

# Multipartite Joining

## Description

Multipartite Joining is a form of graph projection with attribute aggregation. One partite (or semi-partite) set of nodes, selected by a type attribute, is projected across another (semi-) partite set of nodes.

The new graph only has nodes from the first set, and no edges originally in the graph. Edges are created between nodes in the new graph if they shared neighbors that were in the second set. New attributes are created on nodes in the new graph and on edges based on rules for aggregating the attributes on neighbor nodes (shared neighbor nodes, in the case of attributes created on edges).

For instance, imagine a graph of four nodes. Nodes A and B are nodes in the first set. Node C is connected to both A and B, and node D is only connected to A. Perhaps nodes A and B are people, and nodes C and D are grants. Each grant is for \$10,000.

We can do multipartite joining, selecting those sets (via some type attribute), with rules as follows:

```
grant_money_received = grant_amount.total
edge.grant_money_shared = grant_amount.total
```

The result would be a new graph with just nodes A and B. There would be an edge between A and B. Node A, since it has neighbors with total grant\_amount \$20,000, would have a grant\_money\_received attribute valued \$20,000. Node B, since it only has neighbors with total grant\_amount \$10,000, would have a grant\_money\_received attribute valued \$10,000. Since they share one \$10,000 grant, the edge between them would have a grant\_money\_shared attribute valued \$10,000.

The available operations are sum, average, max, min, mode, and count.

## Pros & Cons

This algorithm is very flexible, but slow. It requires a solid understanding of the underlying data to make sensible choices of aggregation functions.

Much of the slowness is because the flexibility in aggregation functions makes it difficult to use fast matrix operations and remain memory efficient.

## Applications

Extracting subnetworks from a larger multipartite network.

## Implementation Details

The aggregation rules must be specified, as above, in a metadata file provided to the algorithm (which is just a java properties file with the appropriate keys and values, and so might be called something like "metadata.properties").

## See Also



The license could not be verified: License Certificate has expired! [Generate a Free license now.](#)