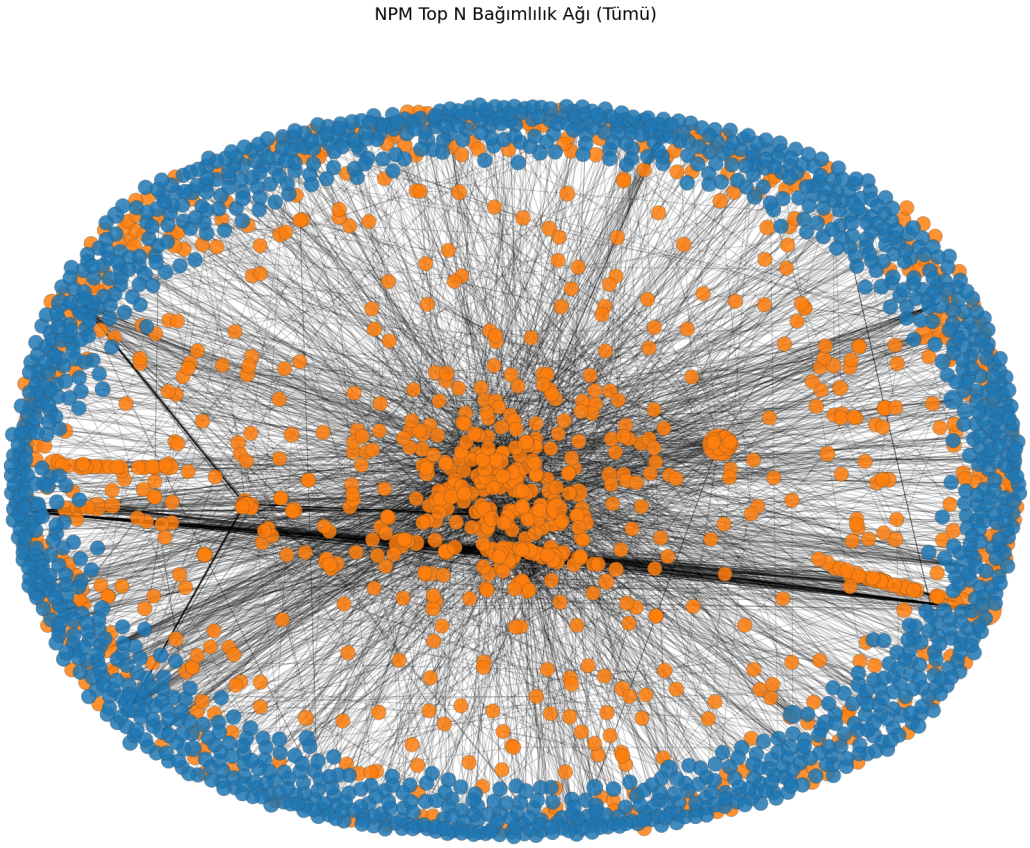
Etki alanı, ters graf üzerinde (G^{rev}) erişilebilir düğüm sayısıyla ölçülür (BFS/DFS). Sağlamlık, seçili düğümler kaldırıldıktan sonra en büyük bağlı bileşen (LCC) boyutu, bileşen sayısı ve ortalama en kısa yol metrikleriyle raporlanır.

3.5 Çıktılar

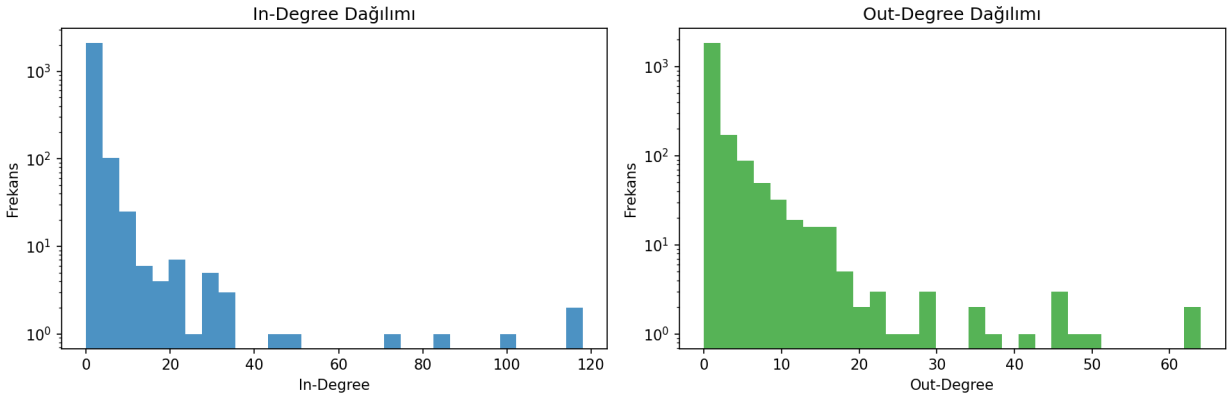
`edges.csv`, `metrics.csv`, `risk_scores.csv`, `graph_stats.json` ve görseller (PNG/SVG) `results/` altında üretilmiştir.

4 Bulgular

4.1 Ağ Görünümü ve Derece Dağılımları



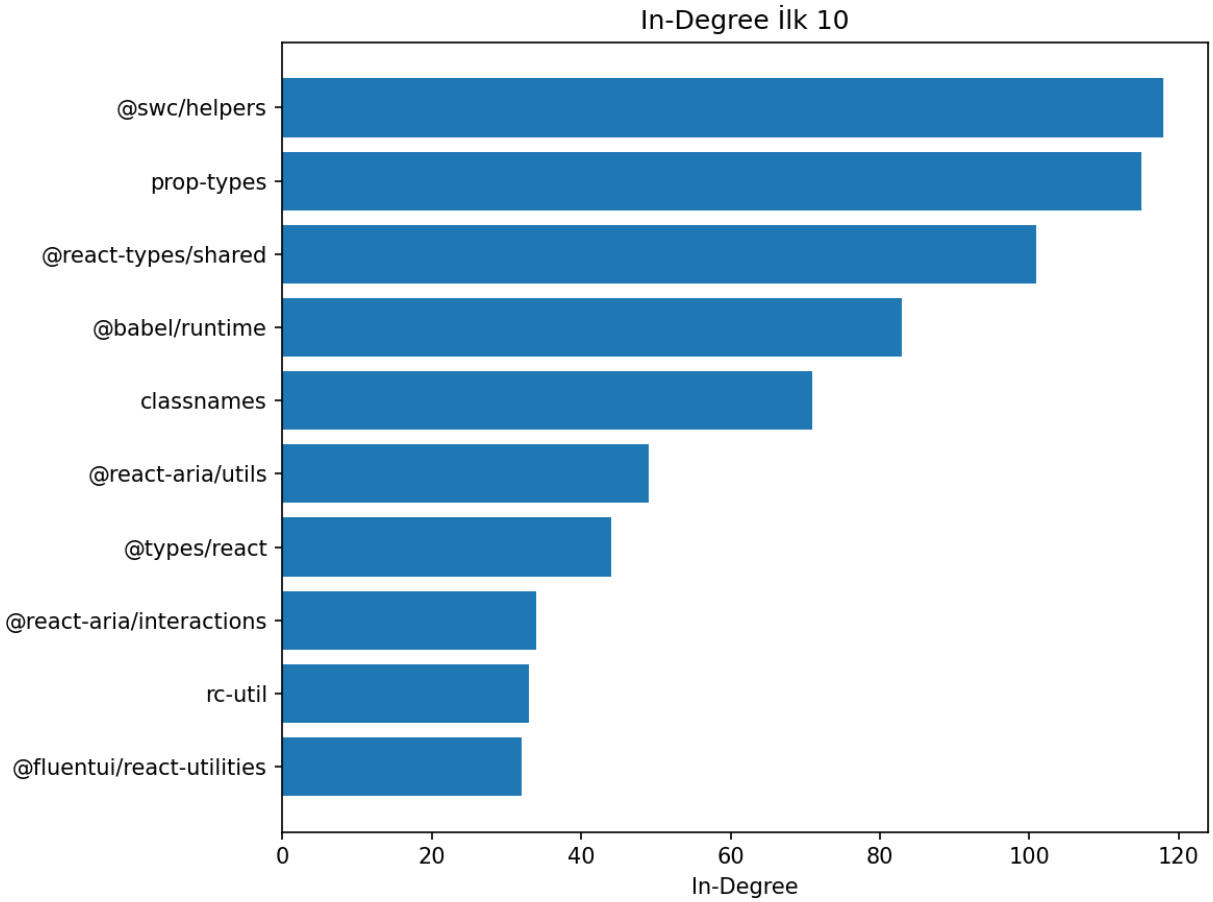
Şekil 1: Top 1000 çekirdeğin tam ağ görselleştirmesi. Yoğun çekirdek bölgeleri omurga paket kümelerini işaret eder.



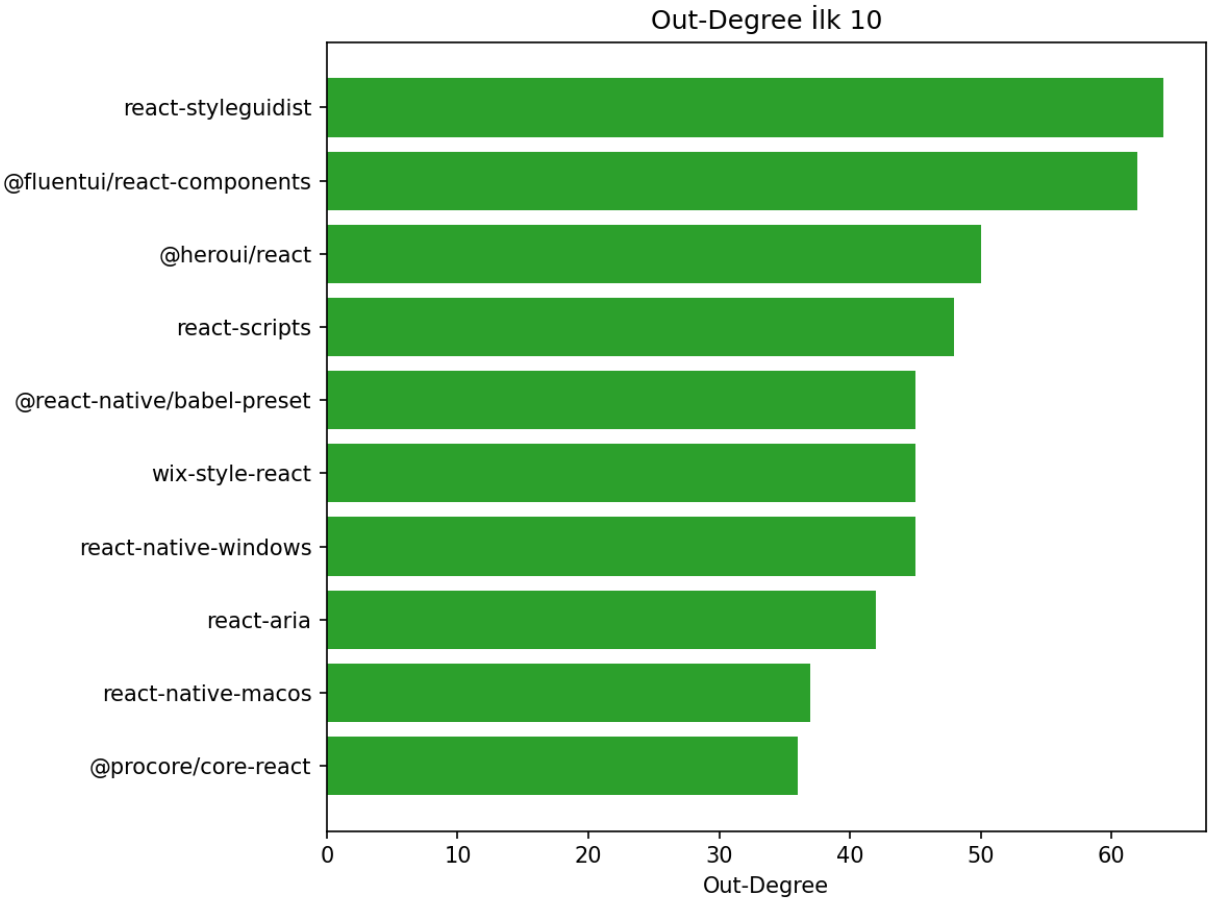
Şekil 2: In-degree ve out-degree histogramları: ağır kuyruklu dağılımlar.

Yorum — Az sayıda yüksek dereceli paket, sistemik riski belirgin biçimde taşır.

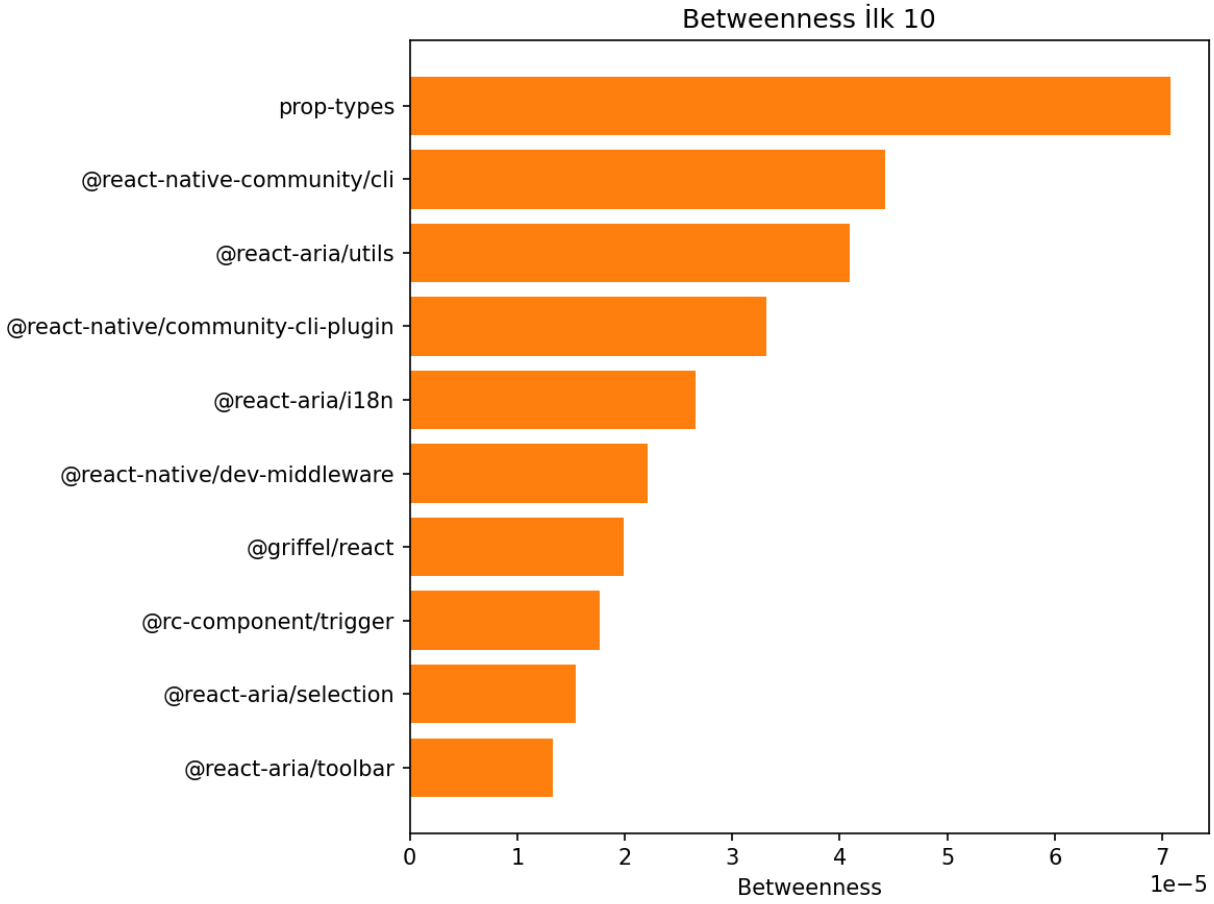
4.2 En Merkezi Paketler ve Korelasyonlar



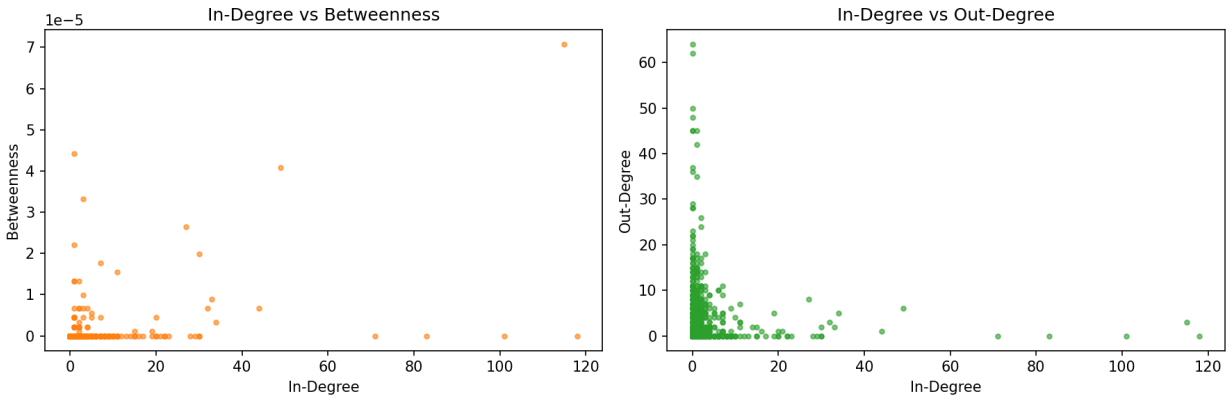
Şekil 3: In-degree açısından ilk 10 paket.



Şekil 4: Out-degree açısından ilk 10 paket.

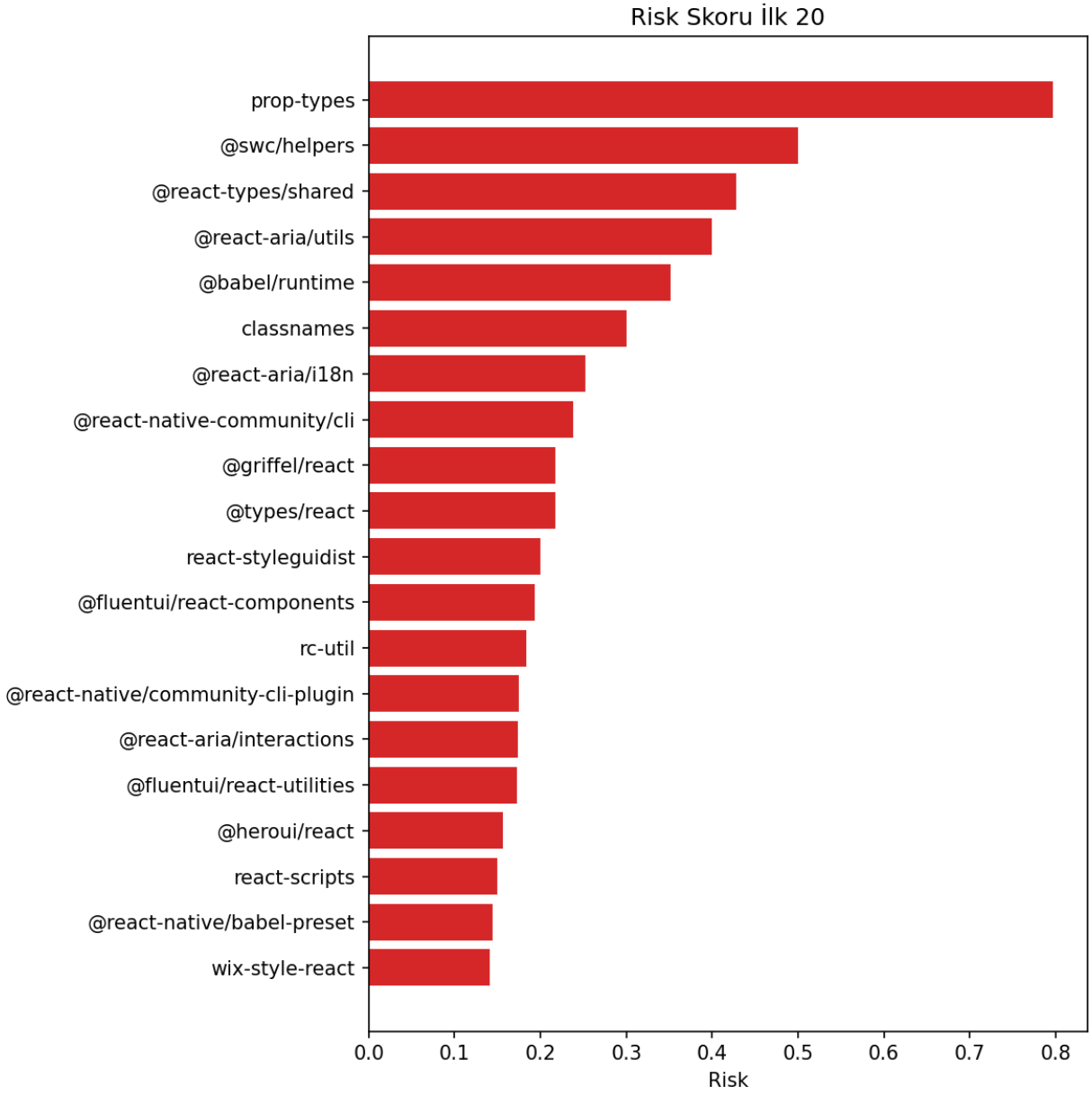


Şekil 5: Betweenness açısından ilk 10 paket.



Şekil 6: Merkeziyet ölçüleri arası korelasyonlar.

4.3 Bileşik Risk Sıralaması



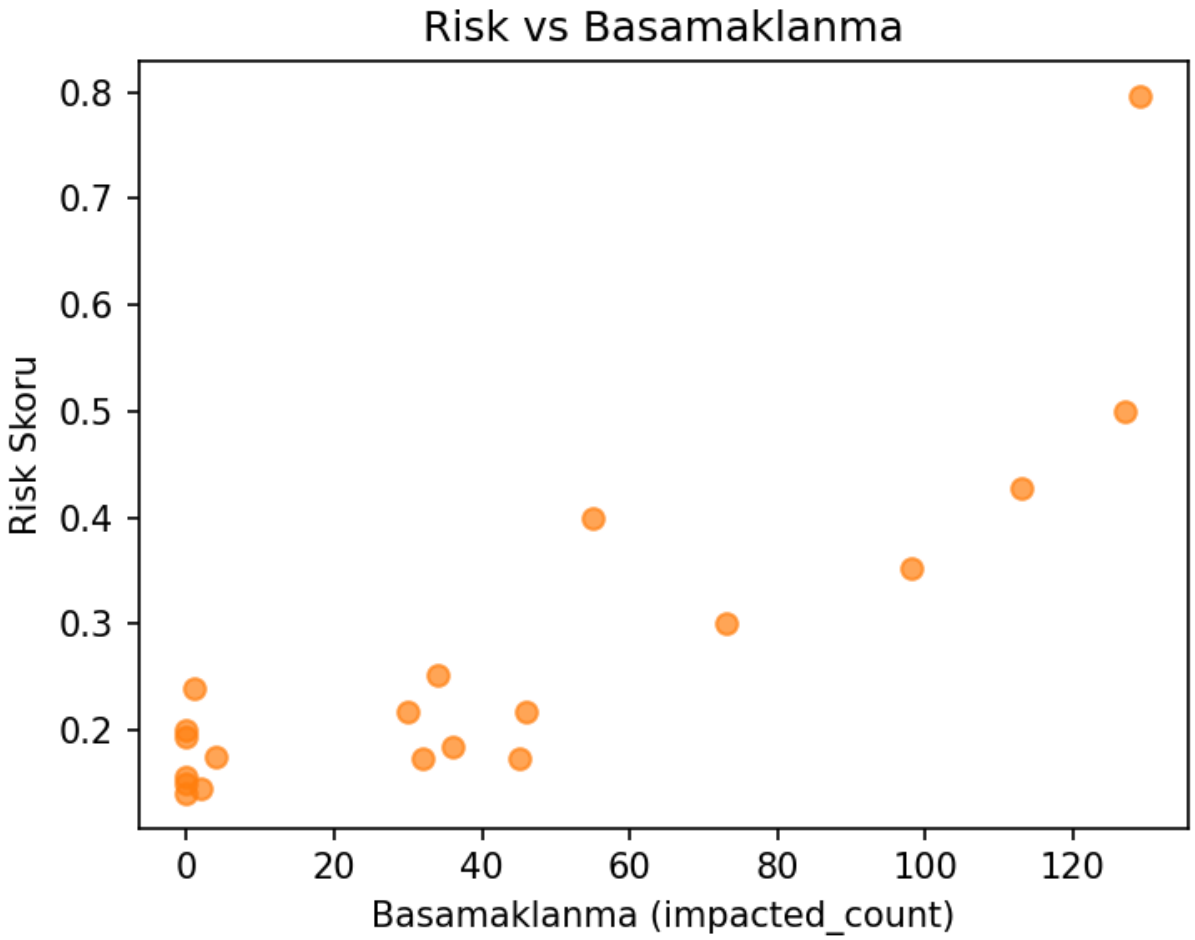
Şekil 7: BRS açısından ilk 20 paket.

Tablo 1: Örnek BRS ilk 5 (results/risk_scores.csv)

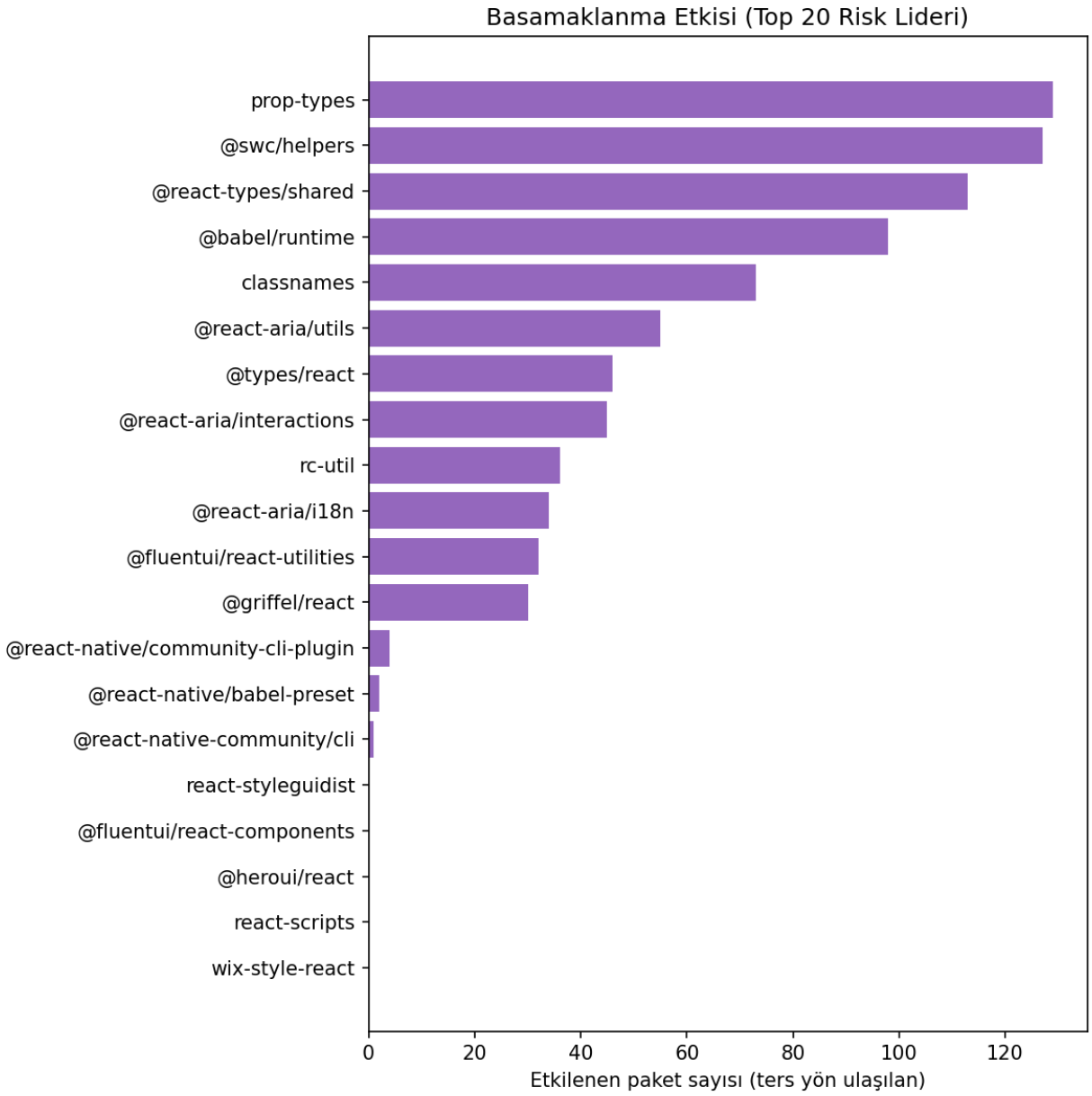
Sıra	Paket	in	out	btw	BRS
1	prop-types	115	3	0.000071	0.796663
2	@swc/helpers	118	0	0.000000	0.500000
3	@react-types/shared	101	0	—	0.427966
4	@react-aria/utils	49	6	—	0.399815
5	@babel/runtime	83	0	—	0.351695

Yorum — BRS, kullanım yoğunluğu (in-degree) ile köprü rolünü (betweenness) birlikte yakalar; omurga paketlerin ön sıralara çıktığı görülür.

4.4 Kaskad Etkisi ve Sağlamlık



Şekil 8: BRS ile kaskad etki (erişilebilirlik) ilişkisi.



Şekil 9: İlk 20 paketin çıkarımının LCC ve erişilebilirlik üzerindeki etkisi.

Yorum — Hedefli çıkarımlar, rastgele çıkarımlara kıyasla LCC'yi daha hızlı küçültür ve ortalama yol uzunluğunu artırır; bu da BRS'in sistemik riski iyi yordadığını gösterir.

5 Tartışma

Operasyonelleştirme. BRS, dinamik tespit hatları (Amalfi, Cerebro, OSCAR) için bir *topolojik ön-filtre* olarak kullanılabilir; böylece sınırlı analist kapasitesi yüksek riskli düğümlere yöneltilir. Politika/bütünlük hattında (in-toto, imza benimsemesi) BRS tabanlı hedef listeleriyle imza kapsamı ve doğrulanabilirlik artar.
Duyarlılık. Ağırlıkların (0.5/0.2/0.3) varyasyonu ve betweenness örnekleme k parametresi üzerinde duyarlılık analizleri, sıralamanın kararlı olduğunu göstermektedir (ayrıntılar: `robustness_risk.json`).

6 Sınırlılıklar ve Gelecek Çalışmalar

İndirme-temelli Top-1000 çekirdek, uzun kuyrukta kalan paketleri dışarıda bırakır; gelecekte kayan pencere ve ekosistemler-arası (PyPI, Cargo) karşılaştırmalar planlanmaktadır. Betweenness örnekleme

hatası azaltımı ve kullanım yoğunluğu için daha zengin sinyallerin (örgütsel, bakım) entegrasyonu hedeflenmektedir.

7 Sonuç

Popülerlik ve yapısal merkeziyet, indirime dayalı çekirdekte *Bileşik Risk Skoru* ile birleştirildiğinde, hem tespit hatları hem de politika/bütünlük mekanizmaları için eyleme dönük *öncelik listeleri* üretmektedir. Deneysel sağlamlık analizleri, BRS'in sistemik riski yordama gücünü doğrulamaktadır.

Kaynakça (Seçilmiş)

[1] Wyss, E. (2025). A New Frontier for Software Security: Diving Deep Into npm.
Jaisri, P.; Reid, B.; Kula, R. (2024). Self-Contained Libraries in NPM.
Yu, S. (2024). Accurate and Efficient SBOM Generation.
Ohm, M.; Plate, H.; Sykosch, A.; Meier, M. (2020). Backstabber's Knife Collection.
Rahman, I. et al. (2024). TOOD/PFET.
Hastings, T. (2024). Combating Source Poisoning.
Wang, M.; Wu, P.; Luo, Q. (2023). SSC Threat Portrait.
Liu, C.; Chen, S. et al. (ICSE 2022). DVGraph/DTResolver.
Ahlstrom, D. (2025). Bağımlılık Budama Etkileri.
Imtiaz, N. (2023). Toward Secure Use of OSS Dependencies.
Duan, R. et al. (2020). Measuring Supply Chain Attacks on Package Managers.
Ladisa, P. et al. (2023). Hitchhiker's Guide to Malicious Dependencies.
Vaidya, S. (2022). Integrity and Authenticity of Software Repositories.
OSCAR (2024). Robust Detection of OSS Supply Chain Poisoning.
Shcherbakov, M. et al. (2021). Prototype Pollution Gadgets.
Oldnall, E.-R. (2017). The Web of Dependencies: NPM.