UNIVERSITÄT LEIPZIG

SCALABLE MATCHING OF REAL-WORLD DATA

Erhard Rahm

Hanna Köpcke, Lars Kolb, Andreas Thor

http://dbs.uni-leipzig.de

Object Matching
(entity resolution, deduplication …)

- Identification of semantically equivalent objects
  - within one data source or between different sources
  - to integrate (merge) them, compare them, improve data quality, etc.
- Original focus on structured (relational) data

Source1: Customer

Cno	*LastName*	*FirstName*	*Gender*	*Address*	*Phone/Fax*
24	Smith	Christoph	M	23 Harley St, Chicago IL, 60633-2394	333-222-6542 / 333-222-6599
493	Smith	Kris L.	F	2 Hurley Place, South Fork MN, 48503-5998	444-555-6666

Source2: Client

CID	*Name*	*Street*	*City*	*Sex*
11	Kristen Smith	2 Hurley Pl	South Fork, MN 48503	0
24	Christian Smith	Hurley St 2	S Fork MN	1

Duplicates in (integrated) web sources: Publication references

3

A survey of approaches to automatic schema matching
E Rahm, PA Bernstein - the VLDB Journal, 2001 - Springer
The VLDB Journal 10: 334–350 (2001) / Digital Object Identifier (DOI) 10.1007/s007780100057
... A survey of approaches to automatic schema matching ... Erhard Rahm 1 , Philip A. Bernstein 2 ... 1 Universitat Leipzig, Institut fur Informatik, 04109 Leipzig, Germany; (e-mail: rahm@ ...
Cited by 2658 - Related articles - All 87 versions

[CITATION] A survey of approaches to automatic schema matching
R Erhard, AB Philip - VLDB Journal, 2001
Cited by 25 - Related articles

[CITATION] A survey of approaches to automatic schema matching
PA Bernstein, E Rahm - VLDB Journal, 2001
Cited by 17 - Related articles - View as HTML

[CITATION] AA survey ofapproaches to automatic schema matching
E RRhm... - The VLDB Journal, 2001
Cited by 2 - Related articles

[CITATION] Bernstein P 八 A Survery of Approaches to Automatic Schema Matching
E Rahm - The International Journal on Very Large Da —
Cited by 2 - Related articles

Duplicates due to
- Order of authors
- Confusion of first and last names
- Extraction errors
- Typos
- ...

Duplicates in web sources: Product offers

4

Canon VIXIA HF S10 Camcorder - 1080p - 8.59 MP - 10 x optical zoom
Flash card, 32 GB, 1y warranty, F/1.8-3.0
The VIXIA HF S10 delivers brilliant video and photos through a Canon exclusive 8.59 megapixel CMOS image sensor and the latest version of Canon's advanced image processor, ...
12 reviews - Add to Shopping List
$975 new
from 52 sellers
Compare prices

Canon (VIXIA) HF S10 iVIS Dual Flash Memory Camcorder
Canon HF S10 iVIS Dual Flash Memory CamcorderSPECIAL SALE PRICE: $899 Display both English/Japanese + we supplu all English manuals in English as PDF. ....
Add to Shopping List
$899.00 new
Made in Japan Online

Canon VIXIA HF S10
Dual Flash Memory High Definition Camcorder The Next Step Forward in HD Video Canon has a well-known and highly-regarded reputation for optical excellence, ...
Add to Shopping List
$999.00 new
Performance Audio
2 seller ratings

Canon VIXIA HF S100 Flash Memory Camcorder
***Canon Video HF S100 Instant Rebate Receive $200 with your purchase of a new Canon VIXIA HF S100 Flash Memory Camcorder. (Price above includes $200 ....
Add to Shopping List
$899.95 new
Arlingtoncamera.com
5 seller ratings

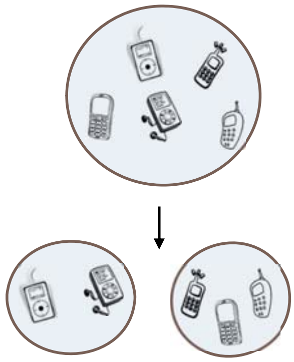
Canon Vixia Hf S10 Care & Cleaning
Care & Cleaning Digital Camera/Camcorder Deluxe Cleaning Kit with LCD Screen Guard Canon VIXIA HF S10 Camcorders Care & Cleaning.
Add to Shopping List
$2.99 new
shop.com
38 seller ratings

Outline

- Motivation
- Existing Frameworks and their Performance
  - Qualitative comparison [DKE'10]
  - Quantitative comparison [VLDB'10]
- Matching of Product Offers
  - Challenges
  - System design with use of extracted features (e.g. product codes)
  - Evaluation
- Parallel Matching in the Cloud
  - Blocking-based Object Matching with MapReduce
  - Load Balancing
    - Block-Split Approach
    - Experimental Results
  - Dedoop tool
- Conclusions & Future Work

Existing Object Matching Approaches

- Many tools and research prototypes
- **Blocking** to reduce search space
  - Group similar objects within blocks based on *blocking key*
  - Restrict object matching to objects from the same block
  - Alternative approach: Sorted Neighborhood



- Combined use of **several matchers**
  - Attribute-level matching
    based on generic or domain-specific similarity functions,
    e.g., string similarity (edit distance, n-gram, TF/IDF, etc.)
  - Context-based matchers
  - Learning-based or manual specification of matcher combination

ER Frameworks 1 (non-learning)*

7

	BN	MOMA	SERF	DuDe	FRIL
Entity type	XML	relational	relational	relational	relational
Blocking key definition	-	-	-	manual	manual
partitioning disjoint overlapping	-	-	-	Sorted Neighborhood	Sorted Neighborhood
Matchers	attribute, context	attribute, context	attribute	attribute	attribute
Matcher combination	numerical	workflow	rules	workflow	workflow

* Koepcke, H.; Rahm, E.: *Frameworks for entity matching: A comparison.* Data & Knowledge Engineering, 2010

ER Frameworks 2 (learning-based)

8

	Active Atlas	MARLIN	Op. Trees	TAILOR	FEBRL	Context-b. F.work	FEVER
Entity type	relational	rel.	rel.	rel.	XML, rel.	rel.	rel.
Blocking key definition	manual	manual	manual	manual	manual	manual	manual
partitioning *disjoint* overlapping	hashing	canopy clustering	canopy cl.	*threshold* Sorted Neighb.	SN	canopy-like	*several,* SN, canopy
Matchers	attribute	attr.	attr.	attr.	attr.	attr., context	attr.
Matcher combination	rules	numerical, rules	rules	numerical, rules	numerical	numerical, rules	workflow
Learners	decision tree	SVM, dec. tree	SVM-like	probab. dec. tree	SVM	diverse	multiple,SVM, dec. tree, ..
Training selection	manual, semi-autom.	manual, semi-autom.	manual	manual	manual, automatic	manual	manual, semi-autom.

Observations from [DKE'10]

- Numerous frameworks with similar functionality regarding blocking and matchers
  - Primarily attribute-level matching for relational sources
  - Manual selection of matchers / attributes
  - Manual specification of blocking keys
- Frequent use of training-based match strategies
  - Mostly manual training
  - Most popular learners: SVM, decision tree
- Heterogeneous, non-conclusive evaluations
  - Different datasets and methodologies
  - Missing specification details, e.g. on training
  - Unclear scalability to larger datasets

VLDB 2010 evaluation: Match tasks

Match task		**Source size (#entities)**		**Mapping size (#correspondences)**		
Domain	Sources	Source 1	Source 2	Full input mapping (cross product)	Reduced input mapping (blocking)	perfect match result
Bibliographic	DBLP-ACM	2,616	2,294	6 million	494,000	2224
	DBLP-Scholar	2,616	64,263	168.1 million	607,000	5343
E-commerce	Amazon-GoogleProducts	1,363	3,226	4.4 million	342,761	1300
	Abt-Buy	1,081	1,092	1.2 million	164,072	1097

[VLDB'10] Koepcke, Thor, Rahm: *Evaluation of entity resolution approaches on real-world match problems*. PVLDB 2010

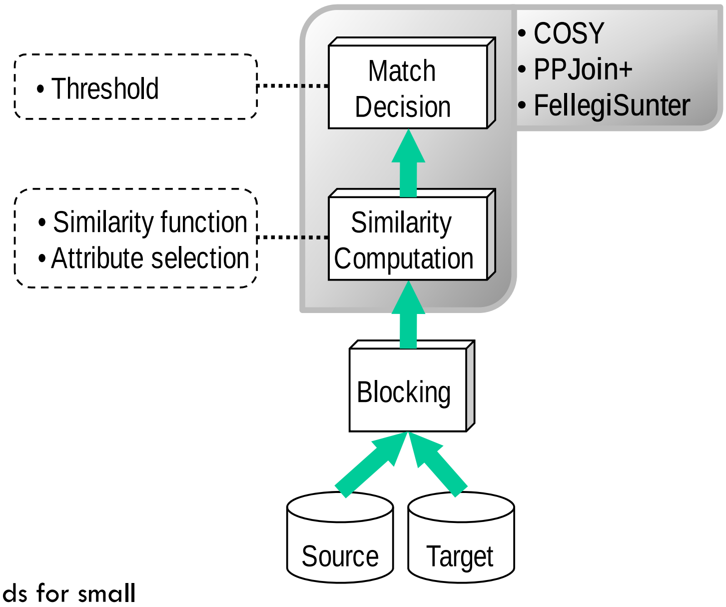
[VLDB'09] Koepcke, Thor, Rahm: Comparative evaluation of entity resolution approaches with FEVER. PVLDB 2009

FEVER

Framework for EValuating Entity Resolution

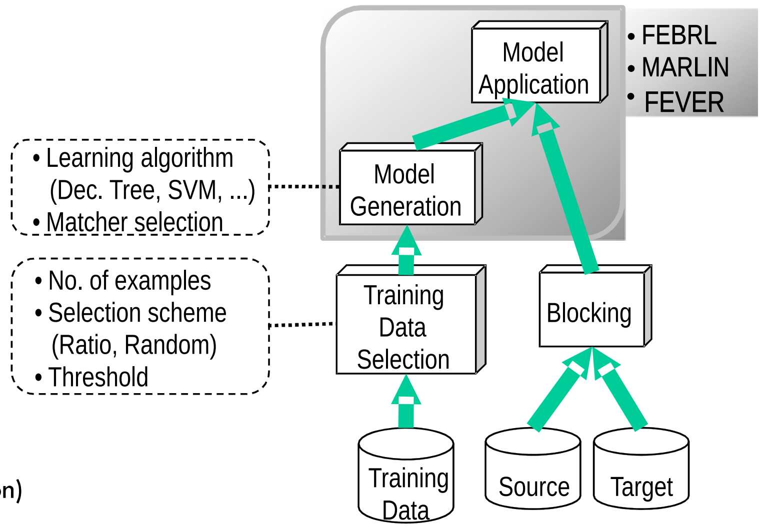
Non-learning approaches

- **COSY** (commercial system)
  - Black box similarity function
  - Overall and attribute level thresholds
- **PPJoin+**
  - Similarity functions: Cosine, Jaccard
  - Threshold
- **FellegiSunter** (FEBRL)
  - Similarity functions: TokenSet, Trigram, Winkler
  - Similarity threshold
- Match configurations
  - Use of 1 or 2 attributes
  - Use of FEVER to optimize thresholds for small subset of input data (500 object pairs)



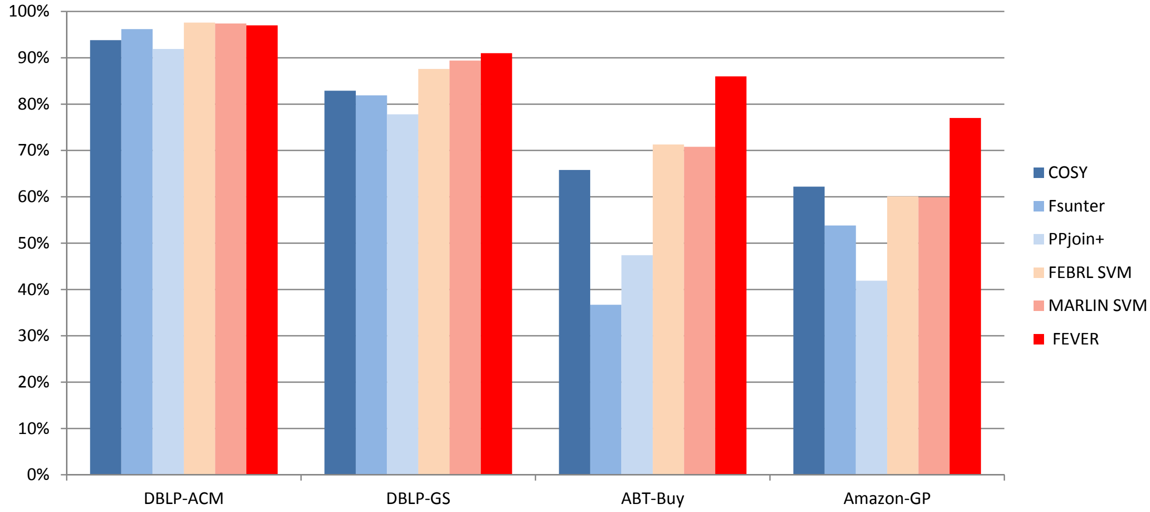
Learning-based approaches

- **FEBRL**
  - 3 matchers: Winkler, Tokenset, Trigram
  - Learning algorithm: SVM
- **MARLIN**
  - 2 matchers: Edit Distance, Cosine
  - Learning algorithms: SVM , decision tree
  - single step vs. two level learning
- **FEVER**
  - Trigram and TF/IDF matchers
  - Majority consensus from 3 learners (SVM , decision tree, logistic regression)
- Match configurations
  - Use of 1 or 2 attributes
  - Small training size (max. 500 object pairs with balanced matches/non-matches)



Quality (F-Measure) Comparison

13



- Bibliographic tasks are simpler than E-commerce tasks
- Learning-based approaches perform best, especially for difficult match problems
  - SVM most promising learner
  - FEVER benefits from majority consensus of 3 learners
- COSY relatively good / PPJoin+ limited to 1 attribute

Efficiency results

14

	Blocked (s)	Cartesian (s)
COSY	1 – 44	2– 434
FellegiSunter	2 – 2,800	17 – >500,000
PPJoin+	<1 – 3	<1 – 7
FEBRL SVM	99-480	1,400 – >500,000
MARLIN SVM	20-380	2,200 – >500,000

- PPJoin+ and COSY very fast, even for Cartesian product
- FellegiSunter slowest non-learning approach
- Learning-based approaches very slow
  - require blocking

Observations

15

- Evaluations reveal big differences regarding match quality and execution times
- Effective approaches: Learning-based approaches,
  COSY (partly)
- Fast approaches: COSY, PPJoin+
- Weak points:
  - Combination of several attributes requires higher tuning/training effort
  - E-commerce tasks could not be effectively solved. More sophisticated methods are needed there
  - Scalability to large test cases needs to be better addressed

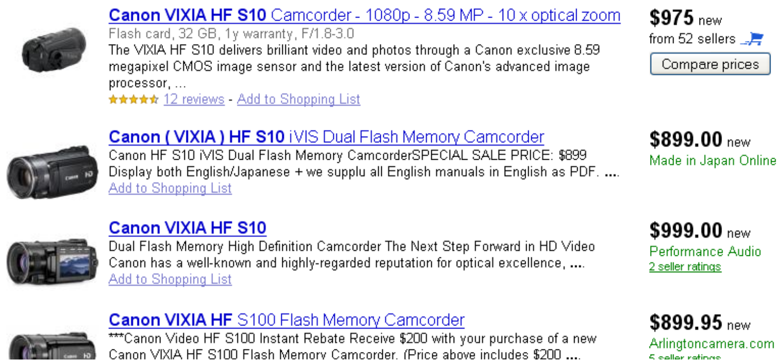
Outline

16

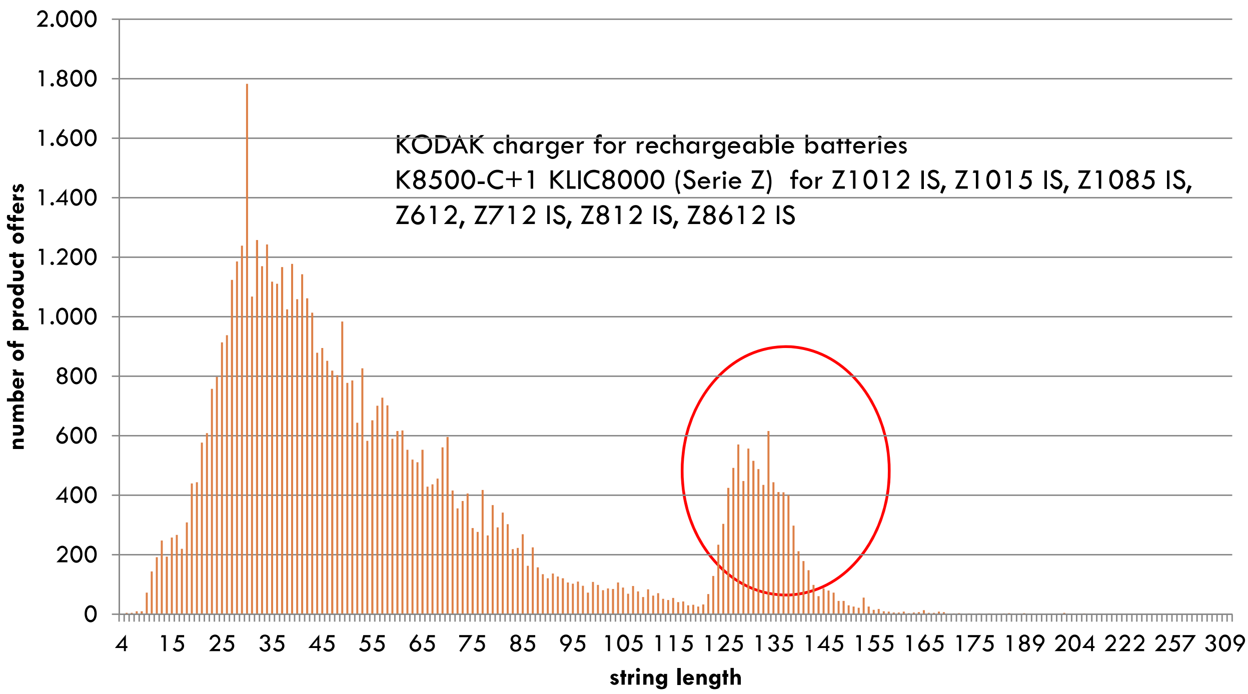
- Motivation
- Existing Frameworks and their Performance
  - Qualitative comparison [DKE'10]
  - Quantitative comparison [VLDB'10]
- Matching of Product Offers
  - Challenges
  - System design with use of extracted features (e.g. product codes)
  - Evaluation
- Parallel Matching in the Cloud
  - Blocking-based Object Matching with MapReduce
  - Load Balancing
    - Block-Split Approach
    - Experimental Results
  - Dedoop tool
- Conclusions & Future Work

Matching product offers: challenges

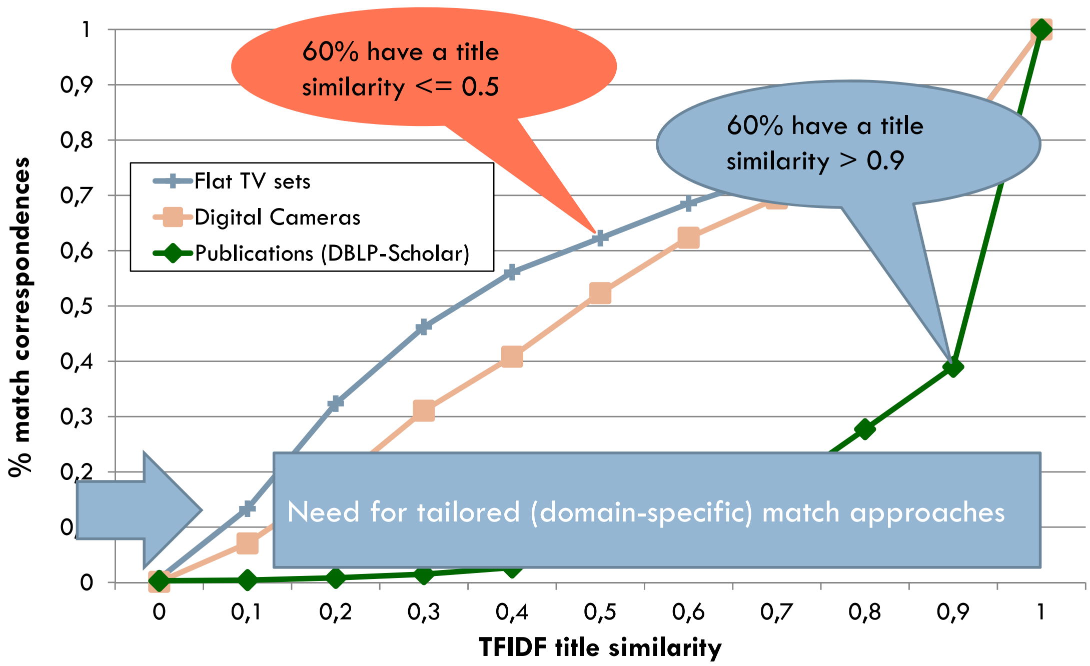
- huge number of offers (many products, many shops)
- many similar but different products
- heterogeneous, shop-specific product categorizations
- frequent changes of products and offers
- few available attributes, not well structured
- product ids (EAN, UPC, GTIN) often unavailable (or misleading)
- poor data quality …



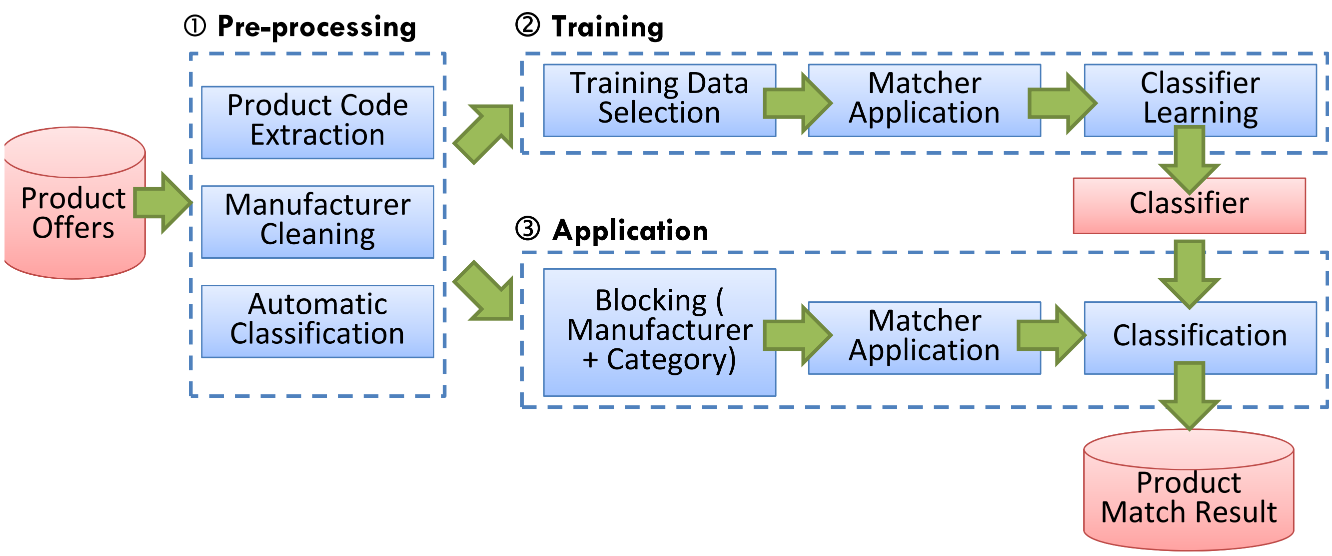
Heterogeneous and verbose strings



Standard string matcher fail



System design*



* Koepcke, Thor, Thomas, Rahm: *Tailoring entity resolution for matching product offers.* Proc. EDBT, 2012

Product code

- Frequent existence of specific product codes for certain products

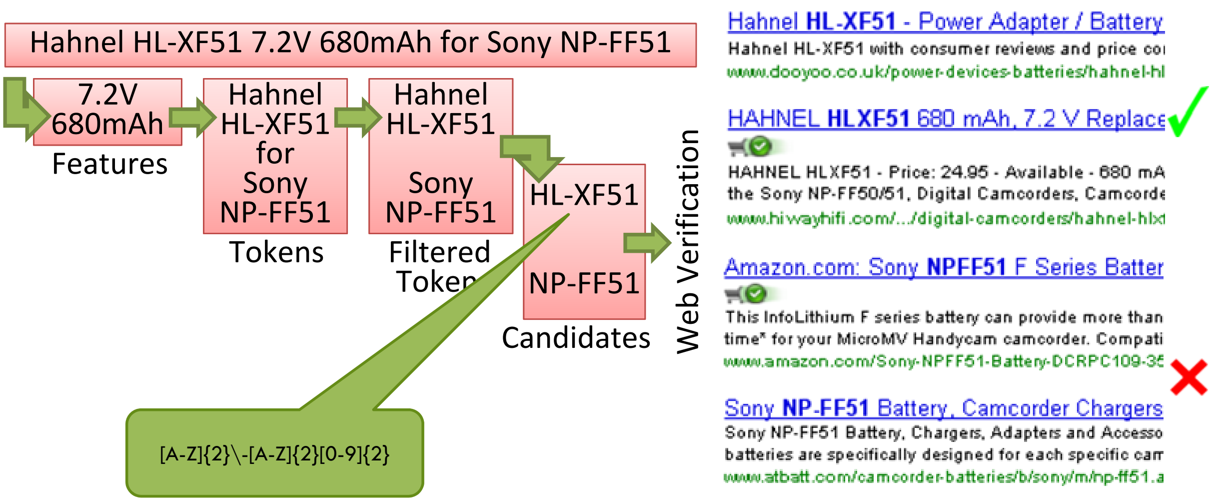
- Product code = manufacturer-specific identifier
  Any sequence consisting of alphabetic, special, and numeric characters split by an arbitrary number of white spaces.

➢ Utilize to differentiate similar but different products.

Canon VIXIA HF S100 Camcorder - 1080p - 8.59 MP

Hahnel HL-XF51 7.2V 680mAh for Sony NP-FF51

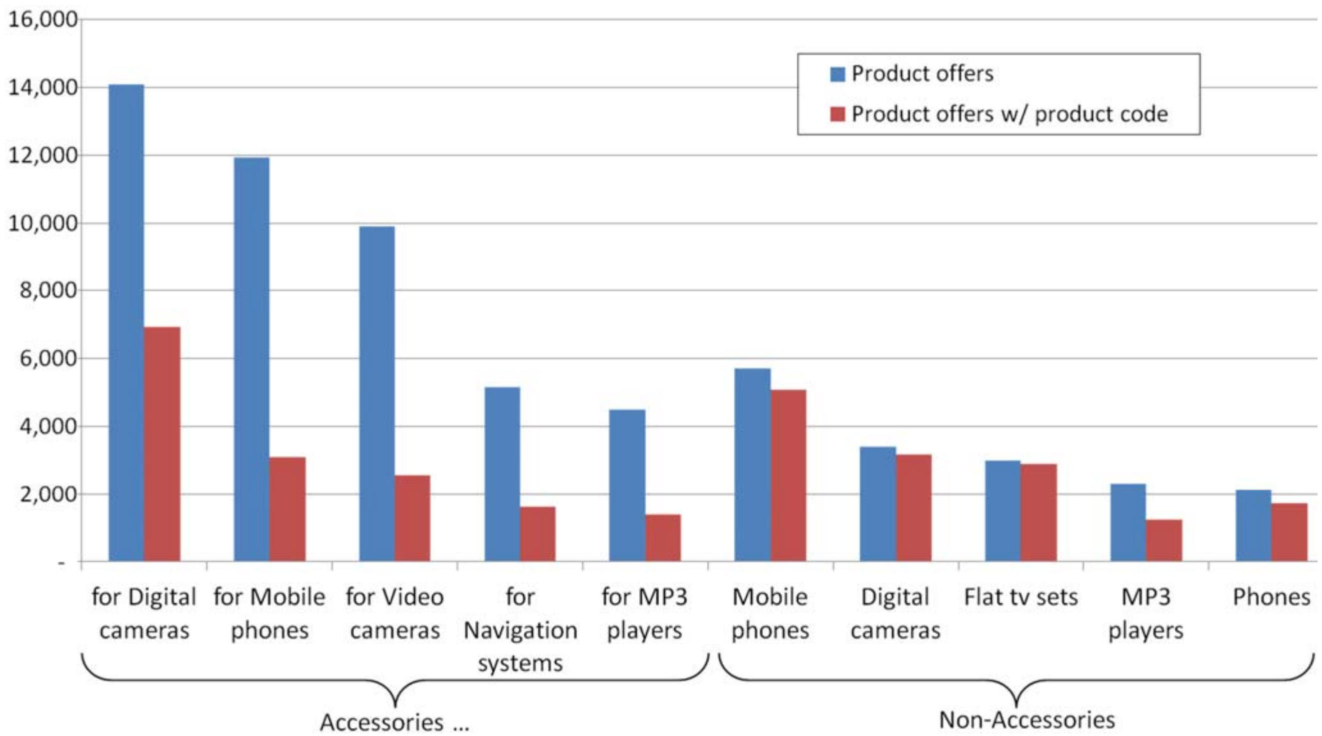
Product code extraction



Evaluation dataset

- 102,182 offers for electronic products and accessory products
- 71 product categories
- Few attributes:
  - Title, description, manufacturer, price
- No clean product reference set
- Offer to offer matching
  - much more challenging than offer-to-product matching

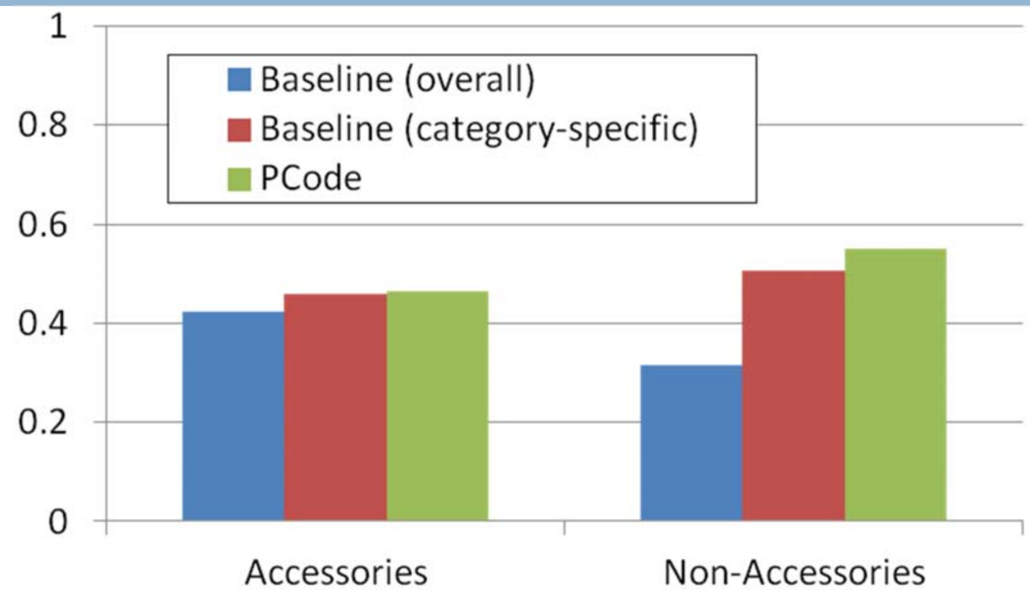
Product code extraction



Quality of product code extraction

	Precision	Recall	F-Measure
Overall	79%	56%	66%
Non-Accessories	79%	64%	71%
Accessories	79%	48%	60%
Mobile Phones	93%	86%	89%

Baseline vs. Product code matching



- Generic string matching on title and description attributes
- EAN-based reference matching

Limitation of EAN-based reference mapping

- Problems of EAN (UPC, GTIN)-based match decisions
  - Different codes for the same product based on the manufacturer's country or target market
  - Existence of offers for different products having the same EAN

Title	EAN
Canon Digital **Ixus 90** IS 10MPix 3fach opt. Zoom 3"	4960999570563
Canon Digital **Ixus 90** IS 10MPix 3fach opt. Zoom 3"	4960999570563
Digital **IXUS 90** IS - Digitalkamera - Kompaktkamera	8714574515588
Canon Digital **IXUS 90** IS Digitalkamera (10 Megapixel, 3-fach opt. Zoom, 3" Display, Bildstabilisator)	8714574515595

→ **Need for manually verified reference mapping**

EAN-based vs. Manual reference mapping

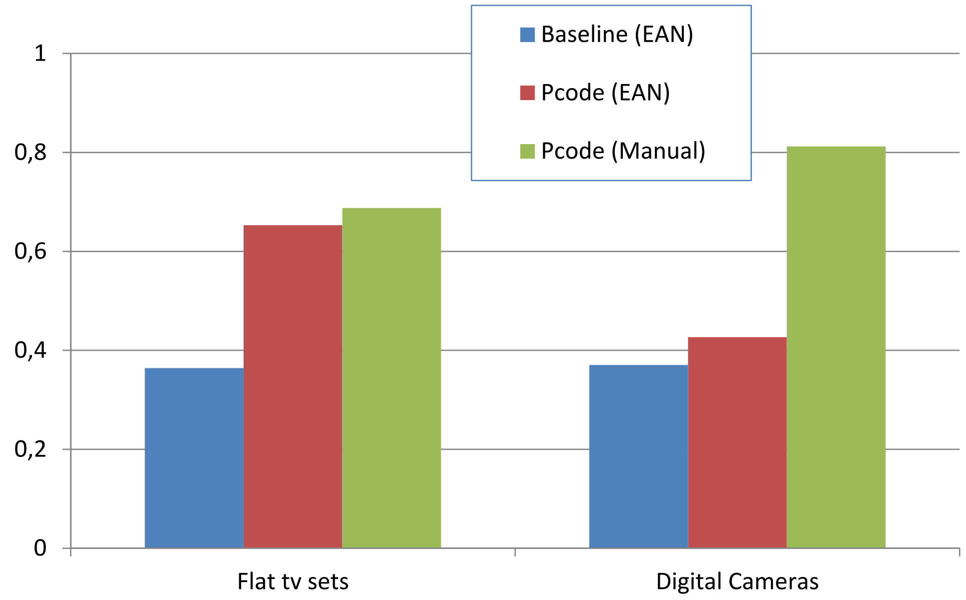
Title	EAN
Canon Digital Ixus 90 IS 10MPix 3fach opt. Zoom 3"	4960999570563
Canon Digital Ixus 90 IS 10MPix 3fach opt. Zoom 3"	4960999570563
Digital IXUS 90 IS - Digitalkamera - Kompaktkamera	8714574515588
Canon Digital IXUS 90 IS Digitalkamera (10 Megapixel, 3-fach opt. Zoom, 3" Display, Bildstabilisator)	8714574515595

EAN-based reference mapping
- 3 clusters
- 1 correspondence

Manually determined mapping
- 1 cluster
- 6 correspondences

Category	Mapping	#Clusters	Average Cluster size	#Corres-pondences
Flat TV sets	EAN-based	1,222	2.5	5,293
	manual	1,103	2.7	6,509
Digital cameras	EAN-based	1,087	3.1	8,375
	manual	504	6.8	32,571

EAN-based vs. Manual reference mapping (evaluation results)



Observations

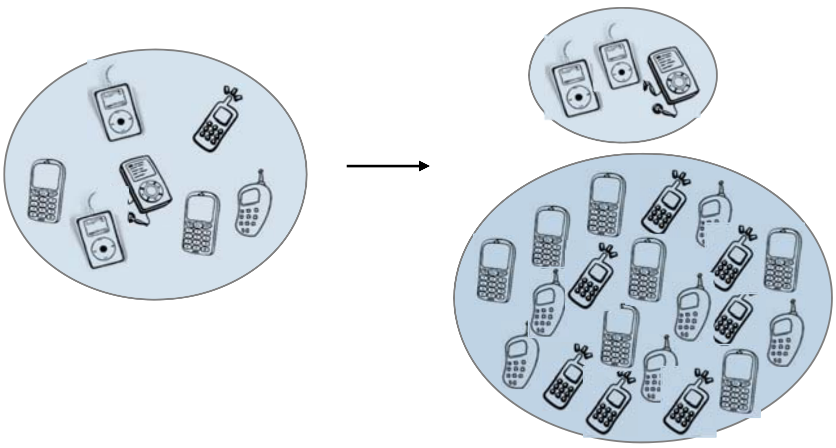
- Product matching requires tailored ER approaches
- Key characteristics of proposed approach
  - Comprehensive preprocessing and data cleaning
  - Pattern-based extraction and web-based verification of product codes
  - Category-specific, learned match strategies
- Limitations of EAN-based reference mappings for evaluation
- Future work:
  - Utilizing further extracted features
  - Matching offers to products

Outline

- Motivation
- Existing Frameworks and their Performance
  - Qualitative comparison [DKE'10]
  - Quantitative comparison [VLDB'10]
- Matching of Product Offers
  - Challenges
  - System design with use of extracted features (e.g. product codes)
  - Evaluation
- Parallel Matching in the Cloud
  - Blocking-based Object Matching with MapReduce
  - Load Balancing
    - Block-Split Approach
    - Experimental Results
  - Dedoop tool
- Conclusions & Future Work

How to speed up object matching?

- **Blocking** to reduce search space



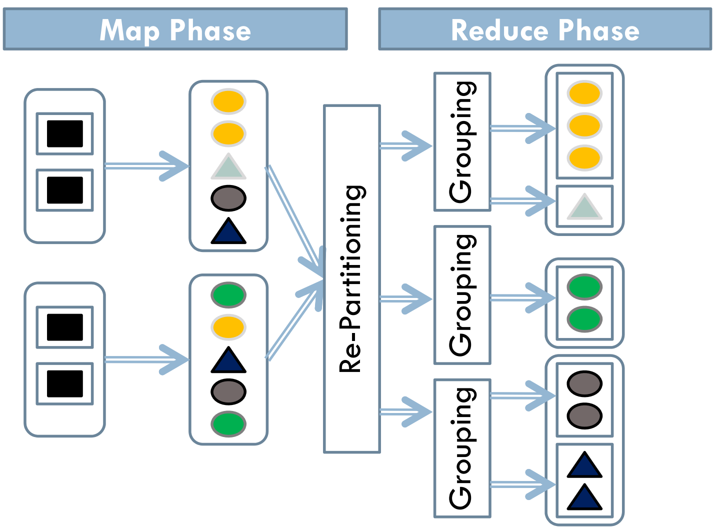
- **Parallelization**
  - Split match computation in sub-tasks to be executed in parallel
  - Exploitation of cloud infrastructures and frameworks like Map/Reduce

MapReduce

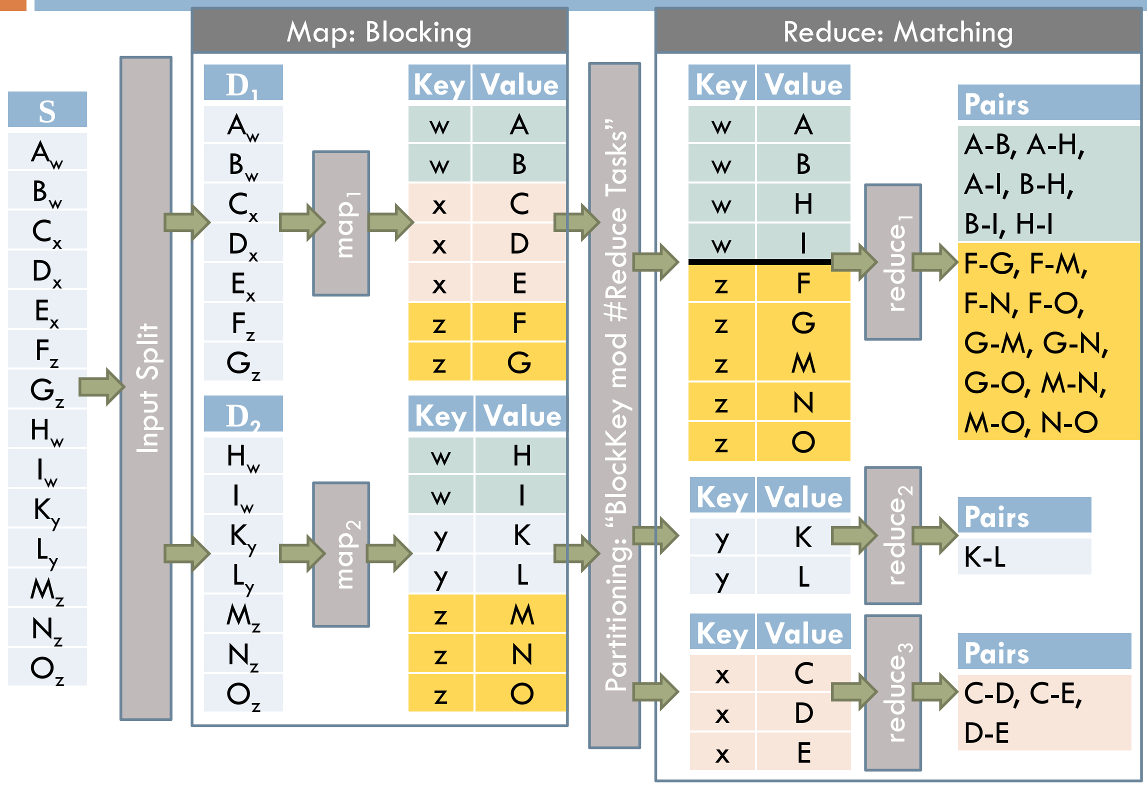
- Programming model for distributed computation
- Dataflow defined by map and reduce functions
  - map: $(key_{in}, value_{in}) \rightarrow list(key_{tmp}, value_{tmp})$
  - reduce: $(key_{tmp}, list(value_{tmp})) \rightarrow list(key_{out}, value_{out})$
- MapReduce framework hides messy details
  - Automatic parallelization
  - Robustness, e.g., handles node failures
  - Scalability
  - …

MapReduce

- **Map** function applied on each input object to generate **key-value pairs**
- Each key-value pair is assigned to a **reduce task**
- **Reduce** function is invoked for each object group with same key



Blocking + MapReduce: Basic scheme



Load Balancing

- Data skew leads to unbalanced workload
  - Large blocks prevent utilization of more than a few nodes
  - Deteriorates scalability and efficiency
  - Unnecessary costs (you also pay for underutilized machines!)
- Key ideas for load balancing
  - Additional MR job to determine blocking key distribution, i.e., number and size of blocks (per input partition)
  - Global load balancing that assigns (nearly) the same number of pairs to reduce tasks

Load Balancing Approaches

37

- Load balancing strategies for parallel object matching with general blocking [ICDE'12]
  - **BlockSplit:** Split large blocks into sub-blocks
  - **PairRange:** Global enumeration and tailored distribution of all pairs
- Variation for Sorted Neighborhood [CSRD'12]

[ICDE'12] Kolb, Thor, Rahm: *Load Balancing for MapReduce-based Entity Matching.* **Proc. Int. Conf. on Data Engineering, 2012**

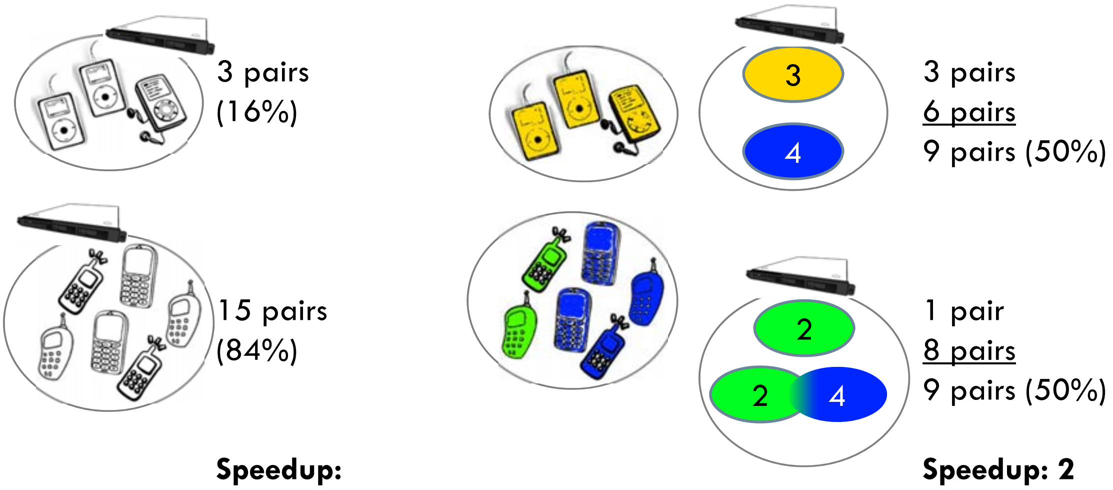
[CSRD'12] Kolb, Thor, Rahm: *Multi-pass Sorted Neighborhood Blocking with MapReduce.* Computer Science - Research and Development, 2012

Block Split: 1 slide illustration

38

- Example: 3 MP3 players + 6 cell phones → 18 pairs (1 time unit)
- Parallel matching on 2 (reduce) nodes

naiive approach **BlockSplit**



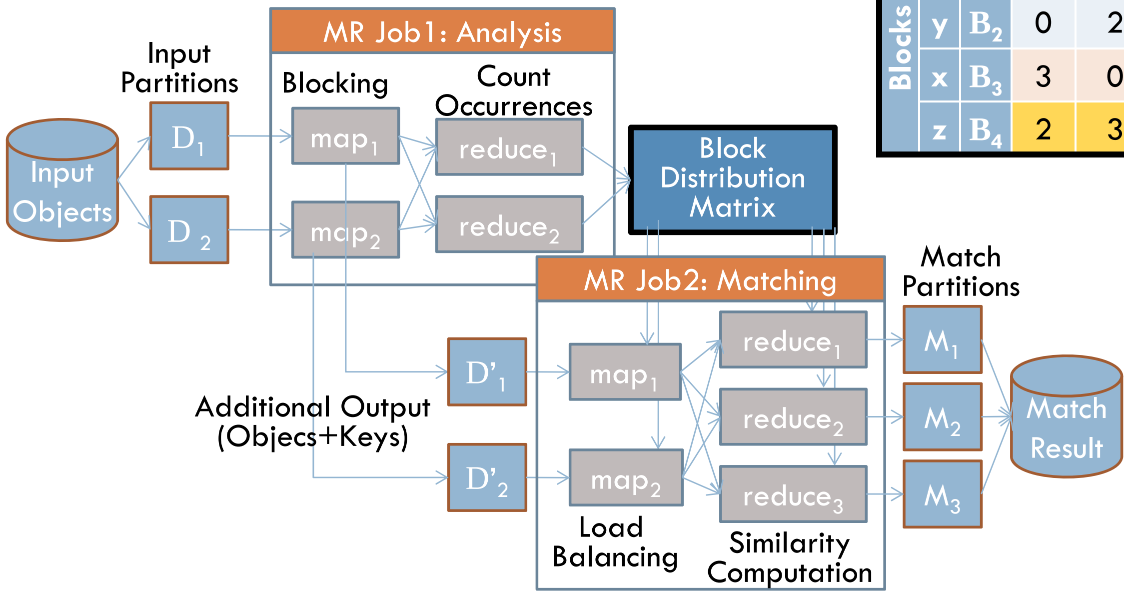
Speedup: 18/15=1.2

Speedup: 2

Load Balancing for MR-based Object Matching

Partition	D_1							D_2							
Object	A	B	C	D	E	F	G	H	I	K	L	M	N	O
Blocking Key	w	w	x	x	x	z	z	w	w	y	y	z	z	z

		Partition		Overall	
		D_1	D_2	#O	#P
w	B_1	2	2	4	6
y	B_2	0	2	2	1
x	B_3	3	0	3	3
z	B_4	2	3	5	10



BlockSplit

- Large blocks split into *m* sub-blocks
  - according to *m* input partitions
  - large if $\#P_{Block} > \#P_{Overall} / \#Reducer$
- Two types of match tasks
  - Single (small blocks and sub-blocks)
  - Two sub-blocks
- Greedy load balancing
  - Sort match tasks by number of pairs in descending order
  - Assign match task to reducer with lowest number of pairs
- **Example**
  - r=3 reduce tasks, split B_4 in m=2 sub-blocks
  - B_4's match tasks: $B_{4.1}$, $B_{4.2}$, and $B_{4.1\times 2}$

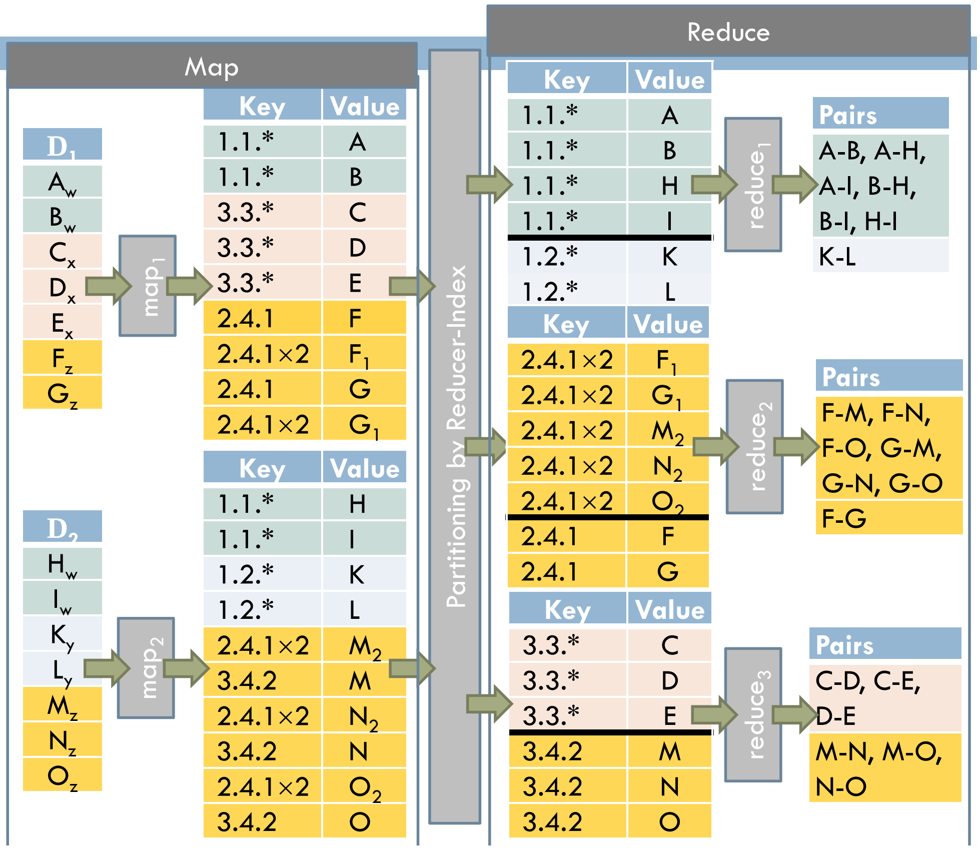
		Partition		Overall	
		D_1	D_2	#O	#P
w	B_1	2	2	4	6
y	B_2	0	2	2	1
x	B_3	3	0	3	3
z	B_4	2	3	5	10

Block Tasks	#P	Reducer
B_1	6	
$B_{4.1\times 2}$	6	
B_3	3	
$B_{4.2}$	3	
B_2	1	
$B_{4.1}$	1	

BlockSplit: MR-Dataflow

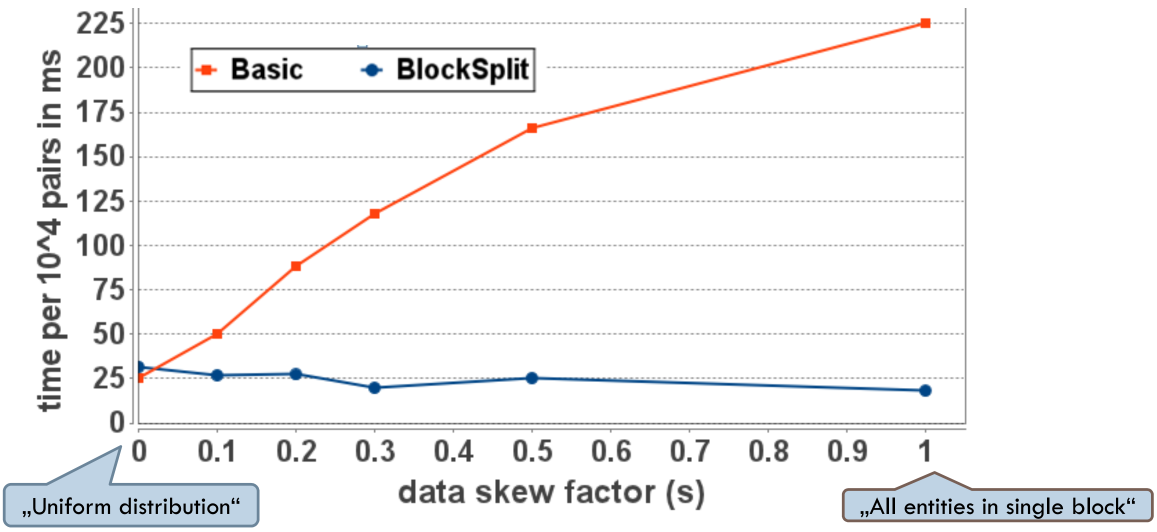
MapReduce Techniques

- MapKey = ReducerIndex + MatchTask
- Replicate objects of sub-blocks



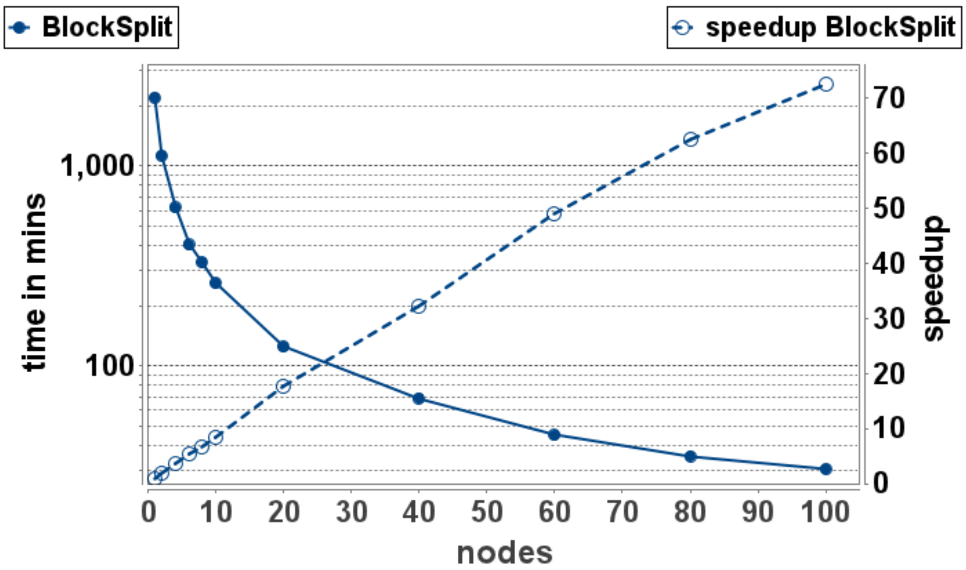
Evaluation: Data Skew

- Evaluation on Amazon EC infrastructure using Hadoop
- Matching of 114.000 product records
- BlockSplit robust against data skew

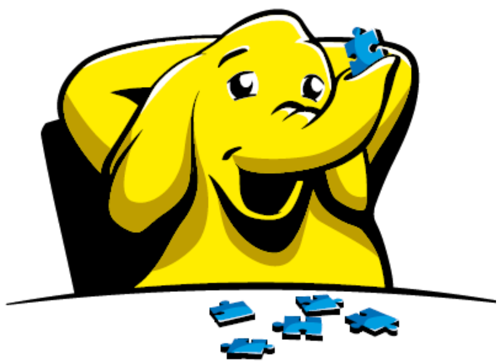


Evaluation: Scalability

- BlockSplit is scalable



Dedoop: Efficient Deduplication with Hadoop

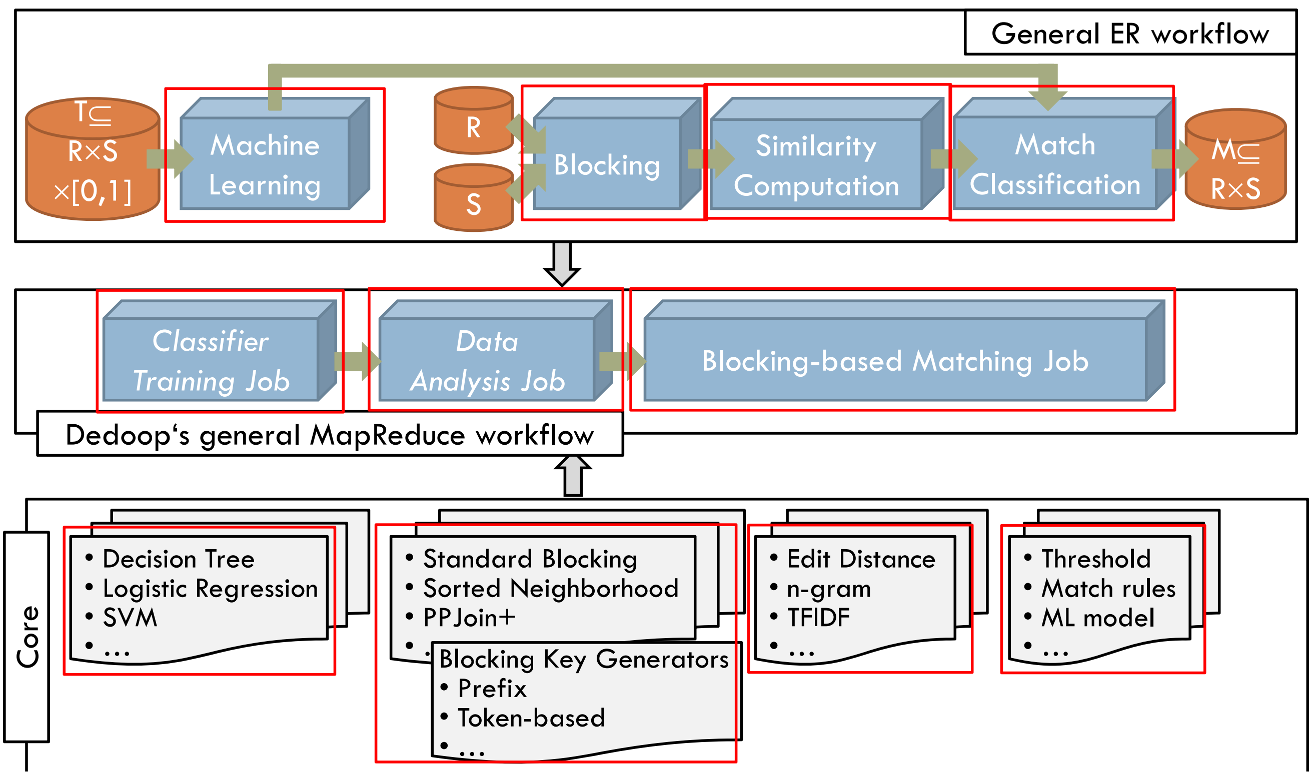


- Parallel execution of Entity Resolution workflows with Hadoop
- Browser-based workflow specification
- Support for powerful match strategies
  - Many blocking and matching techniques
  - Learning-based match strategies
  - Redundancy-free matching for multi-key blocking
- Automatic generation and submission of corresponding Map-Reduce-Workflows
- Support for automatic Load Balancing strategies, e.g. Block-Split
- Progress Monitoring

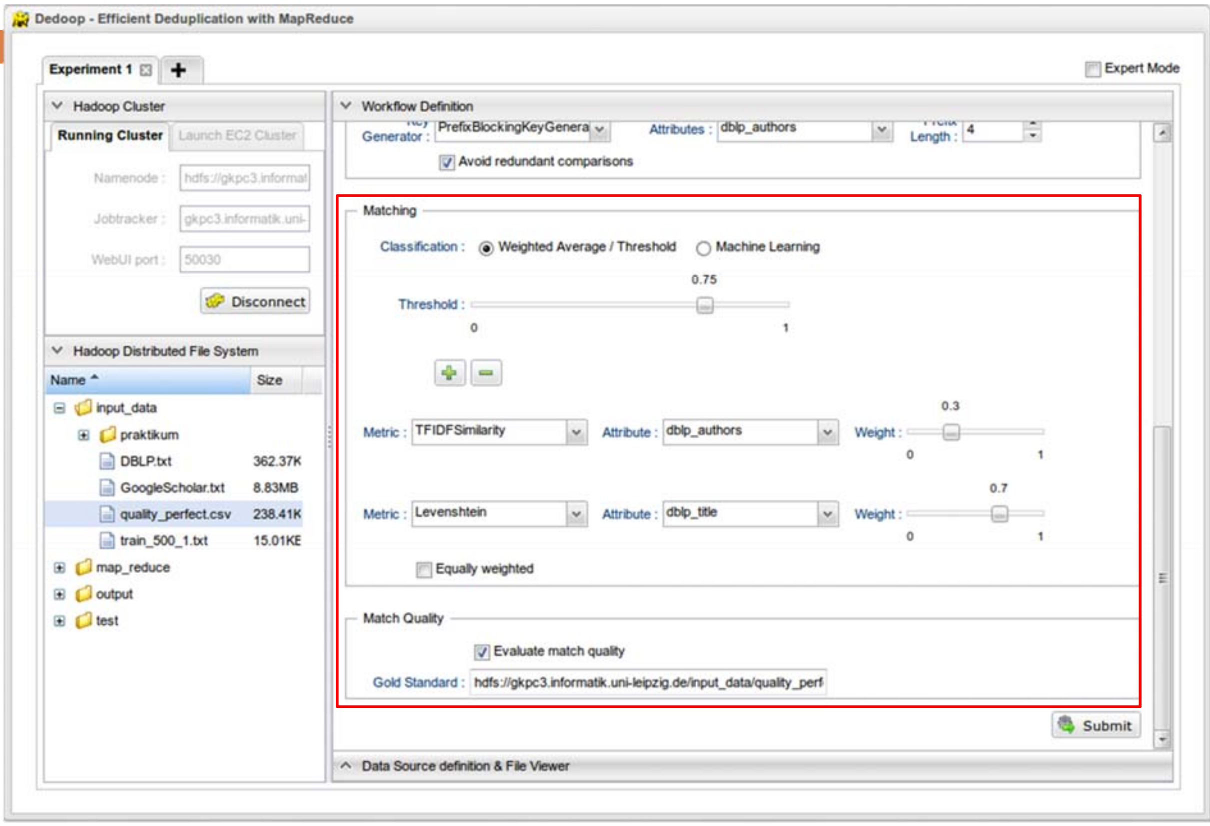
Dedoop (2)

- Significant simplification compared to the specification and use of “hard coded” MapReduce workflows
  - Many MR jobs with tailored map, reduce, part, sort, and group functions
  - Specification of key, value, input format & output format classes
  - Packaging in single jar archive (=Kernel)
  - Workflow execution: `hadoop -jar Kernel.jar <params>`
- Tedious file handling for input / output
  - Copy input data to DFS: `hadoop dfs -copyFromLocal localfile remotedir`
  - Copy output data from DFS to local disk further processing
- Simplification of ernormous parameterization effort
  - Specification and order of MapReduce jobs (“driver classes”)
  - Some workflows require preprocessing jobs (classifier training, IDF index creation)
  - Output/input directories (job_{i+1} consumes output of job_i)
  - Blocking key generation functions, Similarity metrics, and attributes
  - Different handling of different input sources

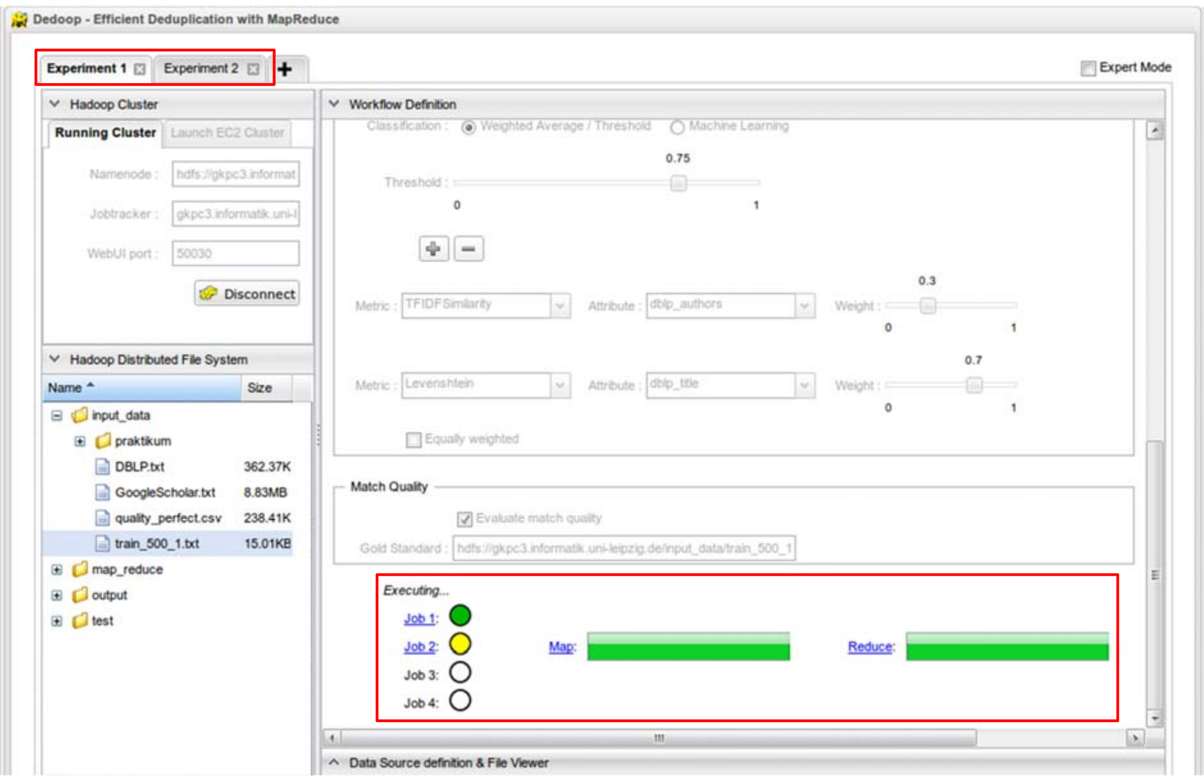
Dedoop Overview



Browser-based workflow specification



Workflow submission & progress monitoring



Outline

Conclusions

- Challenge: Fast and effective object matching for large, real-world (dirty) datasets
- Many useful tools and frameworks, but improvements still needed
- Domain-specific approaches needed for challenging problems such as matching product offers
  - Extensive data preprocessing and cleaning
  - Extraction of match-relevant features such as product codes
  - Multiple match strategies, e.g. per product category
- Cloud-based parallel blocking and matching
  - Straight-forward utilization of MapReduce possible
  - ... but doing it efficiently requires some work
- Effective load balancing approaches such as Block-Split
- Dedoop tool for easy and efficient Hadoop-based matching

Future Work

51

- Principled approach for domain-specific matching
  - Plug-In architecture for different feature extractors and preprocessing steps
  - More support for context-based matchers
- Reduction of manual work needed
  - Preprocessing
  - Configuration effort (matcher selection and combination, etc.)
- More usable learning-based approaches
  - Reduced training effort, e.g. by active learning
  - Improved scalability
- New application areas such as LOD link discovery
  - Combined use of ontology and object matching

Thank you!

References

52

- Kolb, L.; Thor, A.; Rahm, E.: *Dedoop: Efficient Deduplication with Hadoop*. Proc. VLDB Endowment 5(12), 2012 (demo)
- Kolb, L.; Thor, A.; Rahm, E.: *Load Balancing for MapReduce-based Entity Resolution*. Proc. ICDE, 2012
- Kolb, L.; Thor, A.; Rahm, E.: *Multi-pass Sorted Neighborhood Blocking with MapReduce*. CSRD 27(1), 2012
- Koepcke, H.; Thor, A.; Thomas, S., Rahm, E.: *Tailoring entity resolution for matching product offers*. Proc. EDBT, 2012
- Koepcke, H.; Thor, A.; Rahm, E.: *Evaluation of entity resolution approaches on real-world match problems*. Proc. VLDB Endowment 3(1), 2010
- Koepcke, H.; Thor, A.; Rahm, E.: *Learning-based approaches for matching web data entities*. IEEE Internet Computing 14(4), 2010
- Koepcke, H.; Rahm, E.: *Frameworks for entity matching: A comparison*. Data & Knowledge Engineering, 2010