

MASTERARBEIT

Organisation von Bloom-Filtern zur effizienten k-nächste-Nachbarn-Suche in kontextzentrischen sozialen Netzen

Judith Greif



MASTERARBEIT

Organisation von Bloom-Filtern zur effizienten k-nächste-Nachbarn-Suche in kontextzentrischen sozialen Netzen

Judith Greif

Aufgabenstellerin: Prof. Dr. Claudia Linnhoff-Popien

Betreuer: Mirco Schönfeld
Dr. Martin Werner

Abgabetermin: 26. Juli 2016



Hiermit versichere ich, dass ich die vorliegende Masterarbeit selbständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel verwendet habe.

München, den 26. Juli 2016

.....
(*Unterschrift der Kandidatin*)

Abstract

[illegible]

Inhaltsverzeichnis

1	Einleitung	1
2	Hintergrund	3
2.1	Bloom-Filter	3
2.1.1	Grundlagen	3
2.1.2	Mathematische Prinzipien und Operationen	3
2.2	Indexstrukturen	3
2.2.1	B- und B+-Bäume	4
2.3	AMBIENCE	4
3	Vewandte Themen	5
4	Implementierung	7
5	Evaluation	9
6	Zusammenfassung und Ausblick	11
	Literaturverzeichnis	13

1 Einleitung

Die digitale Kommunikation hat im letzten Jahrzehnt einen rapiden Wandel erlebt. Soziale Online-Netzwerke haben an Bedeutung gewonnen und zu neuen Kommunikationsmustern im Internet geführt. Sofortnachrichtendienste und Instant-Messenger beginnen traditionellere Kommunikationsformen wie SMS und Telefonie zu ersetzen. Das Smartphone hat das Mobiltelefon als mobiles Endgerät fast vollständig abgelöst. Feststehende Desktop-Rechner mit einem gleichbleibenden Netzzugang sind außerhalb von Firmen und Bildungseinrichtungen rückläufig. Demgegenüber sind Notebooks und Tablets auf Erfolgsweg¹.

Die Kommunikation im Internet ist jedoch nach wie vor Ende-zu-Ende- beziehungsweise adressbasiert. Das spiegelt sich im Aufbau der bestehenden sozialen Online-Netze wider: Kommunikation basiert darin auf Online-Freundschaft. Mobilität und spezifischer Kontext der Mitglieder werden kaum berücksichtigt. In der Realität verlieren die Webbrowser-Schnittstellen der sozialen Netzwerke jedoch an Bedeutung. So veröffentlichte Facebook 2014 eine Studie zum Nutzungsverhalten von US-Bürgern in einer Multigeräte-Welt². Danach nutzten 60% der Erwachsenen in den USA täglich mindestens zwei Endgeräte, knapp 25% sogar drei Geräte. Mehr als 40% begannen den Tag mit einem Gerät und beendeten ihn mit einem anderen. Das Smartphone ist dabei das Gerät, das am häufigsten mitgenommen wird und eine zentrale Rolle in der Kommunikation per E-Mail und in sozialen Netzen einnimmt. Das Szenario, in dem Alice vor ihrem Desktop-Rechner zu Hause oder im Büro sitzt und über die Webbrowser-Schnittstelle Nachrichten an ihren Facebook-Freund Bob schreibt, gehört der Vergangenheit an. Stattdessen verwendet Alice wohl eher ein Tablet, ein Notebook und ein Smartphone und kommuniziert mit Bob je nach Aufenthaltsort und Kontext ganz unterschiedlich.

So stellt sich die Frage nach einer Neuorientierung von sozialen Online-Netzen: Nicht nur in der praktischen Umsetzung, also durch unterschiedliche Schnittstellen und neue Funktionalitäten, sondern im Sinne eines Paradigmenwechsels auf einer Meta-Ebene. Ein kontextzentrisches soziales Netz³ gründet strukturell nicht auf Ende-zu-Ende-Kommunikation, adressbasiertem Routing und einem gleich bleibenden Netzzugang. Die Kommunikation basiert allein auf Kontext-Ähnlichkeit, nicht auf virtueller Freundschaft. Diese Überlegungen sind z.B. in das soziale Online-Netz AMBIENCE eingeflossen⁴. Nachrichten werden auf Grund von zeitlicher und räumlicher Ähnlichkeit ausgetauscht, Sender und Empfänger bleiben anonym. Ein solches Netz erfordert eine gänzlich andere Kommunikationsstruktur: Nachrichten werden nicht aktiv von einem Sender für einen spezifischen Empfänger

¹Vgl. z.B. <http://www.fairrank.de/blog/wissenswertes-aus-der-branche/847-weiterhin-ruecklaeufige-verkaufszahlen-bei-notebooks-und-desktop-rechnern.html>

²Vgl. <https://www.facebook.com/business/news/Finding-simplicity-in-a-multi-device-world>

³Der Begriff *Context awareness* wurde von Schilit et al. 1994 eingeführt und bezeichnet die Nutzung von Kontextinformationen als Informationsquelle für Anwendungen und Netzwerke. In diesem Zusammenhang ist vor allem *Context awareness* bezüglich des Nutzers von Interesse (vgl. [WDS15] für eine exakte Abgrenzung).

⁴Vgl. ebd. sowie 2 für eine detaillierte Darstellung.

verfasst und an ihn verschickt. Stattdessen kann ein Sender eine Nachricht verfassen und z.B. an einem WiFi-Access Point hinterlegen. Mitglieder des Netzwerks, die sich in der Nähe des Access Points aufhalten, können die dort vorhandenen Nachrichten mit gezielten Anfragen durchsuchen und die zum jeweiligen Kontext ähnlichsten Nachrichten abrufen.

Damit stellt sich die Frage: Wie lassen sich die Nachrichten an einem Host, also z.B. an einem WiFi-Access Point, so organisieren, dass die k ähnlichsten Nachrichten möglichst schnell und effizient gefunden werden? Das ist von entscheidender Bedeutung, wenn ein soziales Netz wachsen und über den Status eines Prototypen hinaus erfolgreich sein soll. Mengentheoretisch handelt es sich dabei um das Problem der k -nächsten-Nachbarn-Suche, die zu einem Anfragepunkt die k ähnlichsten Elemente einer Menge, hier bestehend aus den Nachrichten an einem Host, liefern soll.

Daneben ist die Form der Nachrichten von zentraler Bedeutung. Wie können Multimedia-Nachrichten wie Bilder, Sprachnachrichten oder Textdateien effizient, einheitlich und sicher vor unbefugtem Zugriff hinterlegt und verschickt werden? Gleichzeitig muss irgendwie ihre Ähnlichkeit oder Unähnlichkeit ermittelt werden können, d.h. es muss ein Ähnlichkeitsmaß geben, das der k -nächste-Nachbarn-Suche zu Grunde liegt. AMBIENCE bedient sich dazu einer Bloom-Filter-Konstruktion. Eine Nachricht wird als Menge von Zeichenketten aufgefasst, die mit geeigneten Hashfunktionen in ein Bit-Array fester Länge eingefügt werden. Ähnlichkeit zwischen Nachrichten ist damit als Ähnlichkeit zwischen Bloom-Filtern definiert. Nachrichten werden in Form von Bloom-Filtern kodiert, gespeichert und verglichen. Als Ähnlichkeitsmaß wird dabei die Jaccard-Distanz verwendet, die die Ähnlichkeit zwischen zwei Mengen beschreibt.

Die folgende Arbeit betrachtet daher die Organisation von Bloom-Filtern zur effizienten k -nächste-Nachbarn-Suche in kontextzentrischen sozialen Netzen. Das folgende Kapitel 1 gibt einen Überblick über mengentheoretische und probabilistische Grundlagen, verwendete Datenstrukturen und den Aufbau von AMBIENCE. Kapitel 3 gibt einen Überblick über verwandte Arbeiten und Fragestellungen. Das entwickelte Verfahren wird anschließend in Kapitel 4 ausführlich dargestellt. Kapitel 5 vergleicht die Implementierung mit dem bisherigen, nicht optimierten Ansatz. Das abschließende Kapitel 6 fasst die Arbeit zusammen, verweist auf Grenzen und mögliche zukünftige Arbeiten.

Referenzen bis jetzt: [AT06], [ADI⁺12], [BMS07], [BM04], [BCM02], [DWM10], [HP94], [LC86], [Naf05], [QLC14], [RK14], [SBE⁺12], [Sch13], [SW14], [STT⁺09], [WDS15], [YL02], [Zha12], [ZJW04], [Jan95].

2 Hintergrund

2.1 Bloom-Filter

2.1.1 Grundlagen

Attenuated Bloom Filter: [SS11]: 316 and 318

Counting Bloom Filter: [FCAB00]

Compressed Bloom Filter: [Mit02]

2.1.2 Mathematische Prinzipien und Operationen

2.2 Indexstrukturen

To support query processing and operations in an efficient manner, the internal layer of a database system uses specific data structures and memory methods. These are called *index structures*. They organize the data to support the required operations using its *indices*.

An *index* (also called *directory*) of a file holds information about its structure. A *file* in this context refers to an entire data structure, i.e. an array, a search tree etc.. One can differentiate between three classes of index structures depending on the manner of organization:

1. ***Data-organizing index structures*** are used to organize the actual amount of data. They mostly rely on *search trees*.
2. ***Space-organizing index structures*** are used to organize the space that holds the data. They make use of *dynamic hashing*.
3. ***Hybrid index structures*** are a combination of both classes. They are based on *hash trees*.

There are several requirements for an index structure in order to meet its purpose.

- *Efficient search:* A data query on the index structure should return an answer in optimal time, i.e. the query should be directed to the page or pages that contain the queried data using as little steps as possible.
- *Dynamic insertion, deletion and modification of data sets:* The amount of data to be organized changes over time, leading to alterations in the index structure as well. Any implementation requiring a complete reorganization of the index structure on insertion, deletion or modification of data sets is clearly unacceptable. Any of these operations may therefore only lead to local changes.
- *Local preservation of order:* If there are some data sets the keys of which are successors within the applied order relation (i.e. the less-or-equal relation on non-negative integers), this order should be preserved within the index structure. This holds for search trees but it does not hold for linear hashing. It is clearly of great importance regarding the application scenario in question.

- *Efficient use of space*: This requirement is of great importance for real-world applications. So far the reference implementation *AMBIENCE* has served as a proof of concept. Accordingly the number of messages, i.e. the amount of data to be queried, has been relatively small compared to a real-world scenario. Therefore the memory requirements of any index structure within the current scenario that represents the actual amount of data is unlikely to require vast amounts of memory. However, keeping in mind future application scenarios for *AMBIENCE*, efficient use of space cannot be entirely discarded.

Further requirements include *feasability* and *implementation cost*. Any index structure aiming at a real-world implementation such as *AMBIENCE* naturally has to be feasible, so this requirement will be overlooked in the following. As this work clearly has a scholarly background, not an industrial one, the implementation cost will be disregarded as well. [OW12]

2.2.1 B- und B+-Bäume

[Knu98]

2.3 AMBIENCE

[WDS15].

3 Vewandte Themen

4 Implementierung

5 Evaluation

6 Zusammenfassung und Ausblick

Literaturverzeichnis

- [ADI⁺12] AHLGREN, BENGT, CHRISTIAN DANNEWITZ, CLAUDIO IMBRENDA, DIRK KUTSCHER und BÖRJE OHLMAN: *A Survey of Information-Centric Networking*. Communications Magazine, IEEE, 50(7):26–36, 2012.
- [AT06] AGARWAL, SACHIN und ARI TRACHTENBERG: *Approximating the number of differences between remote sets*. In: *Information Theory Workshop, 2006. ITW '06 Punta del Este*. IEEE, Seiten 217–221, March 2006.
- [BCM02] BYERS, JOHN, JEFFREY CONSIDINE und MICHAEL MITZENMACHER: *Fast Approximate Reconciliation of Set Differences*. In: *BU Computer Science TR*, Seiten 2002–2019, 2002.
- [BM04] BRODER, ANDREI und MICHAEL MITZENMACHER: *Network Applications of Bloom Filters: A Survey*. Internet Mathematics, 1(4):485–509, 2004.
- [BMS07] BAYARDO, ROBERTO, YIMING MA und RAMAKRISHNAN SRIKANT: *Scaling Up All Pairs Similarity Search*. In: *Proceedings of the 16th international conference on World Wide Web*, Seiten 131–140. ACM, 2007.
- [DWM10] DÜRR, MICHAEL, MARTIN WERNER und MARCO MAIER: *Re-Socializing Online Social Networks*. In: *Green Computing and Communications (GreenCom), 2010 IEEE/ACM International Conference on & International Conference on Cyber, Physical and Social Computing (CPSCoM)*, Seiten 786–791. IEEE, 2010.
- [FCAB00] FAN, LI, PEI CAO, JUSSARA ALMEIDA und ANDREI BRODER: *Summary Cache: A Scalable Wide-Area Web Cache Sharing Protocol*. IEEE/ACM Transactions on Networking (TON), 8(3):281–293, 2000.
- [HP94] HELLERSTEIN, JOSEPH und AVI PFEFFER: *The RD-Tree: An Index Structure for Sets*. Technischer Bericht, University of Wisconsin-Madison, Computer Sciences Department, 1994.
- [Jan95] JANNINK, JAN: *Implementing Deletion in B+-Trees*. ACM Sigmod Record, 24(1):33–38, 1995.
- [Knu98] KNUTH, DONALD: *The art of computer programming, Volume 3, Sorting and searching*. Addison Wesley Longman, 1998.
- [LC86] LEHMAN, TOBIN und MICHAEL CAREY: *A Study of Index Structures for Main Memory Database Management Systems*. In: *Proc. VLDB*, 1986.
- [Mit02] MITZENMACHER, MICHAEL: *Compressed Bloom Filters*. IEEE/ACM Transactions on Networking (TON), 10(5):604–612, 2002.
- [Naf05] NAFE, CLEMENS: *Indexierung lokaler Daten in Peer-to-Peer-Netzwerken*. Diplomarbeit, Universität Rostock, 2005.

- [OW12] OTTMANN, THOMAS und PETER WIDMAYER: *Algorithmen und Datenstrukturen*. Spektrum Akademischer Verlag, 5 Auflage, 2012.
- [QLC14] QIAO, YAN, TAO LI und SHIGANG CHEN: *Fast Bloom Filters and their Generalization*. Parallel and Distributed Systems, IEEE Transactions on, 25(1):93–103, Januar 2014.
- [RK14] RUPPEL, PETER und AXEL KÜPPER: *Geocookie: A Space-Efficient Representation of Geographic Location Sets*. Journal of Information Processing, 22(3):418–424, 2014.
- [SBE⁺12] SARWAT, MOHAMED, JIE BAO, AHMED ELDAWY, JUSTIN LEVANDOSKI, AMR MAGDY und MOHAMED MOKBEL: *Sindbad: A Location-(B)ased Social Networking System*. In: *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, Seiten 649–652. ACM, 2012.
- [Sch13] SCHNELL, RAINER: *Getting Big Data But Avoiding Big Brother*. WP-GRLC, 2, 2013.
- [SS11] SAKUMA, HIROSHI und FUMIAKO SATO: *Evaluation of the Structured Bloom Filters Based on Similarity*. In: *Advanced Information Networking and Applications (AINA), 2011 IEEE International Conference on*, Seiten 316–323, März 2011.
- [STT⁺09] SHIRAKI, TORU, YUICHI TERANISHI, SUSUMU TAKEUCHI, KANAME HARUMOTO und SHOJIRO NISHIO: *A Bloom Filter-Based User Search Method Based on Movement Records for P2P Network*. In: *Applications and the Internet, 2009. SAINT '09. Ninth International Symposium on*, Seiten 177–180. IEEE, Juli 2009.
- [SW14] SCHÖNFELD, MIRCO und MARTIN WERNER: *Node Wake-(U)p via OVFS-Coded Bloom Filters in Wireless Sensor Networks*. In: *Ad Hoc Networks*, Seiten 119–134. Springer, 2014.
- [WDS15] WERNER, MARTIN, FLORIAN DORFMEISTER und MIRCO SCHÖNFELD: *AMBIENCE: A Context-Centric Online Social Network*. In: *12th IEEE Workshop on Positioning, Navigation and Communications (WPNC '15)*, 2015.
- [YL02] YANG, CONGJUN und KING-IP LIN: *An Index Structure for Improving Closest Pairs and Related Join Queries in Spatial Databases*. In: *Database Engineering and Applications Symposium, 2002. Proceedings. International*, Seiten 140–149. IEEE, 2002.
- [Zha12] ZHANG, ZHENGHAO: *Analog Bloom Filter: Efficient simultaneous query for wireless networks*. In: *Global Communications Conference (GLOBECOM), 2012 IEEE*, Seiten 3340–3346. IEEE, 2012.
- [ZJW04] ZHU, YIFENG, HONG JIANG und JUN WANG: *Hierarchical Bloom Filter Arrays (HBA): A Novel, Scalable Metadata Management System for Large Cluster-based Storage*. In: *Cluster Computing, 2004 IEEE International Conference on*, Seiten 165–174. IEEE, 2004.