Web Grundlagen zum Spidering

Dr. Christian Herta

May 22, 2009

Outline

Uniform Resource Locator URL

- Jede Seite im Internet wird eindeutig über eine URL identifiziert, z.B.
  - `http://www.christianherta.de/informationRetrieval/index.html`
- Eine URL hat mindestens drei Teile:
  - *scheme*: `http`
  - *hostname* (*fully qualified domain name* - *fqdn*): `www.christianherta.de`
  - *resource*: `/informationRetrieval/index.html` (Minimum: /)
- Optional z.B. Portnummer: 16 bit Zahl zur Identifikation eines Services; für *Webpages* Standard Portnummer 80

DNS/DNS Server

- DNS steht für *Domain Name System*; Dienst im Internet
- DNS-Server (Nameserver) übersetzten Hostnamen in Internet Protocol Adressen (IP-Adressen)
- zwei Typen von IP-Adressen
  - IPv4: 32 bit; z.B. `192.168.1.1`
  - IPv6: 128 bit; z.B. `2001:db8:0:0:0:0:1428:57ab`

Outline

HTTP - Request-Response Modell

- Web-Seiten werden (in der Regel) über das HTTP-Protokoll übertragen
- Zwei Rollen bei HTTP-Transaktionen
  - *Client*: stellt Anfrage (*request*)
  - *Server*: liefert Antwort (*response*)
- HTTP liegt in der Anwendungsschicht (basierend auf der verbindungsorientierten Transportschicht TCP)
- HTTP ist in RFC 2616 spezifiziert

HyperText Transfer Protokol HTTP

- Bestandteile von HTTP *Requests*:
  - Anfragezeile, z.B. `GET /images/logo.gif HTTP/1.1`
  - Header mit Meta-Daten zur Anfrage, z.B. Accept-Language: en
  - Leere Zeile
  - Optionaler Nachrichten-Body
- Bestandteile von *Reponses*: wie ähnlich wie *Request*, aber statt Anfragezeile Status

HTTP-Operationen

- Methoden
  - `GET`: Auslesen des Inhalts einer URL
  - `PUT`: Anlegen oder Überschrieben einer *Ressource*
  - `POST`: Anlegen einer URL (Server bestimmt URL!); Anhängen von Daten an URL
  - `HEAD`: Auslesen von Meta-Daten einer URL
  - `OPTIONS`: Auslesen von HTTP-Operationen einer URL
  - `DELETE`: Löschen einer *Ressource*
- Eigenschaften
  - **Sicher** (Verändert nicht den sichtbaren Zustand des Servers): `GET` und `HEAD`
  - **Idempotent** (Wiederholter Aufruf lässt Server im gleichen Zustand): `GET`, `HEAD`, `PUT` und `DELETE`

HTTP Response Code

- Fehler oder OK-Status werden über einen HTTP-Response Code an den Client zurückgeliefert, z.B.
  - 200: OK
  - 404: not found

Outline

(World Wide) Web als Graph

- Web kann als gerichteter Graph (*directed graph*) gesehen werden
  - Knoten: Web-Seiten
  - Kanten: *Hyper-Links*
- *in-links*: Eingehenden Links zu einer Web-Seite *in-degree* bezeichnet
- *out-links*: von einer Web-Seite ausgehende Links
- die Anzahl der ein- bzw. ausgehenden Links wird als *in-* bzw. *out-degree* bezeichnet
- *anchor text*: Textinhalt zwischen <a>-Tags mit `href`-Attributen

Verteilung des *in-link degree*

- *in-link degree* ist nicht Possion-verteilt, wie es bei reiner zufälligen Gleichverteilung zu erwarten wäre
- *in-link degree Power-Law* verteilt

$$n_i \propto i^{-\alpha} \tag{1}$$

mit

- n_i: Anzahl der Seiten mit *in-link degree* i
- $\alpha = 2.1$

Outline

Hintergrund von Spam

- Wirtschaftlichen Bedeutung von Suchmaschinen
- Potentielle Käufer werden durch Suchmaschinen auf die Seiten von Anbietern geführt
- Anbieter wollen das Käufer auf ihre Seiten geführt werden, d.h. dass ihre Seiten zu relevanten Suchbegriffen bzgl. ihres Produkts in den Ergebnislisten weit oben stehen
- Dies führt zu:
  - Spam und Gegenmaßnahmen (Wettrennen)
  - SEO: *Search Engine Optimazation*

Verschiedene Arten von Spam

- (1.Generation von Spam): Wiederholung von Keywords, teilweise nicht sichtbar über Browser (Buchstaben-Farbe gleich Hintergrundfarbe)
- *Cloaking*: Im HTTP *request* wird erkannt, ob es sich um eine *Crawler-* oder *Browser-Client* handelt, und entsprechend unterschiedlicher Inhalt ausgeliefert
- *Doorway Pages*
- *Link Spamming (Link Farmen)*