Privacy Preserving Record Linkage on Flink

Big Data Praktikum

T. Hornoff M. Franke

Abteilung Datenbanken
Institut für Informatik
Fakultät für Mathematik und Informatik
Universität Leipzig

04.08.2016

Inhalt

Privacy Preserving Record Linkage (PPRL)

- Objekt-Matching mit verschlüsselten Daten
- Verbindung von Datensätzen aus mehreren Datenquellen
- Herausforderungen
  - Schutz personenbezogener Daten
  - Wachsende Datenmengen (Skalierbarkeit)
  - Qualität der Daten
- Zahlreiche Anwendungsgebiete
  - Gesundheitsfürsorge
  - Betrugserkennung
  - Finanzinstitutionen & Banken
  - Unternehmensanwendungen

Apache Flink

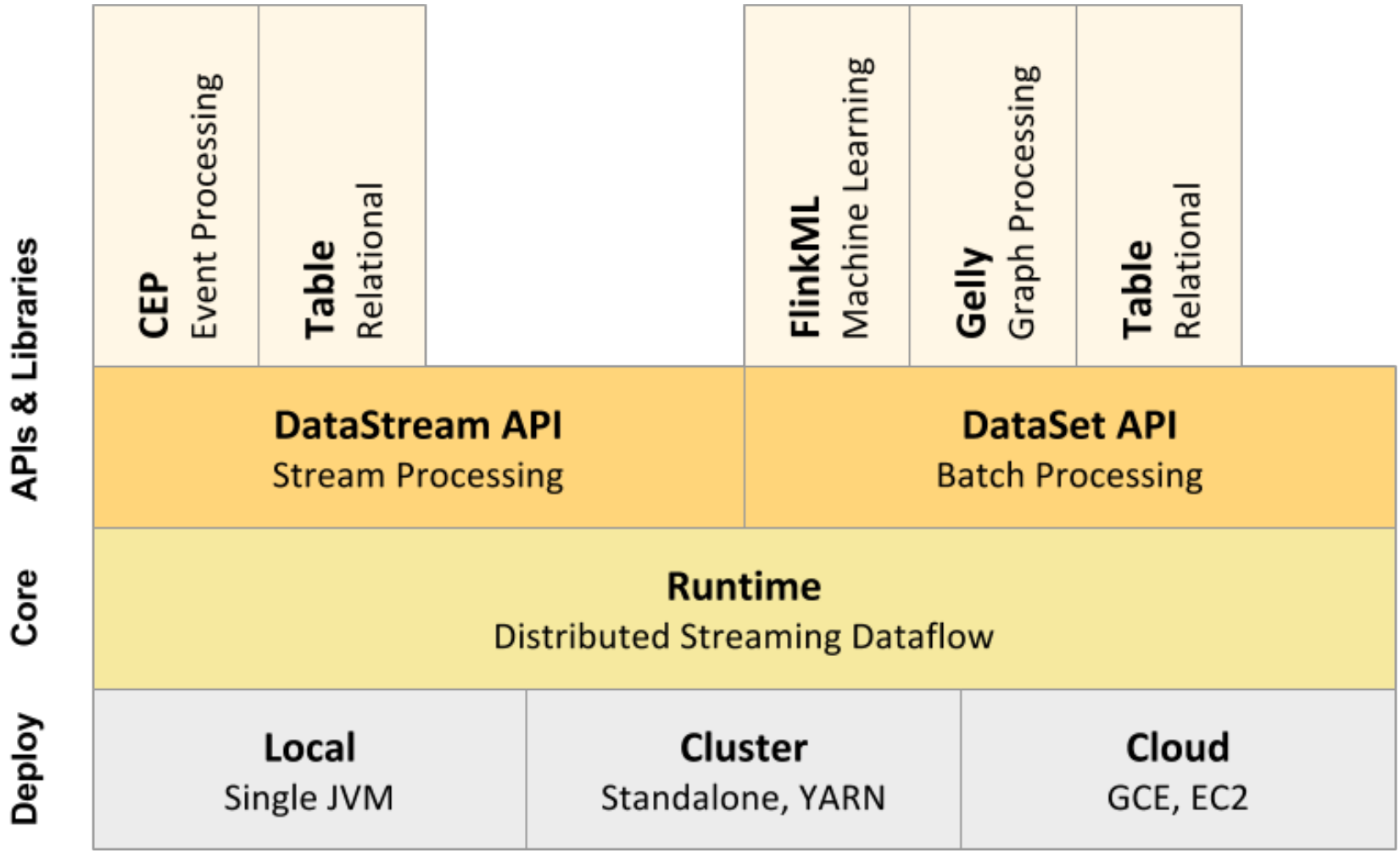
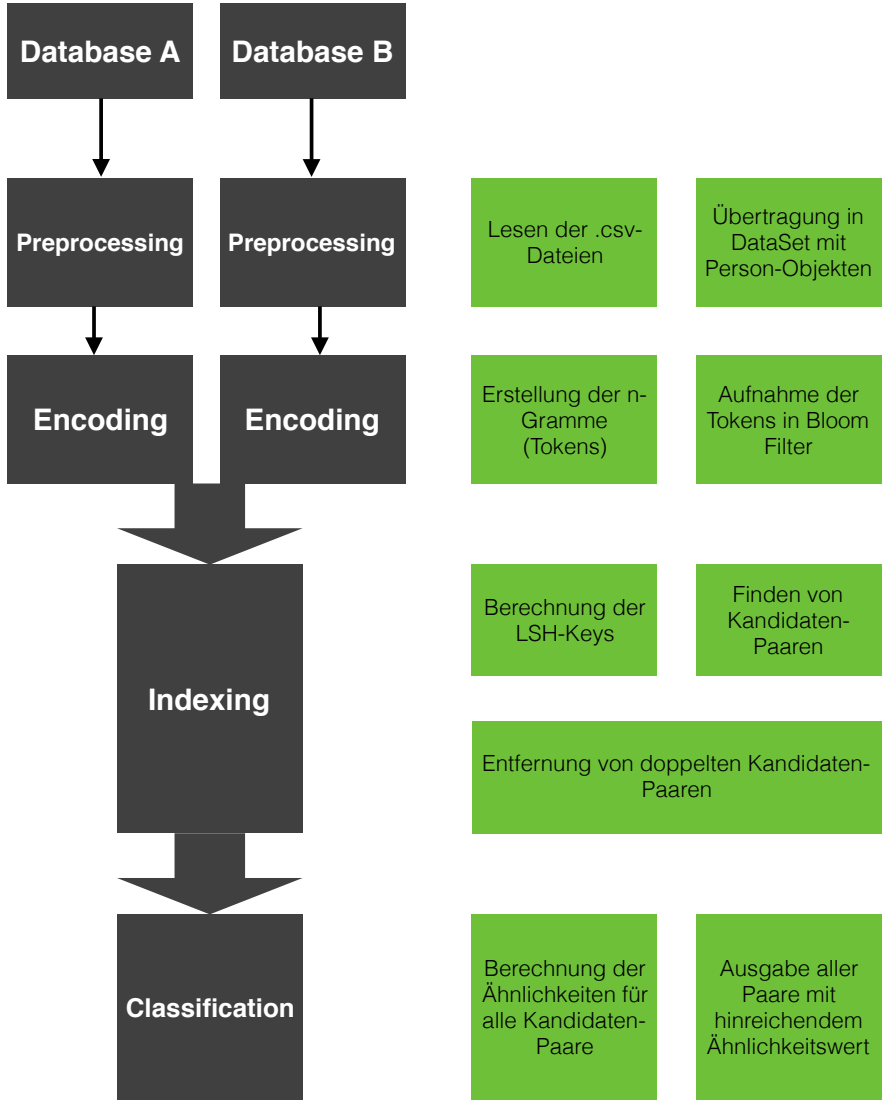


Abbildung: https://flink.apache.org/img/flink-stack-frontpage.png

Inhalt

Gesamt-Prozess



Preprocessing - Lesen der Daten

Lesen der Daten	Lesen der .csv-Dateien	Übertragung in DataSet mit Person-Objekten
Erstellung der Bloom Filter	Erstellung der n-Gramme (Tokens)	Aufnahme der Tokens in Bloom Filter
Blocking mit LSH	Berechnung der LSH-Keys	Finden von Kandidaten-Paaren
Duplikat-entfernung	Entfernung von doppelten Kandidaten-Paaren	
Ähnlichkeits-berechnung	Berechnung der Ähnlichkeit für alle Kandidaten-Paare	Ausgabe aller Paare mit hinreichendem Ähnlichkeitswert

Preprocessing - Lesen der Daten

- Nutzung verschiedener Datensätze
  - North Carolina Voter Registration Database
    - Erreichbar unter `http://dl.ncsbe.gov/`
    - Verschieden große Datensätze
    - Personenbezogene Daten (z.B. Name, Adresse, Alter, Geschlecht, ...)

county_cd	first_name	last_name	middle_name	name_suffix_lbl	res_addr1	res_addr2	res_state	res_city	res_zip	res_zip4	race
DURHA	DEMARCO	HARRIS	DONTEZ		1309 HUDSON AVE	APT B15	NC	DURHAM	27705	3372	B
MECKL	VIOLENA	WORK	ALIVNETTA		1424 PRESSLEY RD	APT 7	NC	CHARLOTTE	28217	934	B
PASQU	WILLIAM	BONDS	JAMES		1403 RIVER RD	LOT 153	NC	ELIZABETH CITY	27909	6721	B
PITT	QUENTIN	ANDREWS	MONTREL		1120 DOROTHY LN		NC	GREENVILLE	27834	9308	B

  - Nutzung von generierten Daten
    - Beginn der Übersetzung eines DataCorruptors[1] in Java mit Flink

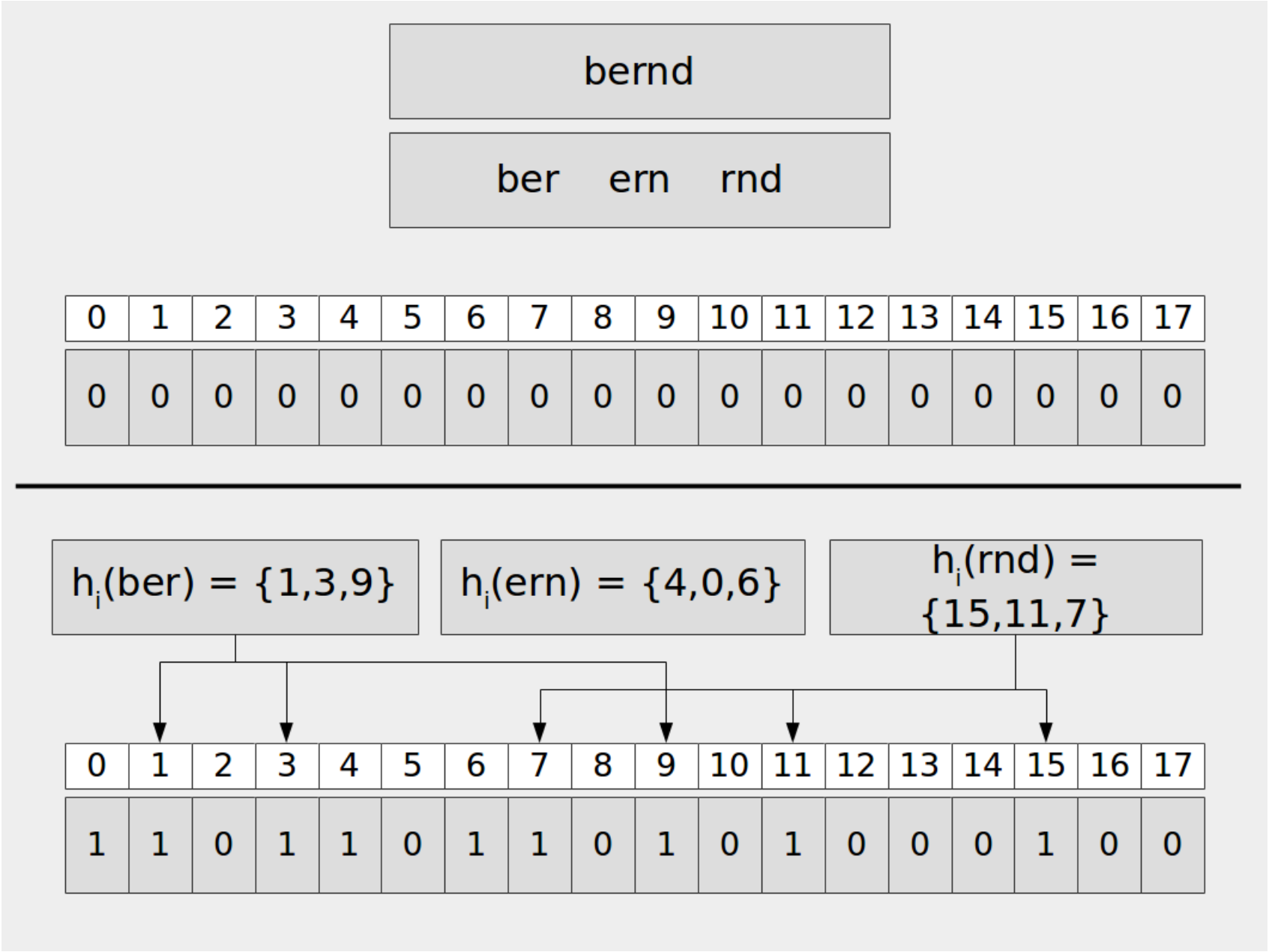
[1] Peter Christen, Dinusha Vatsalan, Flexible and extensible generation and corruption of personal data, ACM, 2013, CIKM '13 Proceedings of the 22nd ACM international conference on Information & Knowledge Management, `http://dl.acm.org/citation.cfm?id=2507815`

Preprocessing - Lesen der Daten: Ergebnis

- Ausgabe der Person-Objekte in der Form (*Person*)

```
Person [id=1:-1642117313, firstName=DEMARCO, middleName=, lastName=HARRIS, addressPartOne=1309 HUDSON AVE, addressPartTwo=, state=, city=,
zip=, genderCode=, age=, birthday=, ethnicCode=]

Person [id=1:1097147679, firstName=PEGGY, middleName=, lastName=TINGLE, addressPartOne=1008 CORBETT AVE NE, addressPartTwo=, state=, city=,
zip=, genderCode=, age=, birthday=, ethnicCode=]

Person [id=1:65241860, firstName=THERESA, middleName=, lastName=PITTMAN, addressPartOne=417 E FRANKLIN ST, addressPartTwo=, state=, city=,
zip=, genderCode=, age=, birthday=, ethnicCode=]

Person [id=1:715220210, firstName=MARGARET, middleName=, lastName=PEARSON, addressPartOne=401 ROBERSON ST, addressPartTwo=, state=, city=,
zip=, genderCode=, age=, birthday=, ethnicCode=]

Person [id=2:-1512168112, firstName=GRADY, middleName=, lastName=WILLIAMS, addressPartOne=821 AVALON RD, addressPartTwo=, state=, city=,
zip=, genderCode=, age=, birthday=, ethnicCode=]
```

Encoding - Bloom Filter



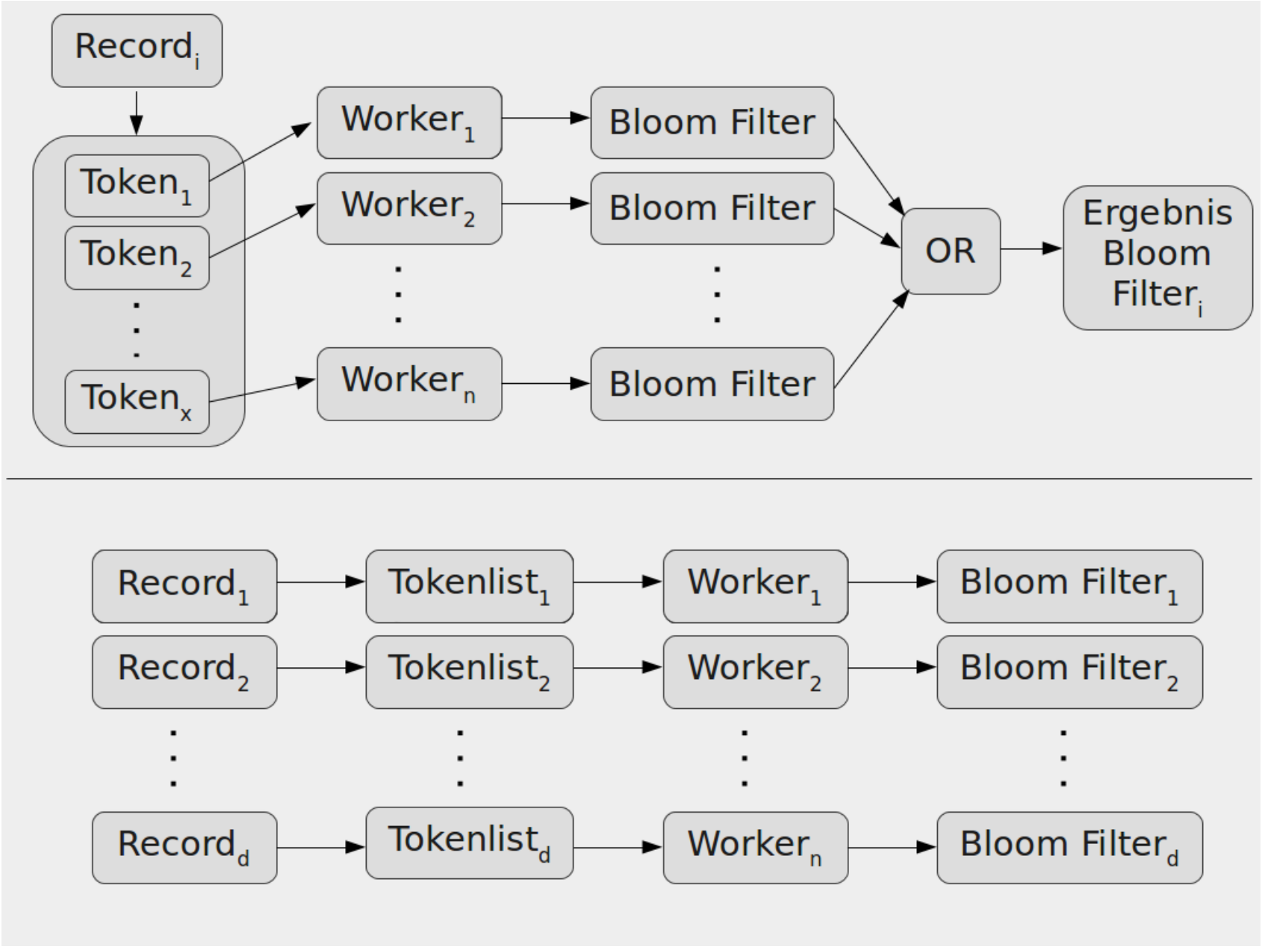
Encoding - Erstellung der Bloom Filter

Lesen der Daten	Lesen der .csv-Dateien	Übertragung in DataSet mit Person-Objekten
Erstellung der Bloom Filter	**Erstellung der n-Gramme (Tokens)**	**Aufnahme der Tokens in Bloom Filter**
Blocking mit LSH	Berechnung der LSH-Keys	Finden von Kandidaten-Paaren
Duplikat-entfernung	Entfernung von doppelten Kandidaten-Paaren	
Ähnlichkeits-berechnung	Berechnung der Ähnlichkeit für alle Kandidaten-Paare	Ausgabe aller Paare mit hinreichendem Ähnlichkeitswert

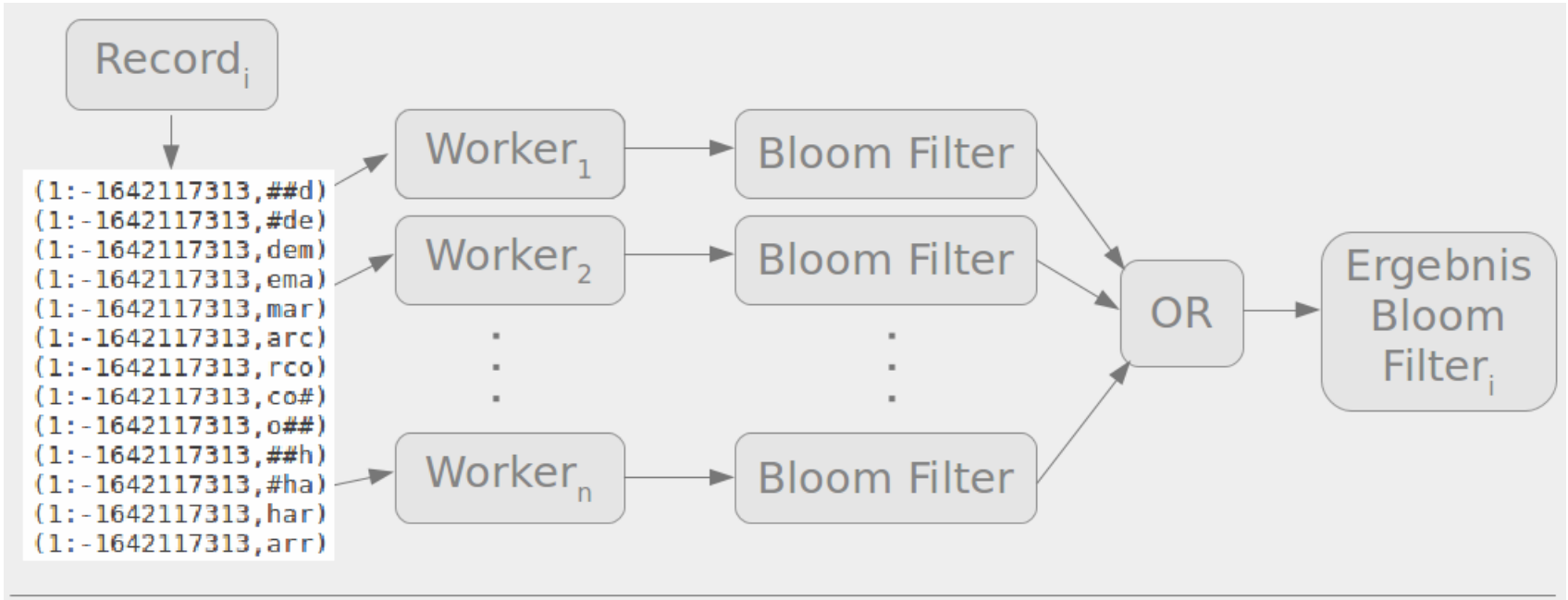
Encoding - Erstellung der Bloom Filter

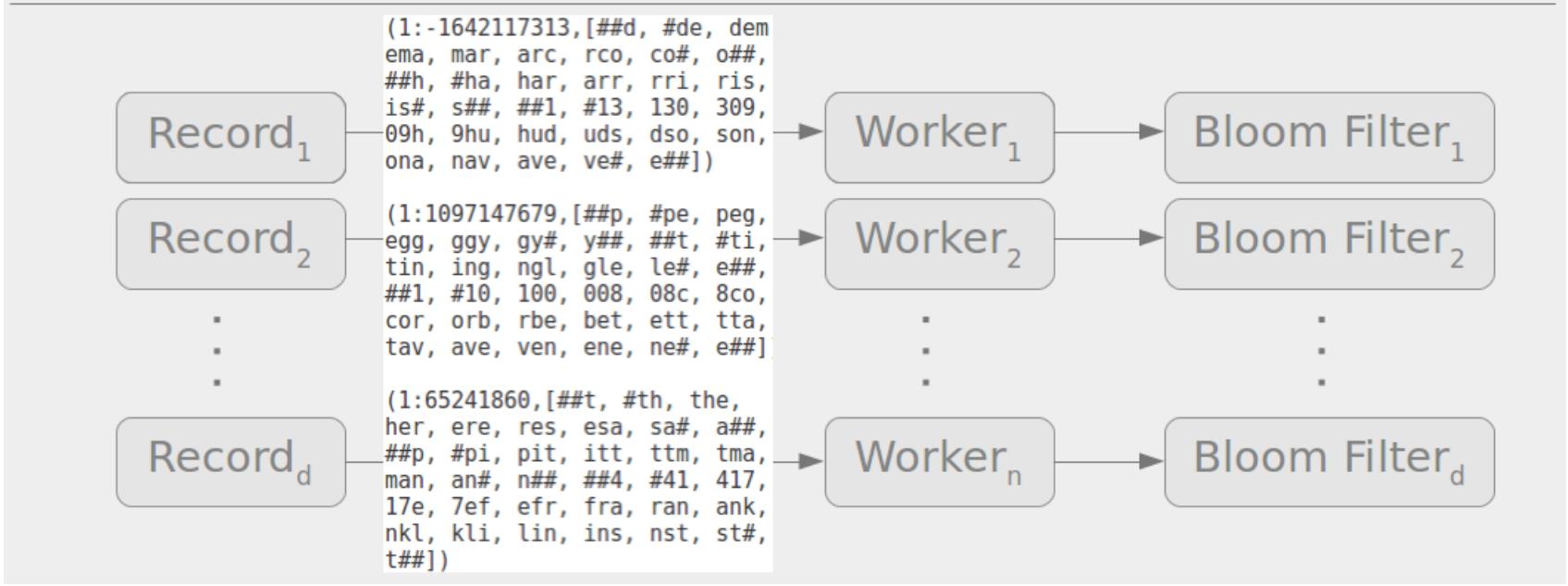
- Bloom Filter Implementierung
  - Einstellbare Länge und Anzahl der Hashfunktionen
- Auswahl der Felder (Person-Attribute)
- Auswahl der Tokenlänge (Größe der n-Gramme)
- 2 Arten der Parallelisierung für Abbildung Token $\mapsto$ Bloom Filter
  - Parallelisierung auf Token-Ebene
  - Parallelisierung auf Record-Ebene

Encoding - Erstellung der Bloom Filter: Parallelisierung



Encoding - Erstellung der Bloom Filter: Parallelisierung



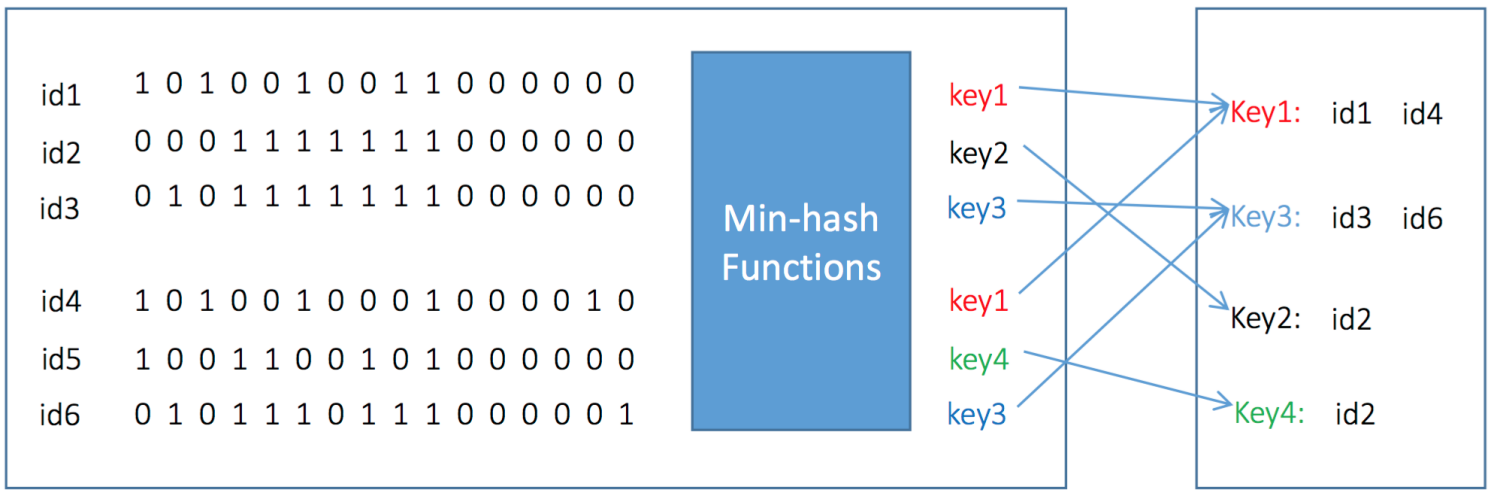


Encoding - Erstellung der Bloom Filter: Ergebnis

- Ausgabe der Bloom Filter in der Form (ID_{Person}, $BloomFilter$)

```
(1:-1764211470,{1, 2, 17, 19, 21, 23, 27, 29, 35, 37, 38, 40, 42, 44, 45, 46, 47, 53, 61, 62, 63, 65, 68, 74, 77, 78, 87, 102, 103, 105, 106, 107, 108, 112,
119, 123, 124, 127, 131, 134, 138, 140, 141, 142, 143, 146, 151, 152, 157, 159, 160, 161, 164, 165, 166, 167, 168, 172, 175, 178, 181, 182, 185, 189, 191, 193,
194, 196, 198, 200, 203, 206, 207, 208, 214, 215, 216, 219, 220, 223, 232, 233, 235, 237, 245, 247, 248, 250, 253, 254, 256, 257, 259, 260, 262, 263, 266, 275,
279, 281, 283, 285, 296, 301, 308, 309, 311, 313, 315, 320, 321, 323, 329, 331, 333, 334, 339, 340, 342, 343, 344, 346, 351, 354, 355, 356, 357, 359, 363, 367,
369, 373, 375, 378, 380, 383, 384, 387, 391, 393, 394, 395, 397, 398, 400, 402, 406, 412, 413, 414, 415, 417, 418, 419, 427, 428, 431, 434, 435, 436, 441, 449,
457, 459, 461, 463, 464, 465, 469, 470, 475, 477, 480, 487, 488, 489, 490, 493, 497, 499, 504, 506, 508, 509, 510, 511, 512, 514, 517, 522, 528, 529, 535, 536,
537, 540, 545, 549, 550, 553, 554, 555, 556, 557, 562, 567, 568, 569, 571, 572, 582, 583, 587, 591, 597, 599, 601, 603, 604, 605, 608, 613, 614, 618, 619, 621,
623, 624, 631, 633, 635, 639, 644, 645, 646, 652, 657, 659, 660, 661, 662, 663, 665, 666, 667, 670, 671, 672, 673, 675, 685, 689, 691, 695, 699, 702, 707, 708,
711, 714, 715, 717, 722, 723, 729, 731, 738, 739, 740, 741, 745, 749, 750, 751, 754, 756, 757, 758, 762, 767, 770, 774, 775, 780, 784, 785, 787, 791, 795, 801,
805, 815, 820, 821, 827, 829, 830, 831, 833, 834, 836, 840, 843, 844, 846, 847, 849, 851, 853, 854, 856, 858, 862, 863, 864, 866, 868, 871, 873, 877, 880, 881,
882, 883, 884, 885, 887, 889, 891, 894, 901, 906, 907, 908, 909, 917, 918, 920, 927, 935, 936, 939, 943, 944, 946, 948, 949, 951, 952, 953, 955, 957, 961, 968,
969, 970, 977, 980, 983, 984, 985, 989, 990})

(1:1383042088,{2, 3, 4, 10, 12, 14, 15, 18, 20, 21, 24, 28, 29, 31, 34, 35, 36, 38, 41, 42, 44, 45, 48, 51, 54, 57, 61, 63, 64, 65, 67, 69, 71, 72, 75, 78, 85,
87, 88, 89, 90, 95, 96, 98, 100, 103, 104, 105, 107, 110, 112, 115, 116, 121, 125, 128, 136, 142, 144, 146, 147, 148, 149, 153, 159, 169, 170, 171, 173, 178,
185, 187, 189, 191, 193, 197, 200, 204, 205, 207, 208, 215, 216, 217, 220, 223, 225, 233, 234, 236, 238, 243, 244, 246, 247, 250, 253, 254, 255, 262, 263, 266,
268, 271, 278, 280, 281, 285, 292, 295, 298, 300, 302, 303, 304, 306, 307, 309, 311, 315, 316, 318, 319, 320, 321, 323, 325, 329, 332, 337, 340, 341, 343, 344,
346, 348, 352, 353, 354, 356, 357, 359, 361, 363, 365, 367, 374, 375, 376, 377, 378, 380, 382, 384, 385, 386, 389, 390, 393, 397, 400, 401, 404, 408, 409, 410,
411, 413, 414, 420, 423, 424, 425, 426, 428, 435, 440, 443, 446, 447, 450, 452, 457, 458, 459, 460, 465, 466, 467, 468, 471, 476, 479, 486, 489, 491, 496, 498,
499, 502, 504, 505, 508, 511, 512, 514, 519, 520, 522, 525, 527, 528, 534, 536, 537, 538, 539, 540, 545, 547, 549, 552, 554, 555, 558, 561, 564, 565, 567, 568,
571, 575, 578, 580, 585, 588, 591, 595, 596, 597, 598, 600, 601, 610, 613, 617, 619, 623, 624, 626, 627, 630, 632, 633, 635, 637, 641, 649, 650, 656, 659, 661,
662, 667, 668, 673, 675, 679, 681, 686, 687, 688, 691, 692, 695, 698, 699, 700, 701, 702, 706, 707, 711, 713, 716, 721, 723, 728, 730, 733, 736, 739, 745, 746,
747, 748, 750, 752, 754, 757, 759, 760, 761, 765, 767, 770, 771, 773, 774, 775, 781, 782, 784, 785, 787, 788, 789, 796, 804, 808, 809, 814, 815, 816, 817, 818,
821, 823, 824, 825, 826, 830, 832, 834, 837, 840, 845, 846, 848, 850, 851, 852, 853, 855, 858, 859, 861, 862, 864, 865, 867, 868, 871, 873, 878, 879, 882, 883,
890, 891, 893, 894, 895, 896, 897, 898, 900, 901, 903, 906, 908, 913, 915, 916, 919, 920, 926, 927, 928, 929, 931, 940, 941, 943, 948, 949, 951, 954, 958, 959,
960, 963, 964, 966, 968, 972, 974, 975, 978, 981, 987, 989, 992, 999})

(1:628522583,{1, 4, 5, 8, 9, 10, 14, 22, 25, 27, 32, 37, 43, 44, 45, 47, 48, 53, 54, 57, 58, 59, 61, 68, 72, 74, 75, 78, 79, 81, 83, 86, 94, 96, 99, 102, 108,
109, 112, 113, 114, 117, 118, 121, 126, 127, 128, 129, 130, 132, 133, 141, 142, 145, 148, 157, 161, 163, 165, 166, 168, 171, 172, 174, 176, 177, 178, 179, 180,
183, 187, 188, 190, 196, 198, 199, 202, 204, 208, 209, 216, 220, 221, 224, 226, 228, 230, 233, 234, 236, 237, 239, 241, 245, 246, 247, 248, 253, 254, 259, 262,
264, 267, 269, 273, 275, 277, 278, 281, 283, 288, 292, 293, 294, 295, 298, 300, 301, 304, 315, 318, 323, 324, 325, 333, 334, 335, 336, 339, 341, 342, 343, 344,
358, 361, 364, 368, 369, 370, 372, 374, 376, 377, 379, 381, 384, 386, 387, 390, 394, 398, 399, 400, 403, 408, 409, 418, 419, 420, 422, 428, 434, 435, 437, 438,
439, 441, 446, 448, 450, 451, 454, 458, 459, 460, 462, 466, 468, 469, 471, 473, 480, 482, 485, 486, 487, 489, 498, 502, 504, 505, 512, 513, 514, 519, 520, 522,
523, 524, 528, 531, 532, 534, 540, 546, 548, 549, 550, 552, 555, 556, 558, 559, 563, 564, 565, 566, 567, 568, 570, 574, 576, 580, 581, 582, 585, 586, 593, 594,
596, 597, 602, 606, 610, 613, 614, 616, 618, 619, 623, 627, 628, 630, 633, 641, 642, 646, 647, 651, 653, 658, 659, 660, 661, 664, 668, 672, 674, 678, 685, 686,
687, 692, 693, 697, 700, 702, 703, 710, 711, 713, 718, 722, 727, 728, 731, 733, 735, 736, 737, 739, 740, 741, 745, 747, 752, 753, 754, 755, 756, 757, 758, 759,
760, 762, 763, 764, 771, 776, 778, 780, 782, 784, 785, 786, 788, 789, 790, 803, 804, 814, 816, 819, 820, 822, 825, 826, 828, 830, 834, 836, 838, 840, 843, 845,
847, 849, 850, 851, 853, 854, 855, 856, 857, 859, 862, 866, 868, 870, 872, 873, 874, 876, 882, 885, 888, 889, 893, 897, 902, 903, 904, 906, 909, 910, 914, 917,
918, 921, 922, 924, 927, 928, 934, 938, 942, 944, 945, 948, 949, 951, 953, 957, 958, 959, 961, 962, 963, 966, 968, 970, 971, 974, 977, 978, 979, 983, 986, 987,
988, 992, 994, 996, 999})
```

Indexing - Locality Sensitive Hashing (LSH)



Indexing - Blocking mit LSH

Lesen der Daten	Lesen der .csv-Dateien	Übertragung in DataSet mit Person-Objekten
Erstellung der Bloom Filter	Erstellung der n-Gramme (Tokens)	Aufnahme der Tokens in Bloom Filter
Blocking mit LSH	**Berechnung der LSH-Keys**	**Finden von Kandidaten-Paaren**
Duplikat-entfernung	Entfernung von doppelten Kandidaten-Paaren	
Ähnlichkeits-berechnung	Berechnung der Ähnlichkeit für alle Kandidaten-Paare	Ausgabe aller Paare mit hinreichendem Ähnlichkeitswert

Indexing - Blocking mit LSH: Ergebnis

- Ergebnis des Blockings in der Form
  $(ID_{LshKey}, Value_{LshKey}, BloomFilter)$

```
(0,{0, 1, 2, 7, 9},BloomFilterWithLshKeys [id=1:-1764211470])
(1,{0, 8},BloomFilterWithLshKeys [id=1:-1764211470])
(2,{0, 1, 7, 8},BloomFilterWithLshKeys [id=1:-1764211470])
(3,{0, 2, 4, 8, 9},BloomFilterWithLshKeys [id=1:-1764211470])
(4,{1, 2, 3, 7, 9},BloomFilterWithLshKeys [id=1:-1764211470])
(0,{0, 1, 2, 4, 7},BloomFilterWithLshKeys [id=1:1383042088])
(1,{2, 3, 5, 6},BloomFilterWithLshKeys [id=1:1383042088])
(2,{},BloomFilterWithLshKeys [id=1:1383042088])
(3,{4, 5},BloomFilterWithLshKeys [id=1:1383042088])
(4,{2, 3, 6, 8, 9},BloomFilterWithLshKeys [id=1:1383042088])
(0,{1, 4, 6, 7, 8, 9},BloomFilterWithLshKeys [id=1:628522583])
(1,{0, 1, 2},BloomFilterWithLshKeys [id=1:628522583])
(2,{5, 6, 8},BloomFilterWithLshKeys [id=1:628522583])
(3,{2, 6, 7, 9},BloomFilterWithLshKeys [id=1:628522583])
(4,{3, 6, 8, 9},BloomFilterWithLshKeys [id=1:628522583])
(0,{3, 7, 8, 9},BloomFilterWithLshKeys [id=1:65241860])
(1,{5, 8},BloomFilterWithLshKeys [id=1:65241860])
(2,{1, 7, 8},BloomFilterWithLshKeys [id=1:65241860])
(3,{0, 2, 4, 5},BloomFilterWithLshKeys [id=1:65241860])
(4,{2, 6, 7, 9},BloomFilterWithLshKeys [id=1:65241860])
(0,{1, 2, 4, 5, 6, 7, 9},BloomFilterWithLshKeys [id=2:-2114639264])
(1,{0, 2, 4, 6, 7, 8},BloomFilterWithLshKeys [id=2:-2114639264])
(2,{0, 1, 7, 8},BloomFilterWithLshKeys [id=2:-2114639264])
(3,{0, 2, 4, 5, 6, 7, 8, 9},BloomFilterWithLshKeys [id=2:-2114639264])
(4,{2, 3, 4, 8},BloomFilterWithLshKeys [id=2:-2114639264])
```

Indexing - Duplikatentfernung

Lesen der Daten	Lesen der .csv-Dateien	Übertragung in DataSet mit Person-Objekten
Erstellung der Bloom Filter	Erstellung der n-Gramme (Tokens)	Aufnahme der Tokens in Bloom Filter
Blocking mit LSH	Berechnung der LSH-Keys	Finden von Kandidaten-Paaren
Duplikat-entfernung	**Entfernung von doppelten Kandidaten-Paaren**	
Ähnlichkeits-berechnung	Berechnung der Ähnlichkeit für alle Kandidaten-Paare	Ausgabe aller Paare mit hinreichendem Ähnlichkeitswert

Comparison - Ähnlichkeitsberechnung

Lesen der Daten	Lesen der .csv-Dateien	Übertragung in DataSet mit Person-Objekten
Erstellung der Bloom Filter	Erstellung der n-Gramme (Tokens)	Aufnahme der Tokens in Bloom Filter
Blocking mit LSH	Berechnung der LSH-Keys	Finden von Kandidaten-Paaren
Duplikat-entfernung	Entfernung von doppelten Kandidaten-Paaren	
Ähnlichkeits-berechnung	Berechnung der Ähnlichkeit für alle Kandidaten-Paare	Ausgabe aller Paare mit hinreichendem Ähnlichkeitswert

Comparison - Ähnlichkeitsberechnung: Ergebnis

- Ergebnis ist Matching Pair in der Form (ID_1, ID_2)

```
(2:-1474380544,1:-1642117313)
(1:715220210,2:-2114639264)
(1:115211210,2:2117779264)
```

Inhalt

Diskussion

- Vielzahl von Parametern
  - Auswahl der Attribute
  - Länge der Bloom Filter
  - Anzahl der Hashfunktionen
  - Länge der n-Gramme
  - Länge der LSH-Keys (#Hashfunktionen pro Hash Family)
  - Anzahl der LSH-Keys (#Hash Families)
  - Schwellwert für Ähnlichkeitsfunktion
- Geeignete Auswahl der Parameter ist von großer Bedeutung
  - Entscheidend für die Qualität und Sicherheit des PPRL
  - Beeinflusst Art und Weise der Parallelisierung
- Mehrere Möglichkeiten der Parallelisierung
- Data Corrupter

Inhalt

Demonstration