Second Workshop on Software Challenges to Exascale Computing

(13th , 14th December 2018, New Delhi)

A presentation on

A review of dimensionality reduction in high-dimensional data using multi-core and many-core architecture

by

Mr. Siddheshwar Vilas Patil

Ph. D. Research Scholar (QIP, AICTE Scheme)

Under the Guidance of

Prof. Dr. D. B. Kulkarni

Registrar & Professor in Information Technology,

Walchand College of Engineering, Sangli, MH, India

(A Government Aided Autonomous Institute)

Outline

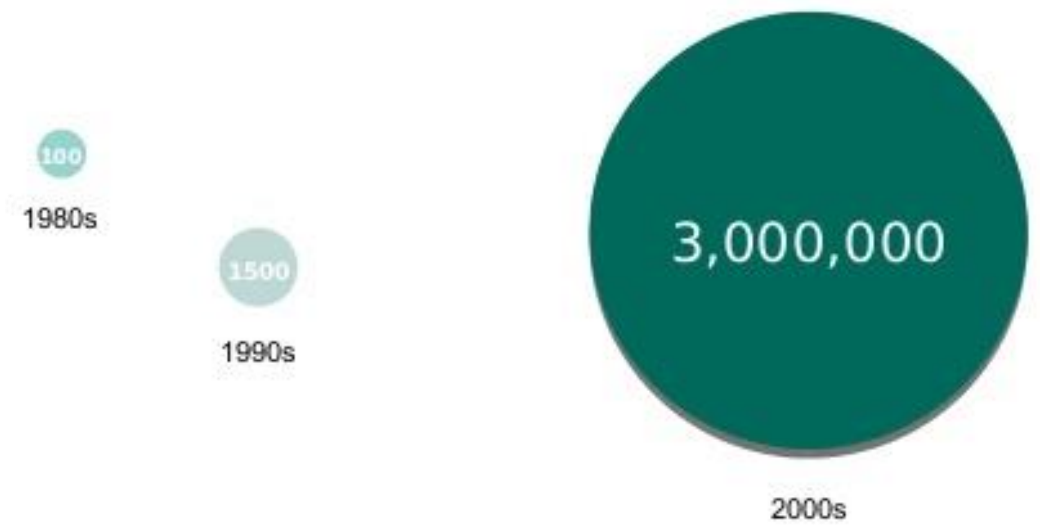
- Introduction
- Dimensionality Reduction
- Literature Review
- Challenges
- Parallel Computing Approaches
- Conclusion
- References

Introduction

- Massive amounts of high dimensional data
- Big Data - Exponential growth and availability of data, 3Vs
- Afterwards, this list was extended with "Big Dimensionality" in Big Data .
- "Curse of Big Dimensionality", is boosted by the explosion of features (thousand or even millions of features)
- Early, Data scientists - **huge number of instances**, while paying **less attention to the features aspect.**

Big Dimensionality

Millions of Dimensions



Example- libSVM Database

- In 1990s, the maximum dimensionality - **62,000**
- In 2000s - **16 million**
- In 2010s - **29 million**
- In this new scenario, it is common now to deal with millions of features, so the existing learning methods need to be adapted.

Summary of high-dimensional datasets

Data set	# samples	# features	#classes
Colon	62	2000	2
Brain tumor	50	10367	4
Leukemia	47	2000	2
Lymphomas	77	5470	2
Prostate	102	1500	2
Epsilon	400000	2000	2
ECBDL14	65003913	630	2
url	1916904	3231961	2
kddb	19264097	29890095	2

Scalability

- Scalability is defined as the effect that an increase in the size of the training set has on the computational performance of an algorithm: accuracy, training time and allocated memory.

Methods to perform DR

- **Missing Values**
- **Low Variance-** Let's think of a scenario where we have a **constant variable** (all observations have the same value) in data set
- Not improve the power of model because it has zero variance
- **High Correlation-** It is not good to have multiple variables of similar information.
- Pearson correlation matrix to identify the variables with high correlation.

Dimensionality Reduction

- **Feature Extraction:** Transforms original features to a set of new features
- More compact and of stronger discriminating power.
- Applications - Image analysis, Signal processing, and Information retrieval

Dimensionality Reduction

- **Feature Selection:** remove the irrelevant and redundant features
- Two features are **redundant** to each other if their values are completely **correlated**
- Irrelevant: contain no information that is useful for the data mining task at hand
- **Feature is relevant** if it contains some information about the target **(removal of this feature will decrease accuracy of classifier)**

Dimensionality reduction

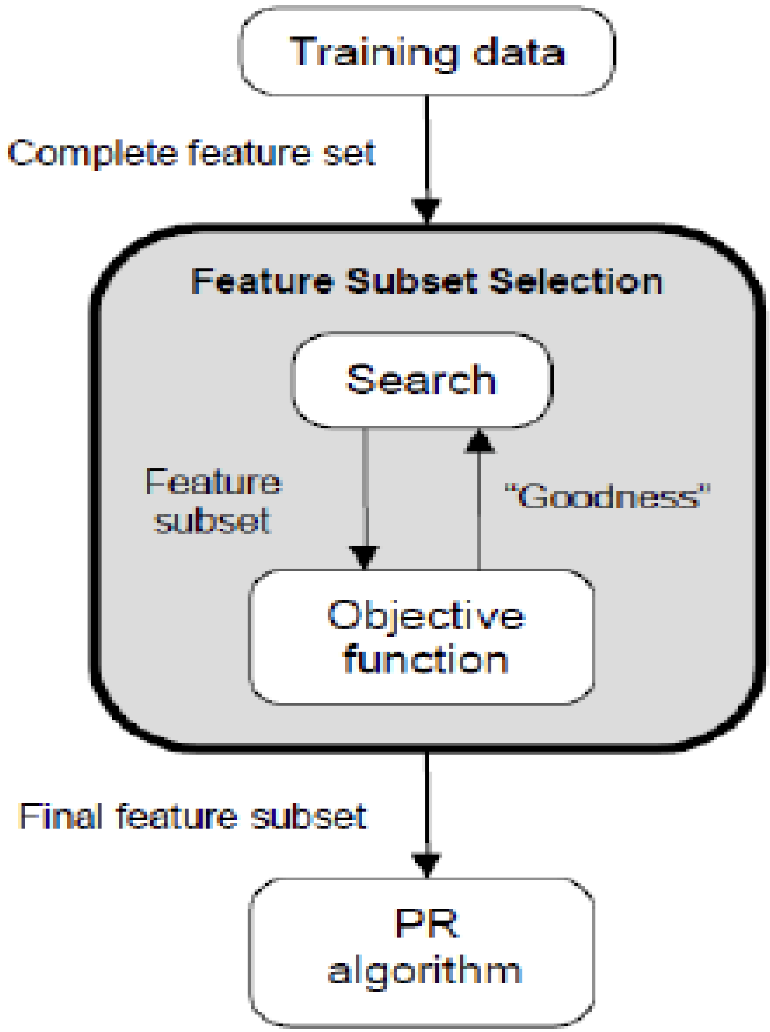
- **Linear Methods:**
  - Principal Component Analysis (PCA)
  - Linear Discriminate Analysis (LDA)
  - Multidimensional Scaling (MDS)
  - Non-negative Matrix Factorization(NMF)
  - Lasso
- **Non-Linear Methods:**
  - Locally Linear Embedding (LLE)
  - Isometric Feature Mapping (Isomap)
  - Hilbert Schmidt Independence Criterion(HSIC)
  - Minimum Redundancy Maximum Relevancy (mRMR)
- Autoencoders (Linear as well Non Linear)

Feature selection methods

- **Individual evaluation** is also known as feature ranking and assesses individual features by assigning them weights according to their degrees of relevance.
- **Subset evaluation** produces candidate feature subsets based on a certain search strategy.
- Compared with the previous best one with respect to this measure.
- While the **individual evaluation is incapable of removing redundant features because redundant features are likely to have similar rankings,** the subset evaluation approach can handle feature redundancy with feature relevance.

Feature Selection Steps

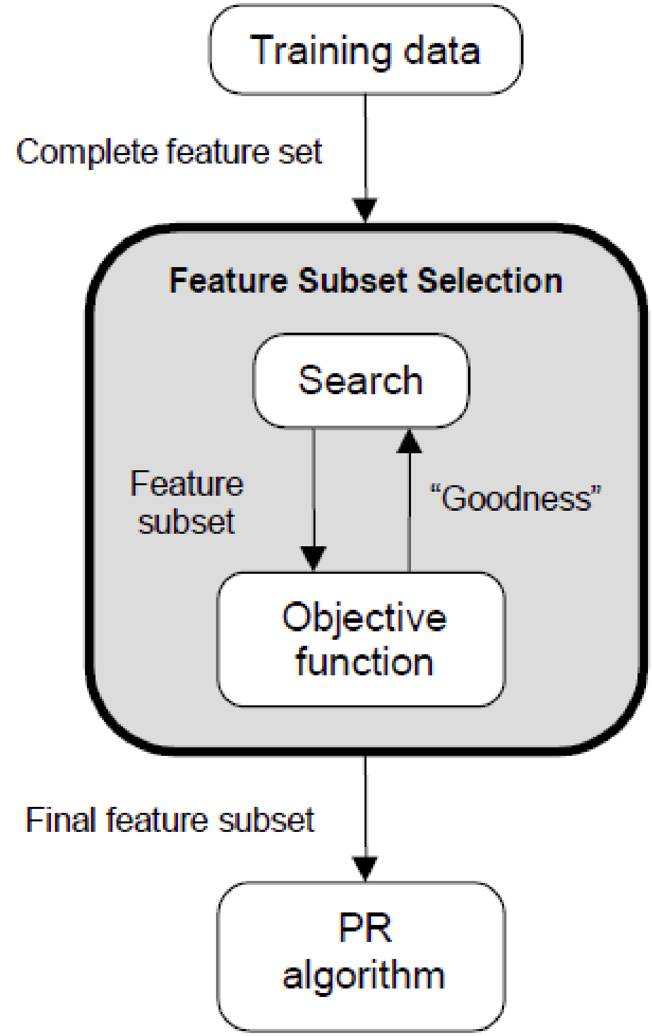
- Feature selection is an **optimization** problem.
- Step 1: **Search** the space of possible feature subsets.
- Step 2: Pick the subset that is optimal or near-optimal with respect to some **criterion**



Feature Selection Steps (Cont'd)

- Search strategies
  - –Exhaustive
  - –Heuristic

- Evaluation Criterion
  - Filter methods
  - Wrapper methods



Search Strategies

- Assuming d features, an exhaustive search would require:
- Examining all possible subsets of size m.
- Selecting the subset that performs the best according to the criterion.
- Exhaustive search is usually impractical.
- In practice, heuristics are used to speed-up search

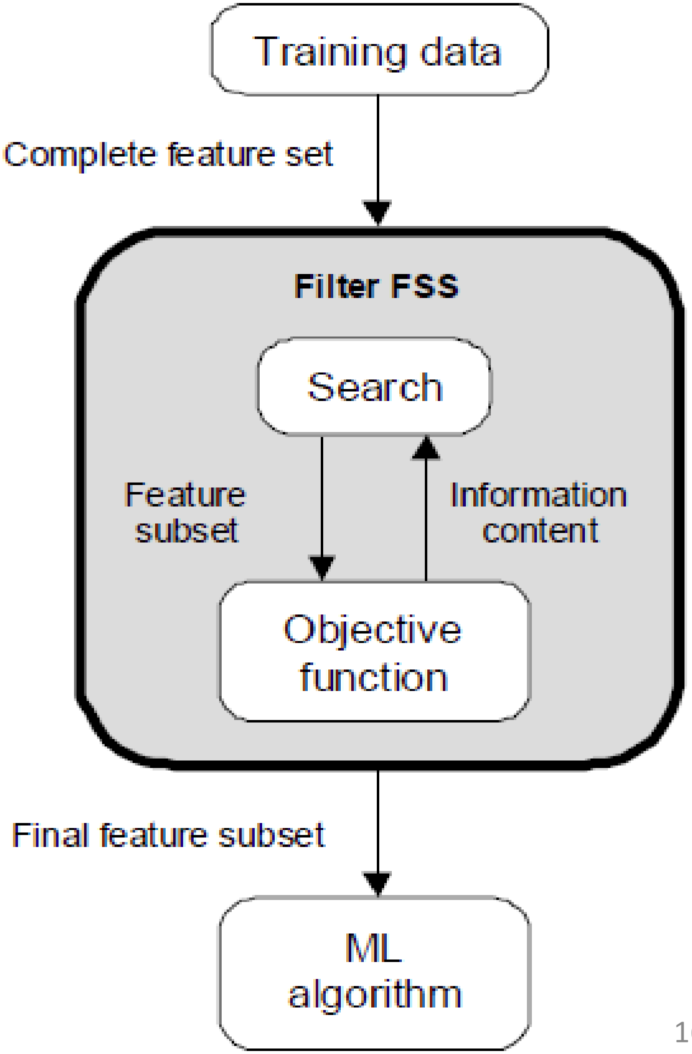
Evaluation Strategies

- **Filter Methods**

– Evaluation is **independent** of the classification method

– The criterion evaluates feature subsets based on their **class discrimination ability (feature relevance):**

- Mutual information or correlation between the feature values and the class labels

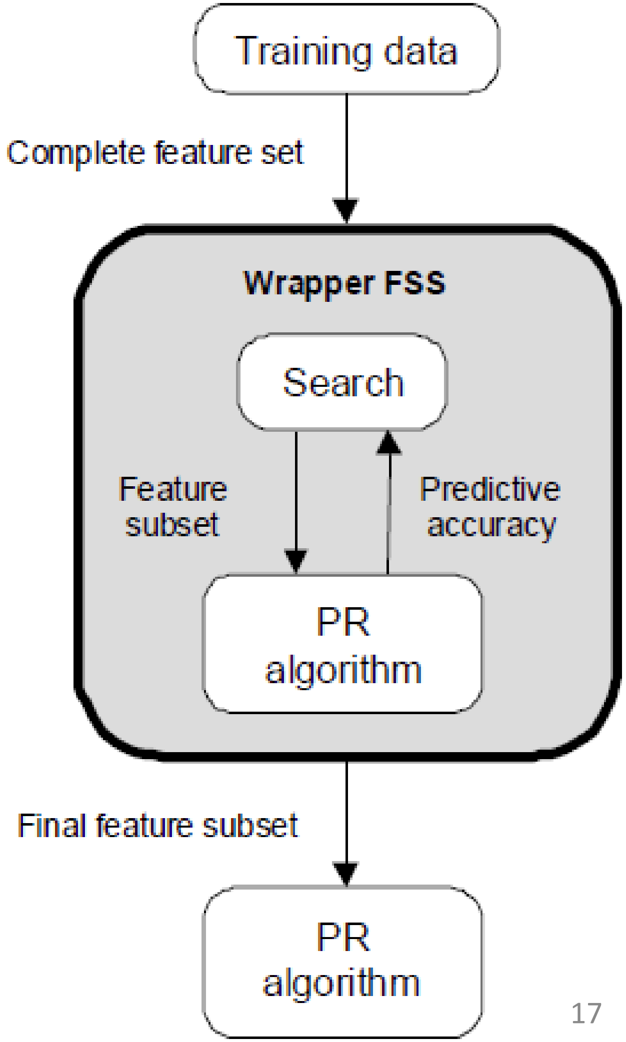


Evaluation Strategies

- **Wrapper Methods**

–Evaluation uses criteria **related** to the classification algorithm.

–To compute the objective function, a **classifier is built** for each tested feature subset and its generalization accuracy is estimated (e.g. cross-validation)



Evaluation Strategies

- Filter based
  - Chi-Squared
  - Information Gain
  - Correlation-Based Feature Selection, CFS
- Wrapper methods
  - recursive feature elimination
  - sequential feature selection algorithms
  - genetic algorithms

Feature Ranking

- Evaluate all d features individually using the criterion
- Select the top m features from this list.

Sequential forward selection (SFS) (heuristic search)

- First, the best **single** feature is selected
- Then, **pairs** of features are formed using one of the remaining features and this best feature, and the best pair is selected.
- Next, **triplets** of features are formed using one of the remaining features and these two best features, and the best triplet is selected.
- This procedure continues until a predefined **number of features are selected.**
- Wrapper methods (e.g. decision trees, linear classifiers) or Filter methods (e.g. mRMR) could be used
- **Sequential backward selection (SBS)**

Advantages of Dimensionality Reduction

- Helps in data compression, and hence reduced storage space.
- It reduces computation time.
- It remove redundant irrelevant features, if any
- Improves accuracy of Classification

Literature Review

- Implementation of the Principal Component Analysis onto High-Performance Computer Facilities for Hyperspectral Dimensionality Reduction: Results and Comparisons

- An Information Theory-Based Feature Selection Framework for Big Data Under Apache Spark

- Ultra High-Dimensional Nonlinear Feature Selection for Big Biological Data

Author	Dimensionality reduction algorithm	Parallel programming model	H/W configuration	Datasets
M. Yamada et al. [7]	Hilbert-schmidt independence criterion lasso with least angle regression	MapReduce framework (Hadoop and apache spark)	Intel xeon 2.4 GHz, 24 GB RAM (16 cores)	P53, Enzyme
Z. Wu et al.[12]	Principal component analysis	MapReduce framework (Hadoop and apache spark), MPI Cluster	Cloud computing (Intel Xeon E5630 CPUs(8 cores) 2.53 GHz, 5GB RAM, 292 GB SAS HDD), 8 slave(Intel Xeon E7-4807 CPUs (12 cores) 1.86 GHz)	AVIRIS cuprite hyperspectral datasets
S. Ramirez - Gallego et al.[2]	Minimum redundancy maximum relevance (mRMR)	MapReduce on apache spark, CUDA on GPGPU	Cluster (18 computing nodes, 1 master node) computing nodes: Intel Xeon E5-2620, 6 cores/processor, 64 GB RAM	Epsilon, URL, Kddb

Author	Dimensionality reduction algorithm	Parallel programming model	H/W configuration	Datasets
E. Martel et al. [4]	Principal component analysis	CUDA on GPGPU	Intel core i7-4790, NVIDIA 32 GB Memory, GeForce GTX 680 GPU	Hyperspectral data
J. Zubova et al. [13]	Random projection	MPI Cluster	-	URL, Kddb
L. Zhao et al. [5]	Distributed subtractive clustering	Cluster platforms	-	Economic data (China)
S. Cuomo et al.[8]	Singular value Decomposition	CUDA on GPGPU	Intel core i7, 8GB RAM, 2.8 GHz, GPU NVIDIA Quadro K5000, 1536 CUDA cores	-
W. Li et al. [9]	Isometric mapping (ISOMAP)	CUDA on GPGPU	Intel core i7-4790, 3.6 GHz, 8 cores, 32GB RAM, GPU Nvidia GTX 1080, 2560 CUDA cores, 8GB RAM	HIS datasets -Indian pines,Salinas , Pavia

Challenges

- Exponential growth in the dimensionality and sample size.
- So, the existing algorithms not always respond in an adequate same way when deal with this new extremely high dimensions.

Challenges

- Reducing data complexity is therefore crucial for data analysis tasks, knowledge inference using machine learning (ML) algorithms, and data visualization
- Ex. Use of feature selection in analyzing DNA microarrays, where there are many thousands of features, and a few tens to hundreds of samples

Challenges

- The time and space cost of learning feature selection/classification algorithms is large and grows fast as the variables increase.
- Large amounts of data are needed for its independence test which makes the problem harder.
- Classification of the high-dimensional data is challenging due to the curse of dimensionality, heavy computational burden and decreasing precision of algorithms

Challenges

- Feature selection methods –
  - full search of the feature space,
  - testing subsets of features
  - evaluating them to find the final solution. The search space consists of the combination of all possible subsets, which for a dataset with *n* features produces a feature space of size 2^n.
- For problems with a large number of features, finding an optimal subset of features is usually intractable **(NP-hard)**

Computing approaches

- Distributed implementation
- Shared memory implementation

Scaling up FS

- Distributed Feature Selection
- Allocating the learning process among several workstations
- Advantages:
  - Reduction in execution time
  - Resources sharing
  - Better performance
- Use of GPGPU

GPGPU Computing and MapReduce

- GPGPUs are shared memory model and MapReduce is distributed computing frameworks aim at different scaling purposes.
- Scalability approaches include vertical and horizontal scaling.
- **Vertical scaling: increasing the processing power, memory, and resources of a single node** in a system (GPGPUs)
- **Horizontal scaling: adds nodes to a system and distributes the workload across them** (Hadoop and Spark MapReduce frameworks)

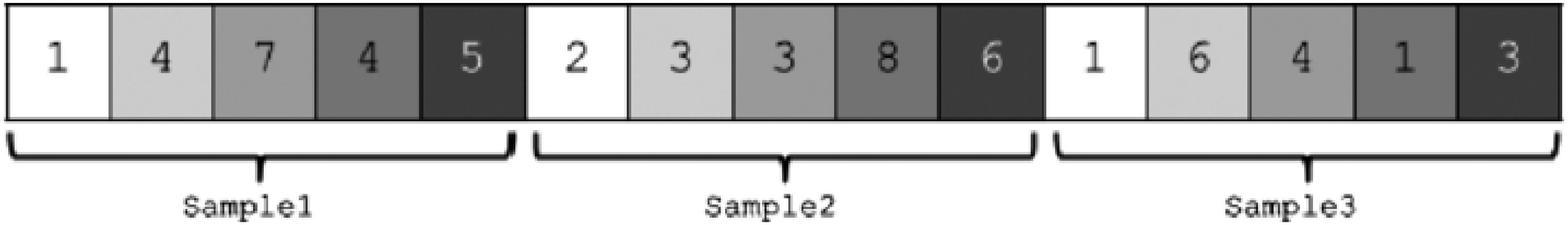
Drawbacks of MapReduce

- Not well suited for **iterative algorithms** due to performance impact of the launch overhead.
- The creation of the jobs, data transfers, and nodes synchronization through the network impose an overhead
- Jobs run in isolation which increases the difficulty of implementing shared communication between intermediate processes.
- it requires a fault tolerant distributed file system, such as the Hadoop distributed file system (HDFS).

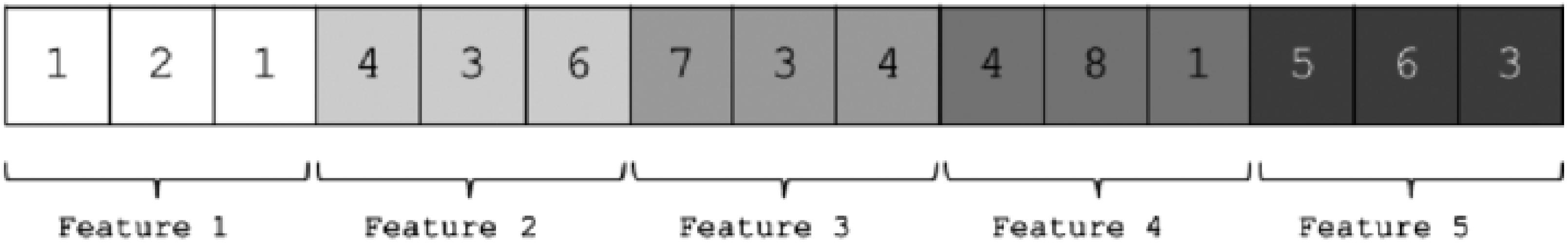
Advantage of GPGPU

- Parallel algorithms running on GPGPUs- achieve up to 100X speedup over similar CPU algorithms
- Very small kernel launch overhead, which permits executing parallel tasks with no delay and obtain almost instant results.
- Scalability to big data is limited due to the GPU memory capacity. Multi-GPU and distributed-GPU solutions combine hardware resources to scale-out to bigger data.

Optimizations: Data-access pattern

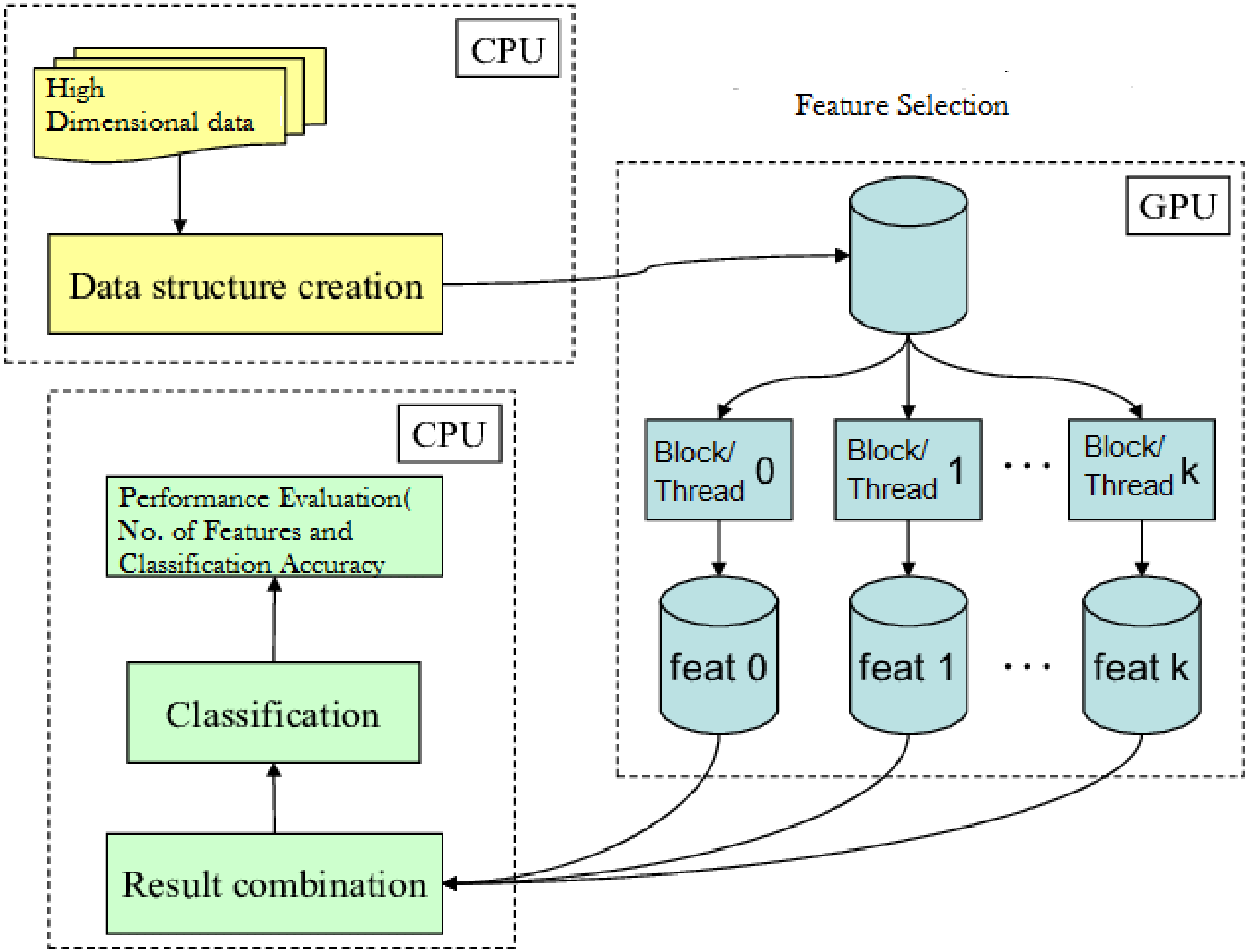


(a) Original data structure



(b) Refactored data structure

General Architecture



Conclusion

- Need to focus on important issues of high dimensionality problems and dimensionality reduction model on it
- High-performance computing approaches are best suitable for solving high dimensional data problems.
- Parallel processing techniques and computational power of multi-core and many-core architecture accelerates the performance for solving high dimensional problems.

References

[1] E. Martel, R. Lazcano, J. Lopez, D. Madronal, R. Salvador, et al., “Implementation of the Principal Component Analysis onto High-Performance Computer Facilities for Hyperspectral Dimensionality Reduction: Results and Comparisons”, Remote Sens, 10, 864, 2018

[2] S. Ramirez-Gallego et al, “An Information Theory-Based Feature Selection Framework for Big Data under Apache Spark”, IEEE Transactions on Systems, Man, and Cybernetics: Systems. 48, 9, 1441-1453, 2018

[3] T. Gao and Q. Ji, “Efficient Markov Blanket Discovery and Its Application”, IEEE Transactions on Cybernetics, vol. 47, no. 5, pp. 1169-1179, May 2017.

[4] A. L. Heureux, K. Grolinger, H. F. Elyamany and M. A. M. Capretz, “Machine Learning With Big Data: Challenges and Approaches”, IEEE Access, vol. 5, pp. 7776-7797, 2017

[5] L. Zhao, Z. Chen, et al., “Distributed feature selection for efficient economic big data analysis”, IEEE Transactions on Big Data , 2018

References

[6] L. Kasun, Y. Yang, G. Huang and Z. Zhang, “*Dimension Reduction With Extreme Learning Machine*”, IEEE Transactions on Image Processing, vol. 25, no. 8, pp. 3906-3918, 2016

[7] M. Yamada *et al*., "Ultra High-Dimensional Nonlinear Feature Selection for Big Biological Data," in *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 7, pp. 1352-1365, 1 July 2018.

[8] Cuomo, S., Galletti, A., Marcellino et al., “On gpu-cuda as preprocessing of fuzzy-rough data reduction by means of singular value decomposition”, Soft Computing 22(5), 1525-1532 , 2018

[9] Li, W., Zhang, L., Zhang, L., Du, B., “GPU parallel implementation of isometric mapping for hyperspectral classification”, IEEE Geoscience and Remote Sensing Letters 14(9), 1532{1536 (2017)

References

[10] T. Mingjie, Y. Yu, W. G. Aref, Q. Malluhi and M. Ouzzani, "*Efficient Parallel Skyline Query Processing for High-Dimensional Data*", IEEE Transactions on Knowledge and Data Engineering, 2018

[11] K. Passi, A. Nour and C. K. Jain, "*Markov blanket: Efficient strategy for feature subset selection method for high dimensional microarray cancer datasets*", IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2017

[12] Z. Wu, Y. Li, A. Plaza, J. Li, F. Xiao and Z. Wei, "*Parallel and Distributed Dimensionality Reduction of Hyperspectral Data on Cloud Computing Architectures*", IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, vol. 9, no. 6, pp. 2270-2278 , 2016

[13] J. Zubova, M. Liutvinavicius, O. Kurasova:, *"Parallel computing for dimensionality reduction",* Communications in Computer and Information Science, vol. 639. Springer, Cham , 2016

Thank You.!