Internet-Algorithmen

Sommersemester 2014

Prof. Dr. Georg Schnitger
Hannes Seiwert

Arbeitsgruppe Theoretische Informatik, Institut für Informatik

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

Übungsblatt 6

Ausgabe: 16.06.2014
Abgabe: 23.06.2014 vor Vorlesungsbeginn

Hinweis: Alle Lösungen sind ordentlich und in lesbarer Schrift zu verfassen. Fasse dich kurz, beschränke deine Erläuterungen und Rechnungen auf die wesentlichen Punkte.

Hinweis: Die Gesamtpunktzahl beträgt 24 Punkte. Darüber hinaus erworbene Punkte werden als Bonuspunkte angerechnet.

Aufgabe 6.1. (2+4+1+5)

Mäuse finden

Gegeben sei ein Datenstrom $D = x_1, \ldots, x_n$ mit Elementen aus einem Universum U. Wir bezeichnen mit $U_1 := \{u \in U | a_u \geq 1\}$ die Menge der Elemente, die mindestens einmal vorkommen. Offensichtlich gilt $|U_1| = H_0$.

Wir suchen nach Elementen, die nur selten im Datenstrom auftauchen. Sei $A := \{u \in U_1 | a_u = 1\}$ die Menge aller Elemente, die genau einmal vorkommen. Es soll $Q := |A|/H_0$ berechnet werden.

a) **Erkläre**, warum hier ein Stichproben-Ansatz mit Reservoir-Sampling nicht weiterhilft.

b) Sei K eine auf U_1 gleichverteilte Zufallsvariable und a_K die Häufigkeit des (zufälligen) Elementes K. **Beschreibe**, wie K und a_K speichereffizient erzeugt werden können. Beachte, dass anfangs nur U bekannt ist, nicht aber U_1.

c) Sei B eine binäre Zufallsvariable mit $B = \begin{cases} 1\,, & \text{falls } a_K = 1 \\ 0\,, & \text{sonst} \end{cases}$

Berechne den Erwartungswert $E[B]$.

d) **Entwirf** einen Algorithmus, der eine $(1+\varepsilon)$-Approximation von Q mit Wahrscheinlichkeit mindestens $1 - \delta$ berechnet und **gib** den benötigten Speicherplatz **an**.

Wenn $\hat{Q}$ die Ausgabe deines Algorithmus ist, soll also gelten: $\text{prob}\left[|\hat{Q} - Q| \geq \varepsilon\, Q\right] \leq \delta$.

Hinweis: Erzeuge mehrere unabhängige Schätzungen, bilde eine geeignete Mittelung und schätze den Fehler z.B. mit den Chernoff-Ungleichungen ab.

Kommentar: In der Praxis ist die Bestimmung von Q hilfreich, um Distributed-Denial-of-Service(DDoS)-Angriffe zu erkennen.

Aufgabe 6.2. (8) *Elephanten finden*

Auf einem Datenstrom $D = x_1, \ldots, x_n$ soll ein Heavy Hitter mit Häufigkeit größer $n/2$ bestimmt werden. Wir wollen eine untere Schranke für den benötigten Speicherplatz einer deterministischen Lösung dieses Problems angeben.

Sei A ein deterministischer Algorithmus, der auf D folgende Ausgabe liefert:
(1) Falls es ein Element u mit Häufigkeit $a_u > n/2$ gibt, gib u aus.
(2) Falls nicht, gib FALSE aus.

Es sei m die Anzahl verschiedener vorkommender Schlüssel. **Zeige**, dass Algorithmus A mindestens $\Omega(m)$ Bits an Speicherplatz benötigt.

Hinweis: Wende das Kommunikationsmodell an; teile also den Datenstrom in zwei Hälften, eine für Alice, eine für Bob. Benutze dann ein Fooling Argument, d.h. zeige, dass Alice für bestimmte verschiedene Eingaben keine identischen Nachrichten verschicken darf. Um den entsprechenden Nachweis zu führen, konstruiere „passende“ Eingaben für Bob.

Für Binomialkoeffizienten kann die Gleichung $\binom{a}{a/2} = \Theta\left(\frac{2^a}{\sqrt{a}}\right)$ benutzt werden.

Aufgabe 6.3. (4+2+2) *Der Zähleralgorithmus*

Gegeben sei ein Datenstrom $D = x_1, \ldots, x_n$ mit Elementen $x_i \in U$.

Wir schauen uns Algorithmus 4.20 zur Bestimmung von Heavy Hitters aus der Vorlesung näher an. Wir arbeiten mit k vielen Countern. Anfangs ist K die leere Menge.

Für jedes eingehende Element x des Datenstroms tun wird folgendes:

(1a) Falls $x \in K$, inkrementiere den Counter C_x um eins.
(1b) Falls $x \notin K$, lege einen neuen Counter C_x mit Wert 1 an und füge x zur Menge K hinzu.
(2) Gilt nun $|K| > k$, dekrementiere alle Counter um eins und entferne alle Elemente mit Zählerstand 0 aus K.

Nachdem der Datenstrom abgehandelt ist: Gib K als Obermenge aller Heavy Hitters aus. Für jedes Element $u \in K$ gib den Zählerstand C_u als Schätzung für die tatsächliche Häufigkeit a_u aus.

Sei $\theta = 1/(k+1)$ und $\hat{a}_u$ die berechnete Schätzung der Häufigkeit a_u.

a) **Zeige**, dass für alle $u \in K$ gilt:
   (i) $a_u - \hat{a}_u \geq 0$
   (ii) $a_u - \hat{a}_u \leq (n - a_u)/k$

b) **Zeige:** Alle Elemente $u \in U$ mit Häufigkeit $a_u > \theta\, n$ werden ausgegeben.

c) Wenn wir $k = 1$ wählen, können wir ein Element mit Häufigkeit größer als $n/2$ also fehlerfrei aufspüren. Warum ist das kein Widerspruch zu Aufgabe 6.2? **Erkläre!**