

Übungsblatt 3

Abgabe: **06. Juni 2007**

Prioritätsclusterer

Implementieren Sie einen allgemeinen Prioritätsclusterer, der das Interface

`de.uka.algo.clustering.algorithms.Algorithm`

erfüllt. Als Eingabe soll eine Berechnungsweise von Prioritäten und ein Qualitätsindex übergeben werden. Der Algorithmus soll dabei wie folgt vorgehen: Es werden alle Kanten mit höchster Priorität berechnet und anschließend aus dem Graphen entfernt. Die Zusammenhangskomponenten definieren die Clusterung. Die finale Clusterung soll diejenige sein, die am besten bewertet wurde (bezüglich dem Qualitätsindex).

Erweitern Sie Ihren Prioritätsclusterer, so dass auch Kanten mit kleinerer Priorität entfernt werden können.

Tipp: Schreiben Sie ein zugehöriges

`de.uka.algo.clustering.algorithms.modules.AlgorithmModule`,

welches Sie in yED einbinden können.

Betweenness

Implementieren Sie einen Wrapper, der die exakte und approximierete Betweenness als Priorität für den obigen Algorithmus zur Verfügung stellt.

Erste Untersuchungen

- (a) Anstatt nach jeder Löschoption die neue Clusterung zu berechnen und zu bewerten, berechnet man die komplette Löschoption und baut die Clusterungen inkrementell auf. Ergibt sich ein Laufzeitunterschied? Was sind mögliche Begründungen.
- (b) Wie stark ist der Samplingeinfluss bei der Approximation von Betweenness?