Rough Sets and Big Data

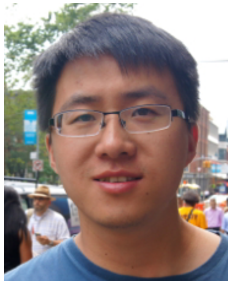
Tianrui LI （李天瑞）
Southwest Jiaotong University, Chengdu, China
trli@swjtu.edu.cn

Lab Website: http://sist.swjtu.edu.cn/CCIT
Personal Website: http://userweb.swjtu.edu.cn/Userweb/trli30/index.htm

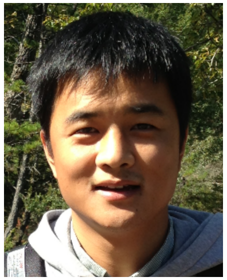
Main co-authors

- Dun Liu (刘盾)
  - SW Jiaotong University

- Hongmei Chen (陈红梅)
  - SW Jiaotong University



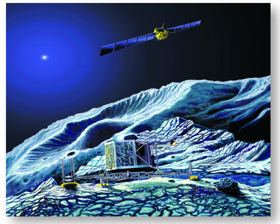
- Junbo Zhang (张钧波)
  - Microsoft Research



- Chuan Luo (罗川)
  - Sichuan University

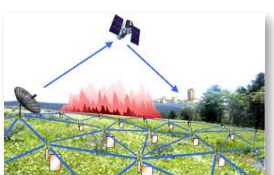
Background of Big Data

Background of Big Data

Scientific instruments
(collecting all sorts of data)

Social media and networks
(all of us are generating data)



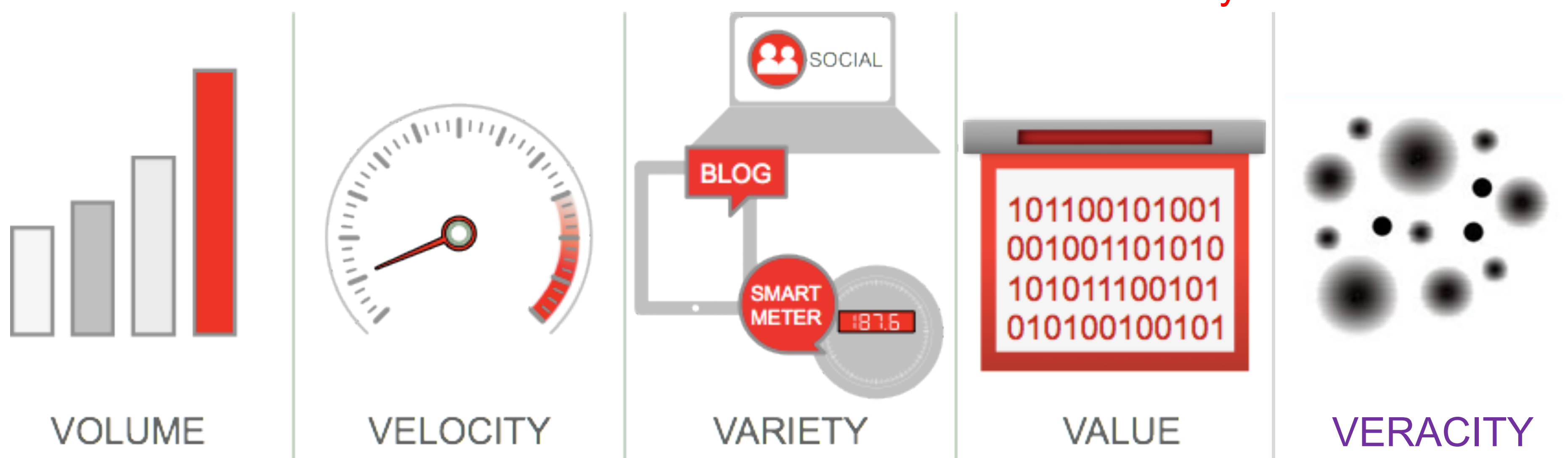

Sensor technology and networks
(measuring all kinds of data)

Mobile devices
(tracking all objects all the time)

Big Data $V^3/V^4/V^5$

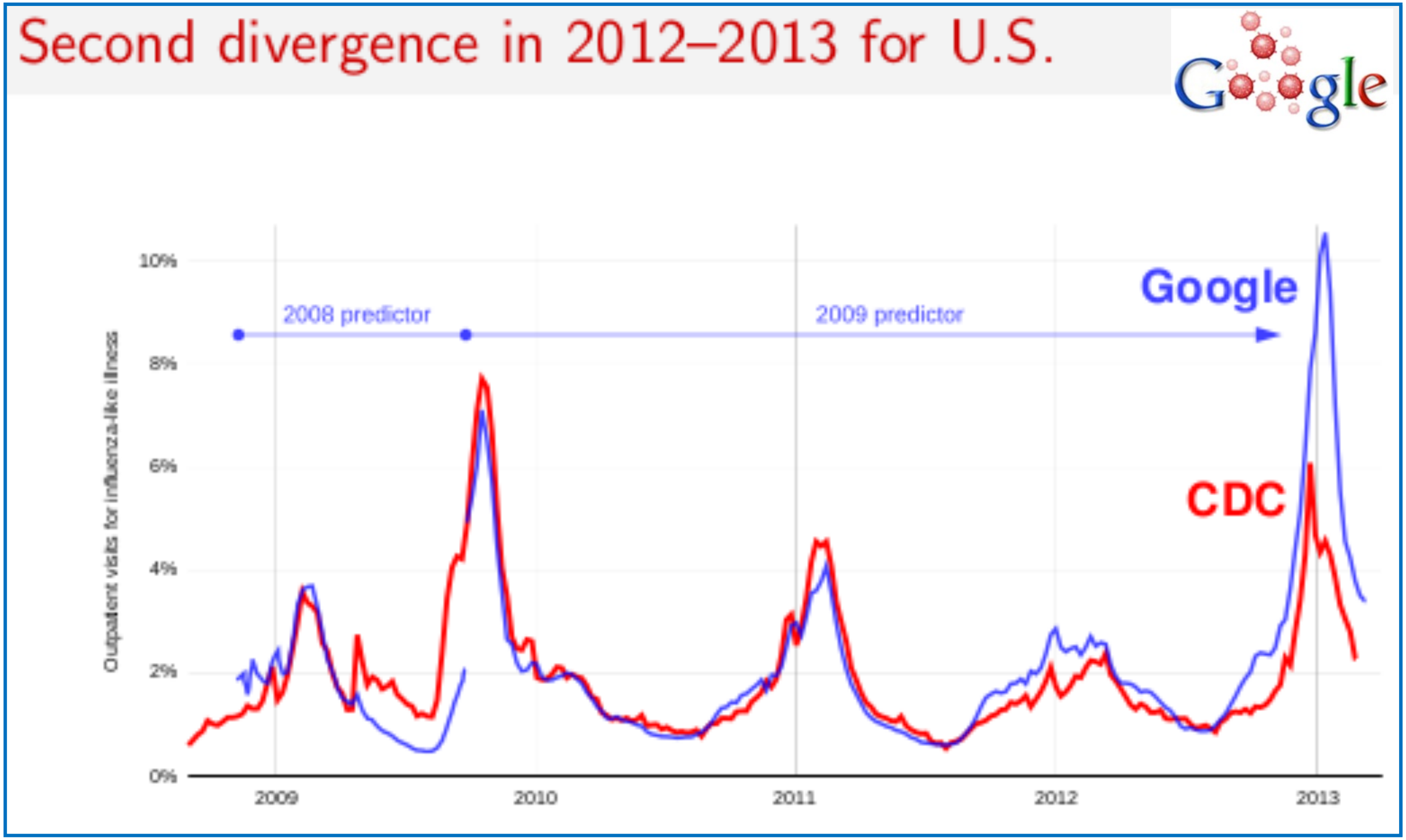
- Volume: Gigabyte (10^9), Terabyte (10^{12}), Petabyte (10^{15}), Exabyte (10^{18}), Zettabytes (10^{21})
- Variety: Structured, semi-structured, unstructured
- Velocity: Dynamic, sometimes time-varying

Data at Rest | Data in Motion | Data in Various Forms | Data with Low Value Density | Data in Doubt

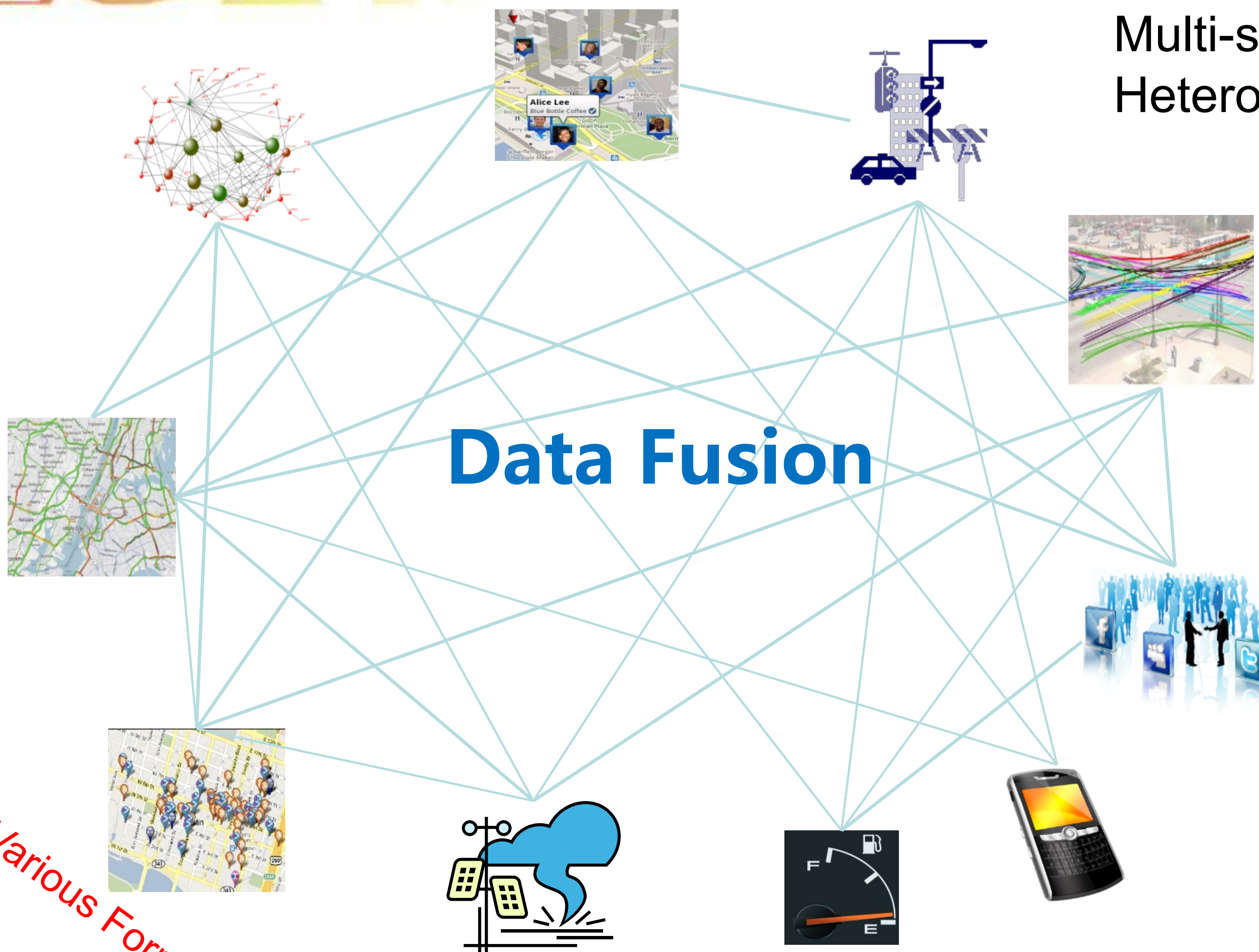


Google Flu Trends

Data in Motion



Multi-source
Heterogeneous

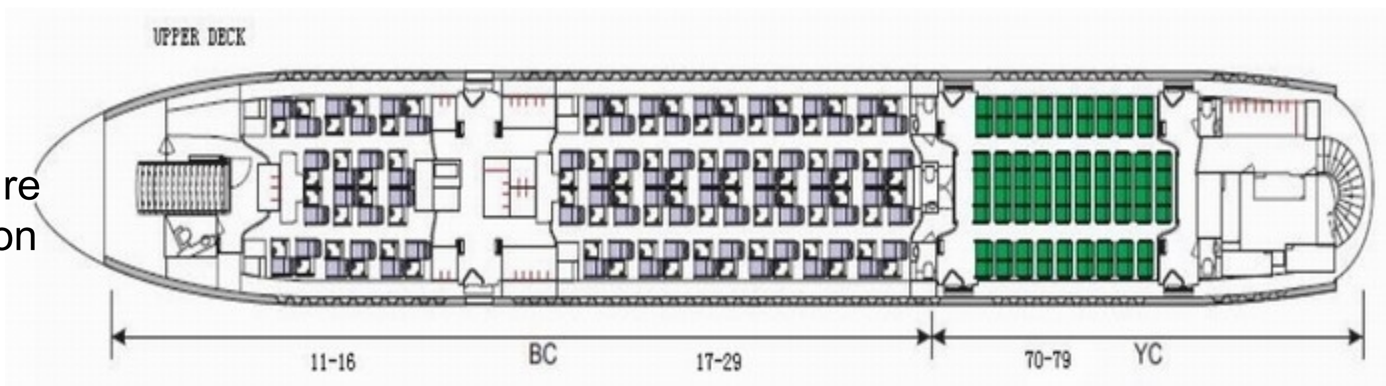


Data in Various Forms

Data with Low Value Density

Evaluation of Safety of HST

Pressure
Vibration
Noise
…



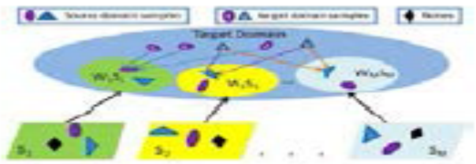
Malaysia Airlines MH370 Flight Incident

Data in Doubt

Granular Computing (GrC)

Granular computing (GrC), as an emerging computational and mathematical theory which describes and processes uncertain, vague, incomplete, and massive information, has been successfully used in knowledge discovery. Following are several representative GrC models.

- Computing with words
- Rough set theory
- Quotient space theory
- Others …



Multi-granularity Pattern Discovery

Multi-granularity Structure Association

Information granules/Granular construction ⇒ Knowledge representation/Pattern discovery/Cross-granular reasoning

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Data-intensive applications, challenges, techniques and technologies: A survey on Big Data

C.L. Philip Chen *, Chun-Yang Zhang

Department of Computer and Information Science, Faculty of Science and Technology, University of Macau, Macau, China

6. Underlying technologies and future researches

The advanced techniques and technologies for developing Big Data science is with the purpose of advancing and inventing the more sophisticated and scientific methods of managing, analyzing, visualizing, and exploiting informative knowledge from large, diverse, distributed and heterogeneous data sets. The ultimate aims are to promote the development and innovation of Big Data sciences, finally to benefit economic and social evolutions in a level that is impossible before. Big Data

6.1. Granular computing

Granular computing

When we talk about Big Data, the first property of it is its size. As *granular computing* (GrC) [142] is a general computation theory for effectively using granules such as classes, clusters, subsets, groups and intervals to build an efficient computational model for complex applications with huge amounts of data, information and knowledge, therefore it is very natural to employ granular computing techniques to explore Big Data. Intuitively, granular computing can reduce the data size into different level of granularity. Under certain circumstances, some Big Data problems can be readily solved in such way.

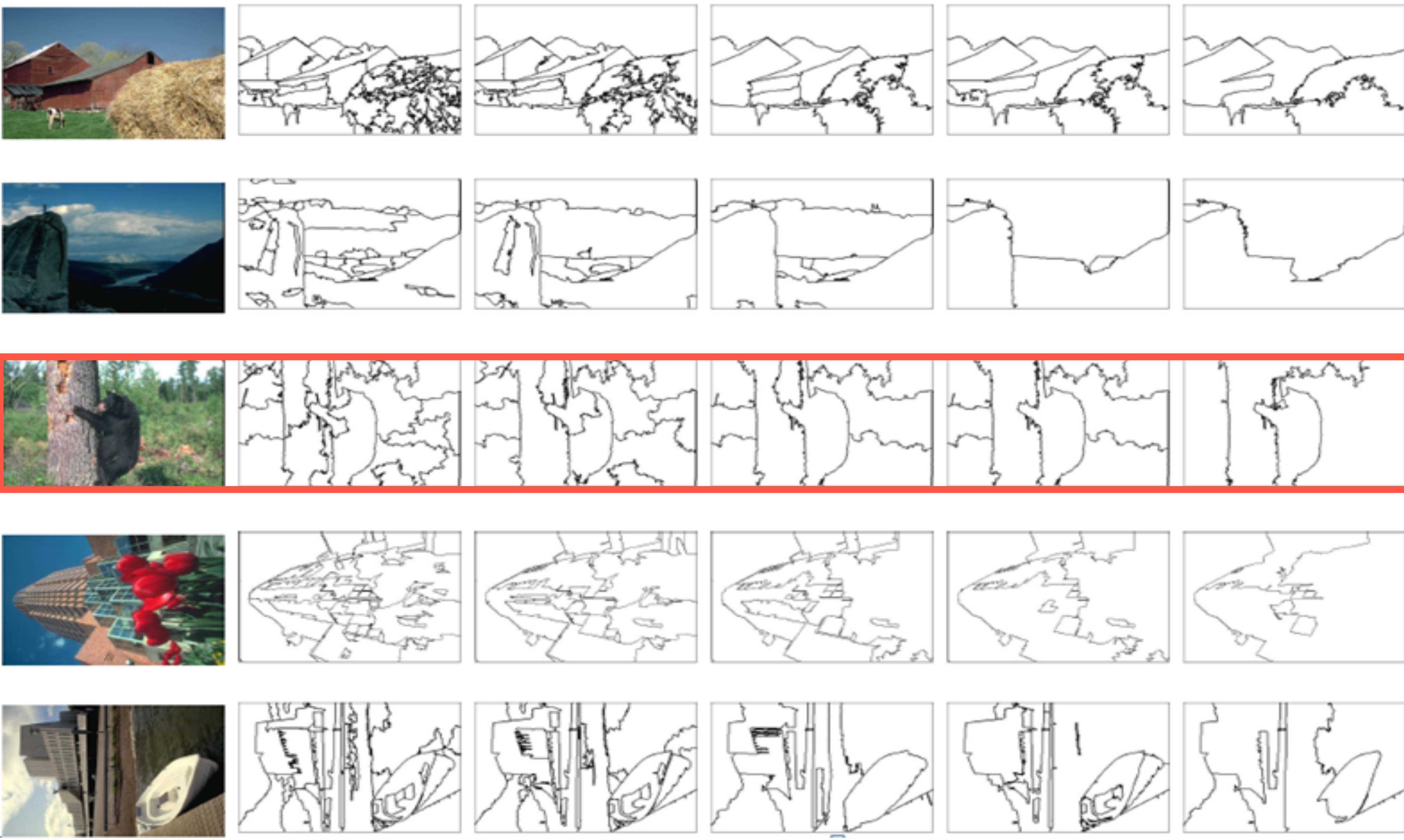
It is very natural to employ granular computing techniques to explore Big Data.

What is GrC

- GrC = Problem solving based on different levels of granularity (detail/abstraction)
  - Level of granularity is essential to human problem solving
- GrC attempts to capture the basic principles and methodologies used by human in problem solving

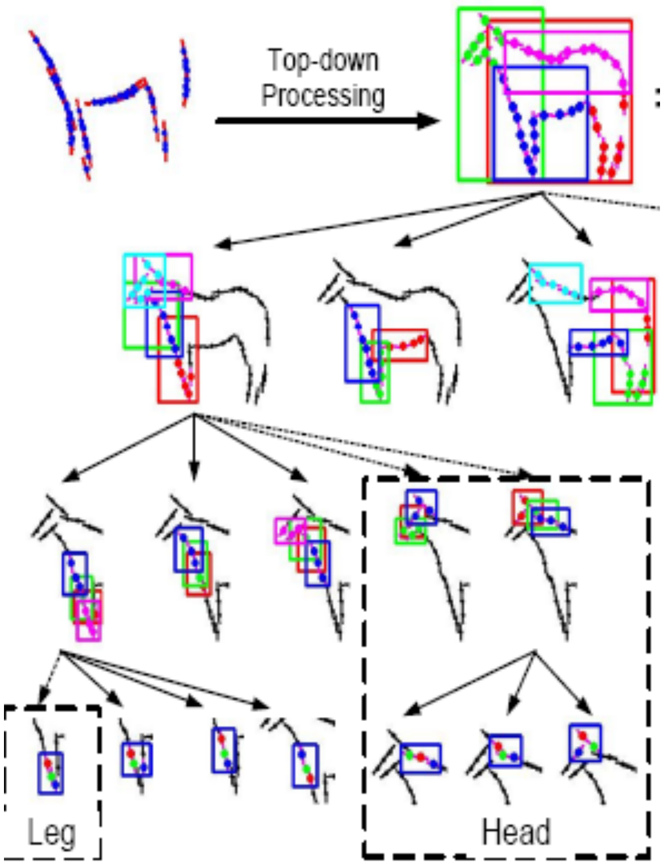
[Yao, Information Science, 2012]

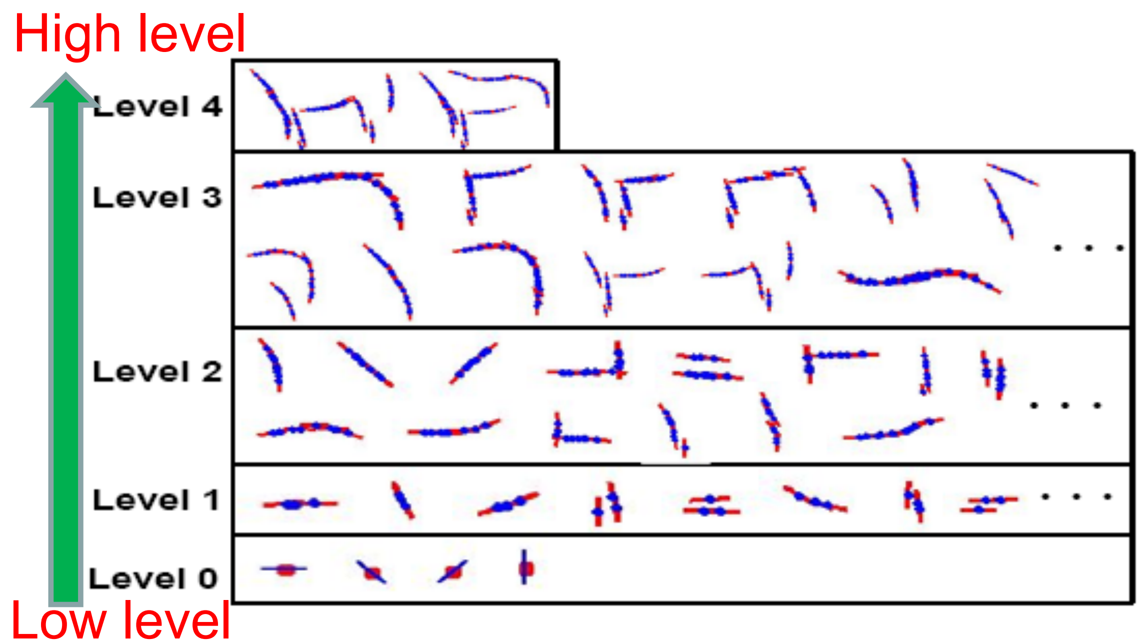
Example: Hierarchical Image Segment



Deep Learning: An Implementation of GrC

A Hierarchical Compositional System for Rapid Object Detection



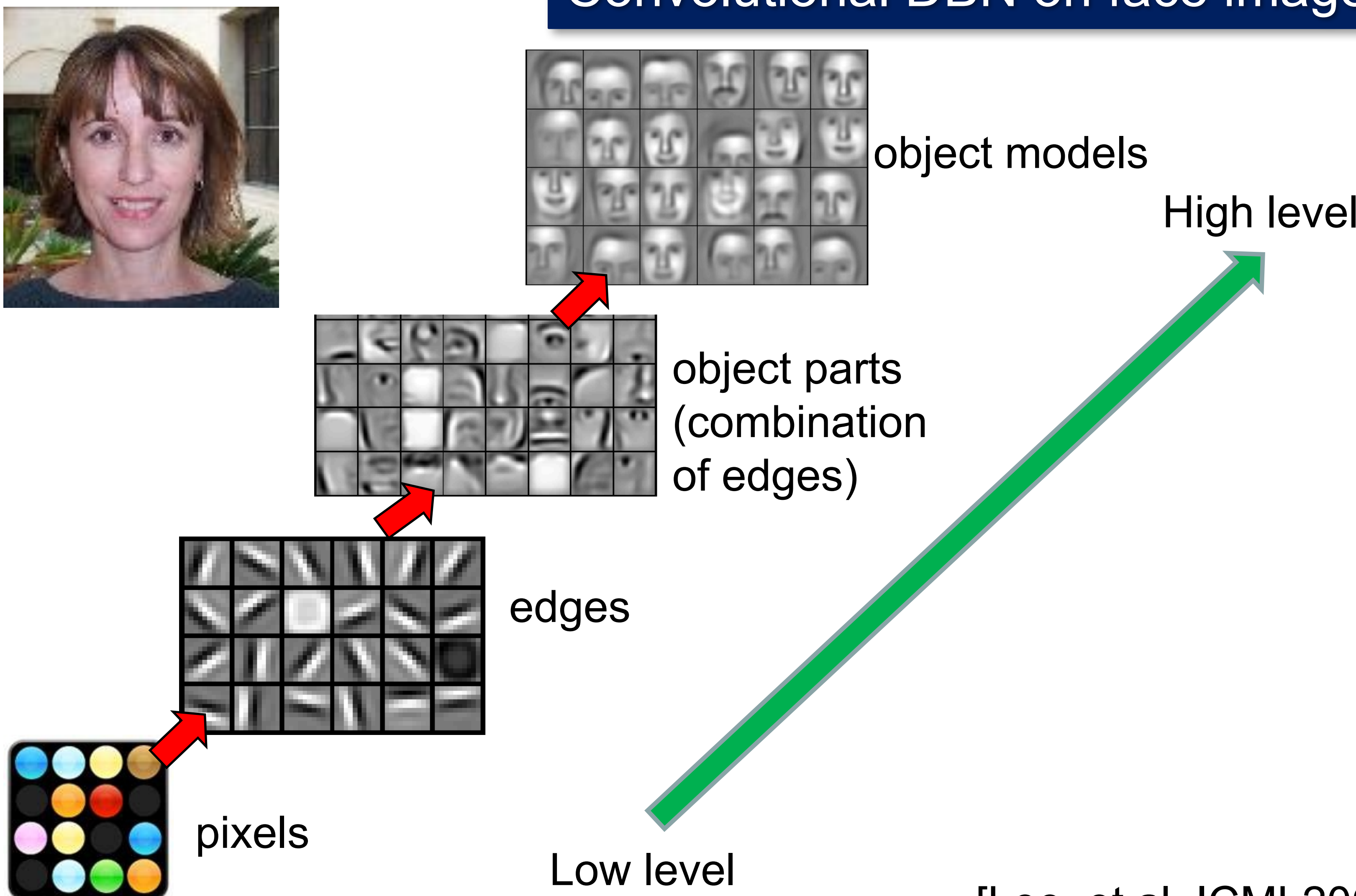


Able to learn #parts at each level

[Long Zhu, Alan L. Yuille, NIPS2005]

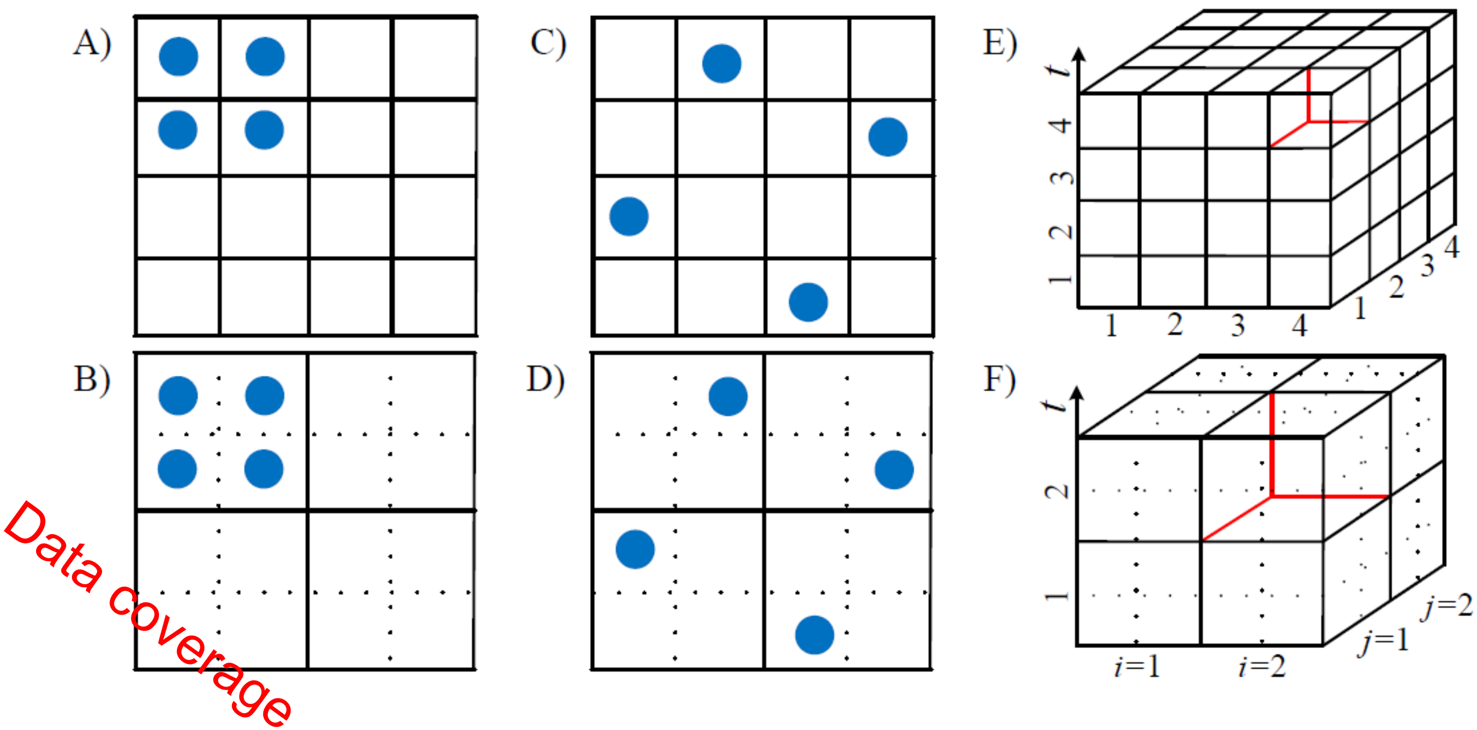
Deep Learning: An Implementation of GrC

Convolutional DBN on face images



[Lee, et al, ICML2009]

GrC in Urban Sensing



A fine-grained partition

A coarse-grained partition

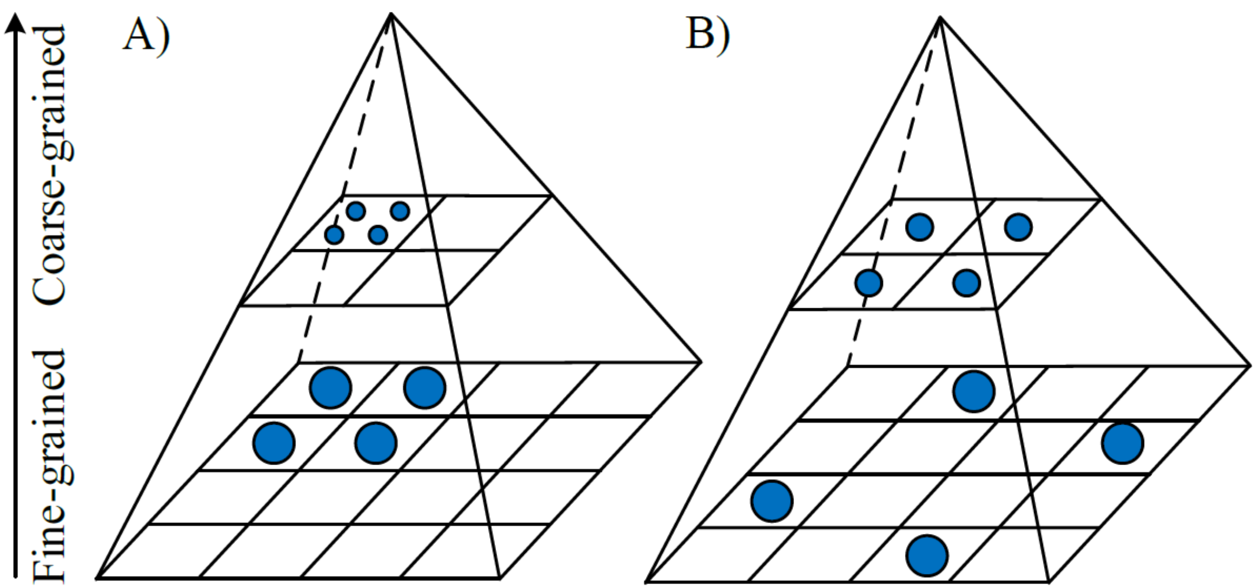
Different granularities of partitions and data distributions

Shenggong Ji, Yu Zheng, Tianrui Li, ***Urban Sensing Based on Human Mobility***. UbiComp 2016

GrC in Urban Sensing

Hierarchical Entropy

$$E(\mathcal{A}) = \sum_{k=1}^{k_{max}} \omega(k) E(\mathcal{A}(k)) / k_{max} \qquad E(\mathcal{A}(k)) = -\sum_{i,j,t} p(i,j,t|k) \log_2(p(i,j,t|k))$$



Data balance

Data balance considering many granularities of partitions

Shenggong Ji, Yu Zheng, Tianrui Li, ***Urban Sensing Based on Human Mobility***. UbiComp 2016

	Attributes		Decision
	Headache	Temperature	Flu
e1	yes	normal	no
e2	yes	high	yes
e3	yes	very_high	yes
e4	no	normal	no
e5	no	high	no
e6	no	very_high	yes
e7	no	high	yes
e8	no	very_high	no

Decision System

Universe: $U=\{x_1, x_2, \ldots, x_n\}$

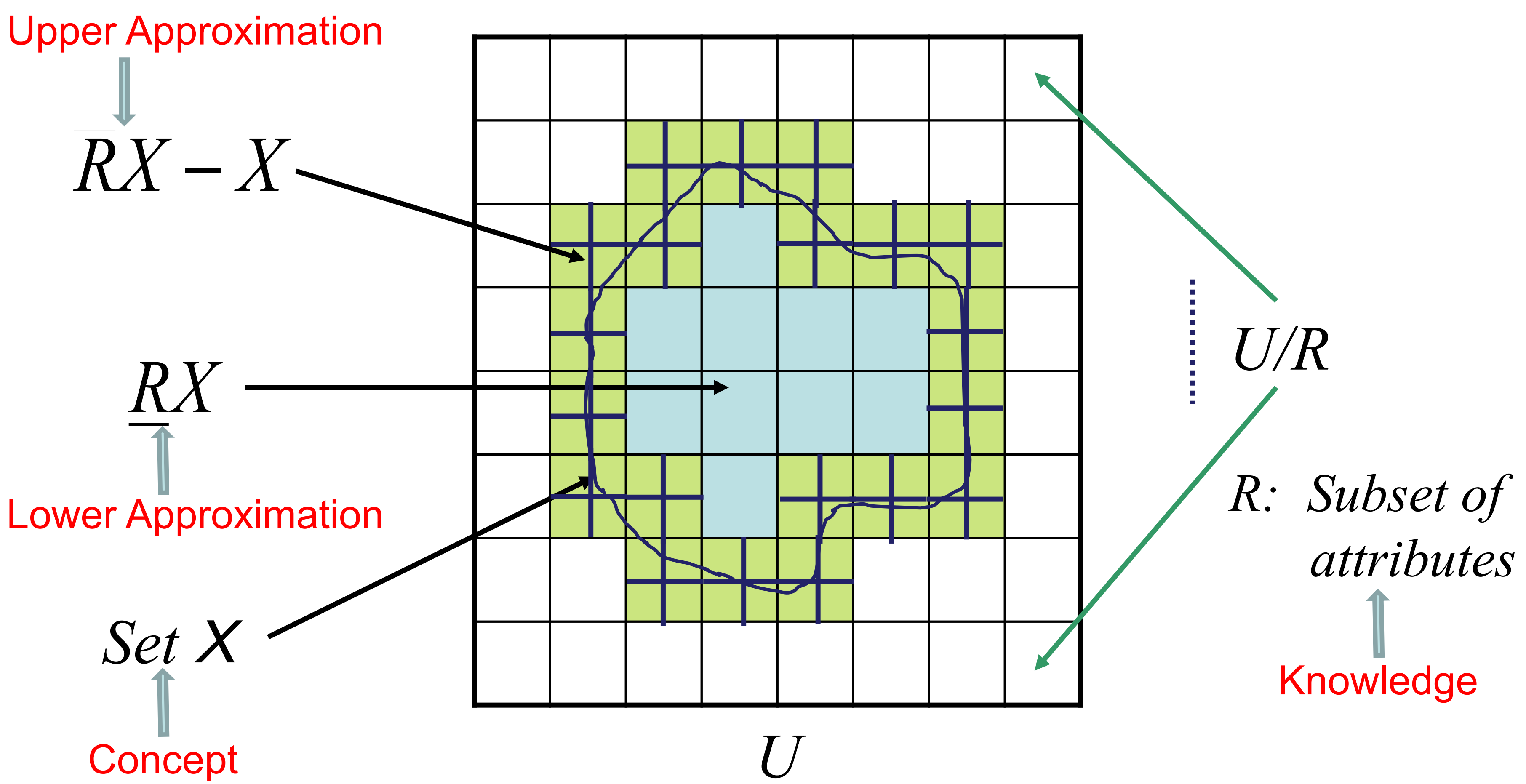
Attributes: $C=\{a_1, a_2, \ldots, a_m\}$

Decision: $U/D=\{d_1, d_2, \ldots, d_k\}$

Information function: $f(x, a)$

A decision system is composed of the universe, attribute sets, decision and information function.

Rough Set Theory (RST)



Calculation of Approximation for Big Data Analysis

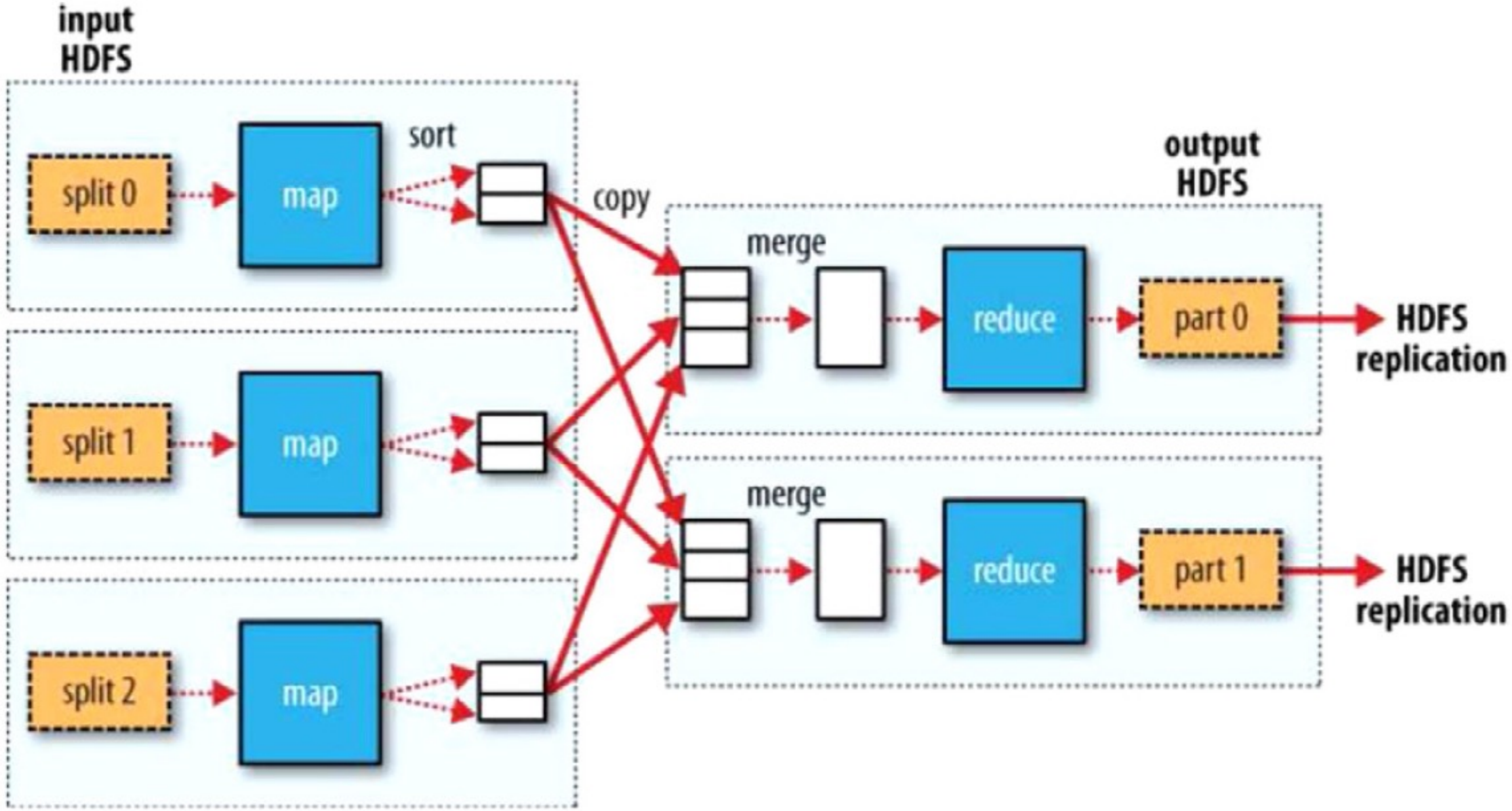
Calculation of Approximation

- A key step in feature selection/attribute reduction in big data
- A fundamental part in rough set-based data analysis
  - Similar to frequent pattern mining in association rules

Our contributions

- A parallel method to compute rough set approximations for big data
- A parallel matrix-based method for computing approximations in incomplete information systems
- A comparison of parallel large-scale knowledge acquisition using rough set theory on different MapReduce runtime systems

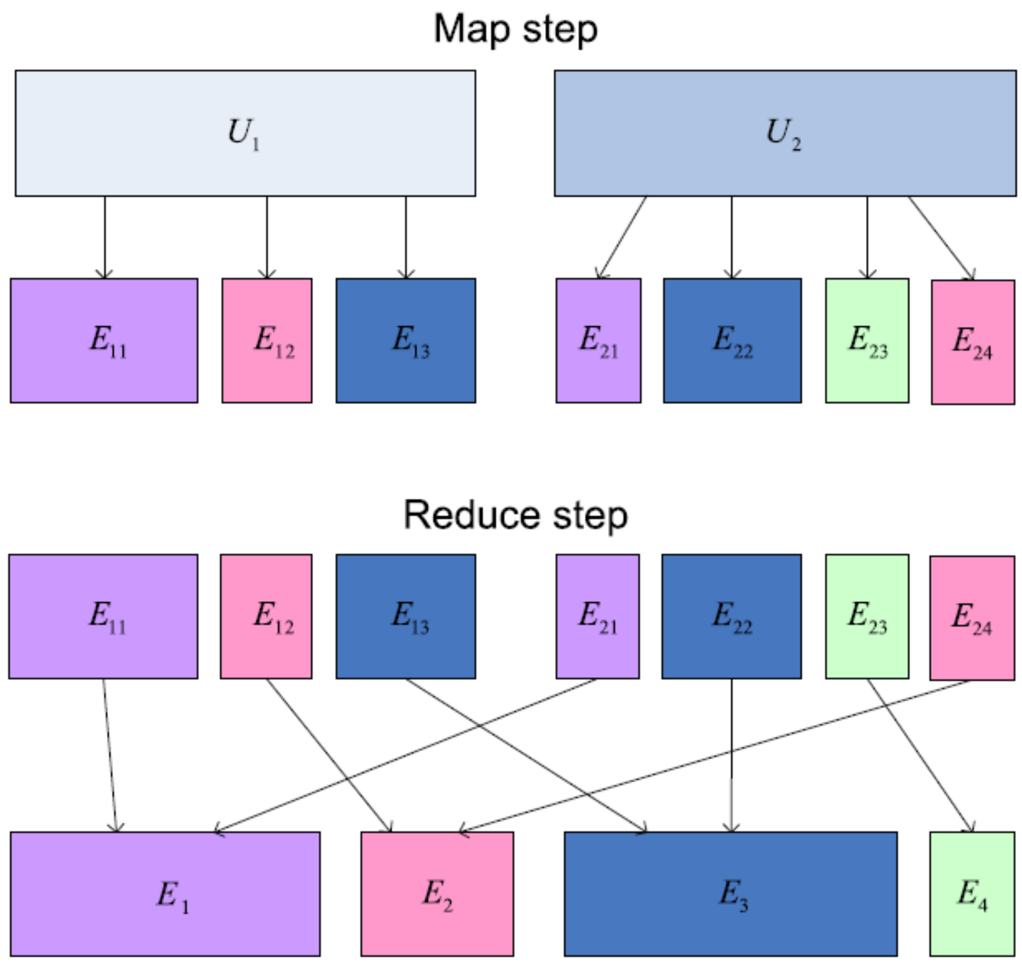
MapReduce



MapReduce: A programming model for processing big data.

A parallel method to compute approximations for big data

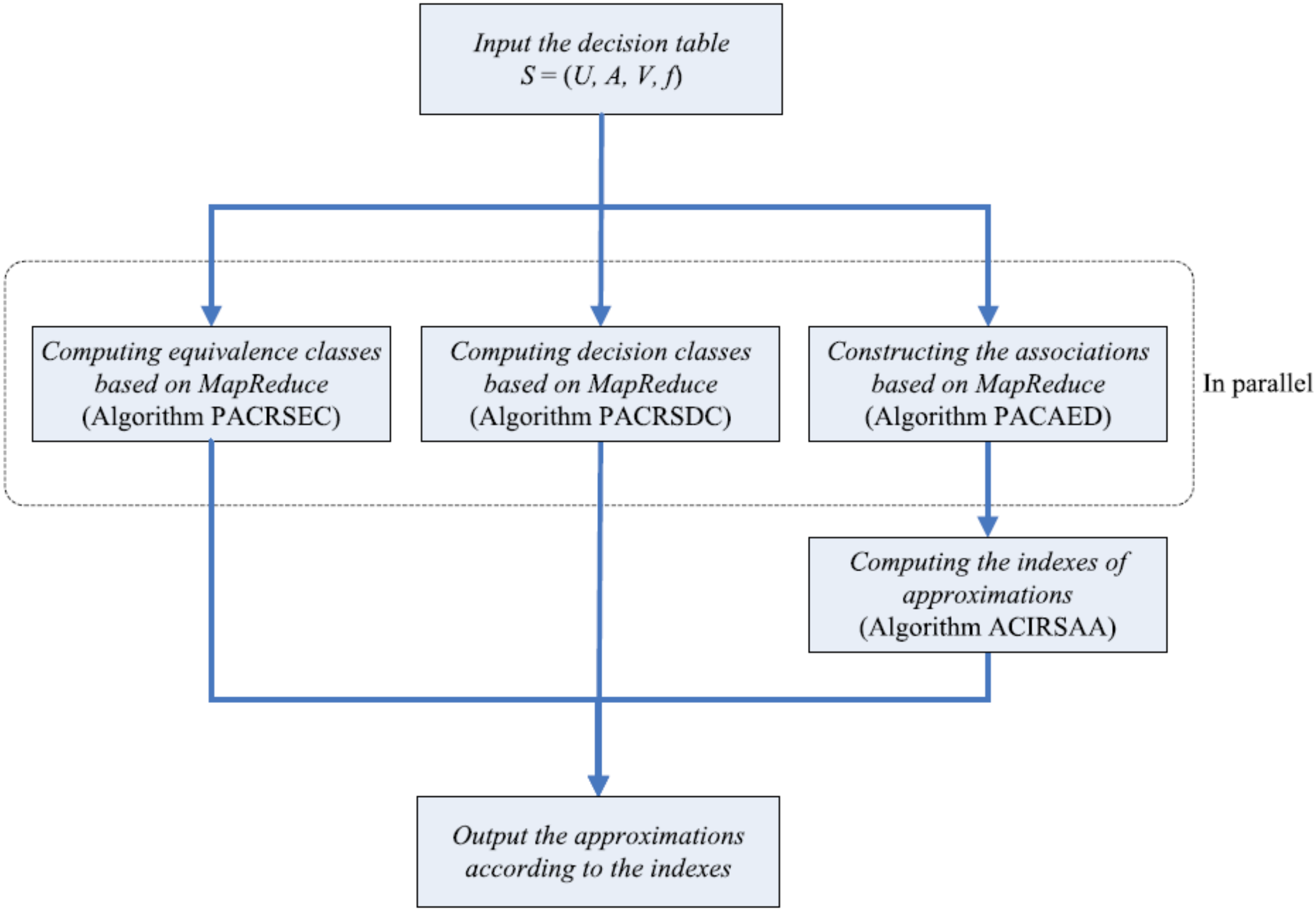
Computing rough set equivalence classes based on MapReduce



Junbo Zhang, Tianrui Li, Da Ruan, et al. ***A parallel method for computing rough set approximations***. Information Sciences, 2012

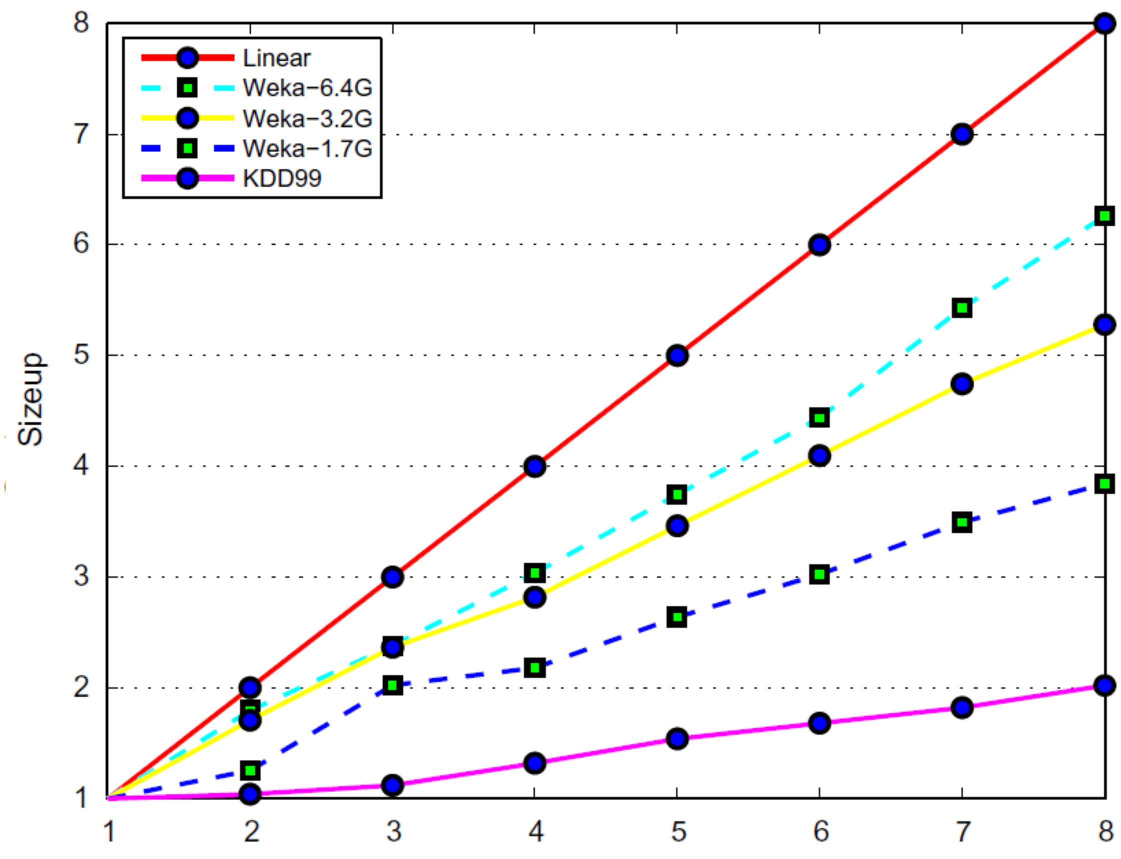
A parallel method to compute approximations for big data

Computing rough set approximations based on MapReduce



Junbo Zhang, Tianrui Li, Da Ruan, et al. ***A parallel method for computing rough set approximations***. Information Sciences, 2012

A parallel method to compute approximations for big data

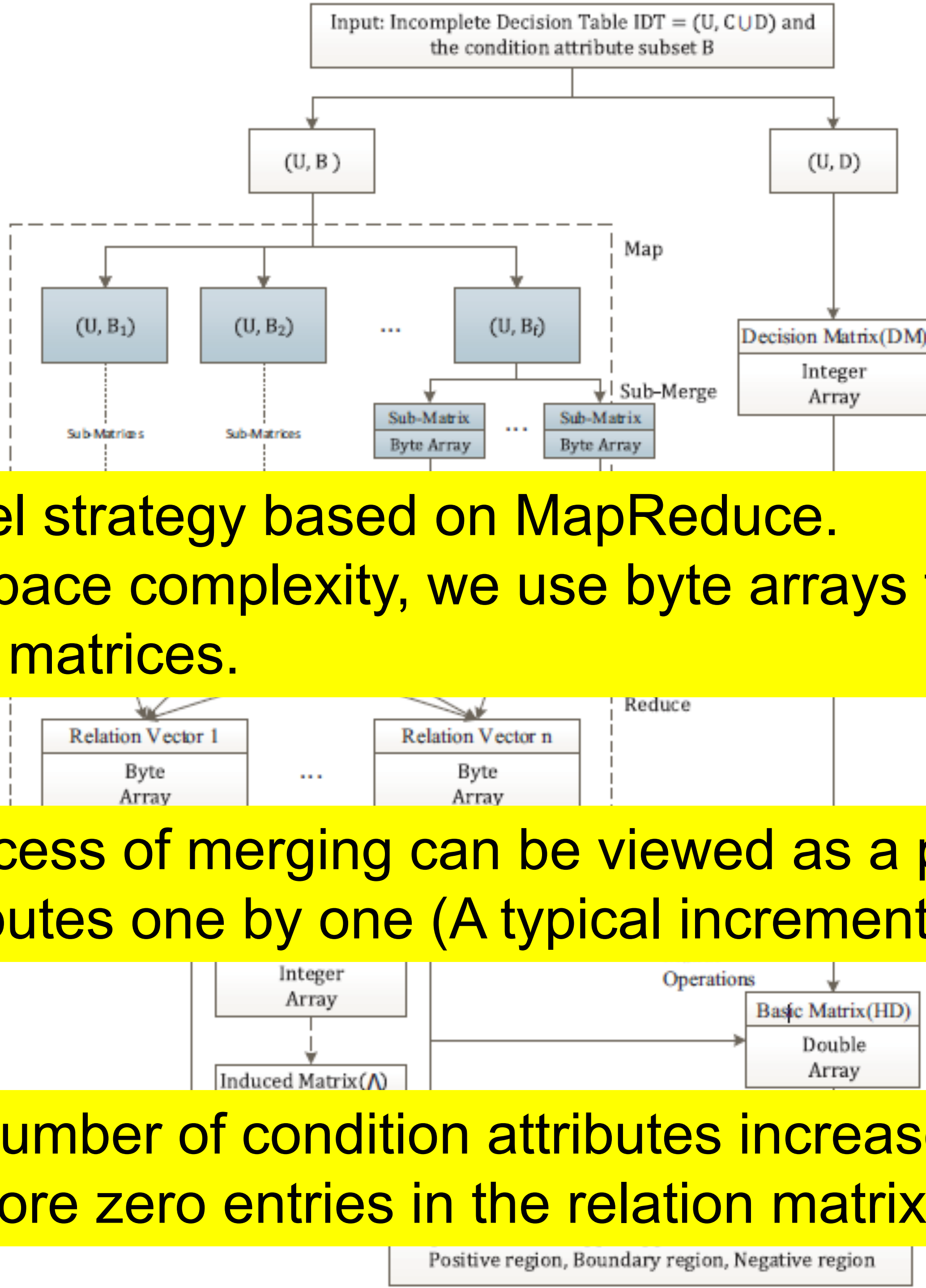


Sizeup measures how much longer it takes, when the size of data set is p-times larger than that of the original data set.

Junbo Zhang, Tianrui Li, Da Ruan, et al. ***A parallel method for computing rough set approximations***. Information Sciences, 2012

- A parallel matrix-based method for computing approximations in incomplete information systems (IIS)
  - S1: A MapReduce-based parallel method to construct the relation matrix is designed for fast computing approximations
    - A Sub-Merge operation is used
  - S2: An incremental method is applied to the process of merging the relation matrices.
    - The relation matrix is updated in parallel and incrementally to efficiently accelerate the computational process.
  - S3: A sparse matrix method is employed to optimize the proposed matrix-based method
    - To further improve the performance of the algorithm.

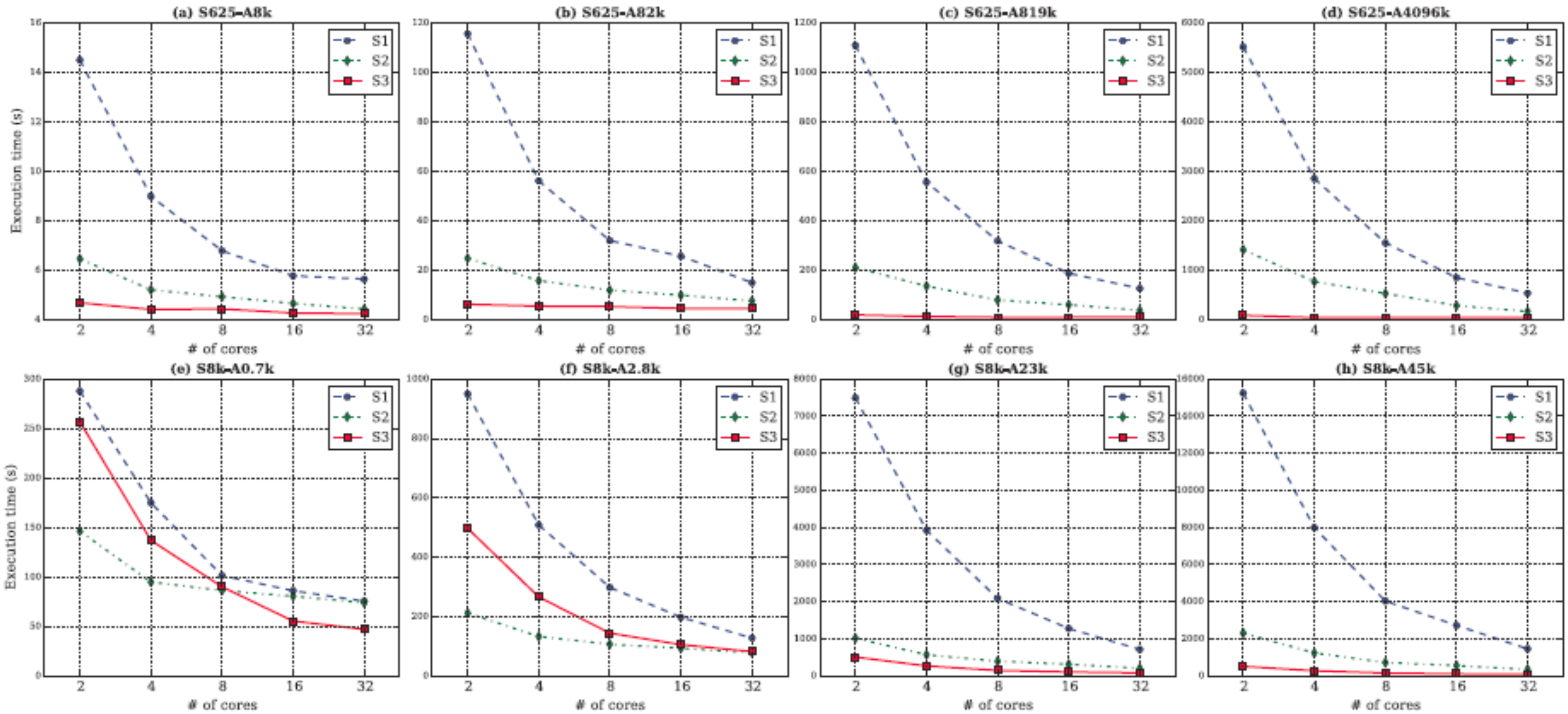
Junbo Zhang, Jian-Syuan Wong, Yi Pan, Tianrui Li, ***A parallel matrix-based method for computing approximations in incomplete information systems***. IEEE Transaction on Knowledge and Data Engineering, 2015



S1: A parallel strategy based on MapReduce.
To reduce space complexity, we use byte arrays to storage the sub-relation matrices.

S2: The process of merging can be viewed as a process of adding attributes one by one (A typical incremental process).

S3: As the number of condition attributes increases, there are more and more zero entries in the relation matrix.



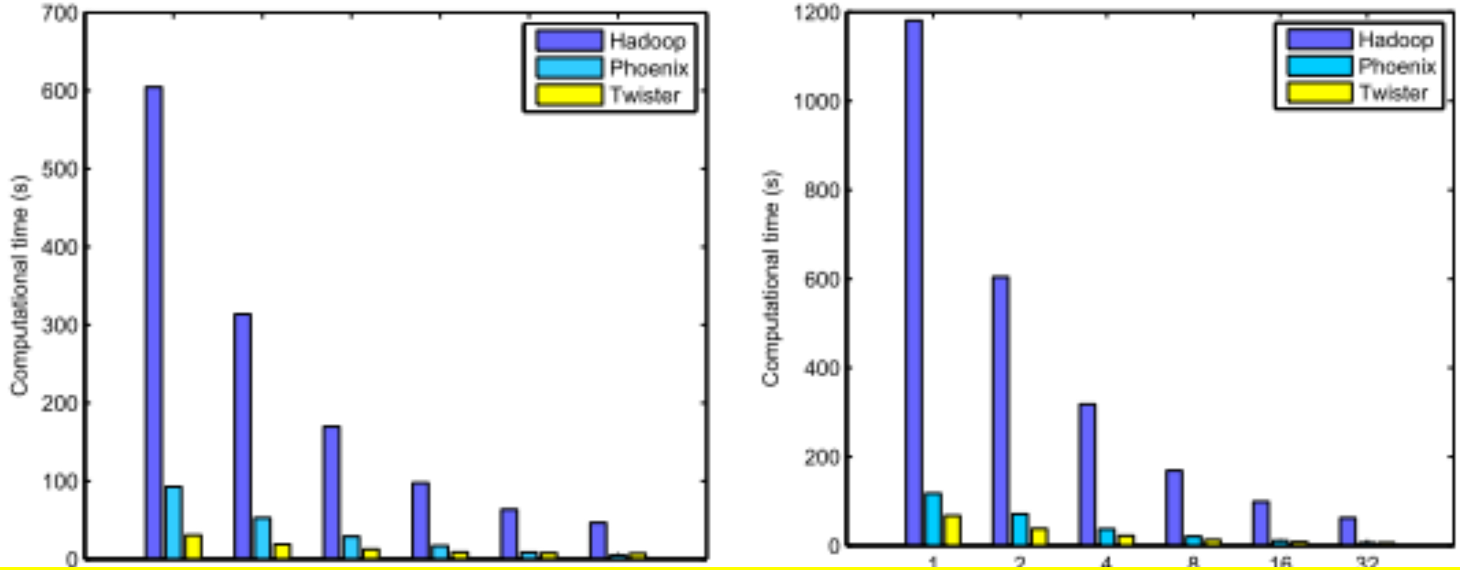
S2 and S3 always have better performance than S1, and, in most cases, S3 outperforms S2.

Junbo Zhang, Jian-Syuan Wong, Yi Pan, Tianrui Li, ***A parallel matrix-based method for computing approximations in incomplete information systems***. IEEE Transaction on Knowledge and Data Engineering, 2015

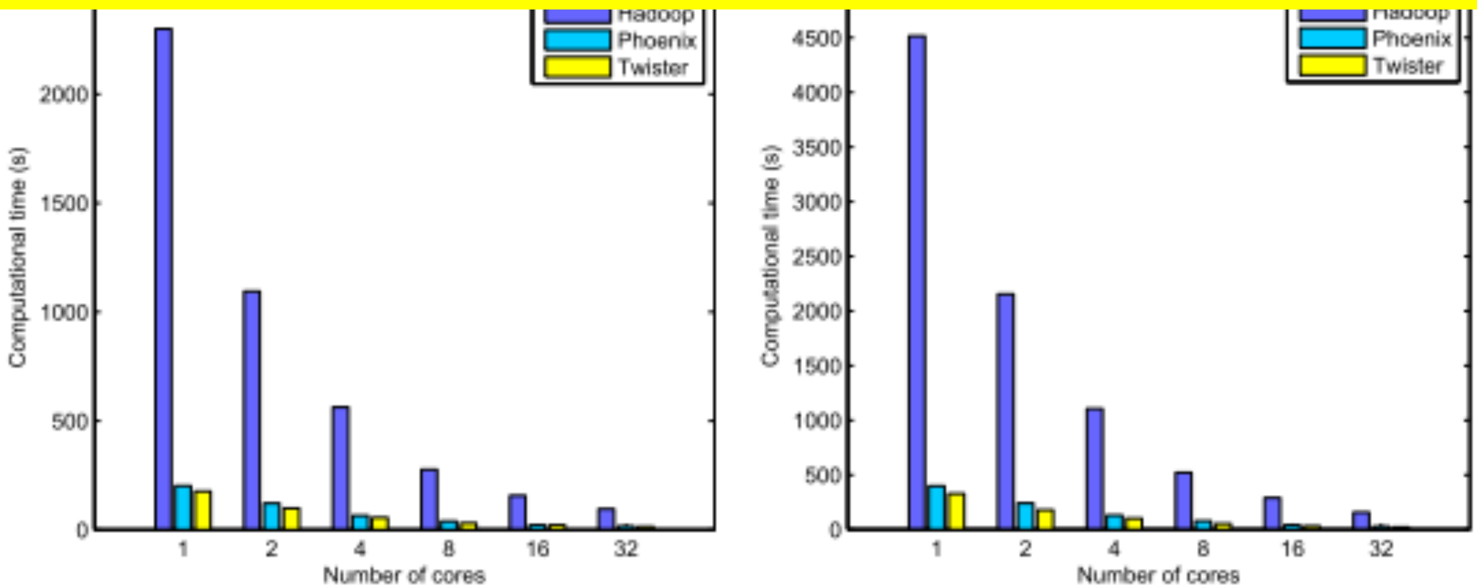
- A Comparison of Parallel Large-scale Knowledge Acquisition Using Rough Set Theory on Different MapReduce Runtime Systems
  - We present parallel large-scale rough set based methods for knowledge acquisition using MapReduce.
  - Experimental results on Hadoop, Phoenix and Twister
    - Computational time is mostly minimum in Twister while employing same cores;
    - Hadoop has the most excellent speedup in the larger data set;
    - Phoenix has the most excellent speedup in the smaller data set.

The Phoenix System for MapReduce Programming

Junbo Zhang, Jian-Syuan Wong, Tianrui Li, Yi Pan, ***A Comparison of Parallel Large-scale Knowledge Acquisition Using Rough Set Theory on Different MapReduce Runtime Systems***. International Journal of Approximate Reasoning, 2014

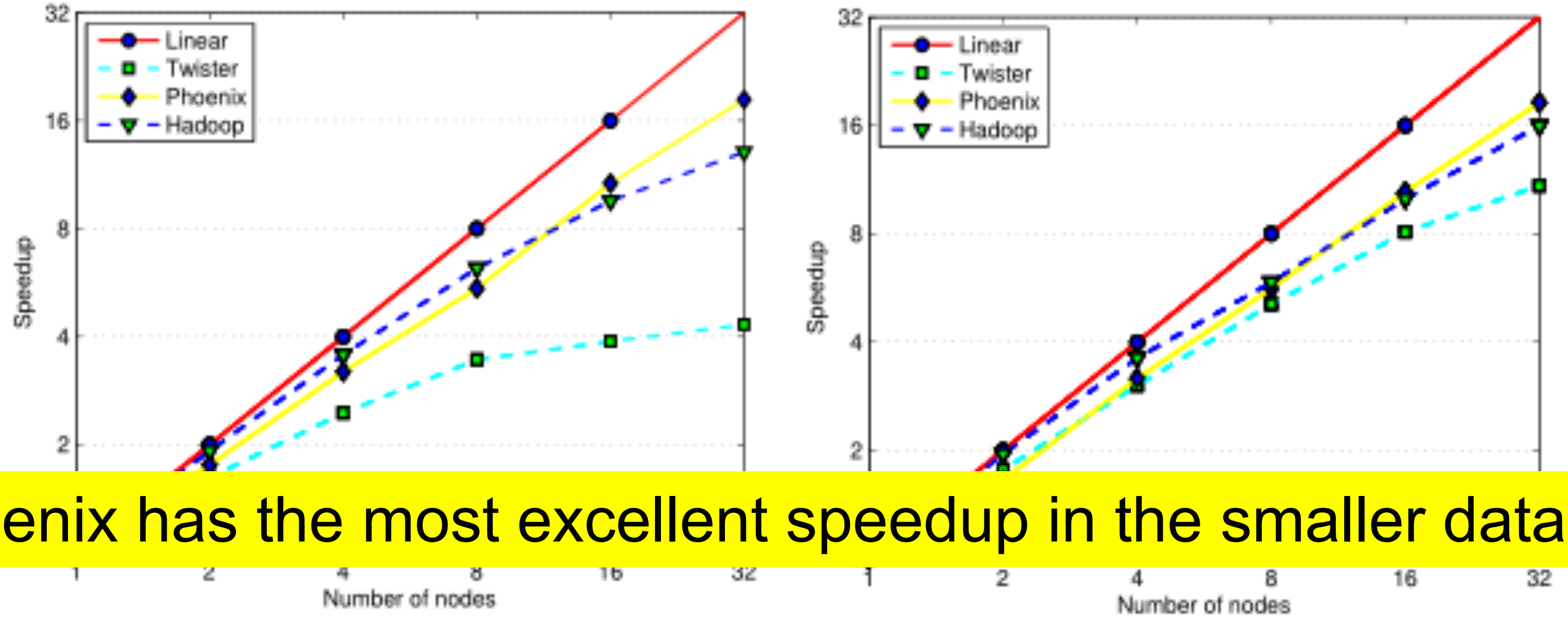


Computational time is minimum in Twister while employing same cores in most cases.



(c) Weka-3.2G

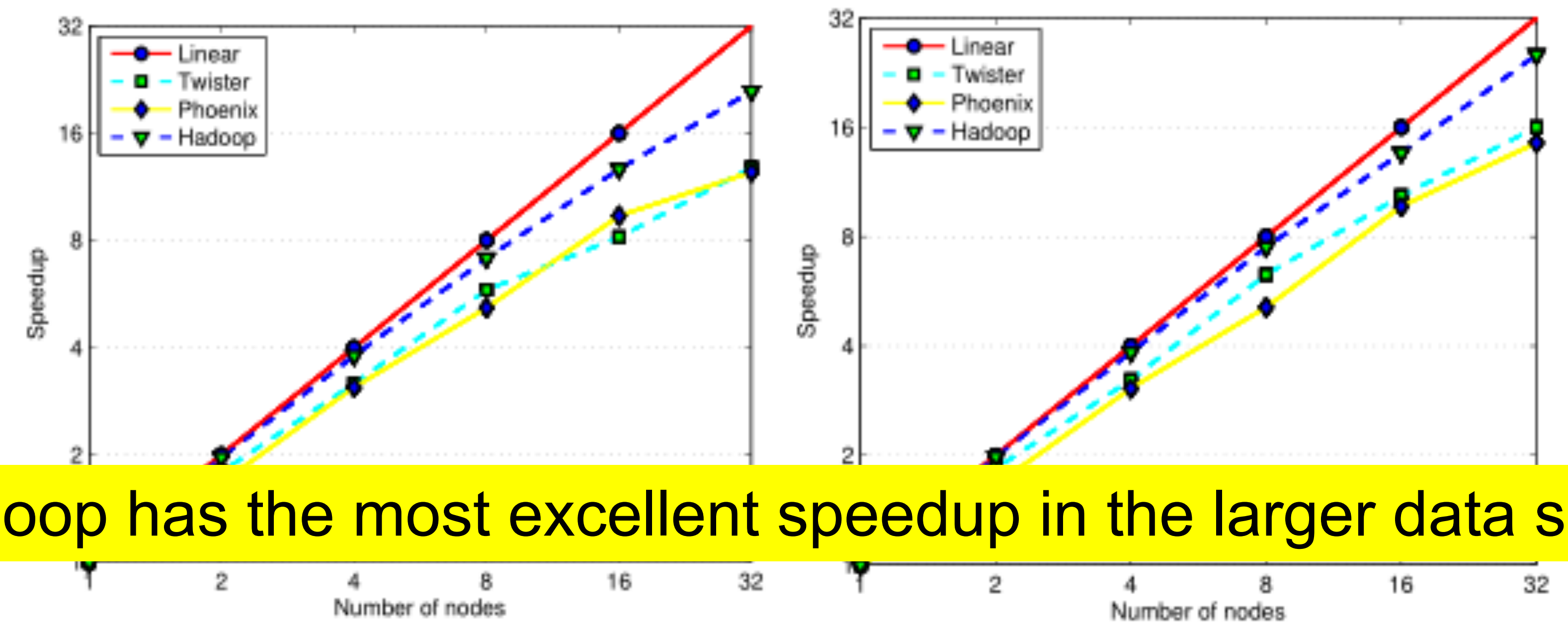
(d) Weka-6.4G



Phoenix has the most excellent speedup in the smaller data set.

(a) KDD99

(b) Weka-1.8G



Hadoop has the most excellent speedup in the larger data set.

(c) Weka-3.2G

(d) Weka-6.4G

Data and Model Parallization Based on RST and GrC for Big Data

Our contributions

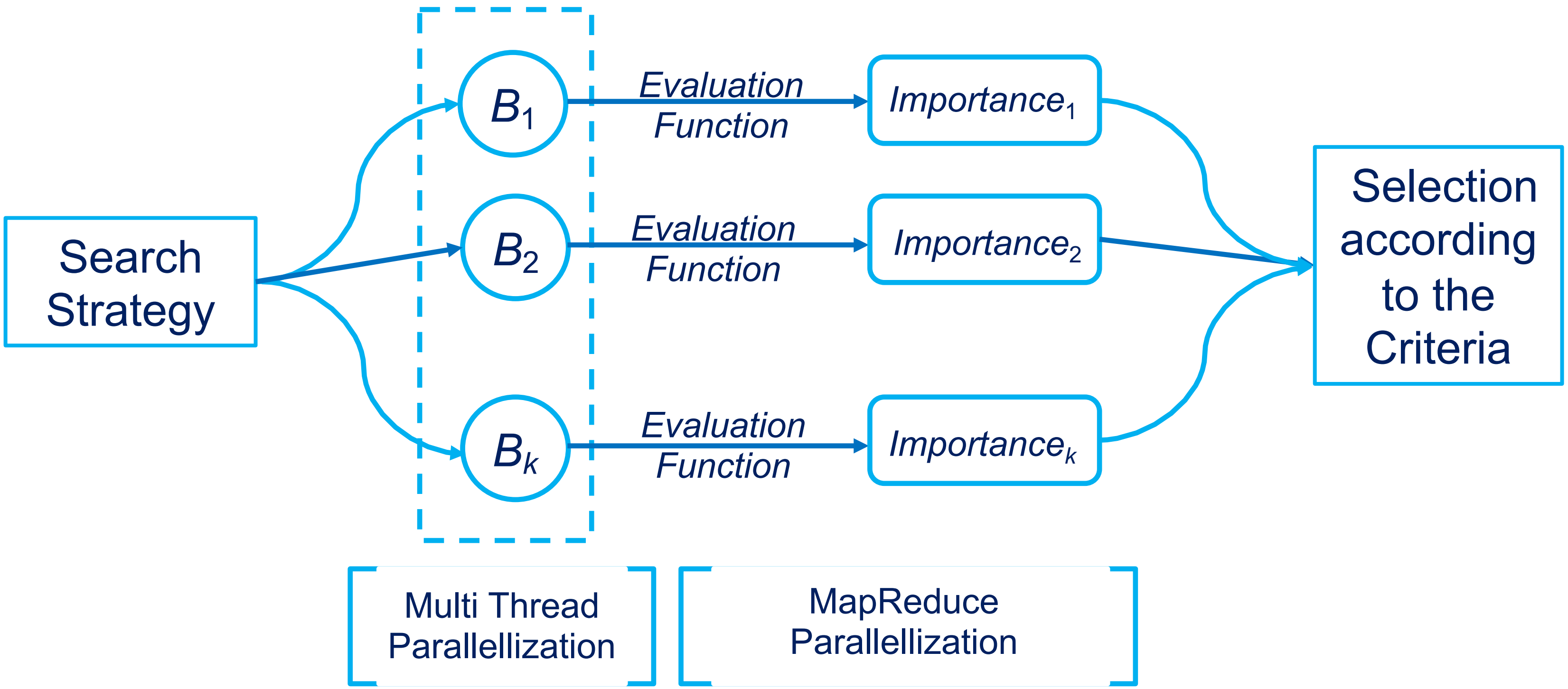
- A unified parallel large-scale framework for computing reduct (feature selection) is presented.
- Its corresponding three parallel methods are proposed, *e.g.*, model parallelism (MP), data parallelism (DP), and model-data parallelism (MDP).
- A unified representation of feature evaluation functions is presented.
- The divide-and-conquer methods for 4 representative evaluation functions are shown.

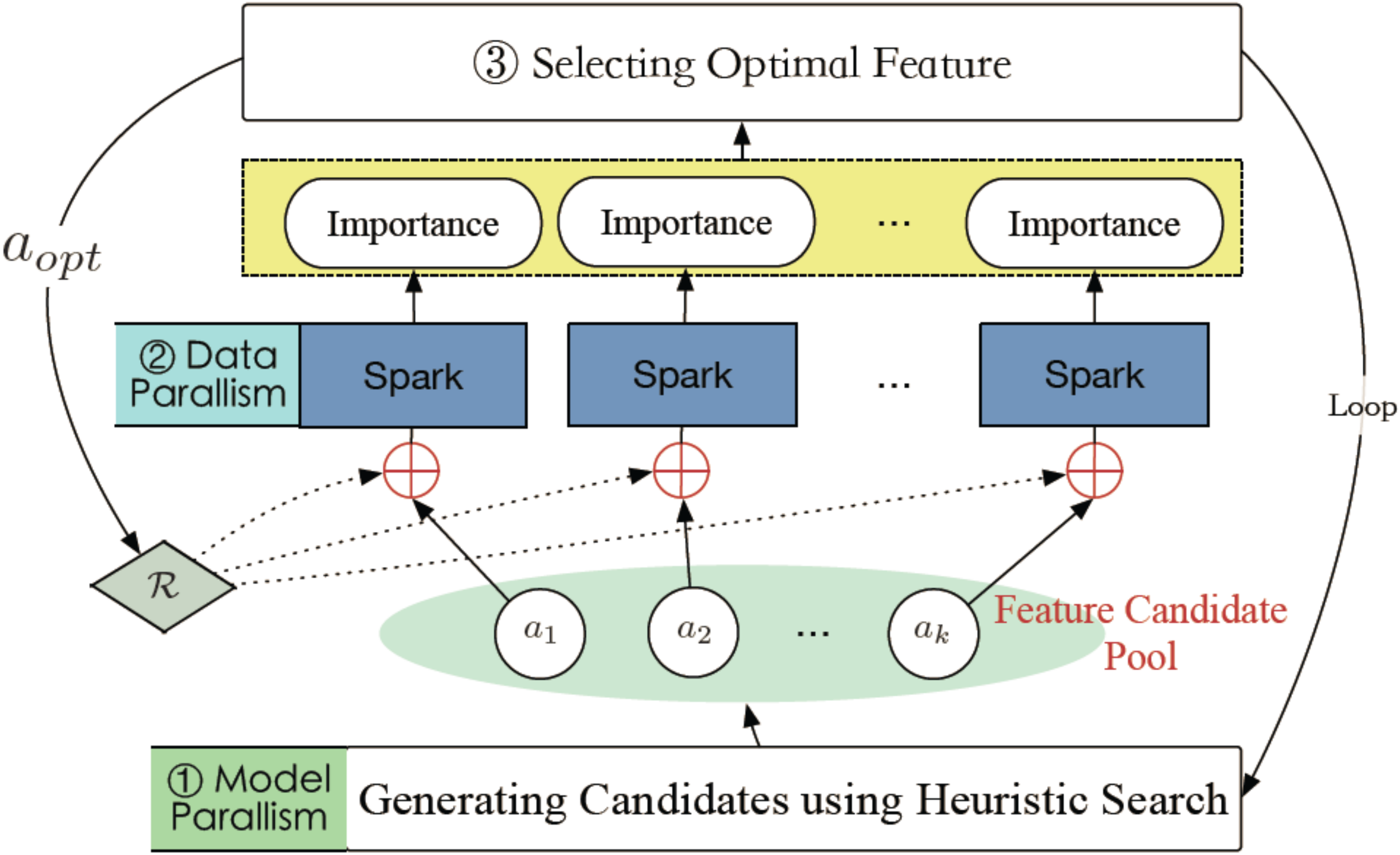
Our contributions

- MapReduce-based and Spark-based Parallel Large-scale Attribute Reduction (PLAR) algorithms are designed.
- GrC theory is introduced for accelerating the process of attribute reduction.
- By combining with MDP, Algorithm PLAR-MDP is presented.

Parallellization Strategy

- Data and Model Parallization





A Parallel Framework for Attribute Reduction

Divide-and-conquer methods

Reduct by PR

Importance of a $\quad \gamma_{B\cup\{a\}}(D) - \gamma_B(D)$

$$\gamma_B(D) = \frac{|POS_B(D)|}{|U|}$$

$$\begin{aligned}
POS_B(D) = \bigcup_{j=1}^{m} \underline{R}(D_j) &= \bigcup_{j=1}^{m}\left(\bigcup_{i=1}^{e}\{E_i \in U/B : E_i \subseteq D_j\}\right) \\
&= \bigcup_{i=1}^{e}\left(\bigcup_{j=1}^{m}\{E_i \in U/B : E_i \subseteq D_j\}\right) \\
&= \bigcup_{i=1}^{e}\{E_i \in U/B : E_i \subseteq D_1 \vee E_i \subseteq D_2 \vee \cdots \vee E_i \subseteq D_m\} \\
&= \bigcup_{i=1}^{e}\{E_i \in U/B : |E_i/D| = 1\}.
\end{aligned}$$

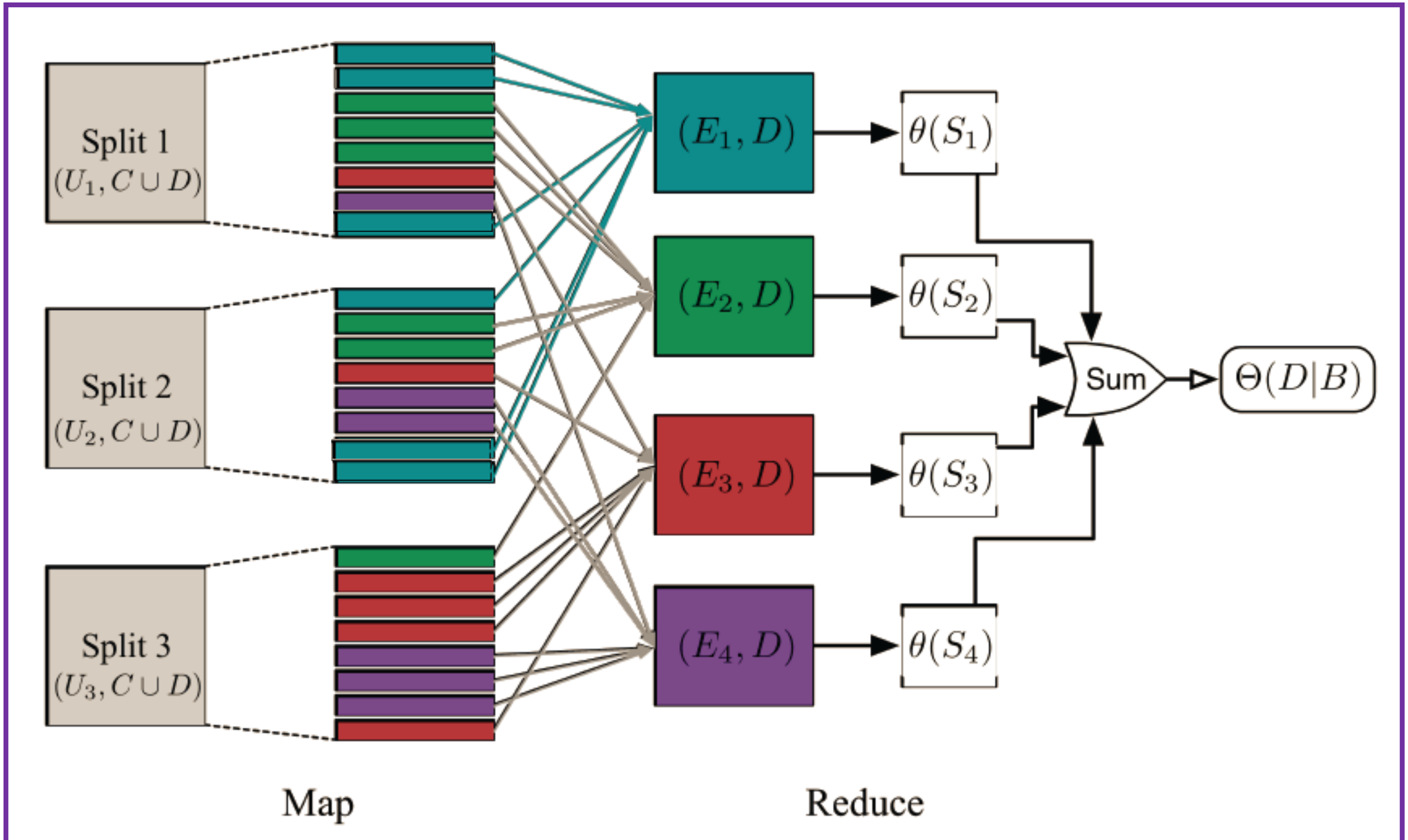
Divide-and-conquer methods

Method	$\Theta(D\|B)$
PR	$\gamma(D\|B) := -\gamma_B(D) = -\frac{\|POS_B(D)\|}{\|U\|}$
SCE	$\mathcal{H}(D\|B) = -\sum_{i=1}^{e} p(E_i) \sum_{j=1}^{m} p(D_j\|E_i) \log(p(D_j\|E_i))$
LCE	$\mathcal{H}_L(D\|B) = \sum_{i=1}^{e}\sum_{j=1}^{m} \frac{\|D_j \cap E_i\|}{\|U\|}\frac{\|D_j^c - E_i^c\|}{\|U\|}$
CCE	$\mathcal{H}_Q(D\|B) = \sum_{i=1}^{e} \left(\frac{\|E_i\|}{\|U\|}\frac{C^2_{\|E_i\|}}{C^2_{\|U\|}} - \sum_{j=1}^{m} \frac{\|E_i \cap D_j}{\|U\|} \right.$

A unified representation

$$\Theta(D|B) = \sum_{i=1}^{e} \theta(S_i)$$

Method	$\theta(S_i)$
PR	$-\frac{\|E_i\| sgn_{PR}(E_i)}{\|U\|}$
SCE	$-\frac{1}{\|U\|}\sum_{j=1}^{m} \|D_{ij}\| \log \frac{\|D_{ij}\|}{\|E_i\|}$
LCE	$\sum_{j=1}^{m} \frac{\|D_{ij}\|(\|E_i\| - \|D_{ij}\|)}{\|U\|^2}$
CCE	$\frac{\|E_i\|^2 \times (\|E_i\|-1)}{\|U\|C^2_{\|U\|}} - \sum_{j=1}^{m} \frac{\|D_{ij}\|^2 \times (\|D_{ij}\|-1)}{\|U\|C^2_{\|U\|}}$



Parallel Large-scale Attribute Reduction based on Hadoop, Spark and MDP model

PLAR-Spark: Parallel Large-scale Attribute Reduction based on Spark

$data \longleftarrow$ spark.textFile("input").map(parseVector()).cache();

$Cands \longleftarrow \{C\} \cup \{C - \{a\} | a \in C\}$;

foreach $B \in Cands$ **do**

  $\Theta(D|B) =$data.Map$\boxed{((\overrightarrow{x}_B, \overrightarrow{x}_D))}$.ReduceByKey($\theta(S_i)$).Sum();

foreach $a \in C$ **do**

  $Sig^{inner}(a, C, D) \longleftarrow \Theta(D|C - \{a\}) - \Theta(D|C)$;

$Core = \{a | Sig^{inner}(a, C, D) > \epsilon, a \in C\}$;

$Reduct \longleftarrow Core$;

while *stopping criterion not met* & $C - Reduct \neq \emptyset$ **do**

  foreach $a \in C - Reduct$ **do**

    $\Theta(D|Reduct \cup \{a\}) =$data.Map$\boxed{((\overrightarrow{x}_{Reduct \cup \{a\}}, \overrightarrow{x}_D))}$.ReduceByKey($\theta(S_i)$).Sum();

  $a_{opt} = \underset{a \in C - Reduct}{\arg\min} \{\Theta(D|Reduct \cup \{a\})\}$;

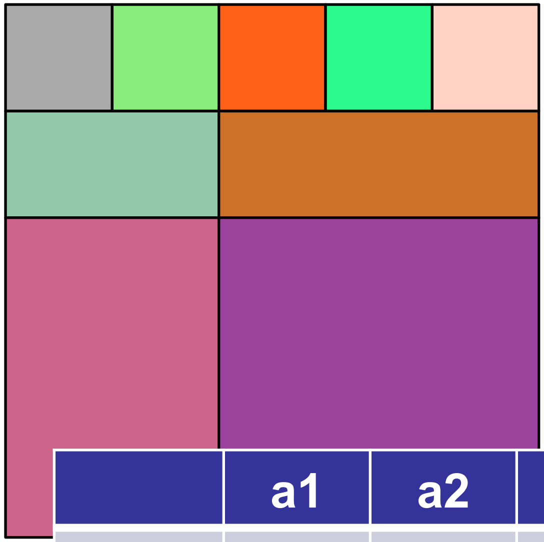
  $Reduct \longleftarrow Reduct \cup \{a_{opt}\}$;

return $Reduct$

Speed up by Granulation

	a1	a2	D
x1	0	0	1
x2	1	1	0
x3	1	1	0
x4	1	0	1
x5	1	0	1
x6	0	0	1
x7	1	0	1
x8	0	0	1
x9	1	1	1

Granulation



Granulation

	a1	a2	D	Num
G1	0	0	1	3
G2	1	1	0	2
G3	1	0	1	3
G4	1	1	1	1

PLAR-MDP: Parallel Large-scale Attribute Reduction based on MDP model

$\boxed{data := G^{(A)}} \longleftarrow$ spark.textFile($input$).map(parseVector()).reduceByKey(add).cache();

$Cands \longleftarrow \{C\} \cup \{C - \{a\} | a \in C\}$;

for $B \in Cands$ **do in parallel** // Model Parallelism

  $\Theta(D|B) = data.\mathrm{Map}\boxed{((\vec{E}_B, (\vec{E}_D, |E_{B \cup D}|)))}\mathrm{ReduceByKey}(\theta(S_i)).\mathrm{Sum}()$; // Data Parallelism

$\boxed{G^{(A)} = \{(\vec{E}_A, |E_A|) : E_A \in U/A\}}$

foreach $a \in C$ **do**

  $Sig^{inner}(a, C, D) \longleftarrow \Theta(D|C - \{a\}) - \Theta(D|C)$;

$Core = \{a | Sig^{inner}(a, C, D) > \epsilon, a \in C\}$;

$Reduct \longleftarrow Core$;

while *stopping criterion not met* & $C - Reduct \neq \emptyset$ **do**

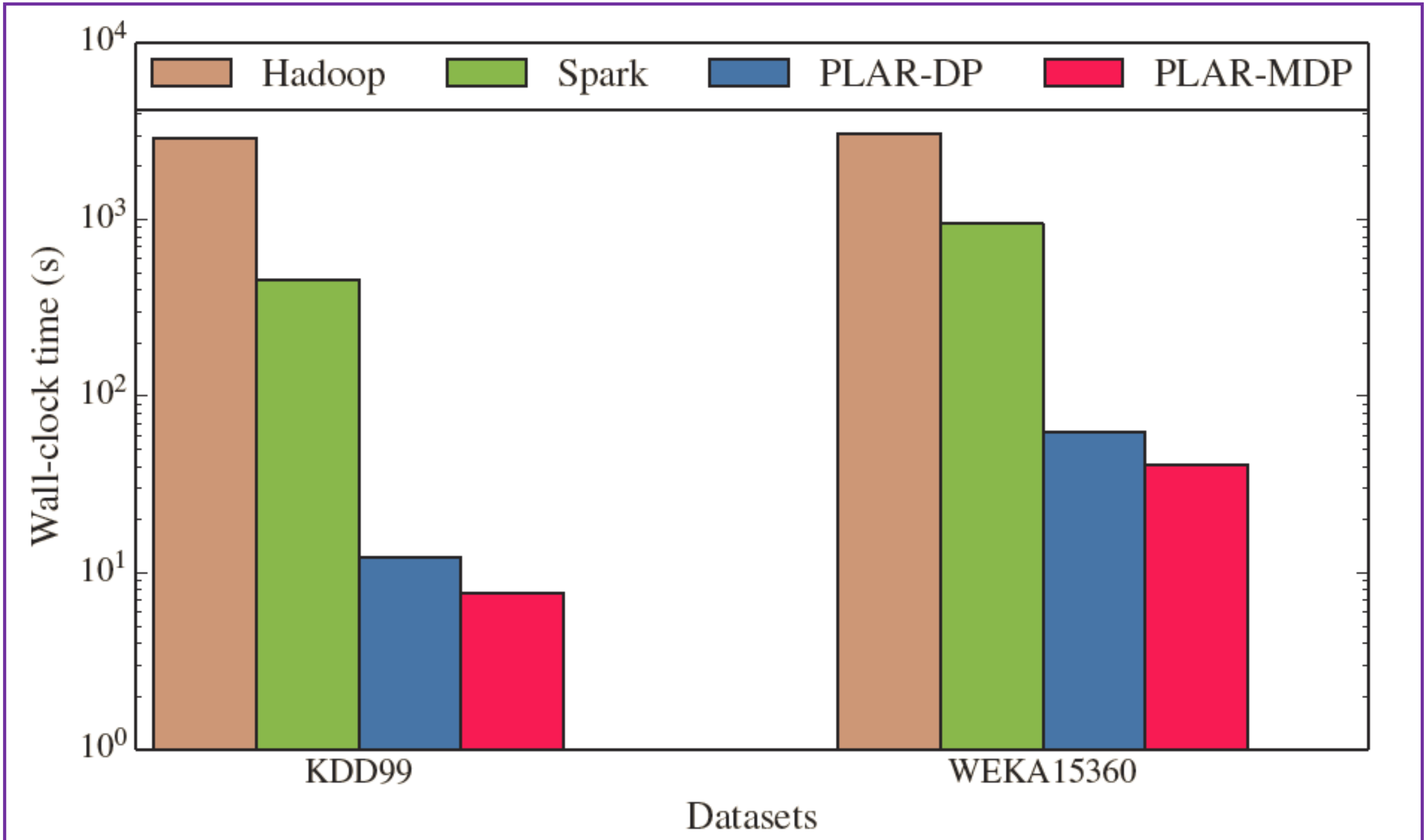
  for $a \in C - Reduct$ **do in parallel** // Model Parallelism

    $\Theta(D|Reduct \cup \{a\}) = data.\mathrm{Map}\boxed{((\vec{E}_{Reduct \cup \{a\}}, (\vec{E}_D, |E_{Reduct \cup \{a\} \cup D}|)))}.\mathrm{ReduceByKey}(\theta(S_i)).\mathrm{Sum}()$;

  $a_{opt} = \arg\min_{a \in C - Reduct} \{\Theta(D|Reduct \cup \{a\})\}$;

  $Reduct \longleftarrow Reduct \cup \{a_{opt}\}$;

return $Reduct$



Parallel Large-scale Attribute Reduction based on Hadoop, Spark and MDP model

Experimental Results

High-Dimension Data

Dataset	Gisette
NoO	6000
NoF	5000

N. of Iteration	PLAR-DP	PLAR-MDP：Degree of Model Parallelism					
		2	4	8	16	32	64
1	6262	3080	1570	885	472	350	371
2	5975	2982	1480	873	465	343	370
3	6261	3059	1497	869	470	344	370
4	6115	3017	1484	877	468	344	369
5	6194	3155	1512	885	465	348	375
Time	30806	15293	7543	4389	2340	1730	1856

Experimental Results

Astronomical big data

Dataset	SDSS
NoO	320000
NoF	5201

	128 Cores	32 Cores
PR	7432	24274
SCE	7312	24181
LCE	7207	24372
CCE	7383	24295

iLgC: Incremental Learning Based on Granular Computing for Evolving Data

The past

The present

The future

Our contributions

- Dynamic maintenance of approximations
  - Variation of the object set
    - New patients' records are added
  - Variation of the attribute set
    - New disease features become available
  - Variation of attribute values
    - The feature values may be revised

Example---Variation of the attribute set

- Dynamic maintenance of approximations in set-valued information systems

Basic vector

$$H(X) = \boxed{\Lambda^{T_B}_{n\times n}} \bullet (\boxed{M^{T_B}_{n\times n}} \bullet G(X))$$

The relation matrix

$$M^{T_P}_{n\times n} = (m_{ij})_{n\times n}$$

The induced diagonal matrix

$$\Lambda^{T_P}_{n\times n} = \mathrm{diag}\left(\frac{1}{\lambda_1}, \frac{1}{\lambda_2}, \ldots, \frac{1}{\lambda_n}\right)$$

Junbo Zhang, Tianrui Li, et al, ***Rough sets based matrix approaches with dynamic attribute variation in set-valued information systems***, Int J of Approximate Reasoning, 2012

Example---Variation of the attribute set

- Dynamic maintenance of approximations in set-valued information systems when adding an attribute set

$$M_{n\times n}^{T_{P\cup Q}} = (m_{ij}^{\uparrow})_{n\times n}$$

$$m_{ij}^{\uparrow} = \begin{cases} 0, & \boxed{m_{ij} = 0} \vee \boxed{(x_i, x_j) \notin T_Q} \\ 1, & m_{ij} = 1 \wedge (x_i, x_j) \in T_Q \end{cases}$$

$$\Lambda_{n\times n}^{T_{P\cup Q}} = \mathrm{diag}\left(\frac{1}{\lambda_1^{\uparrow}}, \frac{1}{\lambda_2^{\uparrow}}, \ldots, \frac{1}{\lambda_n^{\uparrow}}\right)$$

$$\lambda_i^{\uparrow} = \boxed{\lambda_i} - \boxed{\sum_{j=1}^{n} m_{ij} \oplus m_{ij}^{\uparrow}}$$

Junbo Zhang, Tianrui Li, et al, ***Rough sets based matrix approaches with dynamic attribute variation in set-valued information systems***, Int J of Approximate Reasoning, 2012

Example---Variation of the attribute set

- Dynamic maintenance of approximations in set-valued information systems when deleting an attribute set

$$M_{n\times n}^{T_{P-Q}} = (m_{ij}^{\downarrow})_{n\times n}$$

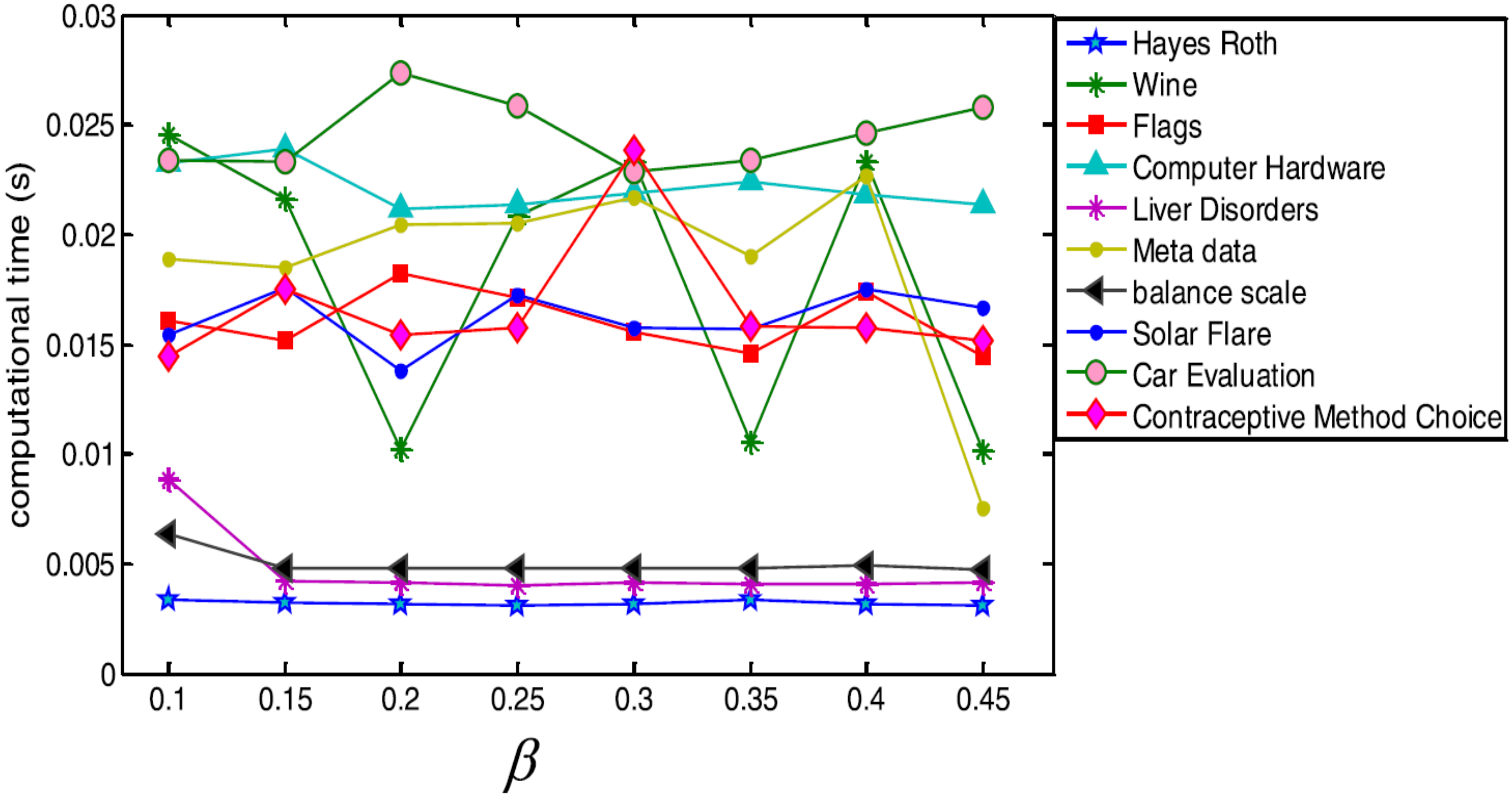
$$m_{ij}^{\downarrow} = \begin{cases} 0, & \boxed{m_{ij} = 0} \wedge \boxed{(x_i, x_j) \notin T_{P-Q}} \\ 1, & \boxed{m_{ij} = 1} \vee \boxed{(x_i, x_j) \in T_{P-Q}} \end{cases}$$

$$\Lambda_{n\times n}^{T_{P-Q}} = \mathrm{diag}\left(\frac{1}{\lambda_1^{\downarrow}}, \frac{1}{\lambda_2^{\downarrow}}, \ldots, \frac{1}{\lambda_n^{\downarrow}}\right)$$

$$\lambda_i^{\downarrow} = \boxed{\lambda_i} + \boxed{\sum_{j=1}^{n} m_{ij} \oplus m_{ij}^{\downarrow}}$$

Junbo Zhang, Tianrui Li, et al, ***Rough sets based matrix approaches with dynamic attribute variation in set-valued information systems***, Int J of Approximate Reasoning, 2012

Example---Variation of the object set

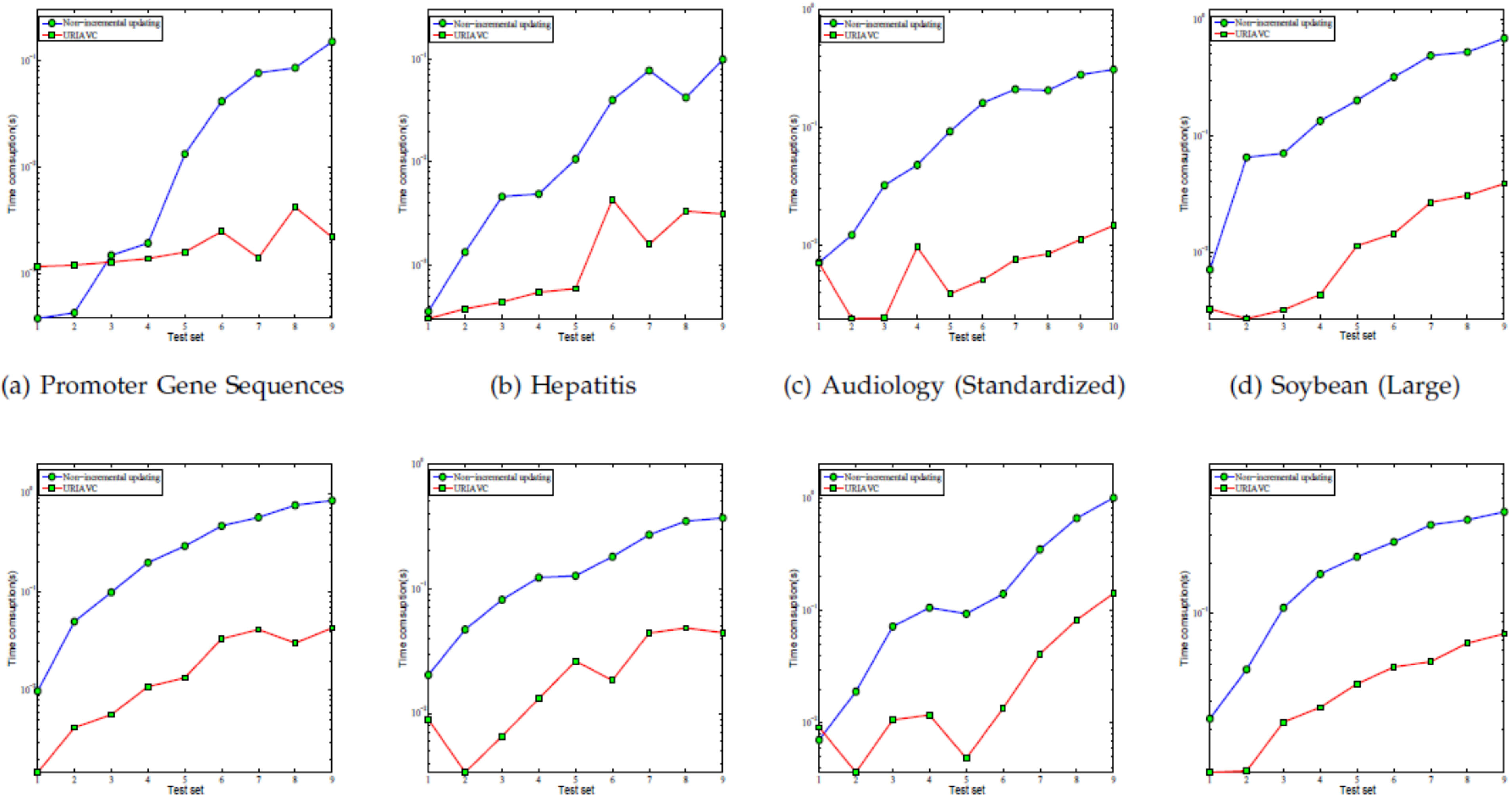


Hongmei Chen, Tianrui Li, et al, ***A Rough-Set Based Incremental Approach for Updating Approximations under Dynamic Maintenance Environments***, IEEE Transaction on Knowledge and Data Engineering, 2013

Example---Variation of the attributes' values

- A rough set-based method for updating decision rules on attribute values' coarsening and refining (AVCR)
  - The definition of minimal discernibility attribute set is presented.
  - Principles of updating decision rules in case of AVCR are discussed.
  - The rough set-based methods for updating decision rules in the inconsistent decision system are proposed.

Hongmei Chen, Tianrui Li, et al, ***A rough set-based method for updating decision rules on attribute values' coarsening and refining***, IEEE Transaction on Knowledge and Data Engineering, 2014



(a) Promoter Gene Sequences (b) Hepatitis (c) Audiology (Standardized) (d) Soybean (Large)

(e) Dermatology (f) Breast Cancer (g) Tic-tac-toe (h) Solar Flare

Fig. 2: The comparison between non-incremental updating and URIAVC

Hongmei Chen, Tianrui Li, et al, ***A rough set-based method for updating decision rules on attribute values' coarsening and refining***, IEEE Transaction on Knowledge and Data Engineering, 2014

Heterogeneous Data Fusion under Composite Rough Sets

Our contributions

- Composite rough sets are proposed to deal with attributes of multiple different types in information systems for data fusion
- A matrix-based incremental method is presented for fast updating the approximations
- A parallel method for computing approximations is designed based on matrix, and implements it on Multi-GPU

Composite rough sets

- There may be attributes of multiple different types in information systems in real-life applications.
- Such information systems are called as composite information systems.

$$
\begin{cases}
U, & \text{a non-empty finite set of objects} \\
A = \bigcup A_k, & \text{a union of attribute sets} \\
 & \text{where } A_k \text{ is an attribute set with the same data type} \\
V = \bigcup\limits_{A_k \subseteq A} V_{A_k}, & V_{A_k} = \bigcup\limits_{a \in A_k} V_a, V_a \text{ is a domain of attribute } a \\
f : U \times A \rightarrow V, & \text{namely, } U \times \bigcup A_k \rightarrow \bigcup V_{A_k} \\
 & \text{where } U \times A_k \rightarrow V_{A_k} \text{ is an information function} \\
 & f(x, a) \text{ denotes the value of object } x \text{ on attribute } a
\end{cases}
$$

Junbo Zhang, Tianrui Li, Hongmei Chen: ***Composite rough sets for dynamic data mining.*** Information Science, 2014

Composite rough sets

- A composite relation is proposed to process attributes of multiple different types simultaneously in composite information systems.

Given $x, y \in U$ and $B = \bigcup B_k \subseteq A$, $B_k \subseteq A_k$,

the composite relation CR_B is defined as

$$CR_B = \left\{ (x,y) | (x,y) \in \bigcap_{B_k \subseteq B} R_{B_k} \right\}$$

Junbo Zhang, Tianrui Li, Hongmei Chen: ***Composite rough sets for dynamic data mining*.** Information Science, 2014

Composite rough sets

- An extended rough set model, called as composite rough sets, is presented.

$$\underline{CR_B}(X) = \{x \in U | CR_B(x) \subseteq X\}$$
$$\overline{CR_B}(X) = \{x \in U | CR_B(x) \cap X \neq \emptyset\}$$

$$\begin{cases} POS_{CR_B}(X) = \underline{CR_B}(X) \\ BND_{CR_B}(X) = \overline{CR_B}(X) - \underline{CR_B}(X) \\ NEG_{CR_B}(X) = U - \overline{CR_B}(X) \end{cases}$$

Junbo Zhang, Tianrui Li, Hongmei Chen: ***Composite rough sets for dynamic data mining.*** Information Science, 2014

Composite rough sets

- To adapt to the dynamic variation of the composite information system
- A matrix-based incremental method is presented for fast updating the approximations when many objects enter into or get out of the composite information system.

Characteristic function

$$G(X) = (g_1, g_2, \ldots, g_n)^T$$

Induced diagonal matrix

$$\Lambda_{n\times n}^{CR_B} = \mathrm{diag}\left(\frac{1}{\lambda_1}, \frac{1}{\lambda_2}, \ldots, \frac{1}{\lambda_n}\right)$$

Relation matrix

$$M_{n\times n}^{CR_B} = \min_{B_k \subseteq B} M_{n\times n}^{R_{B_k}}$$

Basic vector

$$H(X) = \Lambda_{n\times n}^{CR_B} \bullet \left(M_{n\times n}^{CR_B} \bullet G(X)\right) = \Lambda_{n\times n}^{CR_B} \bullet \Omega_{n\times 1}^{CR_B}$$

Junbo Zhang, Tianrui Li, Hongmei Chen: ***Composite rough sets for dynamic data mining.*** Information Science, 2014

Composite rough sets

$$M^{CR_B}_{n^+\times n^+} = \left[\begin{array}{c|c} M^{CR_B}_{n\times n} & P \\ \hline Q & R \end{array}\right] \qquad GD_{n^+\times r^+} = \left[\begin{array}{c|c} GD_{n\times r} & P' \\ \hline Q' & R' \end{array}\right]$$

$$\Omega^{CR_B}_{n^+\times r^+} = \left[\begin{array}{c|c} \Omega^{CR_B}_{n\times r} + P \bullet Q' & M^{CR_B}_{n\times n} \bullet P' + P \bullet R' \\ \hline Q \bullet GD_{n\times r} + R \bullet Q' & Q \bullet P' + R \bullet R' \end{array}\right]$$

$$\Lambda^{CR_B}_{n^+\times n^+} = diag\left(\frac{1}{\lambda^+_1}, \frac{1}{\lambda^+_2}, \ldots, \frac{1}{\lambda^+_{n^+}}\right)$$

$$\lambda^+_i = \begin{cases} \boxed{\lambda_i} + \boxed{\sum_{j=n+1}^{n^+} m_{ij},} & 1 \leqslant i \leqslant n \\ \sum_{j=1}^{n^+} m_{ij}, & n+1 \leqslant i \leqslant n^+ \end{cases}$$

$$\Omega^{CR_B}_{n^+\times r^+} = M^{CR_B}_{n^+\times n^+} \bullet GD_{n^+\times r^+} = \left(\omega^+_{ij}\right)_{n^+\times r^+}$$

$$\omega^+_{ij} = \begin{cases} \boxed{\omega_{ij}} + \boxed{\sum_{k=n+1}^{n^+} m_{ik}d_{kj},} & 1 \leqslant i \leqslant n,\ 1 \leqslant j \leqslant r \\ \sum_{k=1}^{n^+} m_{ik}d_{kj}, & \text{else} \end{cases}$$

Junbo Zhang, Tianrui Li, Hongmei Chen: ***Composite rough sets for dynamic data mining.*** Information Science, 2014

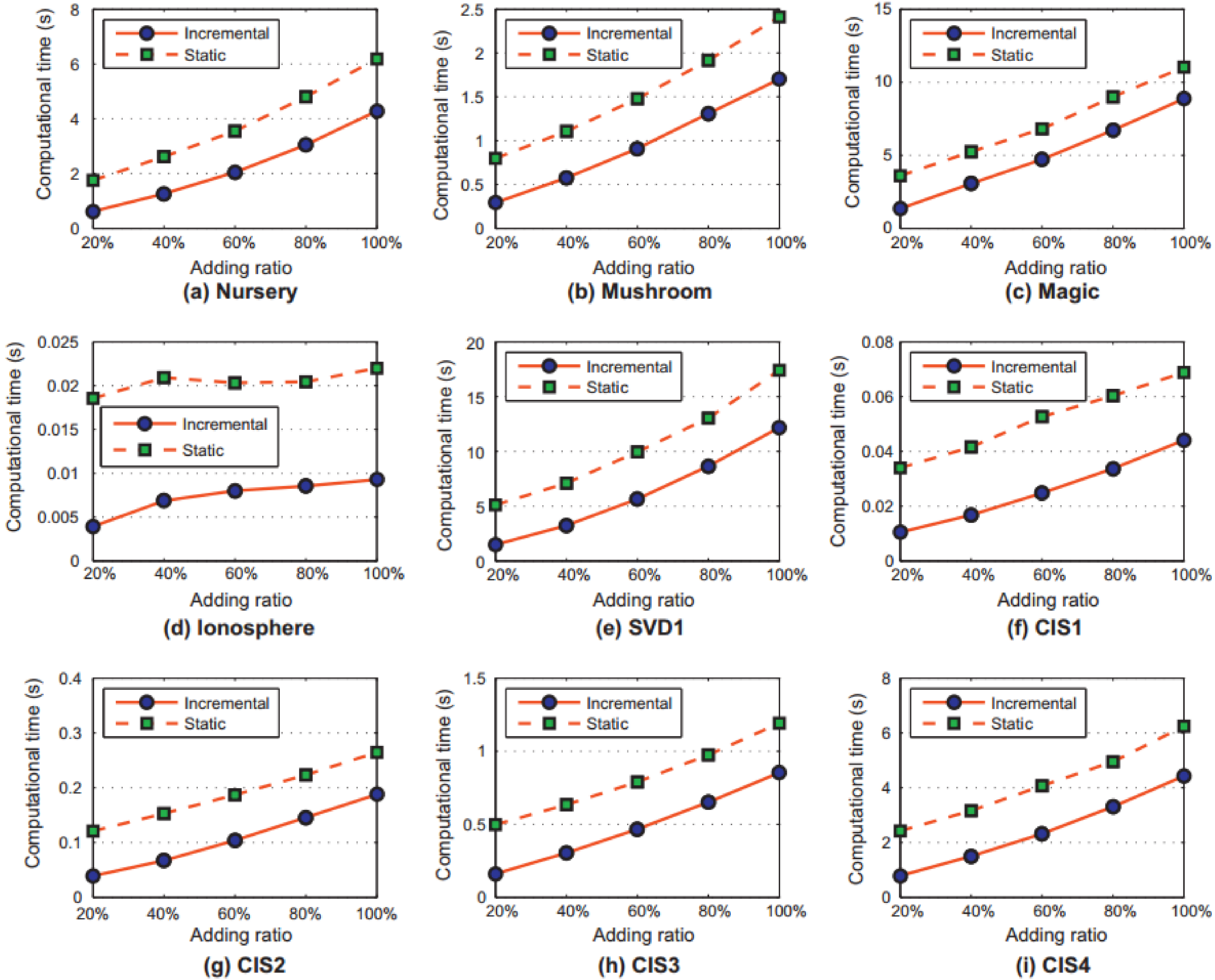


Fig. 1. A comparison of incremental and static algorithms versus the adding ratio of the data.

Matrix representation of approximations

$$\mathbf{U}_B = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n]$$

$$r_{ij} = \mathbf{x}_i \circ \mathbf{x}_j \triangleq \begin{cases} 1, & (x_i, x_j) \in \mathcal{C}_B \\ 0, & (x_i, x_j) \notin \mathcal{C}_B \end{cases}$$

$$\mathbf{R}_{\mathcal{C}_B} = \mathbf{U}_B^\top \circ \mathbf{U}_B \triangleq \begin{bmatrix} \mathbf{x}_1 \circ \mathbf{x}_1 & \mathbf{x}_1 \circ \mathbf{x}_2 & \cdots & \mathbf{x}_1 \circ \mathbf{x}_n \\ \mathbf{x}_2 \circ \mathbf{x}_1 & \boxed{\mathbf{x}_2 \circ \mathbf{x}_2} & \cdots & \mathbf{x}_2 \circ \mathbf{x}_n \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{x}_n \circ \mathbf{x}_1 & \mathbf{x}_n \circ \mathbf{x}_2 & \cdots & \mathbf{x}_n \circ \mathbf{x}_n \end{bmatrix} \in \{0,1\}^{n \times n}$$

$$\mathbf{D} = [\mathbf{D}_1, \mathbf{D}_2, \cdots, \mathbf{D}_m] = (d_{kj})_{n \times m} \in \{0,1\}^{n \times m}$$

$\overline{\mathcal{C}_B}(\mathbf{D}) = \mathbf{R}_B \otimes \mathbf{D}$ $\otimes$ refers to $u_{ij} = \bigvee_{k=1}^{n} (r_{ik} \wedge d_{kj})$

$\underline{\mathcal{C}_B}(\mathbf{D}) = \mathbf{R}_B \odot \mathbf{D}$ $\odot$ refers to $l_{ij} = \bigwedge_{k=1}^{n} (r_{ik} \rightarrow d_{kj})$

Junbo Zhang, Yun Zhu, Yi Pan, Tianrui Li: ***Efficient Parallel Boolean Matrix Based Algorithms for Computing Composite Rough Set Approximations.*** Information Science, 2016

Algorithm and Complexity Analysis

Time complexity

$$\mathcal{O}\left(n \log m + n + n^2 \times |B| + n^2 m + n^2 m\right) = \mathcal{O}\left(n^2(|B| + m)\right)$$

Space complexity

Since m<<n

$$\mathcal{O}(nm + n^2 + nm + nm) = \mathcal{O}\left(n(n+m)\right) \Rightarrow \mathcal{O}(n^2).$$

$\mathbf{D}$、$\mathbf{R}_B$、$\overline{\mathcal{C}_B}(\mathbf{D})$ and $\underline{\mathcal{C}_B}(\mathbf{D})$

$$\boxed{100000^2 \text{ Bytes} = \frac{100000^2}{2^{20}} \text{MB} \approx 9536.7\text{MB}}$$

begin

$\quad \mathbf{D} = (d_{ij})_{n \times m}$;

$\quad \mathbf{R}_B = \mathbf{U}_B^\top \circ \mathbf{U}_B$;

$\quad \overline{\mathcal{C}_B}(\mathbf{D}) = \mathbf{R}_B \otimes \mathbf{D}$;

$\quad \underline{\mathcal{C}_B}(\mathbf{D}) = \mathbf{R}_B \odot \mathbf{D}$;

$\quad$ **Output** $\overline{\mathcal{C}_B}(\mathbf{D})$ and $\underline{\mathcal{C}_B}(\mathbf{D})$;

Junbo Zhang, Yun Zhu, Yi Pan, Tianrui Li: ***Efficient Parallel Boolean Matrix Based Algorithms for Computing Composite Rough Set Approximations.*** Information Science, 2016

Batch Algorithm

begin

  $\mathbf{D} = (d_{ij})_{n \times m}$; // Construct the decision matrix

  for $k \leftarrow 0$ **to** $\left\lceil \frac{n}{T} \right\rceil - 1$ **do**

    $\mathsf{s} \leftarrow kT + 1$; // The `start' index

    $\mathsf{e} \leftarrow \min(kT + T, n)$; // The `end' index

    $\widetilde{U} \leftarrow \{x_{\mathsf{s}}, x_{\mathsf{s}+1}, \cdots, x_{\mathsf{e}}\}$; // The object set on the current data piece

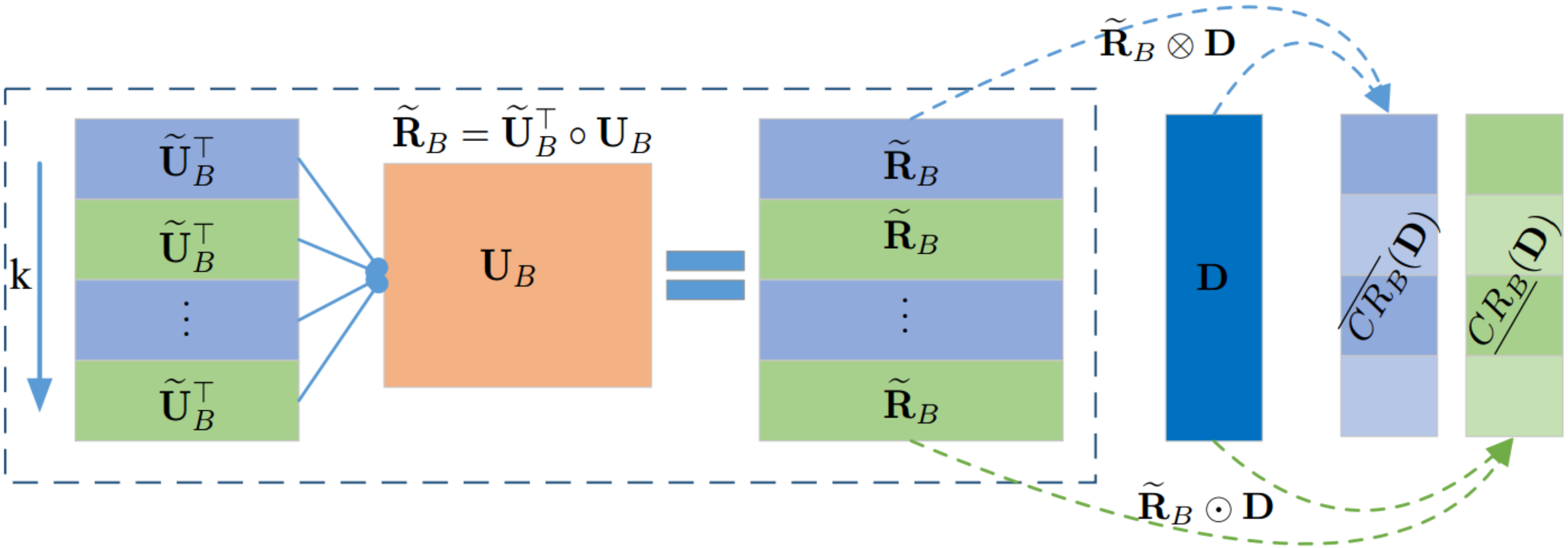
    $$\widetilde{\mathbf{R}}_B = \widetilde{\mathbf{U}}_B^{\top} \circ \mathbf{U}_B = \begin{bmatrix} \mathbf{x}_{\mathsf{s}} \circ \mathbf{x}_1 & \mathbf{x}_{\mathsf{s}} \circ \mathbf{x}_2 & \cdots & \mathbf{x}_{\mathsf{s}} \circ \mathbf{x}_n \\ \mathbf{x}_{\mathsf{s}+1} \circ \mathbf{x}_1 & \mathbf{x}_{\mathsf{s}+1} \circ \mathbf{x}_2 & \cdots & \mathbf{x}_{\mathsf{s}+1} \circ \mathbf{x}_n \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{x}_{\mathsf{e}} \circ \mathbf{x}_1 & \mathbf{x}_{\mathsf{e}} \circ \mathbf{x}_2 & \cdots & \mathbf{x}_{\mathsf{e}} \circ \mathbf{x}_n \end{bmatrix};$$ // Construct the relation submatrix

    $\overline{\mathcal{C}_B}(\mathbf{D})[\mathsf{s}:\mathsf{e}] = \widetilde{\mathbf{R}}_B \otimes \mathbf{D}$; // Calculate the submatrix of the upper approximation

    $\underline{\mathcal{C}_B}(\mathbf{D})[\mathsf{s}:\mathsf{e}] = \widetilde{\mathbf{R}}_B \odot \mathbf{D}$; // Calculate the submatrix of the lower approximation

  Output $\overline{\mathcal{C}_B}(\mathbf{D})$ and $\underline{\mathcal{C}_B}(\mathbf{D})$;

Batch Algorithm



Time complexity

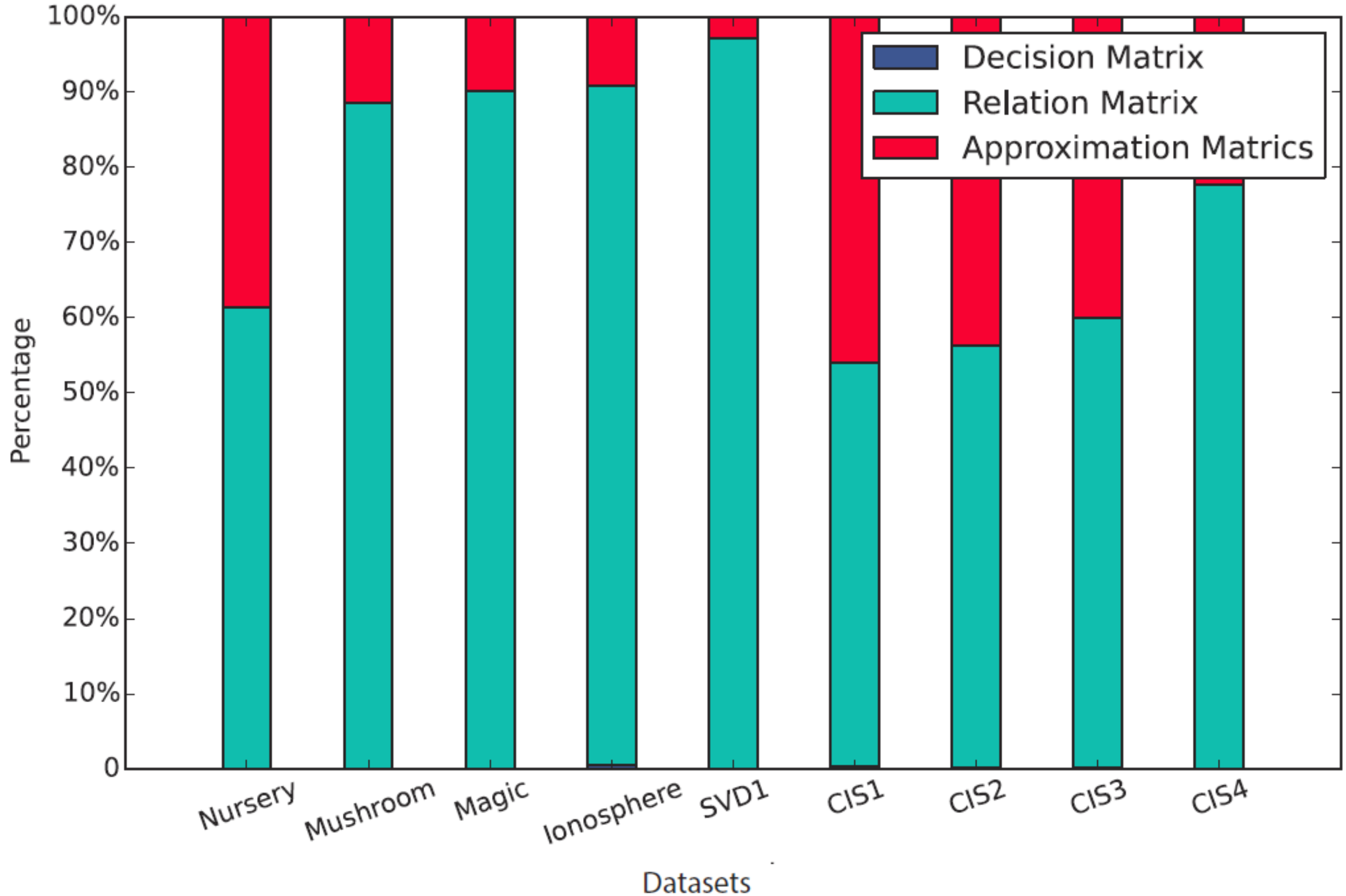
$$\mathcal{O}\left(n \log m + n + n^2 \times |B| + n^2 m + n^2 m\right) = \mathcal{O}\left(n^2(|B| + m)\right)$$

Space complexity

$$\mathcal{O}(nm + Tn + nm + nm) = \mathcal{O}\left(n(T + m)\right)$$

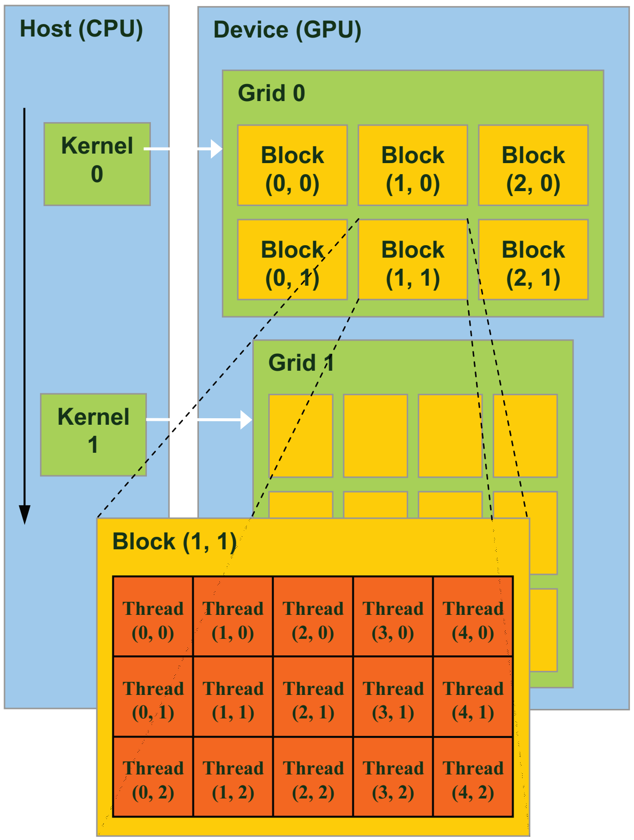
Junbo Zhang, Yun Zhu, Yi Pan, Tianrui Li: ***Efficient Parallel Boolean Matrix Based Algorithms for Computing Composite Rough Set Approximations.*** Information Science, 2016

Bottleneck of Computation



Junbo Zhang, Yun Zhu, Yi Pan, Tianrui Li: ***Efficient Parallel Boolean Matrix Based Algorithms for Computing Composite Rough Set Approximations.*** Information Science, 2016

Speed up by GPU



GPU Program Model

Time complexity

$$CPU(n \log m + n) + GPU\left(n^2(|B| + m)/p\right) + COMM\left(n(|B| + 3m)\right)$$

Multi-GPU

Time complexity

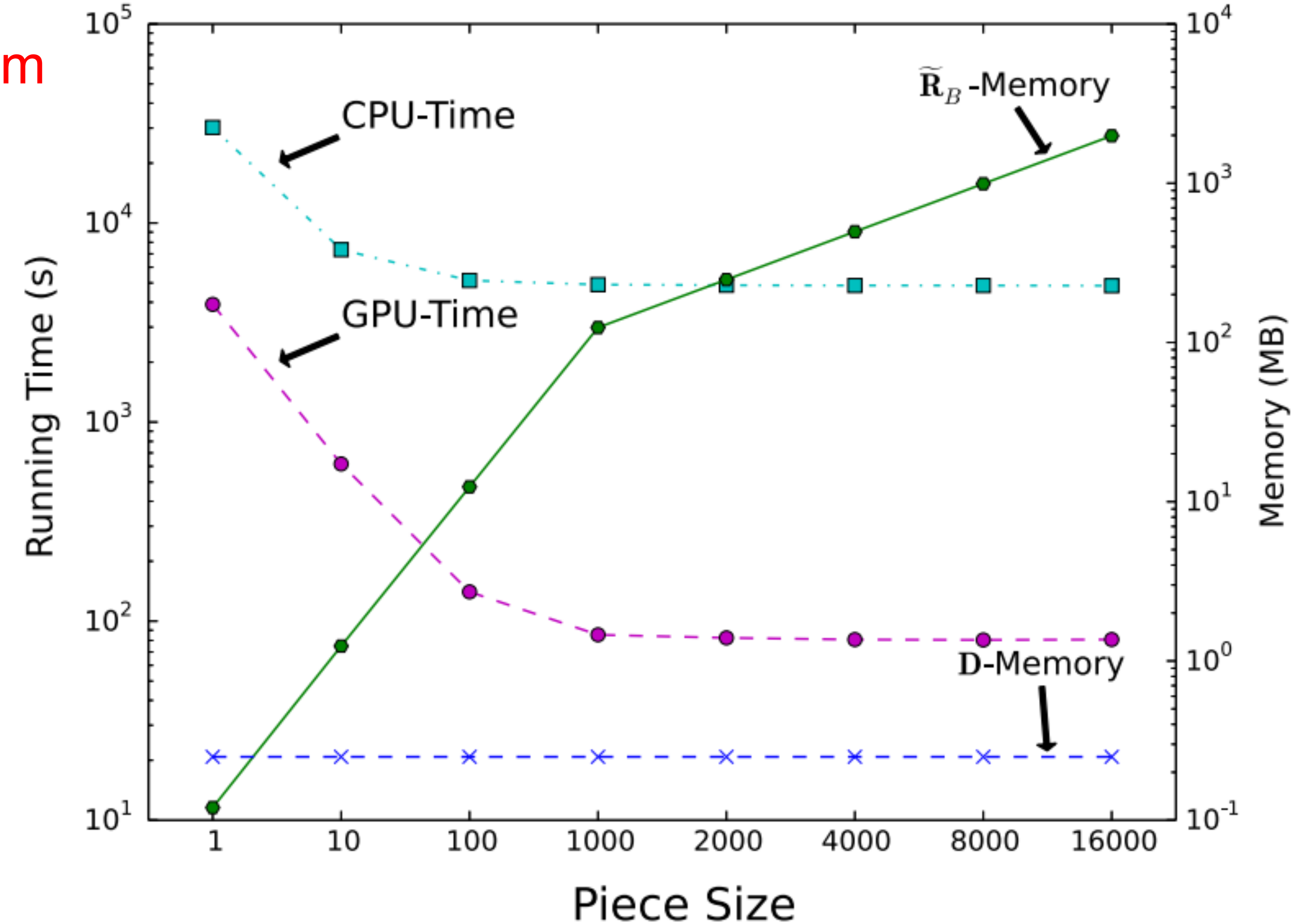
$$CPU(n \log m + n) + GPU\left(\frac{n^2(|B| + m)}{p|G|}\right) + COMM\left(n(|B| + 3m)\right)$$

begin
- [Host] $\mathbf{D} = (d_{ij})_{n \times m}$;
- [Host-to-Device] **Transfer** $\mathbf{D}$, $\mathbf{U}_B$ into global memory;
- **for** $k \leftarrow 0$ **to** $\left\lceil \frac{n}{T} \right\rceil - 1$ **do** `// Execute in GPU`
  - $\mathsf{s} \leftarrow kT + 1$;
  - $\mathsf{e} \leftarrow \min(kT + T, n)$;
  - $\widetilde{U} \leftarrow \{x_{\mathsf{s}}, x_{\mathsf{s}+1}, \cdots, x_{\mathsf{e}}\}$;
  - [CUDA Kernel] $\widetilde{\mathbf{R}}_B = \widetilde{\mathbf{U}}_B^{\top} \circ \mathbf{U}_B$;
  - [CUDA Kernel] $\overline{\mathcal{C}_B}(\mathbf{D})[\mathsf{s} : \mathsf{e}] = \widetilde{\mathbf{R}}_B \otimes \mathbf{D}$;
  - [CUDA Kernel] $\underline{\mathcal{C}_B}(\mathbf{D})[\mathsf{s} : \mathsf{e}] = \widetilde{\mathbf{R}}_B \odot \mathbf{D}$;
- [Device-to-Host] **Transfer** $\overline{\mathcal{C}_B}(\mathbf{D})$ and $\underline{\mathcal{C}_B}(\mathbf{D})$ from global memory;
- [Host] **Output** $\overline{\mathcal{C}_B}(\mathbf{D})$ and $\underline{\mathcal{C}_B}(\mathbf{D})$;

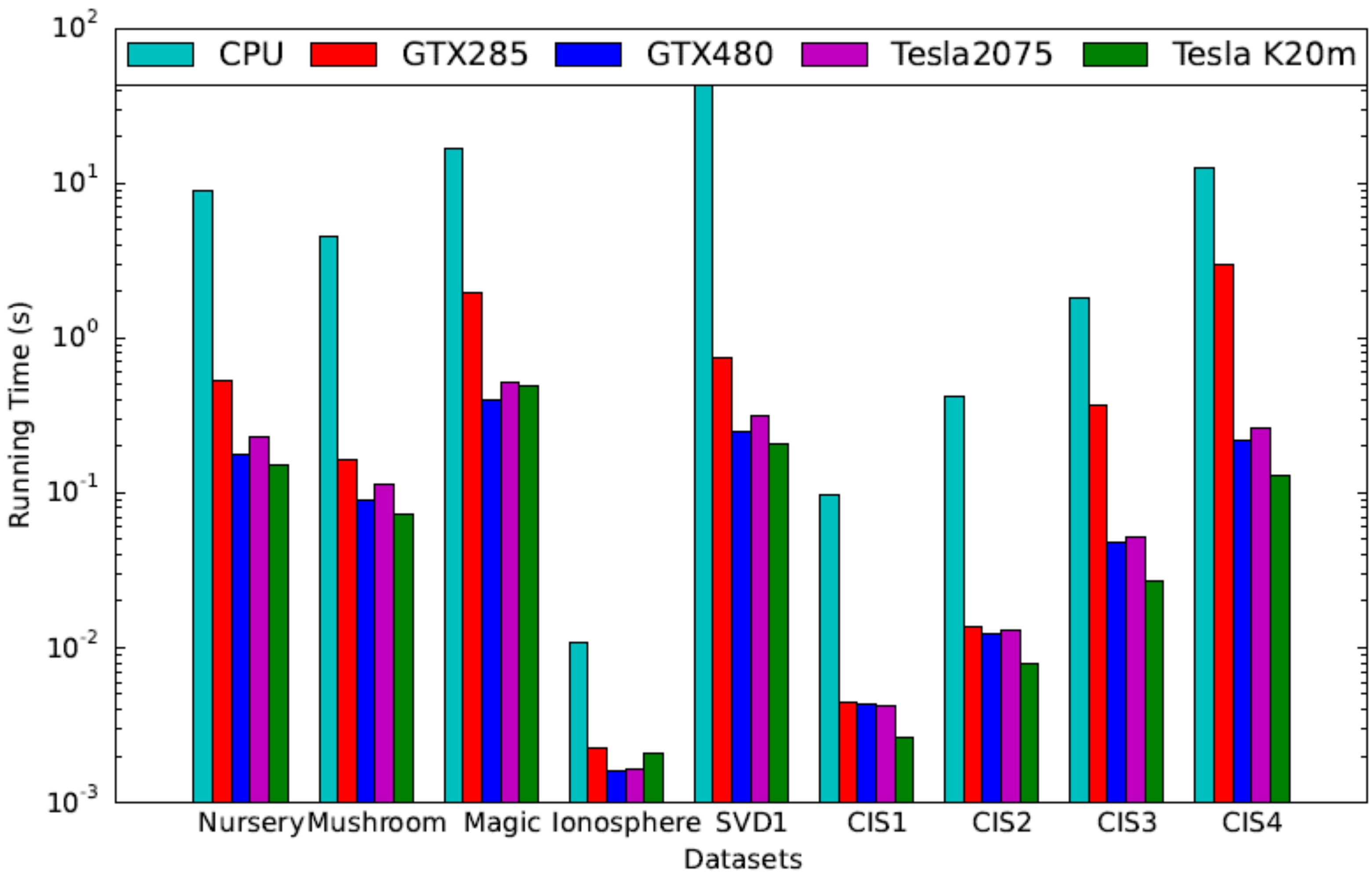
Junbo Zhang, Yun Zhu, Yi Pan, Tianrui Li: ***Efficient Parallel Boolean Matrix Based Algorithms for Computing Composite Rough Set Approximations.*** Information Science, 2016

Experimental Results

Batch Algorithm



Experimental Results



Average running time CPU and four kinds of GPUs with different datasets.

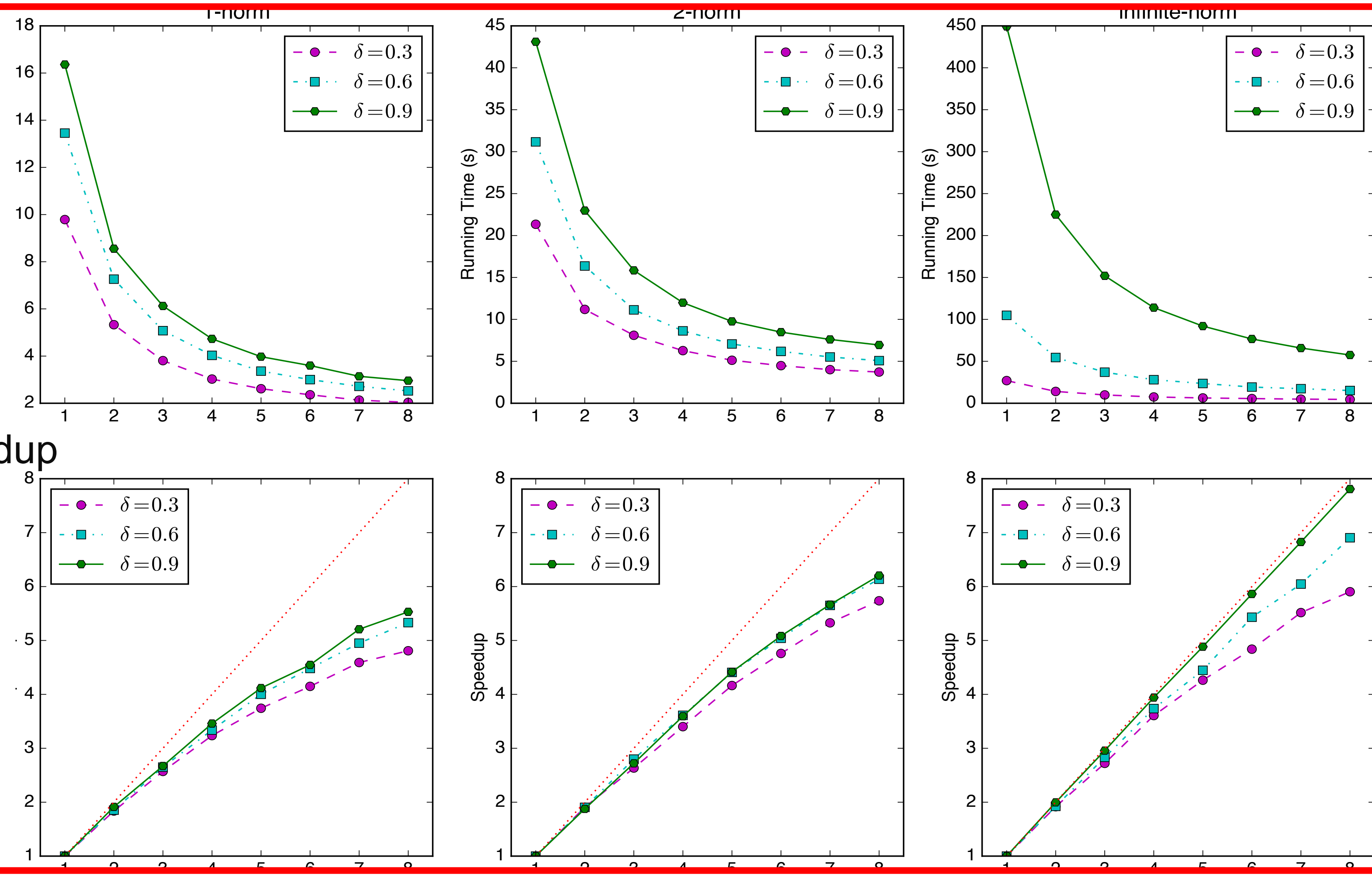
Experimental Results

8 GPUs

Data	p53-new
NoO	31420
NoF	5408

Time

Speedup



Uncertainty Information Processing under Three-way Decisions in GrC

Our contributions

- Dynamic maintenance of three-way decision rules
- Incremental three-way decisions with incomplete information
- Three-way decisions in dynamic decision-theoretic rough sets

Veracity of Big Data

- Veracity of Big Data refers to the biases, noise and abnormality in data
  - Is the data that is being stored, and mined meaningful to the problem being analyzed?
- Veracity deals with uncertain or imprecise data
  - Understanding the uncertainty in the data

Dynamic Maintenance of Three-Way Decision Rules

- The basic idea of TWD is to classify a set of objects into three regions, where three-way decision rules can be derived directly.
- As the data changed continuously, the three regions of a decision will be changed inevitably, while the induced three-way decision rules can be changed avoidably.
- The dynamic maintenance principles of three-way decision rules with incremental object are investigated.

Dynamic Maintenance of Three-Way Decision Rules

$$\Re_P(C): \quad Des([x]) \rightarrow Des(C), \; for \; [x] \subseteq POS_{(\alpha,\bullet)}(C),$$
$$\Re_B(C): \quad Des([x]) \rightarrow Des(C), \; for \; [x] \subseteq BND_{(\alpha,\beta)}(C),$$
$$\Re_N(C): \quad Des([x]) \rightarrow Des(C), \; for \; [x] \subseteq NEG_{(\bullet,\beta)}(C),$$

Table 1. Updating patterns of the conditional probability.

	Patterns	$[x]$	C	$Pr(C\|[x])$
1.	$\mathbf{x} \in [x] \wedge \mathbf{x} \in C$	$[x] \cup \{\mathbf{x}\}$	$C \cup \{\mathbf{x}\}$	$\frac{\|C \cap [x]\|+1}{\|[x]\|+1}$
2.	$\mathbf{x} \notin [x] \wedge \mathbf{x} \in C$	$[x]$	$C \cup \{\mathbf{x}\}$	$\frac{\|C \cap [x]\|}{\|[x]\|}$
3.	$\mathbf{x} \in [x] \wedge \mathbf{x} \notin C$	$[x] \cup \{\mathbf{x}\}$	C	$\frac{\|C \cap [x]\|}{\|[x]\|+1}$
4.	$\mathbf{x} \notin [x] \wedge \mathbf{x} \notin C$	$[x]$	C	$\frac{\|C \cap [x]\|}{\|[x]\|}$

(1) If $[x] \subseteq POS^{(t)}_{(\alpha,\bullet)}(C)$, then

$$\Re_P^{(t+1)}(C) = \Re_P^{(t)}(C);$$
$$\Re_B^{(t+1)}(C) = \Re_B^{(t)}(C);$$
$$\Re_N^{(t+1)}(C) = \Re_N^{(t)}(C).$$

(2) If $[x] \subseteq BND^{(t)}_{(\alpha,\bullet)}(C)$, then

(a) if $Pr(t+1) \geq \alpha$, then

$$\Re_P^{(t+1)}(C) = \Re_P^{(t)}(C) \cup (Des([x]) \rightarrow Des(C));$$
$$\Re_B^{(t+1)}(C) = \Re_B^{(t)}(C) - (Des([x]) \rightarrow Des(C));$$
$$\Re_N^{(t+1)}(C) = \Re_N^{(t)}(C).$$

if

$$\Re_B^{(t+1)}(C) = \Re_B^{(t)}(C);$$
$$\Re_N^{(t+1)}(C) = \Re_N^{(t)}(C).$$

(3) If $[x] \subseteq NEG^{(t)}_{(\alpha,\bullet)}(C)$, then

(a) if $Pr(t+1) \geq \alpha$, then

$$\Re_P^{(t+1)}(C) = \Re_P^{(t)}(C) \cup (Des([x]) \rightarrow Des(C));$$
$$\Re_B^{(t+1)}(C) = \Re_B^{(t)}(C);$$
$$\Re_N^{(t+1)}(C) = \Re_N^{(t)}(C) - (Des([x]) \rightarrow Des(C));$$

(b) if $\beta < Pr(t+1) < \alpha$, then

$$\Re_P^{(t+1)}(C) = \Re_P^{(t)}(C);$$
$$\Re_B^{(t+1)}(C) = \Re_B^{(t)}(C) \cup (Des([x]) \rightarrow Des(C));$$
$$\Re_N^{(t+1)}(C) = \Re_N^{(t)}(C) - (Des([x]) \rightarrow Des(C)).$$

(c) if $Pr(t+1) \leq \beta$, then

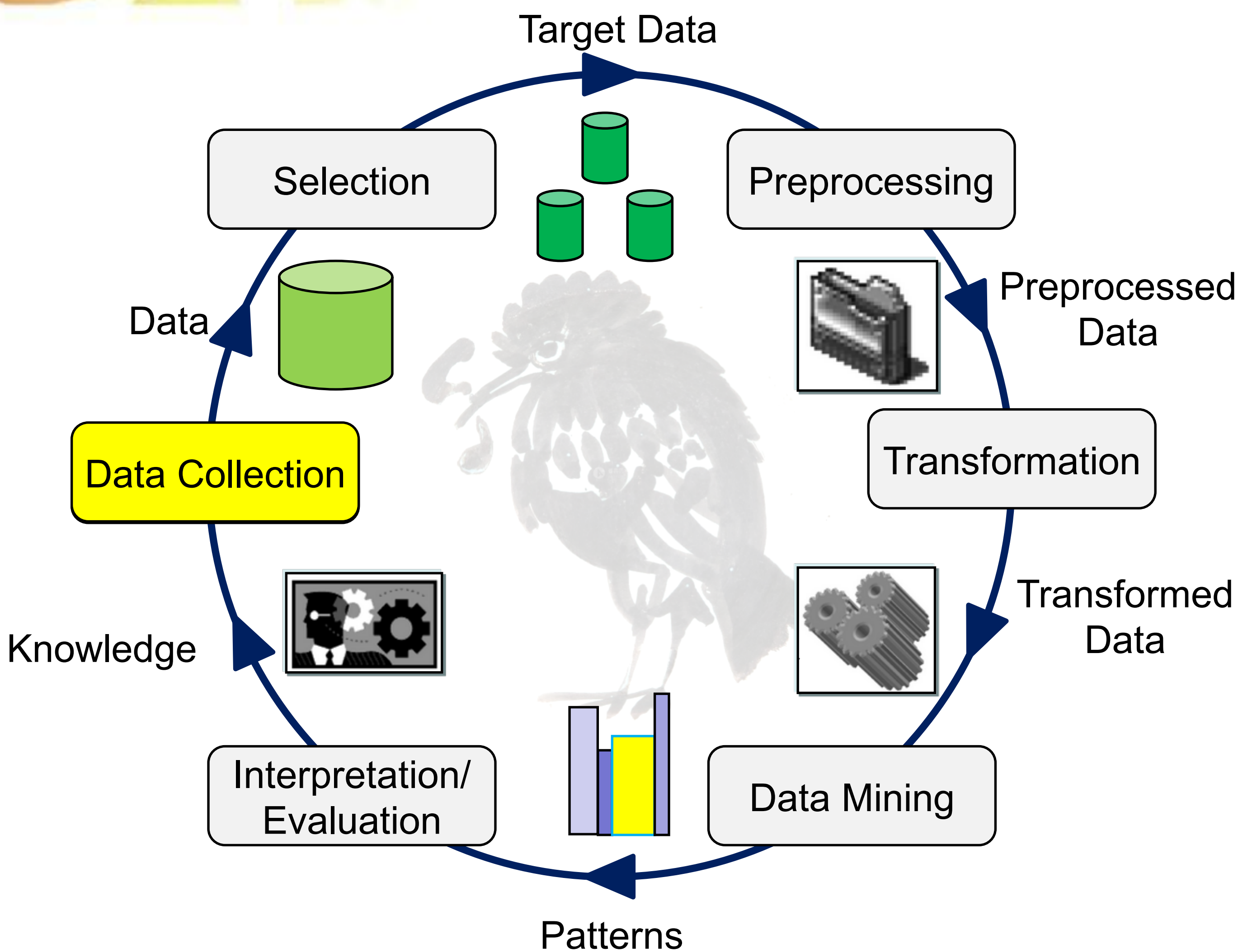
$$\Re_P^{(t+1)}(C) = \Re_P^{(t)}(C);$$
$$\Re_B^{(t+1)}(C) = \Re_B^{(t)}(C);$$
$$\Re_N^{(t+1)}(C) = \Re_N^{(t)}(C).$$

Knowledge Discovery under RST and GrC

Our contributions

- An updated KDD process model is presented
- iRoughSet: Incremental learning based on rough set theory
- RSHadoop: A rough set toolkit for massive data analysis on Hadoop

An updated KDD process model

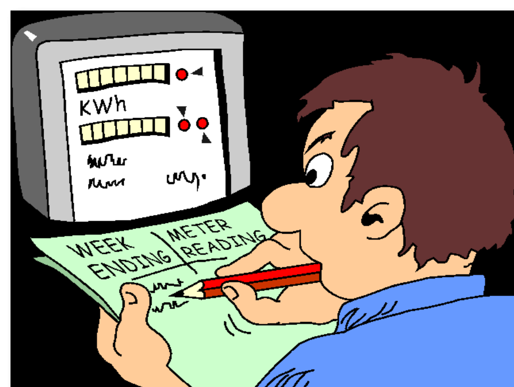


An updated process model of KDD (Li and Ruan, 2007)

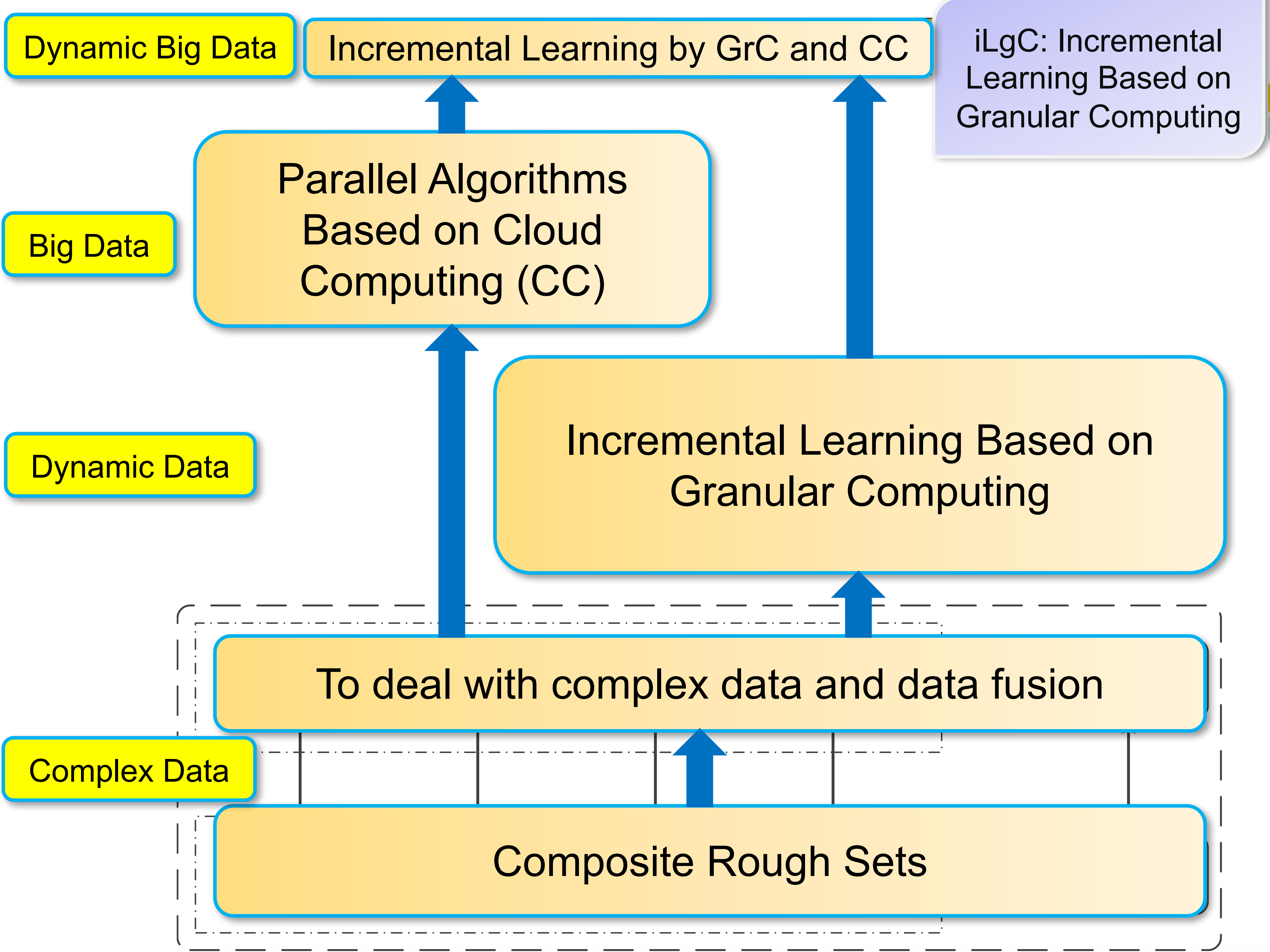
Tianrui Li, Da Ruan, ***An extended process model of knowledge discovery in database***, J. of Enter Inform Management, 2007

An updated KDD process model

- It incorporates data collection in the KDD process to provide a framework to support KDD applications better
  - Data collection directly affects mining results
  - Mining results may improve data collection



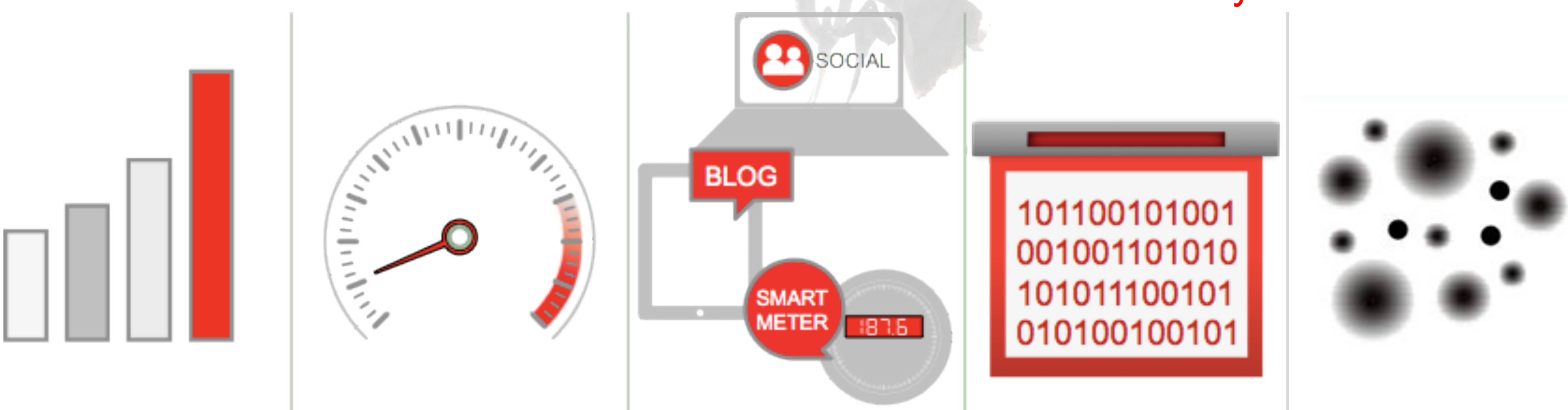
Tianrui Li, Da Ruan, ***An extended process model of knowledge discovery in database***, J. of Enter Inform Management, 2007


Dynamic Big Data
Incremental Learning by GrC and CC
iLgC: Incremental Learning Based on Granular Computing
Big Data
Parallel Algorithms Based on Cloud Computing (CC)
Dynamic Data
Incremental Learning Based on Granular Computing
To deal with complex data and data fusion
Complex Data
Composite Rough Sets

- iRoughSet: Incremental learning based on rough set theory
  - http://sist.swjtu.edu.cn:8080/ccit/project/iroughset.html
- RSHadoop: A rough set toolkit for massive data analysis on Hadoop
  - It is designed large-scale knowledge discovery based on rough set theory
  - http://sist.swjtu.edu.cn:8080/ccit/project/rshadoop.html

Our Solutions--PICKT

Parallel/Cloud Computing	Incremental Learning	Composite Rough Set Models	Knowledge Discovery	Three-Way Decisions
Data at Rest	Data in Motion	Data in Various Forms	Data with Low Value Density	Data in Doubt



Tianrui Li, Chuan Luo, Hongmei Chen, Junbo Zhang: PICKT: A Solution for Big Data Analysis. RSKT 2015

Reference

- A decision-theoretic rough set approach for dynamic data mining. **IEEE Transactions on Fuzzy Systems**, 2015
- A parallel matrix-based method for computing approximations in incomplete information systems, **IEEE Transactions on Knowledge and Data Engineering**, 2015
- A rough set-based method for updating decision rules on attribute values' coarsening and refining, **IEEE Transactions on Knowledge and Data Engineering**, 2014
- A rough-set based incremental approach for updating approximations under dynamic maintenance environments. **IEEE Transactions on Knowledge and Data Engineering**, 2013

Reference

- Efficient Parallel Boolean Matrix Based Algorithms for Computing Composite Rough Set Approximations, **Information Sciences**, 2016
- Fast algorithms for computing rough approximations in set-valued decision systems while updating criteria values, **Information Sciences**, 2015
- Incremental Update of Approximations in Dominance-based Rough Sets Approach under the Variation of Attribute Values, **Information Sciences**, 2015
- Dynamic Maintenance of Approximations in Set-valued Ordered Decision Systems under the Attribute Generalization. **Information Sciences**, 2014
- Composite Rough Sets for Dynamic Data Mining. **Information Sciences**, 2014
- A Parallel Method for Computing Rough Set Approximations. **Information Sciences**, 2012

Reference

- Matrix approach to decision-theoretic rough sets for evolving data, **Knowledge-based Systems**, 2016
- Update of Approximations in Composite Information Systems, **Knowledge-based Systems**, 2015
- Incremental Updating Approximations in Probabilistic Rough Sets under the Variation of Attributes, **Knowledge-based Systems**, 2015
- Incremental Updating Approximations in Dominance-based Rough Sets Approach under the Variation of the Attribute Set, **Knowledge-based Systems**, 2013
- Incremental Updating Approximations in Dominance-based Rough Sets Approach under the Variation of the Attribute Set, **Knowledge-based Systems**, 2013
- Maintenance of approximations in incomplete ordered decision systems while attribute values coarsening or refining, **Knowledge-based Systems**, 2012

Reference

- Incorporating logistic regression to decision-theoretic rough sets for classifications, **International Journal of Approximate Reasoning**, 2014
- A Comparison of Parallel Large-scale Knowledge Acquisition using Rough Set Theory on Different MapReduce Runtime Systems. **International Journal of Approximate Reasoning**, 2013
- Dynamic Maintenance of Approximations in Dominance-based Rough Set Approach under the Variation of the Object Set, **International Journal of Intelligent Systems**, 2013
- Rough Sets Based Matrix Approaches with Dynamic Attribute Variation In Set-valued Information Systems, **International Journal of Approximate Reasoning**, 2012
- Neighborhood rough sets for dynamic data mining. **International Journal of Intelligent Systems**, 2012

Reference

- 云平台下基于粗糙集的并行增量知识更新算法, **软件学报**, 2015
- 对象集变化时近似集动态维护的矩阵方法, **计算机研究与发展**, 2013
- ST-MVL: Filling Missing Values in Geo-sensory Time Series Data, **IJCAI 2016**
- Urban Sensing Based on Human Mobility, **UbiComp 2016**
- Forecasting Fine-Grained Air Quality Based on Big Data, **KDD 2015**
- Supervised Deep Learning with Auxiliary Networks, **KDD 2014**

Reference

- **Special Issue**: Three-way Decisions and Granular Computing, **KBS**, 2016
- **Special Issue**: Computational Technique in Data Science, **IJIS**, 2015
- **Special Issue**: Advances on Rough Sets and Knowledge Technology, **FI**, 2014
- **Special Issue**: Fuzziness in Systems Modelling, **IJGS**, 2013
- **Special Issue**: Computational Intelligence for Policy Making and Risk Governance, **IJUFKBS**, 2012
- **Special Issue**: Computational Intelligence in Data Analysis, **IJCIS**, 2012
- **Special Issue**: New Trends on Intelligent Decision Support Systems, **KBS**, 2012
- **Special Issue**: Computational Intelligence in Decision Making, **IJCIS**, 2011

Special Issues

谢　谢

THANK YOU