

Use Cases

ImmortalGraph
Durchmesser eines Graphen zu Zeitpunkt t (= Suche nach dem längsten Shortest Path)
Freunde gefunden seit Jahr XY (= wie viele Kanten sind in einem bestimmten Zeitraum angefügt worden?)
Welche Person hat den größten Einfluss im Netzwerk und wie verändert sich dieser (= Centrality – Degree, Closeness oder Betweenness)
GraphInc
Shortest Paths
Connected Component finden durch Label Propagation
Wurden durchgeführt, Laufzeit-Diagramm – nach dem Motto „das ist unser Modell, damit gehen folgende Algorithmen
TGraph
Verkehrs- und Straßennetz: shortest path zwischen 2 Knoten zu unterschiedlichen Tageszeiten
Tink
Shortest path earliest arrival time (starte an einem Knoten u, was ist die frühestmögliche Ankunftszeit an Knoten v?)
Shortest path fastest path (starte an Knoten x, über welchen Pfad gelangt man am schnellsten zu Knoten z?)
Temporal betweenness (Wie wichtig ist Knoten x im Netzwerk?)
Temporal closeness
Chronos
Durchmesser des Graphen (finde den längsten shortest path zu Zeitpunkt t)
Wie verändert sich PageRank für jeden einzelnen Knoten in einem bestimmten Zeitintervall?
SSSP auf Wikipedia Ref Graph
Kineograph
PageRank
TunkRank (berechnet Einfluss von User X, anschließend global aggregator, um Liste von Top K zu erhalten)
Shortest Paths
GraphTAU
PageRank
GraphTides (2018)
Connections in a Social Network (Rang je User, welchen Einfluss dieser hat)
Blockchain Transaction (transaction/wallet graph – z.B. durchschnittl. Betrag einer Transaktion)
Naiad und Tornado
WCC
Shortest Paths
PageRank

- ➔ Allg. Graph-Algorithmen wie SSSP
- ➔ PageRank, TunkRank
- ➔ Centrality Kennzahlen und Durchmesser

Als Basis muss es möglich sein folgende Kennzahlen zu berechnen:

- **Degree der Knoten**
- **Anzahl der Knoten, die von einem Knoten aus erreicht werden können (CC)**
- **Länge des SP zwischen 2 Knoten**
- **Summe der Gewichte der Kanten auf einem Pfad**

Telekom-Use Case:

- Zeitraum, in dem irgendein Vertragsverhältnis besteht (Kunde ist Telekom-Kunde generell)
- Kunde hat über den Zeitraum mehrere unterschiedliche Verträge (anfangs nur Internet und Festnetz, dann Internet+Festnetz+Fernsehen)
- Kunde hat Adresse (was ist bei Umzug?)
- Tarife ändern sich, sowohl innerhalb eines Vertrages (1. Jahr billiger als 2. Jahr) als bei Vertragswechsel
- Verschiedene Modellierungs-Szenarien:
 - a) eine Kante für eine kontinuierliche Kundenbeziehung, Kunden-Knoten mit Attributen, die sich ändern können (z.B. Tarif, Preis, Adresse) – dann müssten die Attribute eine Art Listen sein, die für die jeweils gültigen Zeiträume, die entsprechenden Werte liefern
 - b) je Vertrag eine Kante zum Kunden-Knoten. Kante mit Gültigkeitszeitraum und Knoten mit entsprechenden Attribute und Werten -> je Vertragsänderung neue Kanten/neuer Knoten (evtl. im Graph doppelt gespeicherte Daten, wie Name und nicht veränderte Adresse)
 - c) sobald neuer Kunde entsteht: neue Kante/neuer Knoten mit Stammdaten (was sind Stammdaten?), der nicht wieder gelöscht werden kann und von diesem Knoten aus neue Kanten (mit Gültigkeitszeitraum) jeweils für unterschiedliche Verträge [Stammdaten z.B. Name, Kundennummer, Adresse Wohnsitz // jeweilige Vertragsdaten z.B. Lieferadresse, Tarif, Preis]
- mögliche Datenabfragen: AVG Preis aller Magenta M Kunden, SUM aller Einnahmen für PLZ 86150, welche Stadt in Deutschland MAX Einnahmen.

Kumulative Aggregation abgeleitet aus [5]:

- Wie hat sich die Summe der Ausgaben aller Kunden innerhalb der letzten 2 Jahre monatliche entwickelt? (Zeitraum von 24 Monaten, 24 Stichtage/Zeitpunkte) wenn man

keine historische Auswertung machen will, müsste man fortlaufend Sum-AGG am Ende eines jeden Monats speichern

- Der neue Aggregatswert bei kumulativen Aggregaten kann direkt aus dem vorherigen gespeicherten Wert abgeleitet werden.

Selektive Aggregate nach [5]:

- Was ist der Preis maximale Preis zu jedem Zeitpunkt?
- Selektive Aggregate können nicht auf Basis des alten Aggregatwertes berechnet werden, wie es bei kumulativen Aggregaten der Fall ist. Als Beispiel dient MAX-Funktion. Wird der Max-Wert ungültig, muss der zweitgrößte Wert bekannt sein. Wird auch dieser ungültig, muss der drittgrößte Wert bekannt sein usw. Häufig wird als Lösungsansatz hierzu empfohlen eine Liste von Top-K Werten zu speichern, wie z.B. in [5] beschrieben.