Humboldt-Universität zu Berlin
Institut für Informatik
Lehrstuhl Logik in der Informatik
Prof. Dr. Nicole Schweikardt

21. Juni 2016

Big Data Analytics in Theorie und Praxis

Sommersemester 2016

Theorie-Übungsblatt 2

Zu bearbeiten bis 28. Juni 2016

Aufgabe 1: **(10 + 10 + 10 = 30 Punkte)**

(a) Arbeiten Sie die Details dazu aus, wie das Vektor-Matrix-Produkt $Y' = Y \cdot P$ zur Berechnung des Page-Ranks ausgerechnet werden kann (siehe „Zweiter Ansatz“ im Vorlesungsskript). Gehen Sie hierbei insbesondere darauf ein, wie dies mit *MapReduce* gelöst werden kann und analysieren Sie die Güte Ihres Ansatzes.

(b) Das im Skript unter „Zweiter Ansatz“ beschriebene Verfahren schickt alle Teilsummen $Z_{1,t}, \ldots, Z_{\ell,t}$ an denselben Rechner, der dann die Gesamtsumme Z_t berechnen soll. Beschreiben Sie eine alternative Methode, die die Berechnung dieser Summe parallel auf mehreren Rechnern durchführt und dabei die im Rechencluster vorhanden Rechner so gut wie möglich nutzt. Geben Sie auch an, wie dies mit *MapReduce* umgesetzt werden kann und analysieren Sie die Güte Ihres Ansatzes.

(c) Gehen Sie von $n = 10^{12}$ Webseiten, durchschnittlich 20 Links pro Webseite und einem benötigten Speicherplatz von 4 Byte für jeden Eintrag im Eingabe-Vektor Y und jeden Eintrag der Page-Rank-Matrix P aus. Gehen Sie davon aus, dass jeder Clusterrechner einen zur Berechnung verfügbaren Arbeitsspeicher von 20 Gigabyte besitzt. Wie viele Rechner werden im Cluster benötigt, damit die jeweiligen Teilstücke von Y und P (dabei P in der in der Vorlesung behandelten kompakten Speicherung) in den Arbeitsspeicher des jeweiligen Clusterrechners passen? Eine gerundete Rechnung ist ausreichend, und es kann davon ausgegangen werden, dass die Werte in P nicht ungünstig verteilt sind.

Aufgabe 2: **(4 + 6 + 10 + 5 = 25 Punkte)**

Seien G_n und G'_k zwei vollständige gerichtete Graphen mit je n bzw. k Knoten, deren Knotenmengen disjunkt sind. D.h.

- $G_n = (V_n, E_n)$ mit $V_n = \{x_1, x_2, \ldots, x_n\}$ und $E_n = \Big\{(x_i, x_j) : i, j \in \{1, \ldots, n\}\Big\}$,
- $G'_k = (V'_k, E'_k)$ mit $V'_k = \{y_1, y_2, \ldots, y_k\}$ und $E'_k = \Big\{(y_i, y_j) : i, j \in \{1, \ldots, k\}\Big\}$,
- $V_n \cap V'_k = \emptyset$, $|V_n| = n$ und $|V'_k| = k$.

Sei nun G die Vereinigung der beiden vollständigen Graphen G_n und G'_k mit einer zusätzlichen Kante von x_1 nach y_1, d.h.

- $G = (V, E)$ mit $V = V_n \cup V_k$ und $E = E_n \cup E_k \cup \{(x_1, y_1)\}$.

(a) Bestimmen Sie für alle Knoten aus G die Page-Ranks für den Dämpfungsfaktor $d = 0$.

(b) Bestimmen Sie für alle Knoten aus G die Page-Ranks für den Dämpfungsfaktor $d = 1$.

(c) Sei nun d ein beliebiger Dämpfungsfaktor mit $0 < d < 1$. Geben Sie die Knoten von G sortiert nach der Größe ihrer Page-Ranks an (je höher der Page-Rank ist, desto weiter vorn soll der Knoten in Ihrer Sortierung stehen). Begründen Sie Ihre Antwort.

(d) Betrachten Sie nun den etwas allgemeineren Fall, dass G_n und G'_k beliebige *stark zusammenhängende* gerichtete Graphen auf den Knotenmenge $V_n = \{x_1, \ldots, x_n\}$ und $V'_k = \{y_1, \ldots, y_k\}$ sind. Der Graph G sei wie oben als die Vereinigung der beiden Graphen G_n und G'_k mit einer zusätzlichen Kante von x_1 nach y_1 definiert.

Welche Aussage können Sie für diesen Graphen G und einen beliebigen Dämpfungsfaktor d mit $0 < d < 1$ über den durchschnittlichen Page-Rank der Knoten aus V_n im Vergleich zu dem durchschnittlichen Page-Rank der Knoten aus V'_k treffen? Begründen Sie Ihre Aussage.

Aufgabe 3: (13 + 12 = 25 Punkte)

Ein Anbieter einer Internetseite w möchte seiner Webseite zu einem höheren Page-Rank verhelfen. Sei $0 < d < 1$ der Dämpfungsfaktor und sei n die Gesamtzahl der Internetseiten inklusive w. Wir gehen in dieser Aufgabe davon aus, dass w nur auf sich selbst verlinkt.

(a) In einem ersten Versuch legt der Anbieter $c \geq 1$ neue Seiten an, die alle ausschließlich auf die Seite w verweisen. Zeigen Sie, dass sich der Page-Rank der Seite w dadurch um maximal $(d - \frac{1}{n}) \cdot \frac{c}{n+c}$ erhöht, d.h.: wenn die Webseite w vorher den Page-Rank p_w hatte, so hat sie nach dem Hinzufügen der c Seiten einen Page-Rank von höchstens $p_w + (d - \frac{1}{n}) \cdot \frac{c}{n+c}$. Gehen Sie dabei der Einfachheit halber von der Annahme aus, dass sich bei jeder der $n-1$ ursprünglich außer w vorhandenen Webseiten der Page-Rank nicht erhöht (diese Annahme müssen Sie nicht beweisen).

(b) Als Alternative überlegt der Anbieter, zusätzlich noch Links zwischen den c neu angelegten Seiten einzufügen, so dass die c neuen Seiten einen vollständigen gerichteten Graphen bilden (Definition siehe Aufgabe 1). Ist dies eine gute Idee? Begründen Sie Ihre Antwort.

Aufgabe 4: (20 Punkte)

In der Vorlesung wurde die „Suche nach der fehlenden Zahl“ betrachtet, bei der man als Eingabe einen Datenstrom von $n-1$ unterschiedlichen Zahlen aus der Menge $\{1, 2, \ldots, n\}$ erhält, bei dem diese $n-1$ Zahlen eine beliebige Reihenfolge haben können. Ziel ist, die fehlende Zahl aus der Menge $\{1, 2, \ldots, n\}$ zu ermitteln. In der Vorlesung wurde ein Algorithmus vorgestellt, der höchstens $\lceil 2 \log n \rceil$ Speicherbits benötigt. Und es wurde gezeigt, dass *jeder* Algorithmus, der das Problem löst, mindestens $\lceil \log n \rceil$ Speicherbits benötigt. Schließen Sie die Lücke zwischen der unteren Schranke $\lceil \log n \rceil$ und der oberen Schranke $\lceil 2 \log n \rceil$ der zur Problemlösung benötigten Speicherbits. D.h.: Geben Sie für jede natürliche Zahl $n \geq 1$ eine geeignete natürliche Zahl $s(n)$ an und

- beschreiben Sie einen Datenstromalgorithmus, der bei Eingabe eines Stroms von $n-1$ verschiedenen Zahlen aus $\{1, \ldots, n\}$ die fehlende Zahl berechnet und dafür höchstens $s(n)$ Speicherbits benutzt,
- weisen Sie nach, dass Ihr Algorithmus tatsächlich die fehlende Zahl korrekt bestimmt und höchstens $s(n)$ Speicherbits benutzt, und
- beweisen Sie, dass es keinen Datenstromalgorithmus geben kann, der das Problem löst und dazu mit weniger als $s(n)$ Speicherbits auskommt.