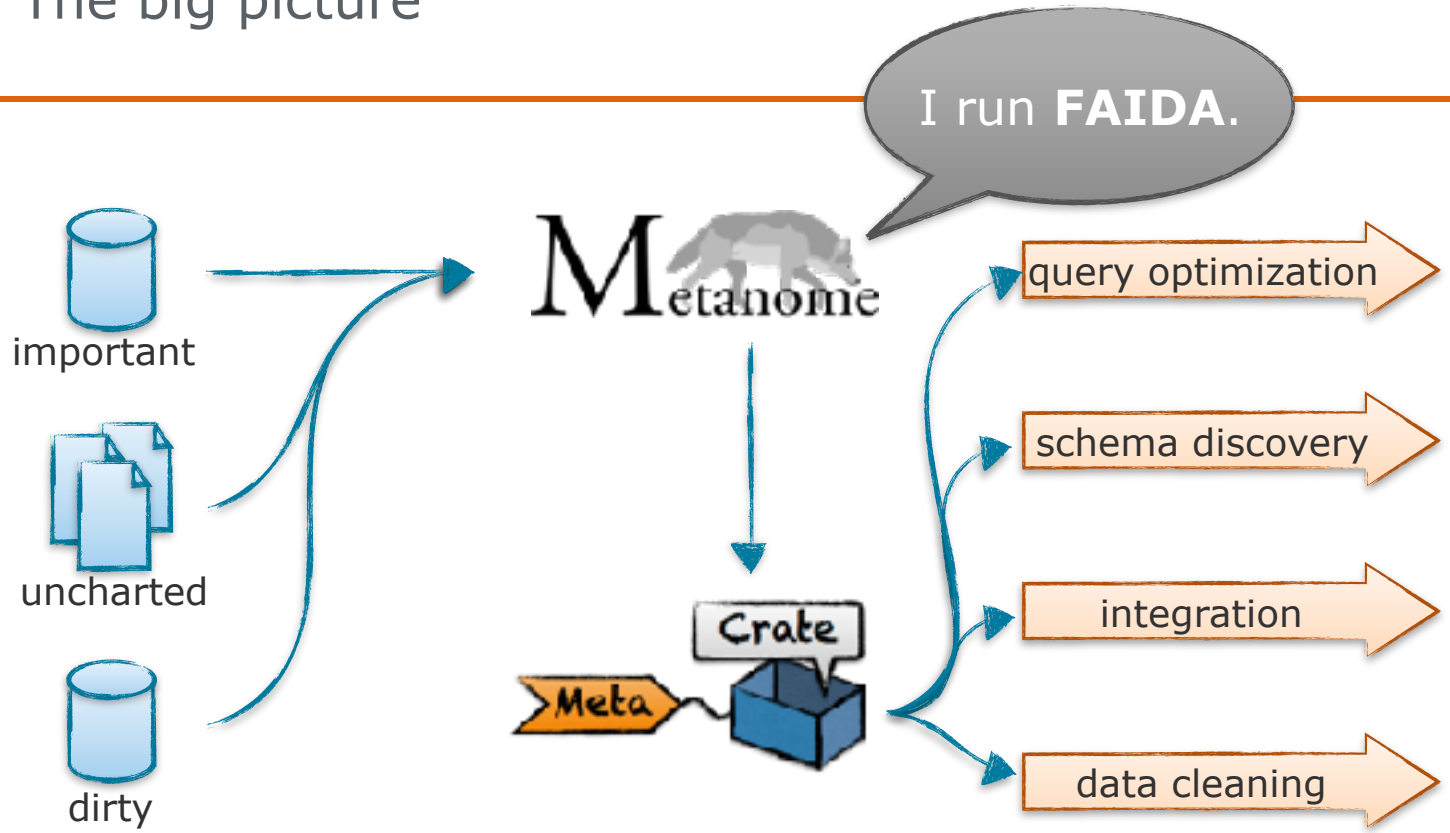
Fast Approximate Discovery of Inclusion Dependencies

Sebastian Kruse, Thorsten Papenbrock, Christian Dullweber, Moritz Finke, Manuel Hegner, Martin Zabel, Christian Zöllner, Felix Naumann

BTW 2017

Data Profiling - The big picture



Inclusion dependencies - What is that?

translations

word1	lang1	type1	word2	lang2	type2
hut	en	noun	Hütte	de	noun
hat	en	noun	Hut	de	noun
has	en	verb	hat	de	verb

words

word	lang	type	freq
hut	en	noun	0.001
hat	en	noun	0.002
has	en	verb	0.01
Hütte	de	noun	0.001
Hut	de	noun	0.002
hat	de	verb	0.01

translations[word1] ⊆ words[word]

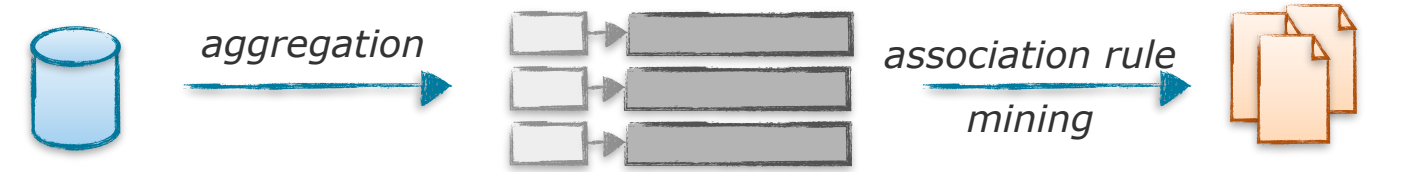
translations[word1, lang1, type1] ⊆ words[word, lang type]

translations[type1] ⊆ words[type]

- schema discovery
- integration
- data cleaning
- query optimization

Fast Approximate Discovery of Inclusion Dependencies

Inclusion dependencies - Why is discovery challenging?



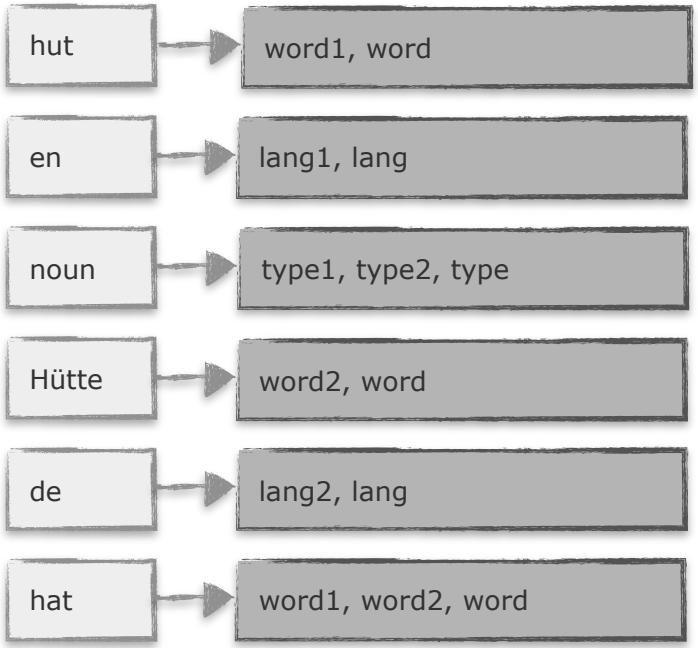
Inclusion dependencies - Why is discovery challenging?

translations

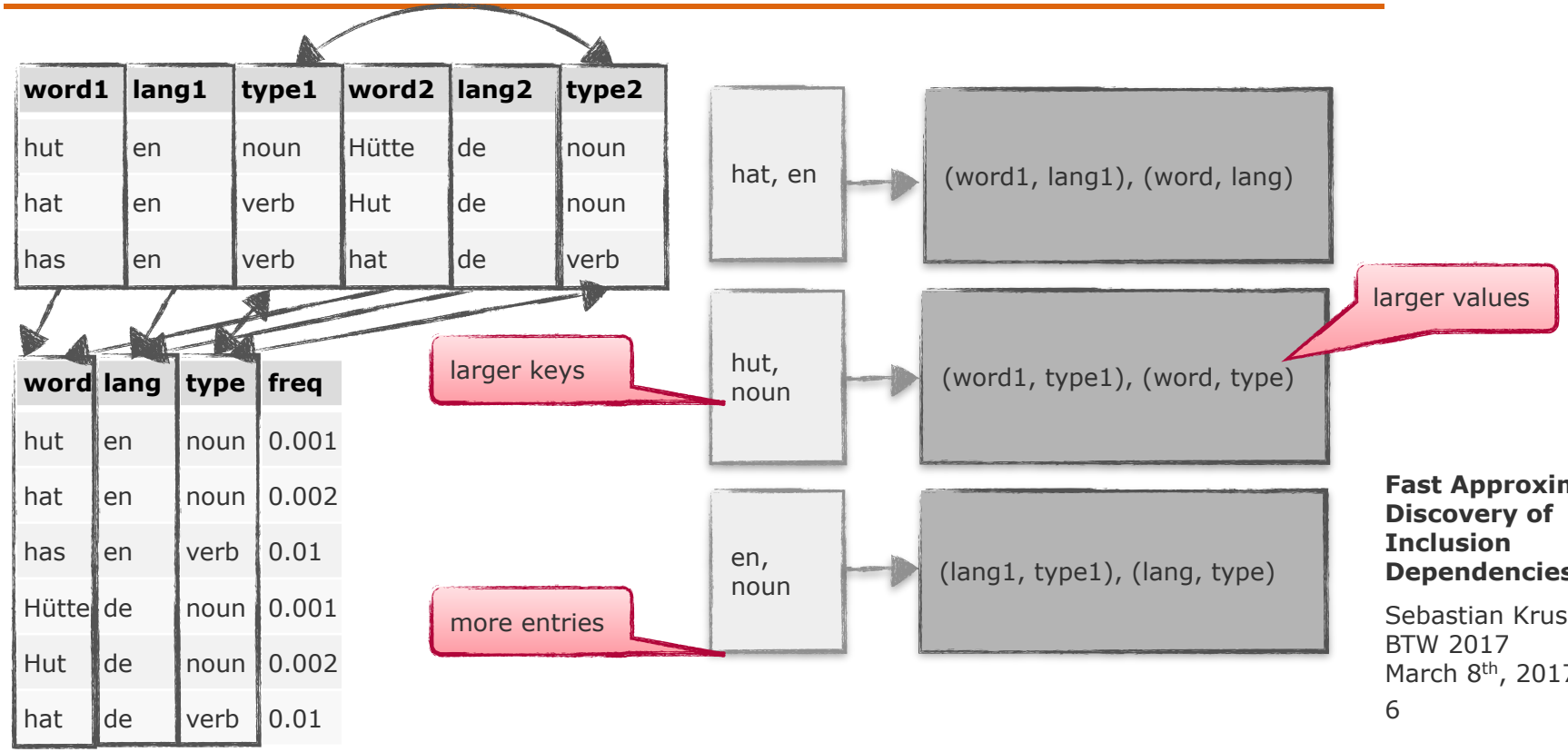
word1	lang1	type1	word2	lang2	type2
hut	en	noun	Hütte	de	noun
hat	en	noun	Hut	de	noun
has	en	verb	hat	de	verb

words

word	lang	type	freq
hut	en	noun	0.001
hat	en	noun	0.002
has	en	verb	0.01
Hütte	de	noun	0.001
Hut	de	noun	0.002
hat	de	verb	0.01



Inclusion dependencies - Why is discovery challenging?



Fast Approximate Discovery of Inclusion Dependencies

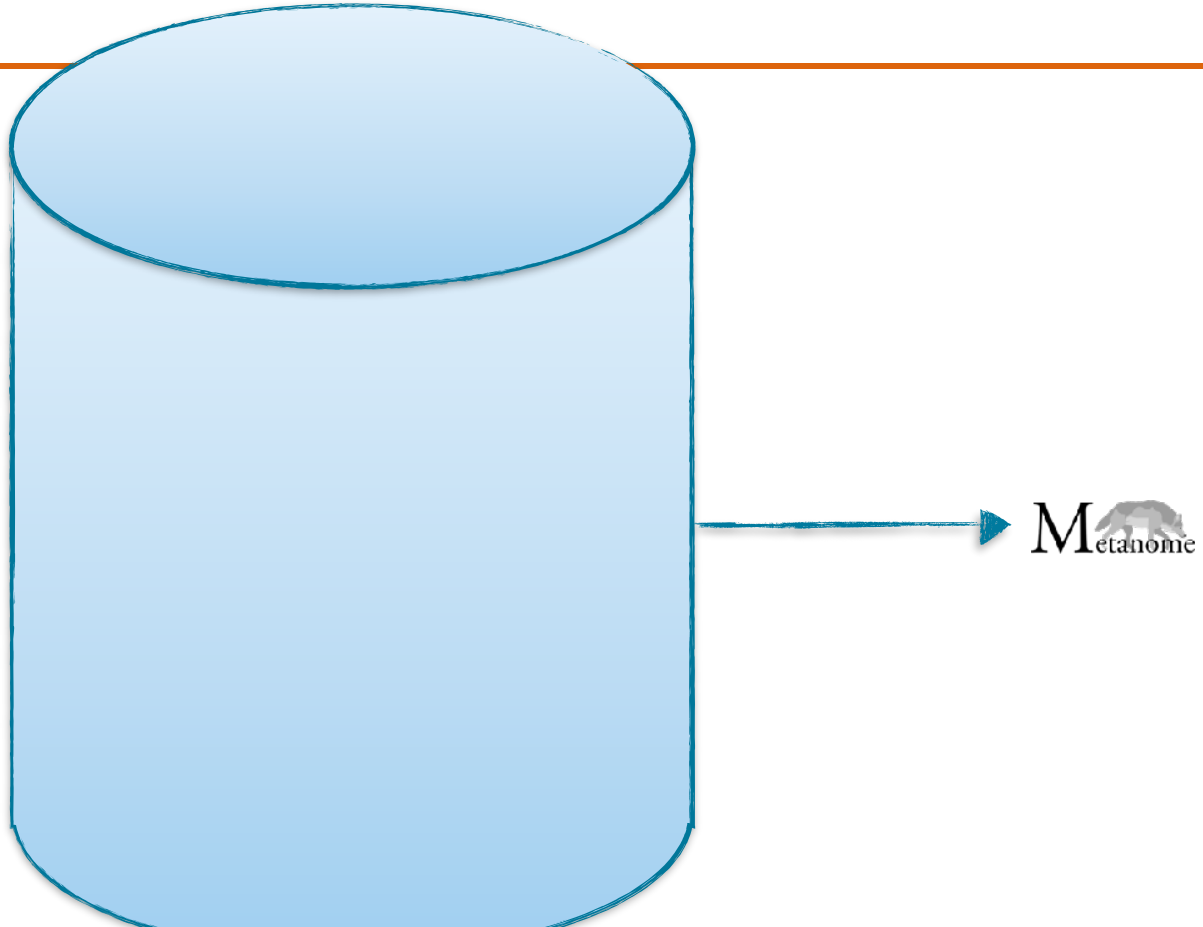
Inclusion dependencies - Inverted index creation

- In-memory mapping

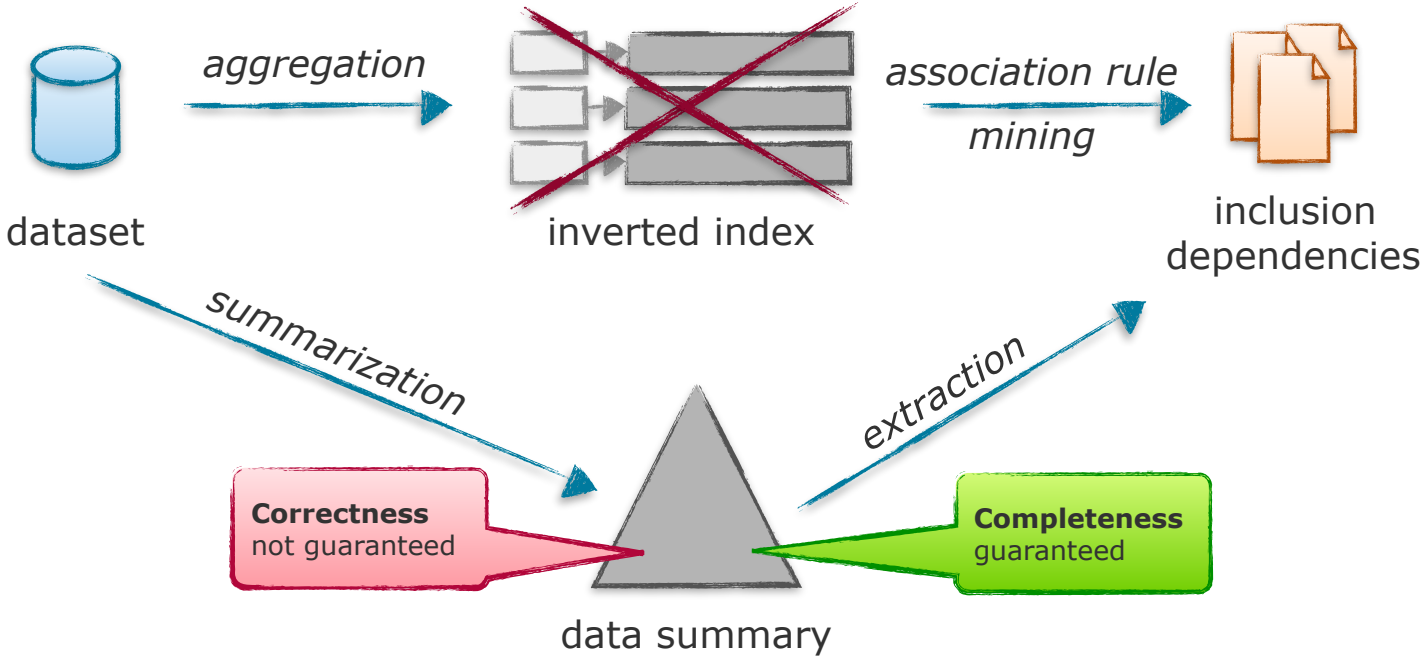
  Fabien De Marchi, Stéphane Lopes, and Jean-Marc Petit. ”Efficient algorithms for mining inclusion dependencies." *EDBT, 2002.*

- Sort-merge join
  - Jana Bauckmann, Felix Naumann, Ulf Leser. "Efficiently detecting inclusion dependencies." *ICDE, 2007.*
- Adaptive hash partitioning
  - Thorsten Papenbrock, Sebastian Kruse, Jorge-Arnulfo Quiané-Ruiz, and Felix Naumann. "Divide & conquer-based inclusion dependency discovery." *PVLDB, 2015.*
- Distributed aggregation
  - Sebastian Kruse, Thorsten Papenbrock, and Felix Naumann. "Scaling Out the Discovery of Inclusion Dependencies." *BTW, 2015.*

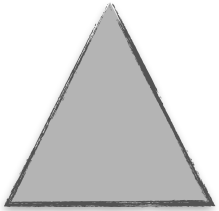
Inclusion dependencies - Inverted index creation



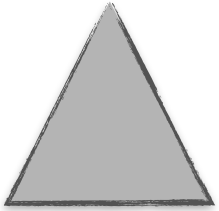
Inclusion dependencies - An approximate approach



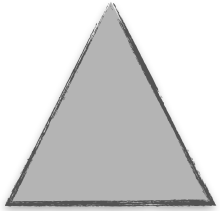
Inclusion dependency approximation - Data summaries



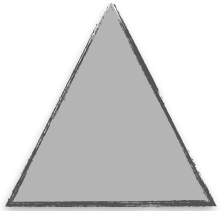
Bloom filters



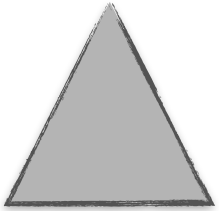
Bottom-k sketches



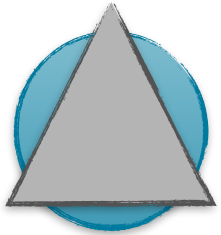
HyperLogLog



Wavelets



Sampling



Hashing

Inclusion dependencies - Why is discovery challenging?

translations

word1	lang1	type1	word2	lang2	type2
hut	en	noun	Hütte	de	noun
hat	en	noun	Hut	de	noun
has	en	verb	hat	de	verb

translations

word1	lang1	type1	word2	lang2	type2
3bef	331b	beef	20aa	32f2	beef
0f8a	331b	beef	cafe	32f2	beef
983	331b	fad3	0f8a	32f2	fad3

words

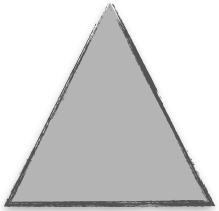
word	lang	type	freq
hut	en	noun	0.001
hat	en	noun	0.002
has	en	verb	0.01
Hütte	de	noun	0.001
Hut	de	noun	0.002
hat	de	verb	0.01

hash

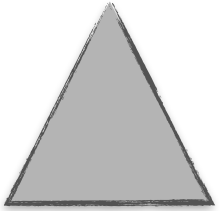
words

word	lang	type	freq
3bef	331b	beef	99ab
0f8a	331b	beef	0af2
983f	331b	fad3	4dc4
20aa	32f2	beef	99ab
cafe	32f2	beef	0af2
0f8a	32f2	fad3	4dc4

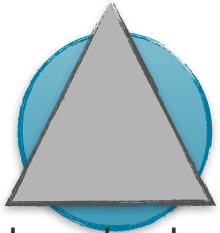
Inclusion dependency approximation - Data summaries



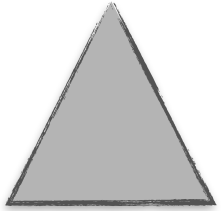
Bloom filters



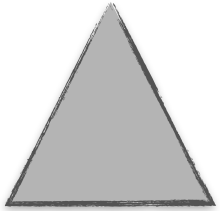
Bottom-k sketches



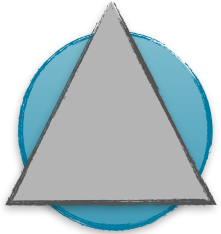
HyperLogLog



Wavelets

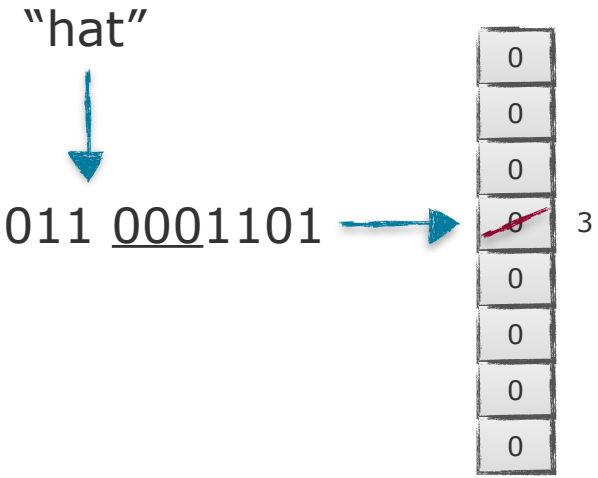


Sampling



Hashing

HyperLogLog - Brief introduction



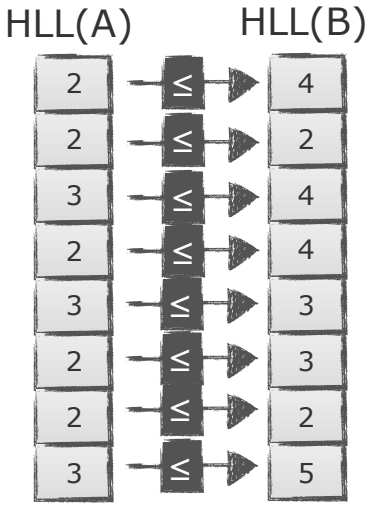
- **IND A ⊆ B is valid**

  ⇔ the values in A are a subset of the values in B

  ⇔ A∪B = B

  ⇔ **|A∪B| = |B|**

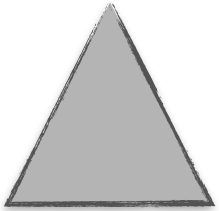
HyperLogLog - Detecting INDs



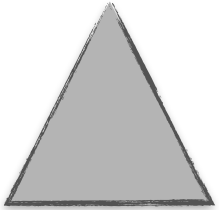
- **IND A ⊆ B is valid**

  ⇐ for every bucket index i
  $HLL(A)_i \leq HLL(B)_i$

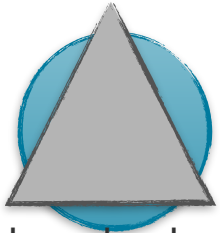
Inclusion dependency approximation - Data summaries



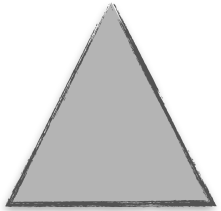
Bloom filters



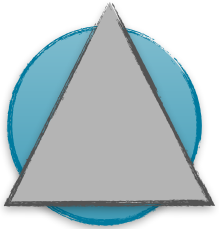
Bottom-k sketches



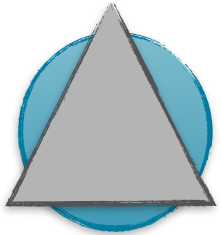
HyperLogLog



Wavelets



Sampling



Hashing

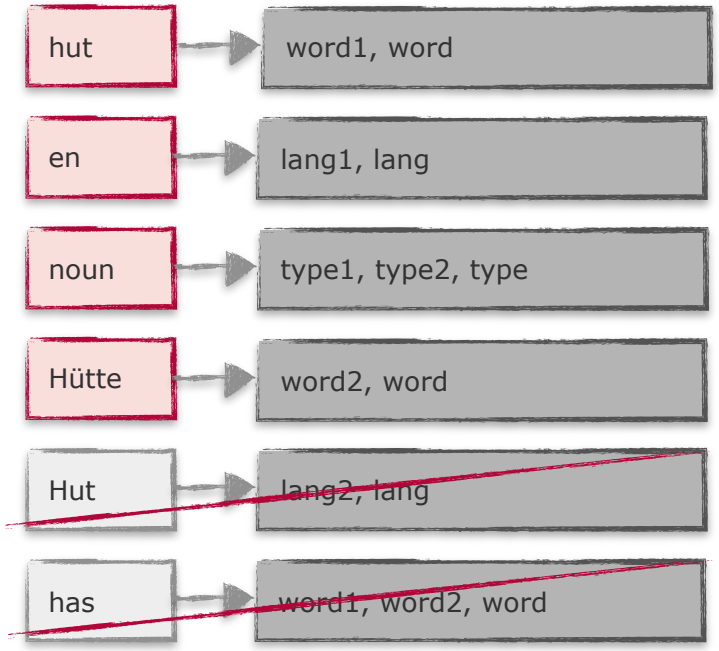
Sampling - Principles

translations

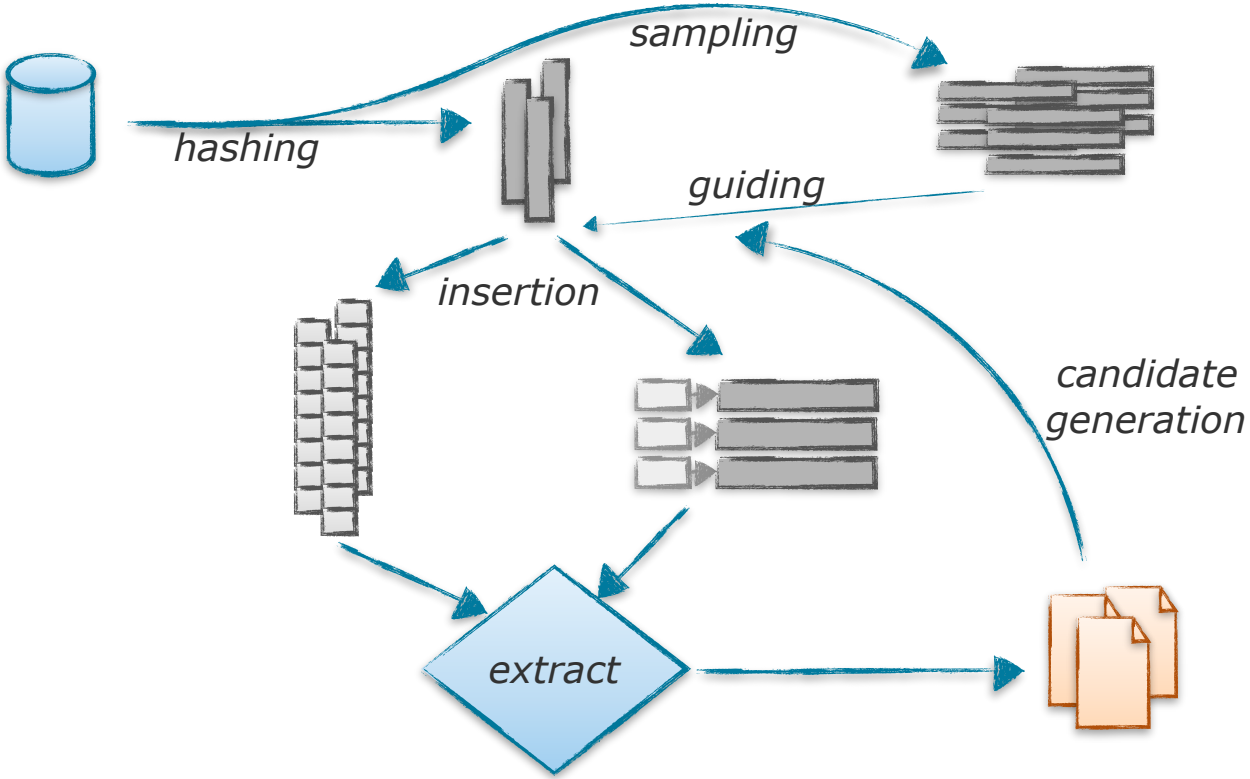
word1	lang1	type1	word2	lang2	type2
hut	en	noun	Hütte	de	noun
hat	en	verb	Hut	de	noun
has	en	verb	hat	de	verb

words

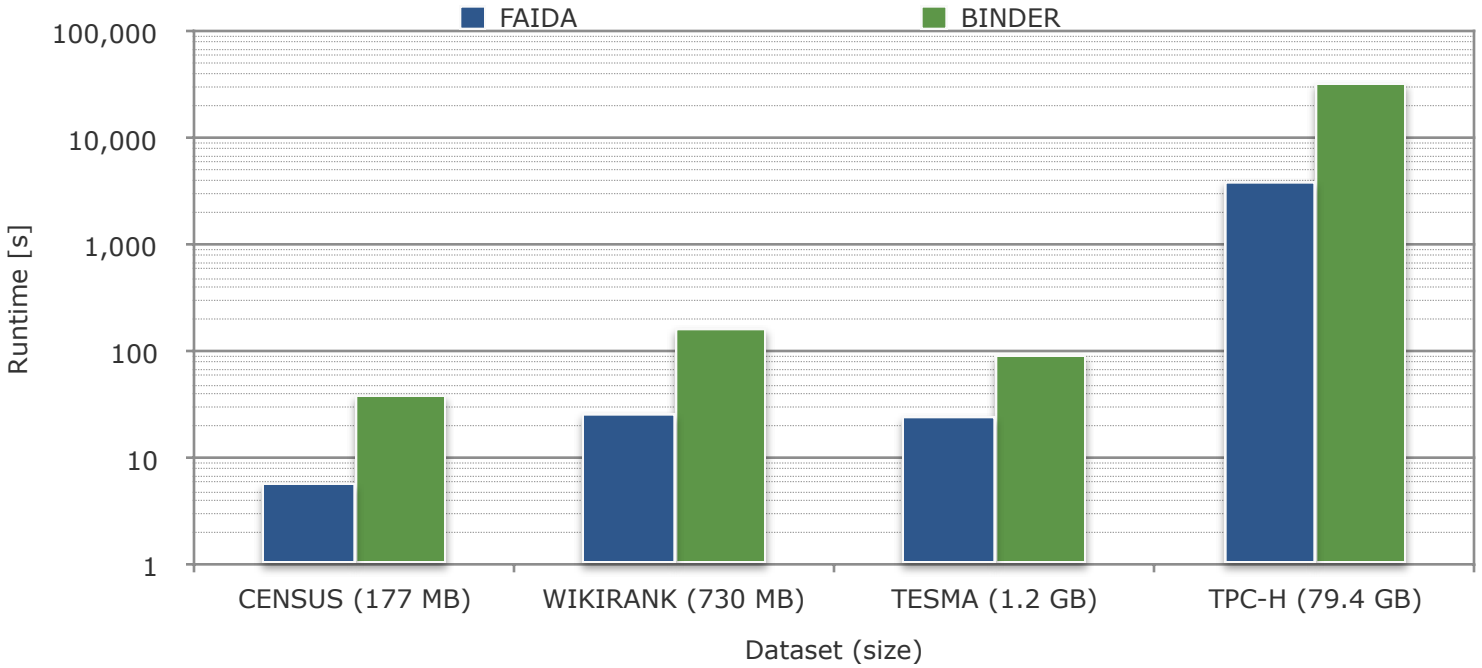
word	lang	type	freq
hut	en	noun	0.001
hat	en	noun	0.002
has	en	verb	0.01
Hütte	de	noun	0.001
Hut	de	noun	0.002
hat	de	verb	0.01



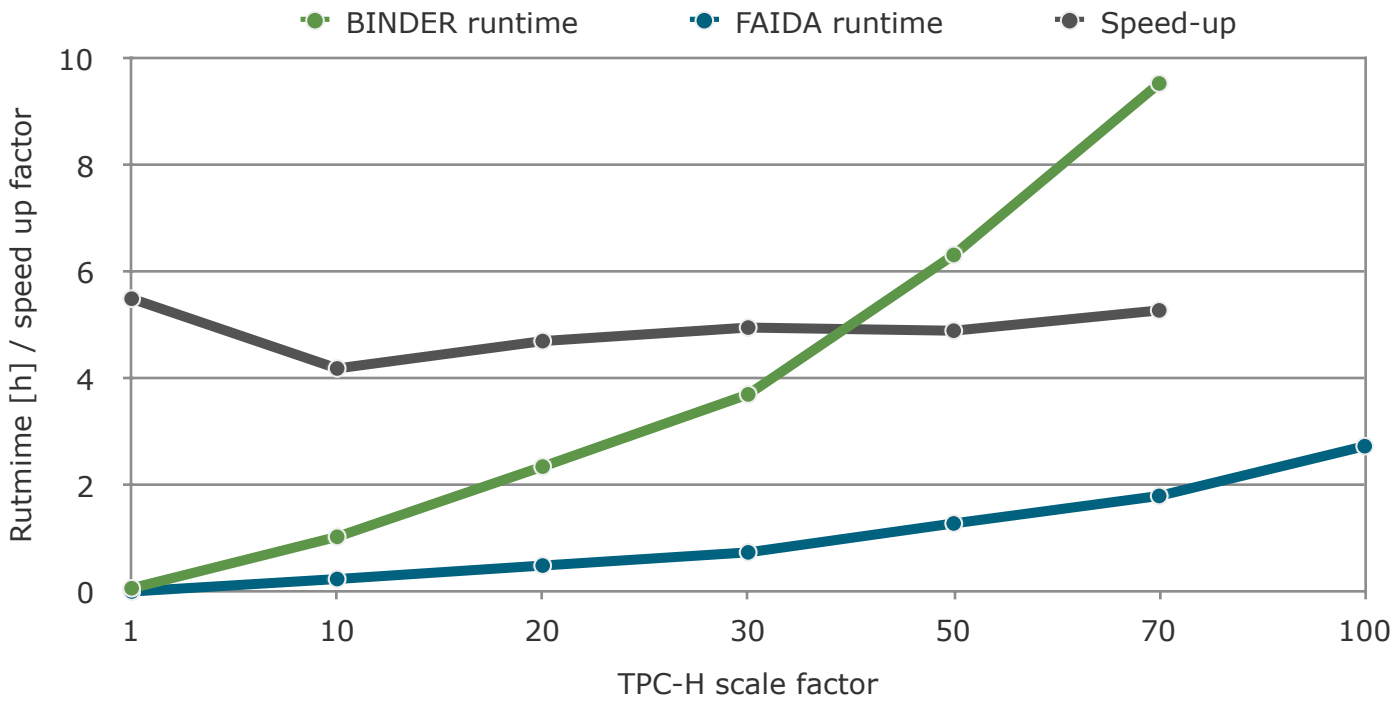
Inclusion dependencies - FAIDA



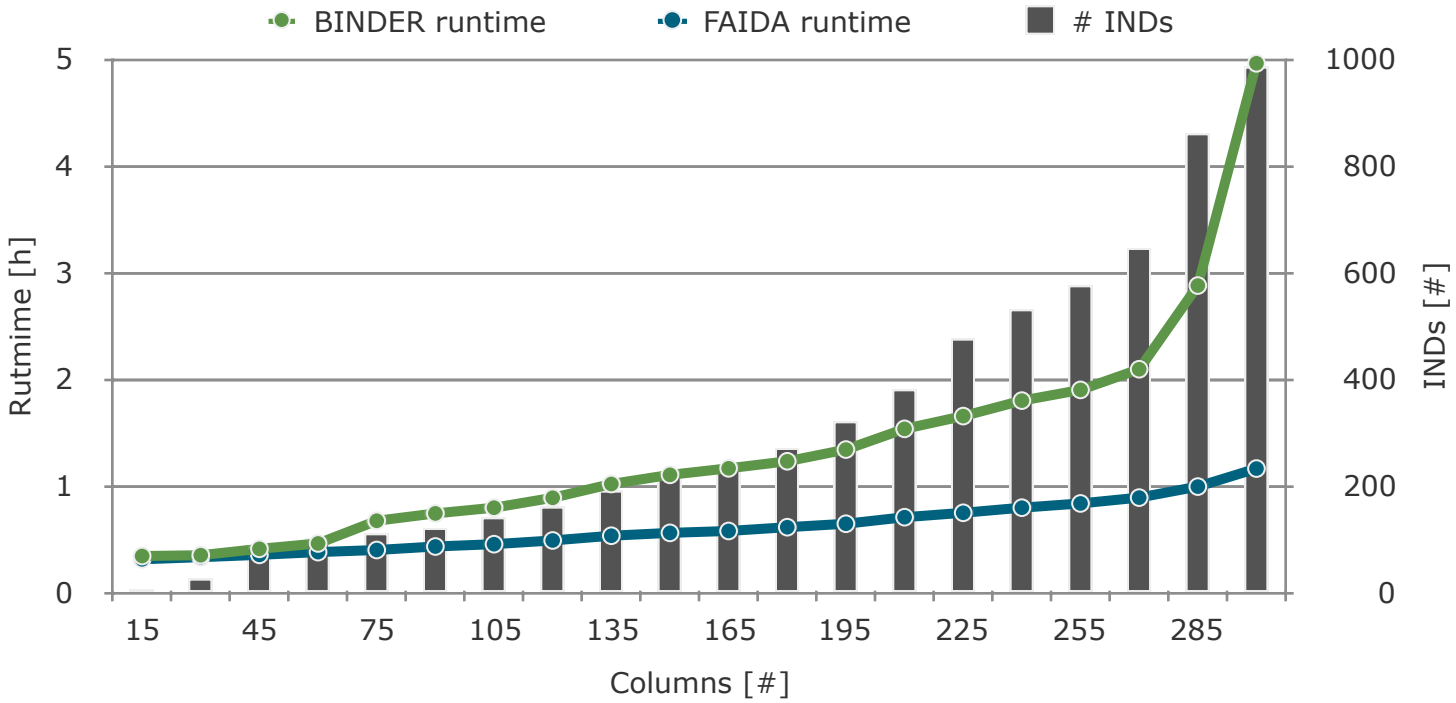
Evaluation - Performance on different datasets



Evaluation - Tuple scalability



Evaluation - Column scalability



Evaluation - Error rate

HLL precision

Sampling goal	10%	1%	0.1%
1	6.000	0.082	0.024
10	0.243	0.047	0.012
100	0.094	0.000	0.000
1,000	0.036	0.000	0.000
10,000	0.000	0.000	0.000

This was our default value.

Fast Approximate Discovery of Inclusion Dependencies - Questions?