Plagiatsdetektor

Textindexierung

Daniel Hoske und Stefan Walzer

10. Juni 2013

1-zu-n Plagiatsproblem

- Gegeben: Dokumentmenge D, Query-Dokument q
- Gesucht: Dokumente $d \in D$, die zu q „ähnlich“ sind (Plagiate)

$\Rightarrow$ Ähnlichkeitsmaß $\delta(d, q)$

- Wollen erkennen: Clone (global), Ctrl+C (lokal), Find-Replace, Remix, Mashup

Grober Ablauf

3 Phasen:

- Normalisierung von q
- Plagiatssuche:
  - Ranking: global
  - Fingerprinting: lokal
- Visualisierung ähnlicher Stellen

Orientiert an Hoad und Zobel, 2003

Gliederung

1. Normalisierung

2. Plagiatssuche

3. Visualisierung & Test

4. Erweiterungen

Normalisierung

- Eingabe: unstrukturierter englischer Text q

- Ausgabe: normalisierter englischer Text $\widetilde{q}$, Index von $\widetilde{q}$ nach q

Normalisierung

- Eingabe: unstrukturierter englischer Text q
- parse in Wörter (z.B. `[[:alnum:]]+`) und Sätze
- wandele in Kleinbuchstaben um
- entferne Stoppwörter (the, of, may, …)
- entferne Zitate (zwischen “” oder ‘’)
- (normaliere Beugungen und Synonyme)
- Ausgabe: normalisierter englischer Text $\widetilde{q}$, Index von $\widetilde{q}$ nach q

Gliederung

1. Normalisierung

2. Plagiatssuche

3. Visualisierung & Test

4. Erweiterungen

Plagiatssuche: Ranking

- Ansatz: Plagiate benutzen Wörter mit ähnlichen Frequenzen
- Beispiel für ein Ähnlichkeitsmaß basierend auf Wortfrequenzen:

$$\frac{1}{1+|f_d - f_q|} \sum_{t \in q \cap d} \frac{\log(N/f_t)}{1+|f_{d,t} - f_{q,t}|}$$

- weitere Ad-hoc-Vorschläge für Maße in Hoad und Zobel 2003
- *Invertierter Index* auf D mit Wortfrequenzen
- Laufzeit: $O\left(\sum_{t \in q} f_t\right)$

Plagiatssuche: Fingerprinting

- baue kompakte Beschreibungen (*m minutiae*, Ganzzahlen) der Dokumente und vergleiche diese
- minutia $\approx$ Hash eines Teilstrings
- Ähnlichkeitsmaß = Anzahl gleicher minutiae
- Hashfunktion gibt ähnlichen Texten ähnliche Werte (siehe z.B. Ramakrishna und Zobel, 1997)
- Substringauswahl: alle r-Gramme, ...
- Laufzeit: verschieden...

Gliederung

1. Normalisierung

2. Plagiatssuche

3. Visualisierung & Test

4. Erweiterungen

Visualisierung

- HTML-Ausgabe mit Markierung von Übereinstimmungen
- Ranking: Wörter mit ähnlichen Frequenzen
- Fingerprinting: Substrings mit übereinstimmenden Hashes (oder Sätze, die diese enthalten)

Test

- Scorenormalisierung: $\widehat{\delta}(q,d) := \delta(q,d)/\delta(q,q)$
- vergleiche Werte von verschiedenen Strategien
- greife Beispiele raus (plausibel oder nicht)
- Welche Dokumentsammlungen? (Guttenplag, ...)

Gliederung

1. Normalisierung

2. Plagiatssuche

3. Visualisierung & Test

4. Erweiterungen

Erweiterungen

Exakter Vergleich:

- Suffixarrays (vielleicht Onlinealgorithmus nach Ukkonen)

$\Rightarrow$ spezifischer, weniger robust

Erweiterungen

Exakter Vergleich:

- Suffixarrays (vielleicht Onlinealgorithmus nach Ukkonen)

$\Rightarrow$ spezifischer, weniger robust

1-zu-o Plagiatsproblem:

- erkenne stilistische Veränderungen in einem Dokument

$\Rightarrow$ weist auf Abschreiben hin

Erweiterungen

Exakter Vergleich:

- Suffixarrays (vielleicht Onlinealgorithmus nach Ukkonen)

$\Rightarrow$ spezifischer, weniger robust

1-zu-o Plagiatsproblem:

- erkenne stilistische Veränderungen in einem Dokument

$\Rightarrow$ weist auf Abschreiben hin

Vorberechnung und Visualisierung nicht wesentlich erweiterbar...

Ende

Vielen Dank für eure Aufmerksamkeit.
Habt ihr noch irgendwelche Anmerkungen oder Fragen?