



Semantics – Uses for Dependencies

29.06.2017
Felix Naumann

Overview

- 1. Key Detection**
2. Foreign Key Detection
3. Normalization
4. Optimization
5. Summarization



Felix Naumann
Data Profiling
Summer 2017

Room for thought

- Which properties do we expect from a key?
- Some ideas
 - UCC with no null values
 - Attribute name: ...key..., ...id..., ...PK..., etc.
 - Position in schema
 - Multi-column UCC: Togetherness
 - Data type, distribution
 - Incremental values
 - Sorted
 - Existing Index
 - Is referenced by many INDs
 - Or INDs that look like FKs
 - Joined often in given workload

Overview

1. Key Detection
- 2. Foreign Key Detection**
3. Normalization
4. Optimization
5. Summarization



Felix Naumann
Data Profiling
Summer 2017

Problem: Automatic Determination of Foreign Keys

- Given
 - Relational schema
 - Database instance of that schema
 - Complete set of (observed) inclusion dependencies
 - Attributes A and B with $R[A] \subseteq S[B]$ (in short $A \subseteq B$)
- Find
 - All foreign key constraints: attributes A and B with A references B
- Difficulty
 - Foreign keys are not intrinsic to data, but defined by humans
 - Discover semantics
- An aside: Even INDs cannot be „discovered“

Characterizing foreign keys

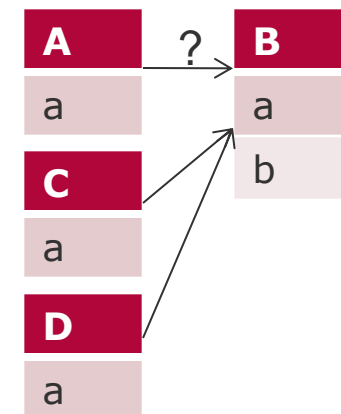
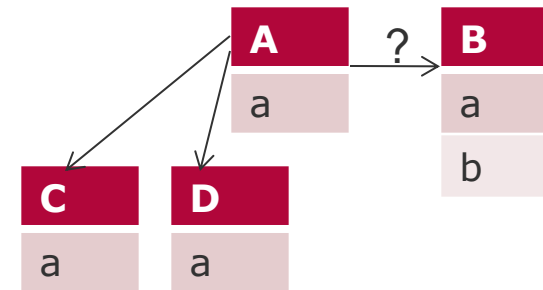
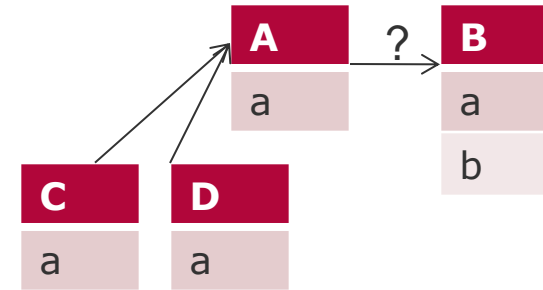
- Find set of characteristic features
 - Easily verifiable
 - Carefully developed
 - Not necessarily independent
- Then apply a machine learning approach to classify INDs as FKs

- Notation
 - Let $s(A)$ denote set of distinct values in attribute A.
 - Let $name(A)$ denote the label of attribute A.

- Source: Alexandra Rostin, Oliver Albrecht, Jana Bauckmann, Felix Naumann, Ulf Leser: A Machine Learning Approach to Foreign Key Discovery. In: WebDB 2009

Features

- **DependentAndReferenced**
 - Counts how often the dependent attribute A appears as referenced attribute in the set of all INDs.
 - Usually, a foreign key is not also a primary key that is referenced as foreign key by other tables.
- **MultiDependent**
 - Counts how often A appears as dependent attribute in the set of all INDs.
 - If $s(A)$ is contained in the set of values of many other attributes, the likelihood for each of these INDs being a FK is decreased.
- **MultiReferenced**
 - Counts how often B appears as referenced attribute in the set of all INDs.
 - Often, primary keys are referenced by more than one foreign key.



Felix Naumann
Data Profiling
Summer 2017

Features

■ DistinctDependentValues

- The cardinality of $s(A)$.
- Usually, attributes that are foreign keys contain at least some different values.

A	?	B
a	→	a
a		b
a		c
a		d
a		e

■ ValueLengthDiff

- Difference between the average value length (as string) in $s(A)$ and $s(B)$.
- Usually, average length of the values is similar whenever foreign keys reference a non-biased sample of the primary keys.

A	?	B
abab	→	abab
abab		b
abab		c
c		d
d		e

Felix Naumann
Data Profiling
Summer 2017

Features

■ Coverage

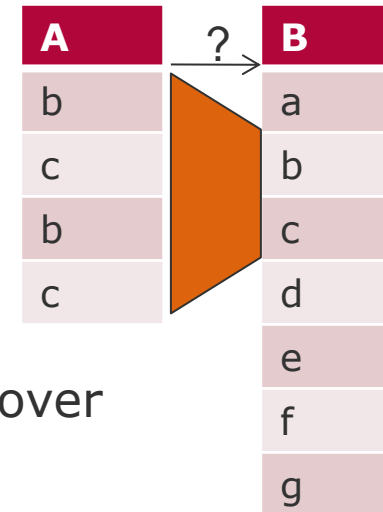
- The ratio of values in $s(B)$ that are covered by $s(A)$ compared to all values in $s(B)$.
- Usually, foreign keys cover a considerable number of primary key values.

■ OutOfRange

- Percentage of values in $s(B)$ that are not within $[\min(s(A)), \max(s(A))]$.
- Usually, the dependent values should be evenly distributed over the referenced values.
- Mostly, less than 5% of values outside of range

■ TableSizeRatio

- Ratio of number of tuples in A and number of tuples in B.
- Usually in life sciences databases, table sizes do not differ wildly



Features

■ ColumnName

- Similarity between *name(A)* and *name(B)*, also considering the name of the table of which B is an attribute.

SG_BIOENTRY.TAX_OID
→ SG_TAXON.OID

COURSE.STUDENT
→ STUDENT.ID

■ TypicalNameSuffix

- Checks whether *name(A)* ends with a substring that indicates a foreign key.
- „id“, „key“, „nr“, etc.

SG_SEQFEATURE.ENT_OID
→ SG_COMMENT.ENT_OID

CUSTOMER.C_NATIONKEY
→ NATION.N_NATIONKEY

FILMTEXTE.FILMTEXTTYPNR
→ FILMTEXTTYPEN.FILMTEXTTYPNR

Learning to classify based on features

- Four (supervised) machine learning methods
 - Naive Bayes
 - Support Vector Machine
 - J48 decision tree
 - Decision tables
- Implementation as provided by WEKA: <http://www.cs.waikato.ac.nz/ml/weka/>
- Cross validation at database level (F-Measure):

Test database	Naive Bayes	SVM	J48	DecisionTab	Avg
UniProt	0.86	0.92	0.84	0.8	0.855
Filmdienst	0.80	0.86	0.86	0.93	0.817
Movielens	0.71	0.71	1.0	0.8	0.805
SCOP	1.0	1.0	1.0	1.0	1.0
TPC-H	0.86	0.90	0.95	0.95	0.915
Average	0.846	0.78	0.930	0.896	

Felix Naumann
Data Profiling
Summer 2017

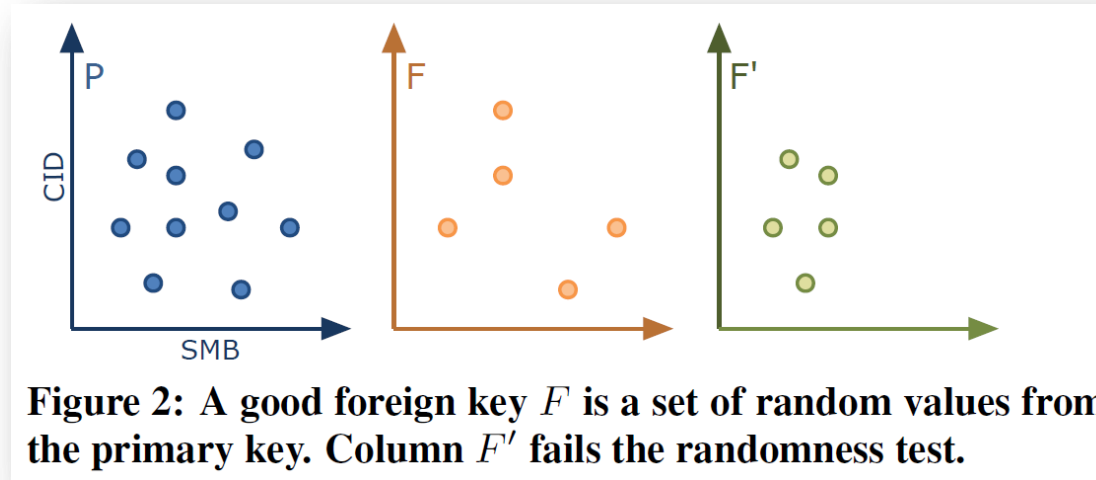
Multi-Column Foreign Key Discovery

- New feature: „Randomness“
 - To subsume several other features
 - Intuition: Given ordered key values, foreign key values form a (nearly) uniform random sample of key values
 - „expected to be “sprinkled” uniformly throughout the ordered set”
 - „It is highly unlikely that a database instance is designed such that a foreign key is a biased sample of the respective primary key (e.g., a prefix or a suffix in the ranked order).“

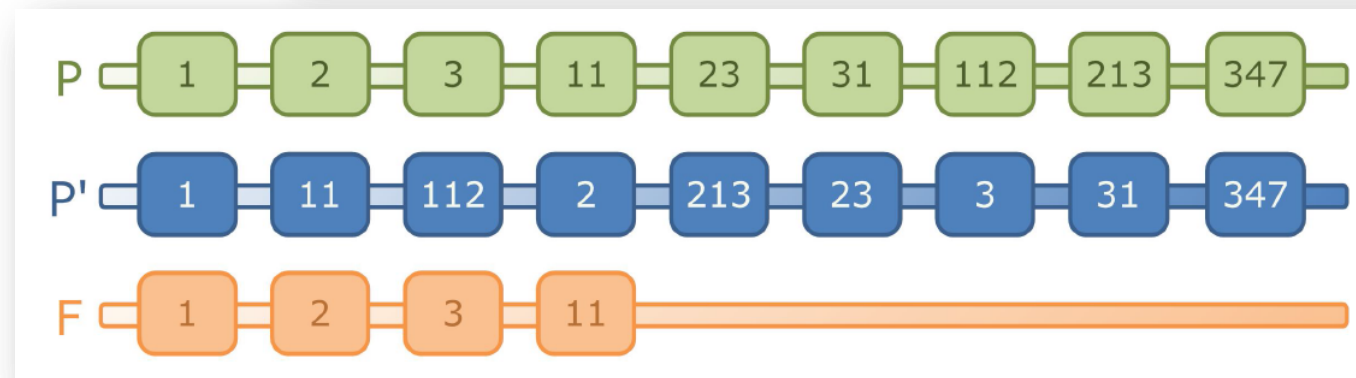
- Source: Meihui Zhang, [Marios Hadjieleftheriou](#), [Beng Chin Ooi](#), [Cecilia M. Procopiuc](#), [Divesh Srivastava](#):
On Multi-Column Foreign Key Discovery. [PVLDB 3\(1\)](#): 805-814 (2010)

Defining Randomness

- Given two sets of values (tuples) F and P , test the statistical hypothesis that the distinct values (tuples) in F have the same underlying distribution as the distinct values (tuples) in P .



- Domain order is needed



Wilcoxon rank-sum test for unary INDs

- Create $F \uplus P$ (multiset semantics)
- Sort the elements and assign ranks
 - Assign mean rank for duplicate values
- Compute sum of ranks of values in F

- If sum is (too) small, most values in F are in a prefix of P .
- If sum is (too) large, most values in F are in a suffix of P .

- Test works only in one dimension.

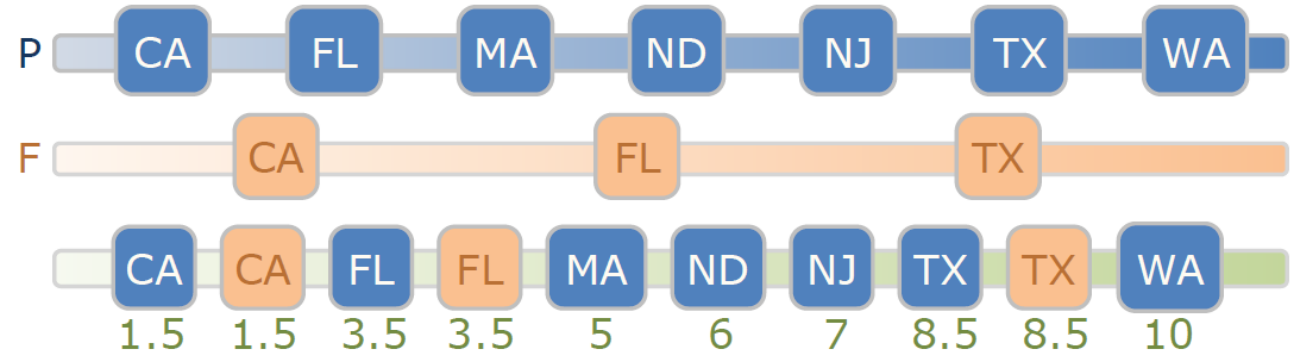
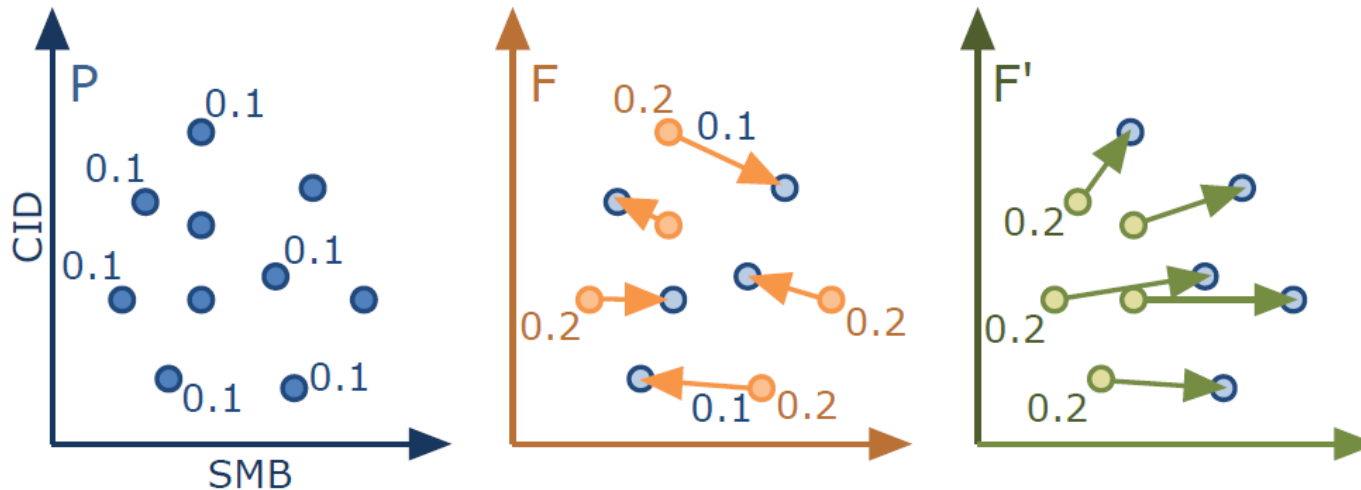


Figure 4: The Wilcoxon test: 1. Sort values in multi-set $F \cup P$; 2. Assign ranks; 3. Compute the rank-sum of values in F (13.5 in this example).

Earth Movers Distance for multiple dimensions

- Minimal cost to transform one set into the other
 - Normalize points in P and F: each value shall have same weight; sum of weights is 1.



- Problems
 - EMD requires metric distance measure
 - Even for numeric values, different ranges result in different EMDs
 - Incomparable across different pairs, e.g. for ranking

Uniform distance measure for string and numeric columns

- Idea: Use distance between ranks of values in P column.
- Single column: Distance between two values in F and P is absolute difference between their ranks in P
- Multi-columns: Sum of single-dimension rank distances
 - Manhattan distance
- Bias across multiple candidates if P has many more values than P': Normalize distance by number of values.

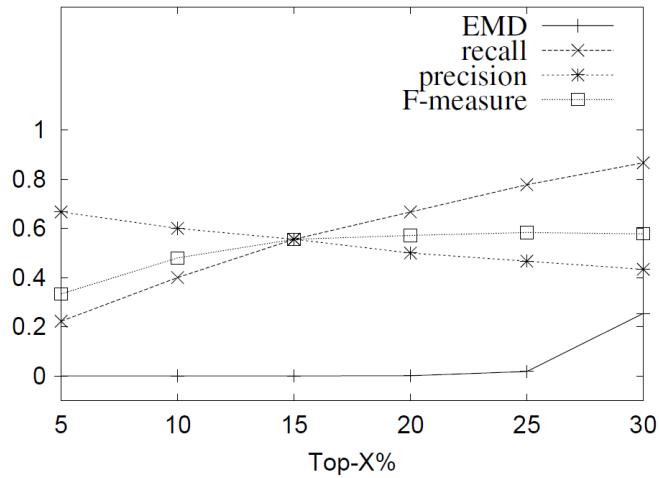
□ In effect:

DEFINITION 2 (QUANTILE DISTANCE). *Given a multi-column set X consisting of n columns, a total order in each column X_i , a function $q_i(x)$ that returns the quantile order of value x in column X_i , and two tuples $v, w \in X$, the quantile distance is*

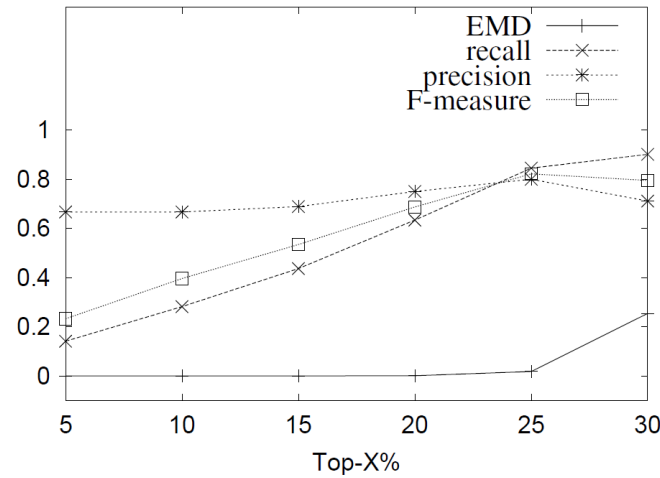
$$d(v, w) = \sum_{1 \leq i \leq n} |q_i(v) - q_i(w)|.$$

- Final normalization is needed if number of dimensions is different
 - Comparing unary candidate P and F with n-ary candidates.

Evaluation

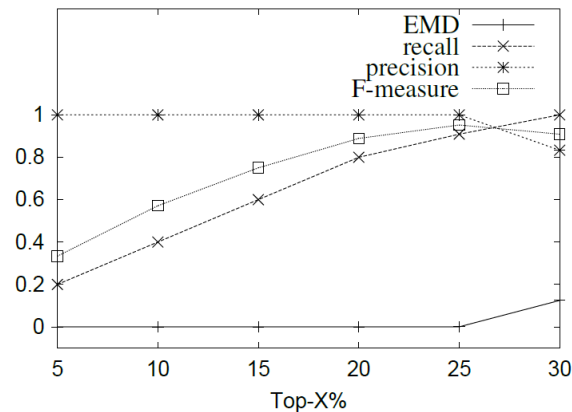


(a) TPC-E.

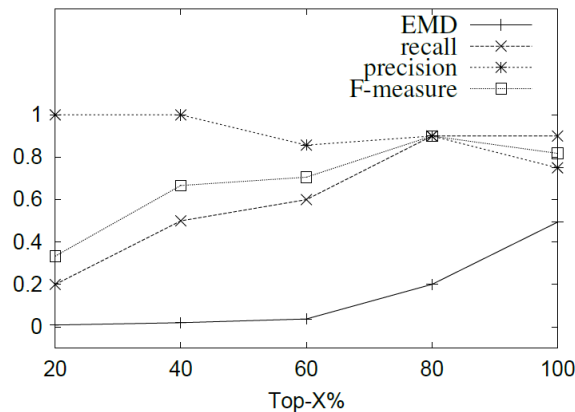


(b) TPC-E extended.

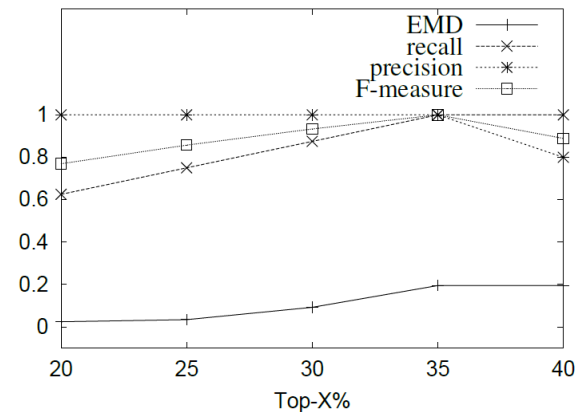
Allowing for symmetric and transitive constraints



(a) TPC-H.



(b) Wikipedia.



(c) IMDB.

Felix Naumann
Data Profiling
Summer 2017

Randomness implies previous features

1. A foreign key should have significant cardinality
 - Implication clear
2. A foreign key should have good coverage of the primary key
 - Implication clear
3. A foreign key should not be at the same time a primary key for too many other foreign keys
 - „Probability that a substantial number of columns F' are random samples of F , without any real correlation between F and F' , is very small.“
4. The set of values of a foreign key should not be a subset of too many primary keys
 - “If F is a random sample of P , and F is a random sample of some other column P' with the same underlying distribution as P , then P and P' are clearly highly correlated. It is unlikely that a large number of such correlated columns P' exist.
5. The average length of the values in foreign/primary key columns should be similar (mostly for strings)
 - Implication clear
6. The primary key should have only a small percentage of values outside the range of the foreign key
 - Implication clear
7. The column names of foreign/primary keys should be similar.

Overview

1. Key Detection
2. Foreign Key Detection
- 3. Normalization**
4. Optimization
5. Summarization

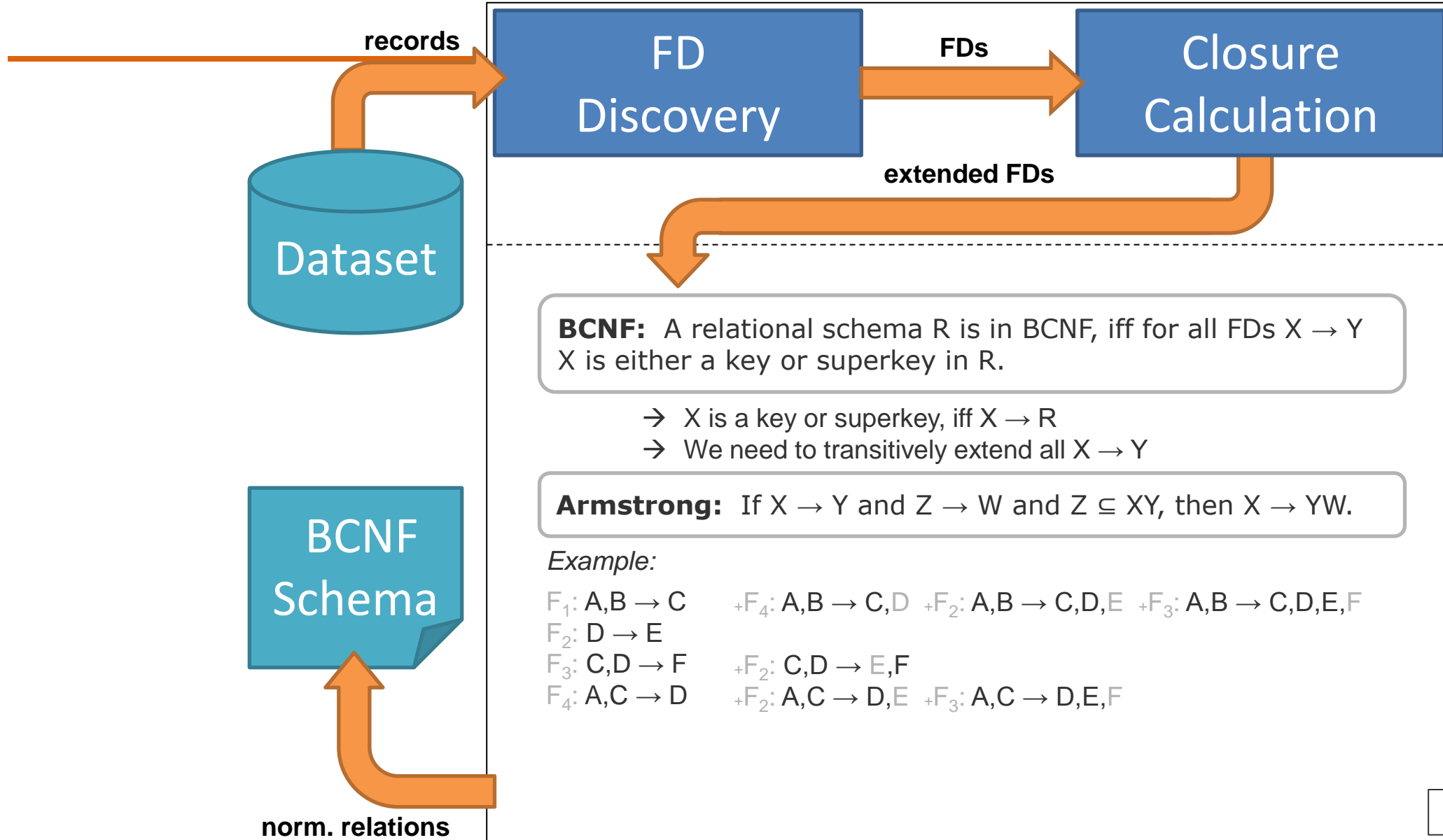


Felix Naumann
Data Profiling
Summer 2017

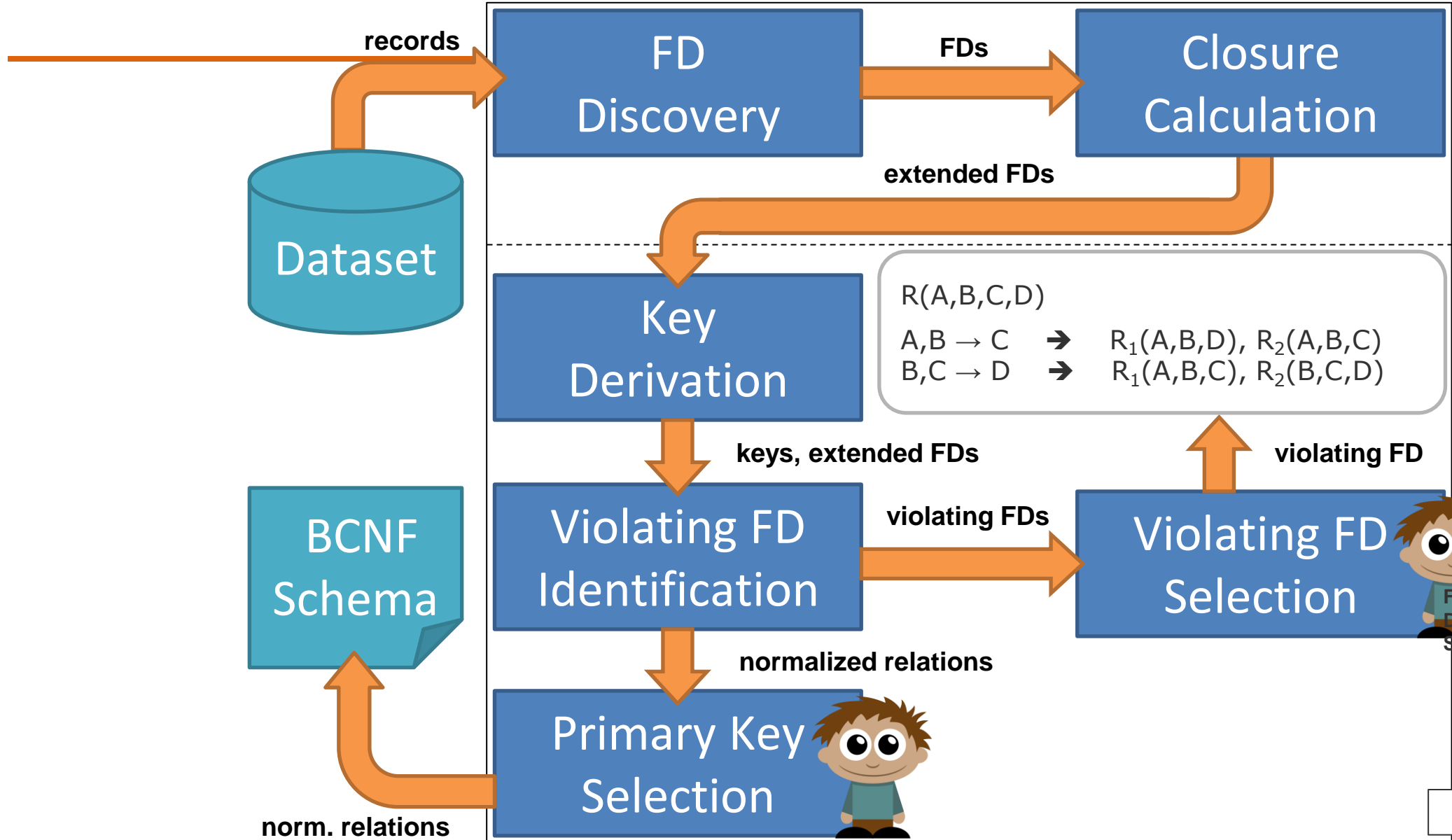
Thorsten Papenbrock, Felix Naumann,
A Hybrid Approach to Functional Dependency Discovery.
SIGMOD Conference 2016: 821-833

Architecture

Data-driven Normalization



Architecture Data-driven Normalization

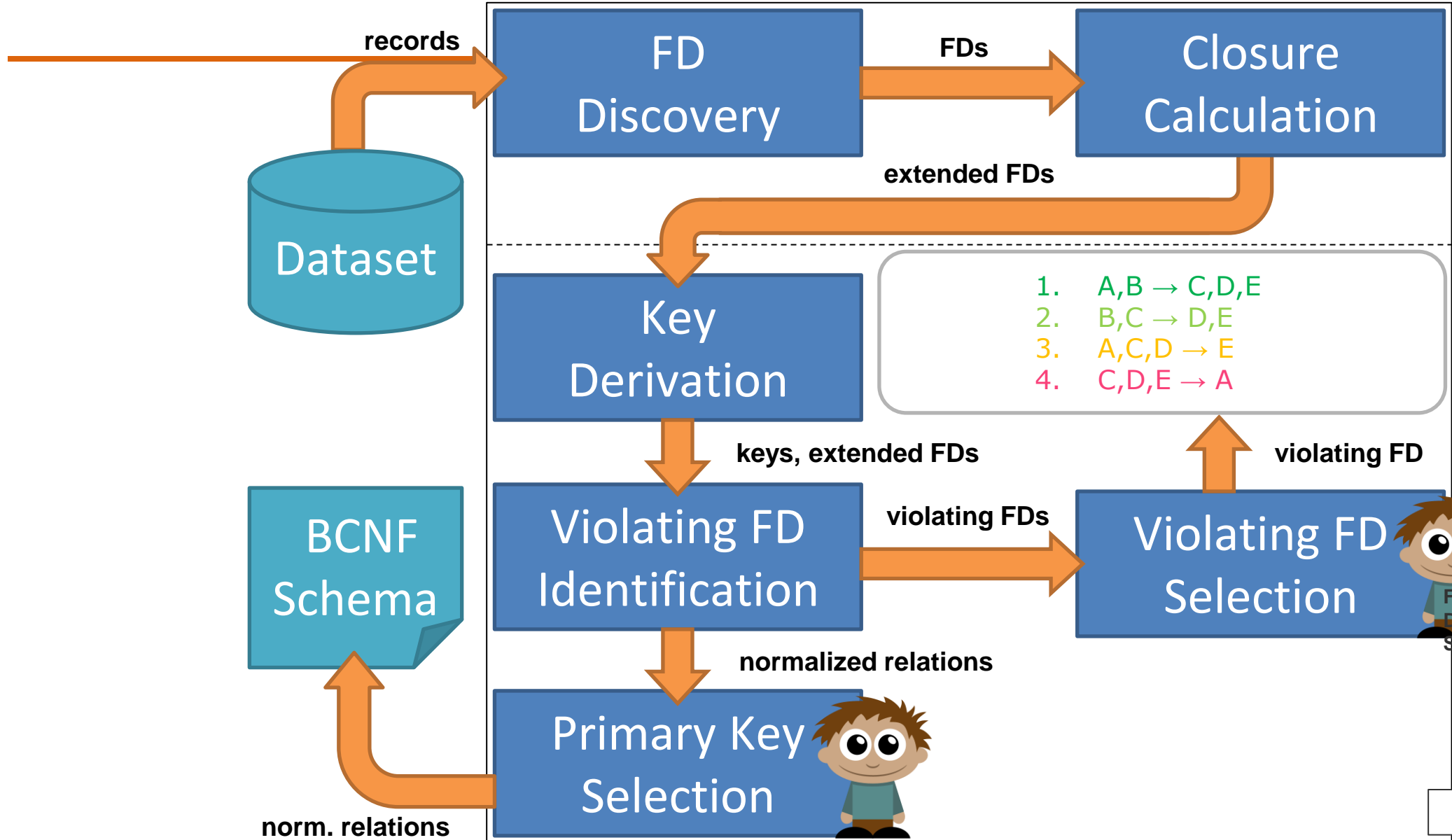


$R(A,B,C,D)$
 $A,B \rightarrow C \rightarrow R_1(A,B,D), R_2(A,B,C)$
 $B,C \rightarrow D \rightarrow R_1(A,B,C), R_2(B,C,D)$




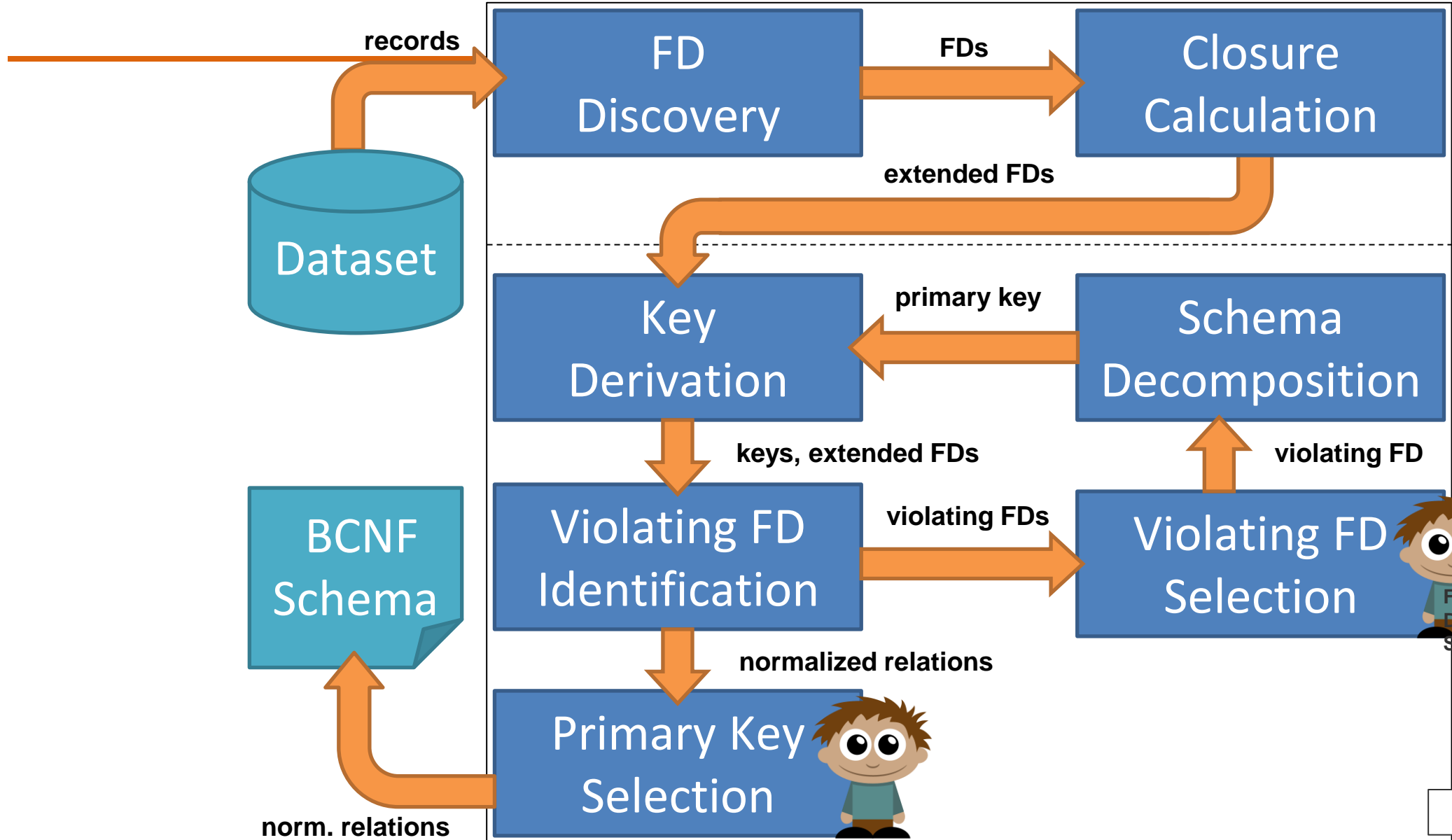
Felix Naumann
Data Profiling
Summer 2017

Architecture Data-driven Normalization



Felix Naumann
Data Profiling
Summer 2017

Architecture Data-driven Normalization



Felix Naumann
Data Profiling
Summer 2017

Primary Key Selection

- $\text{score}(\text{UCC}) = \text{lengthScore}(\text{UCC}) + \text{valueScore}(\text{UCC}) + \text{positionScore}(\text{UCC})$
- $\text{lengthScore}(\text{UCC})$
 - Semantically correct keys are usually shorter than spurious keys in their number of attributes $|X|$: $\frac{1}{|X|}$
- $\text{valueScore}(\text{UCC})$
 - The values in primary keys are typically short, because they serve to identify records and usually do not contain much business logic: $\frac{1}{\max(1, |\max(X)| - 7)}$
- $\text{positionScore}(\text{UCC})$
 - Key attributes are typically located left and without non-key attributes between them: $\frac{1}{2} * \left(\frac{1}{|left(X)| + 1} + \frac{1}{|between(Y)| + 1} \right)$

Violating FD Selection

- score(FD)

- lengthScore(FD) + valueScore(FD) + positionScore(FD) + duplicationScore(FD)

- lengthScore

- X should contain possibly few and Y possibly many attributes: $\frac{1}{2} * \left(\frac{1}{|X|} + \frac{|Y|}{|R|-2} \right)$

- valueScore

- X should hold possibly short values: $\frac{1}{\max(1, |\max(X)|-7)}$

- positionScore

- The attributes in X and Y should be close in the schema: $\frac{1}{2} * \left(\frac{1}{|\text{between}(X)|+1} + \frac{1}{|\text{between}(Y)|+1} \right)$

- duplicationScore

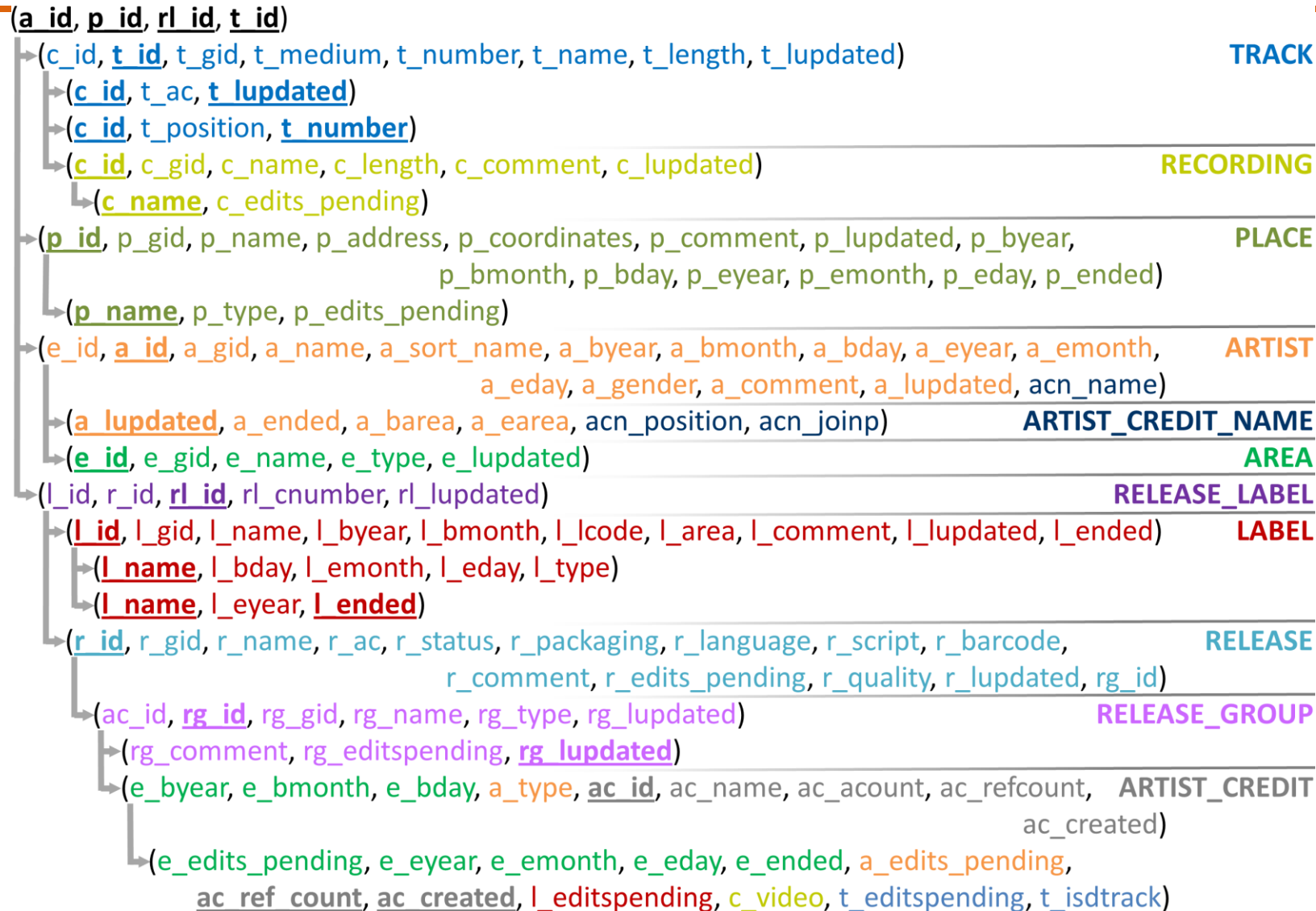
- X and Y should contain possibly much redundancy: $\frac{1}{2} * \left(2 - \frac{|\text{uniques}(X)|}{|\text{values}(X)|} - \frac{|\text{uniques}(Y)|}{|\text{values}(Y)|} \right)$

Experimental Evaluation



Felix Naumann
Data Profiling
Summer 2017

Experimental Evaluation



Felix Naumann
Data Profiling
Summer 2017