

Bloom Filter

1. Idee des Bloomfilters

Mithilfe des Bloomfilter kann schnell festgestellt werden ob ein Element in einer Gruppe vorhanden ist oder nicht. Generell wird dazu ein weitere Schritt vor der eigentlichen Abfrage eingefügt, in dem die Gruppe von Elementen bzw. die Datenstruktur gefiltert wird.

Die Repräsentation einer Liste mit Elementen die ein Kriterium erfüllen (Zur Gruppe gehören oder nicht) wird repräsentiert durch ein Bit-Array. Ist ein Element in der Gruppe also ist das Element an der Stelle 1 wird das Element an der Stelle des Hashwertes auf 1 gesetzt.

1. Vorteile

- Keine False Negatives möglich.
- Kann Performance stark verbessern.
- Eignet sich für sehr grosse Datenmengen, weil Daten selbst nicht gespeichert werden müssen.

2. Nachteile

- Eignen sich nur für spezifische Probleme, bei denen festgestellt werden soll ob ein Element zu einer Gruppe gehört oder nicht.
- Zusätzlicher Schritt bei der Erstellung einer neuen Zusätzlichen Datenstruktur nötig. Aber e

2. Konkretes Beispiel aus der Praxis

Ungültige Datenbankabfragen auf nicht existierende Zeilen in einer Datenbank vermeiden. Dadurch wird die Performance verbessert,

da jeder Zugriff auf eine nicht existierende Datenbankzeile Ressourcen braucht.

Ein konkretes Beispiel dafür ist Google Hochleistungs Datenbank System Bigtable.

3. Beschreibung Transformation Hashwert -> Indexwert

Mithilfe der in der Aufgabenstellung vorgegebenen Hashfunktion (murmur3_123) erhalten wir die Hashwerte der jeweiligen Worte. Dieser Hashwert kann allerdings grösser als der gewählte Filtergrösse m sein. Deshalb wird die Modulofunktion auf ihn angewendet und der Rest als entsprechender Wert gesetzt.

4. Beschreibung Fehlerwahrscheinlichkeitstest

Da keine False Negatives möglich sind, können wir die Fehlerwahrscheinlichkeit einfach anhand der False Positives im Verhältnis zu allen getesteten Wörtern ausrechnen.

Die Ergebnisse bei uns waren