Challenges of Big Data

Volume
- Data quantity
- Few Terabytes
- Huge Transaction
- Log Tables

Velocity
- Data Speed
- Batch Processing
- Stream Data
- Real/Near-time Processing

Variety
- Data Types
- Structured
- Unstructured
- Semi structured

Veracity
- Trust and integrity
- Trustworthiness
- Authenticity
- Availability
- Accountability

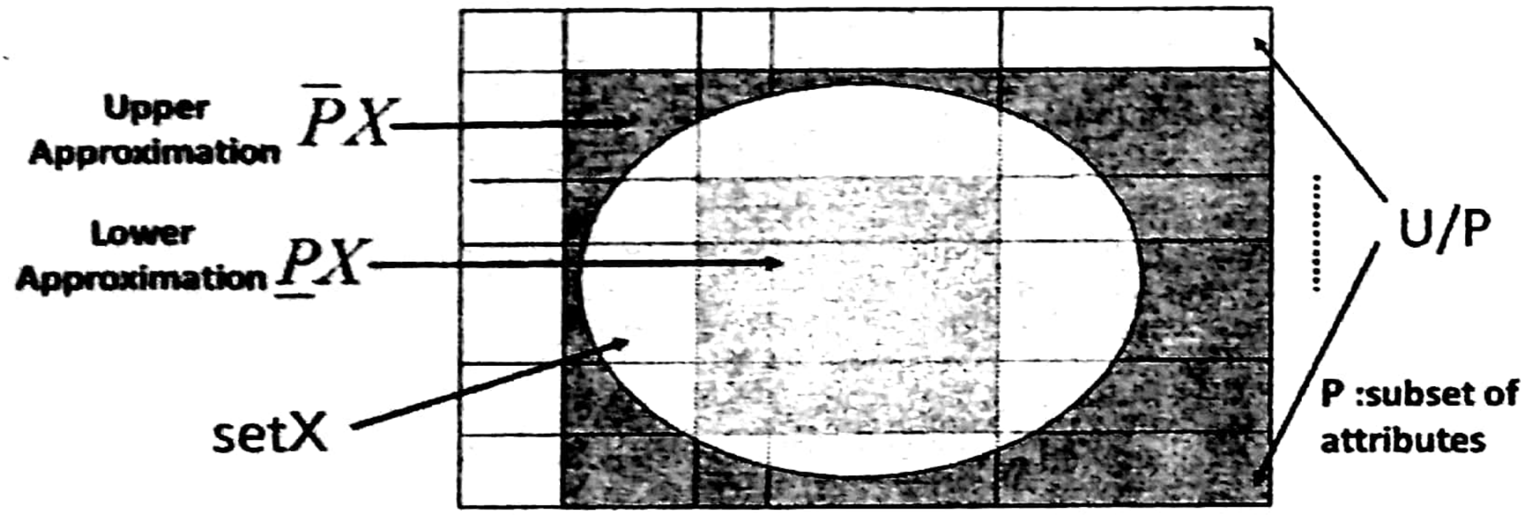
Value
- Valuable Information
- Time Complexity
- Cost of Processing
- Co-relations

Preliminaries

Rough Set Theory

- Mathematical tool for both feature selection and knowledge discovery [4].

- Derives information from data itself without requiring any preliminary or additional information about data.



U/P : Partitioning whole set of objects (universe) with respect to subset of attributes P

Reduct

Example

[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
U1	Yes	Yes	Normal	No
U2	Yes	Yes	High	Yes
U3	Yes	Yes	Very-high	Yes
U4	No	Yes	Normal	No
U5	No	No	High	No
U6	No	Yes	Very-high	Yes

Reduct1 = {Muscle-pain,Temp.}

[illegible]	[illegible]	[illegible]	[illegible]
U1,U4	Yes	Normal	No
U2	Yes	High	Yes
U3,U6	Yes	Very-high	Yes
U5	No	High	No

Reduct2 = {Headache, Temp.}

[illegible]	[illegible]	[illegible]	[illegible]
U1	Yes	Normal	No
U2	Yes	High	Yes
U3	Yes	Very-high	Yes
U4	No	Normal	No
U5	No	High	No
U6	No	Very-high	Yec

Positive Region

$P, Q \subseteq A$, equivalence partitions over U, then a P positive region[12] of Q can be defined as

$$POS_P(Q) = \cup_{X \subseteq U/Q} \underline{PX}$$

P and Q are the equivalence partitions with respect to set of conditional attribute set P and decision attribute set Q.

The P-positive region of X contains those objects that can be certainly classified in the set X.

Degree of Dependency

The Degree of dependency (k) of Q on P can be defined as

$$k = \gamma_P(Q) = \frac{|pos_P(Q)|}{|U|}$$

Q is totally and partially dependent on P, if $k=1$ and if $0 < k < 1$ respectively. For $k=0$, Q is independent of P [12].

Positive Region

Set of condition Attribute

$P = \{b, c\}$

Set of Objects

$X = \{0,1,2,3,4\}$

Partition1	Partition2	Partition3	Partition4	Partition5
0	1	2	3	5
4	6			
	7			
b = 0, c = 2	b = 1, c = 1	b = 0, c = 0	b = 1, c = 0	b = 2, c = 0

Equivalence Partitions of whole set of objects (U) with respect to attribute set P is $U/P = \{\{0,4\},\{1,6,7\},\{2\},\{3\},\{5\}\}$

Positive Region

Set of decision Attribute

$Q = \{e\}$

Set of Objects

$X = \{0,1,2,3,4\}$

Partition1	Partition2	Partition3
0	1	2
	3	4
	6	5
		7
e = 0	e = 2	e = 1

Equivalence Partitions of whole set of objects (U) with respect to attribute set Q is $U/Q = \{\{0\},\{2,4,5,7\},\{1,3,6\}\}$

Positive Region

Partition of conditional attribute set is fully belongs to the partitions of decision attribute set.

$$U/P = \{\{0,4\},\{1,6,7\},\{2\},\{3\},\{5\}\}$$

$$U/Q = \{\{0\},\{2,4,5,7\},\{1,3,6\}\}$$

Final Positive region Set: $POS_P(Q) = \{2,3,5\}$

Certainty, Coverage, Strength

- An Information system $J = (U, C \cup D)$
  $U = \{u_1, \ldots\ldots, u_x\}$ is the set of data objects,
  $C = \{c_1, \ldots., c_n\}$ is the set of condition attributes
  and $D = \{d\}$ is the one-element set with decision attribute or class label attribute.

- **Certainty[4]** $cer_x(C,D) = \frac{|C(x) \cap D(x)|}{|C(x)|}$

Degree of membership x to the decision class $D(x)$, given C.

- If $cer_x(C, D) = 1$, then will be called a *certain decision* rule;
- if $0 < cer_x(C, D) < 1$ the decision rule will be referred to as an *uncertain decision rule* [4].

Certainty, Coverage, Strength

- **Coverage[4]** $cov_x(C,D) = \frac{|C(x) \cap D(x)|}{|D(x)|}$

Degree of membership of x to condition class $C(x)$, given D.

- **Support[4]** $supp_x(C,D) = |C(x) \cap D(x)|$

- **Strength[4]** $\sigma_x(C,D) = \frac{supp_x(C,D)}{|U|} = \frac{\textit{Support for a particular rule}}{\textit{Total number of objects in Unive}}$

Certainty, Coverage, Strength

- **Coverage[4]** $cov_x(C,D) = \frac{|C(x) \cap D(x)|}{|D(x)|}$

Degree of membership of x to condition class $C(x)$, given D.

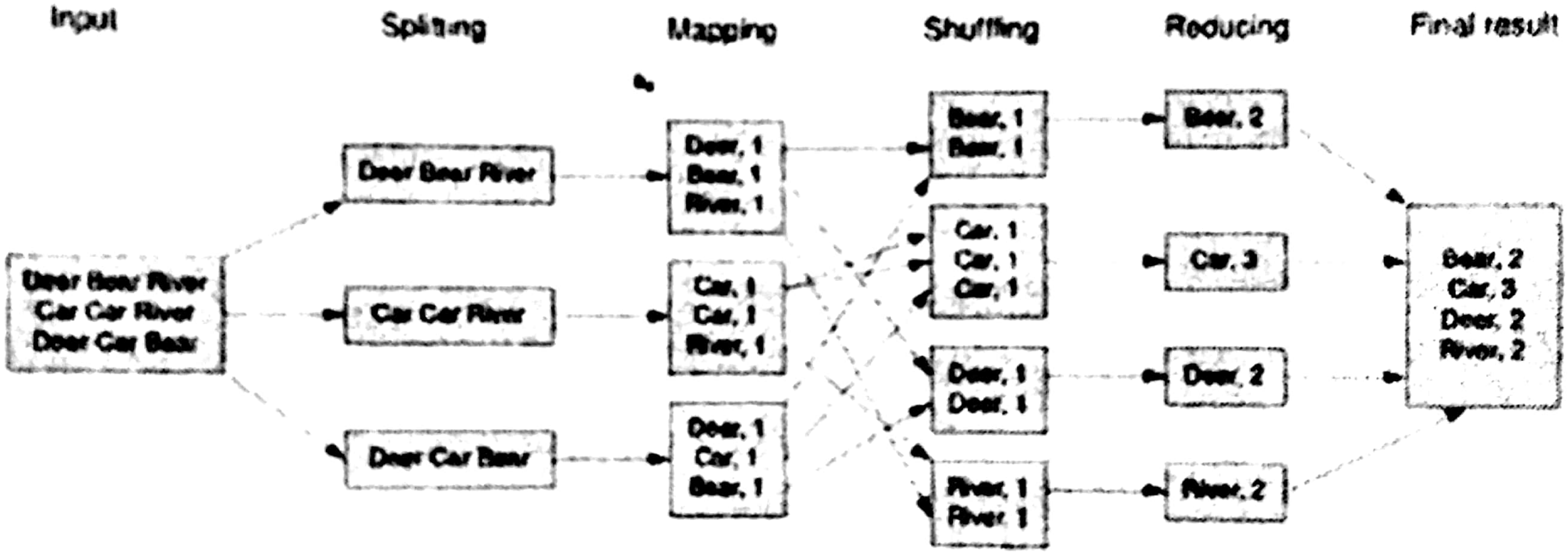
- **Support[4]** $supp_x(C,D) = |C(x) \cap D(x)|$

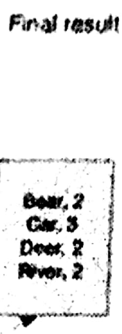
- **Strength[4]** $\sigma_x(C,D) = \frac{supp_x(C,D)}{|U|} = \frac{Support\ for\ a\ particular\ rule}{Total\ number\ of\ objects\ in\ Universe}$

MapReduce Framework

- MapReduce [2]: Framework for Parallel Processing

The overall MapReduce word count process





Proposed Method

•Extract important features from large dataset by using multiple nodes in parallel. Eradicates uncertainty and redundant dataset by using Positive region and Degree of dependency.

•Divide the task into Mapper, Combiner, Reducer and compute reduct.

Mapper

Step1:Distribute the data with *{key,value}* pair.

Step2:Compute the cardinality of *{key,value}* pair in parallel on different nodes.

Decision Table

	Conditional Attributes				Decision Attribute
x ∈ U	a	b	c	d	e
1	1	5	10	6	1
2	1	5	11	6	2
3	2	5	10	6	3
4	2	5	11	6	4
5	3	5	10	6	5

Mapper

- Distribute the dataset in different nodes to run in parallel
- Consider the condition attribute $R_{mod} = \{a\}$
- Partitioning the whole objects U with respect to {a}

$$U / R_{mod} = \{\{1,2\},\{3,4\},\{5\}\}$$

- Partitioning the whole objects U with respect to decision attribute{

$$U / \{e\} = \{\{1\},\{2\},\{3\},\{4\},\{5\}\}$$

x ∈ U	a	b	c	d	e
1	1	5	10	6	1
2	1	5	11	6	2
3	2	5	10	6	3
4	2	5	11	6	4
5	3	5	10	6	5

Mapper

- Projects the *{Key, Value1}* pair with respect to decision attribute {e}.

- Considered an extra *value2* to compute the dependency of partitions of conditional attribute sets on partitions of decision attribute sets.

- Initializes all the *Key K* with respect to *Value1* based on decision attribute.

$$R_{mod} = \{a\}$$

$$U / R_{mod} = \{\{1,2\},\{3,4\},\{5\}\}$$

$$U / \{e\} = \{\{1\},\{2\},\{3\},\{4\},\{5\}\}$$

Key	Value 1	Value 2
{1,2}	{1}	1
{1,2}	{2}	1
{3,4}	{3}	1
{3,4}	{4}	1
{5}	{5}	1

Reducer

- Reduce the number of valid *{key,value1}* pair with respect to same decisions
- Computes the final cardinalities of all the elements of the partition $U \backslash R_{mod}$ which belongs to definite decision class $U \backslash \{e\}$.
- Discards those objects which have partial dependency.
- Calculates the summation of all *value2*.

$$R_{mod} = \{a\}$$

$$U / R_{mod} = \{\{1,2\},\{3,4\},\{5\}\}$$

$$U / \{e\} = \{\{1\},\{2\},\{3\},\{4\},\{5\}\}$$

Key	Value 1	Value 2
{1,2}	-1	0
{3,4}	-1	0
{5}	**{5}**	**1**
TOTAL		**1**

Compute Reduct

Step1: Calculate the degree of dependency of each attribute in parallel based on positive region.

Step2: Find the attribute with a minimum degree of dependency value and include that attribute in Reduct set.

Step3: Compute the degree of dependency for all the combinations of Reduct set element with other attributes i parallel until the degree of dependency is equal to 1 and include this combination in Reduct set.

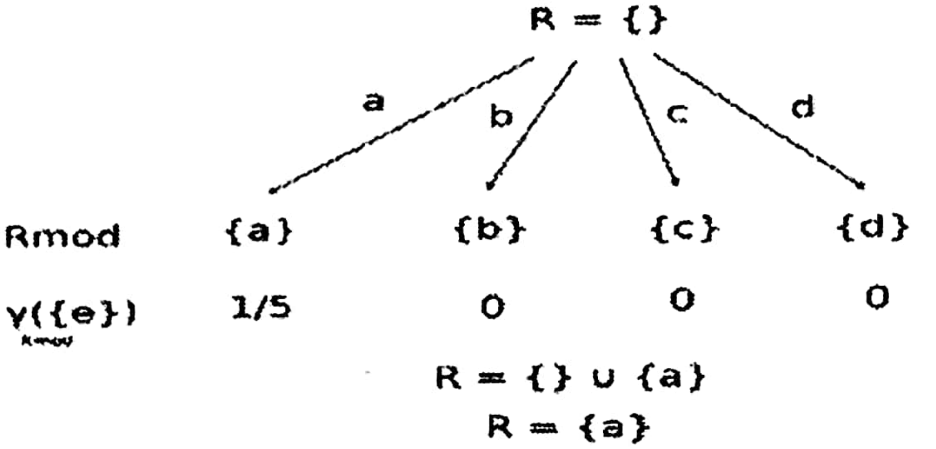
Compute Reduct

Positive Region set : {*5*}
Number of objects in positive region =*1*
Total number of Objects=*5*
Degree of Dependency= *1/5*

Key	Value 1	Value 2
{1,2}	-1	0
{3,4}	-1	0
{5}	**{5}**	**1**

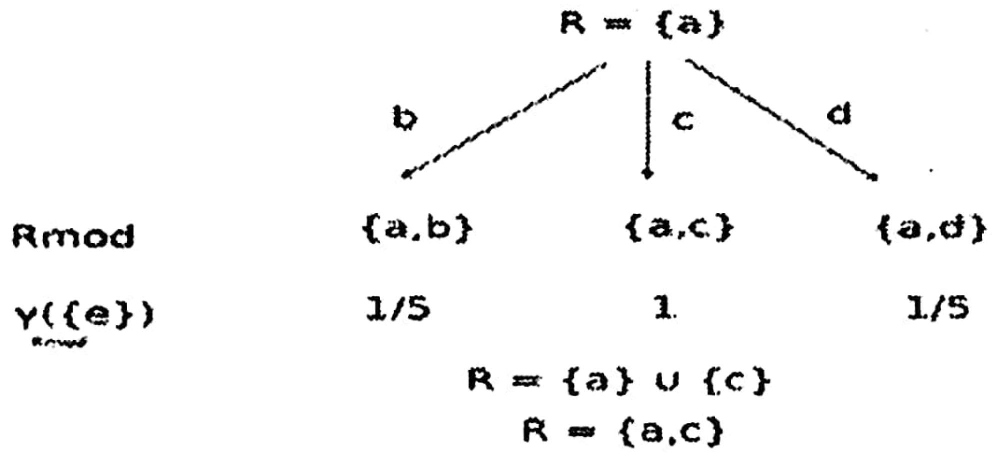
Compute Reduct

- Calculate Degree of dependency with respect to each conditional attribute in parallel in different nodes
- Minimum Degree of dependency for attribute $a = 1/5$
- First element in Reduct set is $\{a\}$



Compute Reduct

- Combinations of Reduct set element with other conditional attributes are $\{a,b\},\{a,c\},\{a,d\}$
- Calculate the Degree of dependency for all combinations in parallel.
- Degree of dependency *1* for the combination $\{a,c\}$
- Final Reduct set is $\{a,c\}$



Time Complexity Analysis

The complexities of each of the functions are:

Function	*Time complexity*
HeuristicMapper	$O(\|C\|)$
Sorting keys (HeuristicMapRed)	$O((\|U\| * \|C\|) * log(\|U\| * \|C\|))$
MaximizerReducer	$O(\|U\| * \|C\|)$
SummerMapper	$O(1)$
Sorting keys (SummerMapRed)	$O((\|U\| * \|C\|) * log(\|U\| * \|C\|))$
SummerReducer	$O(\|U\|)$
Calculating the best x	$O(\|C\|)$
Total per iteration	$O(\|U\| * (\|C\| * log(\|C\| + \|U\|)))$

Final complexity may be expressed as:

$$O(|U| * (|C| * log(|C| + |U|)) * K)$$

where $K = |R|$ is size of the reduct set

Scanned by CamScanner

References

[1] V. López, S. del Río, J. M. Benítez, and F. Herrera, "Cost-sensitive linguistic fuzzy rule based classification systems under the MapReduce framework for imbalanced big data," Fuzzy Sets and Systems, vol. 258, pp. 5-38, 2015.

[2] A. Srinivasan, T. A. Faruquie, and S. Joshi, "Data and task parallelism in ILP using MapReduce," Machine Learning, vol. 86, pp. 141-168, 2011.

[3] Guyon, Isabelle, and A. Elisseeff. "An Introduction to Variable and Feature Selection."Journal of Machine Learning Research, vol. 3, pp. 1157-1182, 2003.

[4] Z. Pawlak, "Rough sets," International Journal of Computer & Information Sciences, vol. 11, pp. 341–356, 1982.

[5] Shang, C., Shen, Q., "Rough Feature Selection for Neural Network Based Image classification," International Journal of Image and Graphics 2, pp. 541-555, 2002.