New Subject Proposal
Dept. of Computer Science and Engineering
IIT Kharagpur.

Course Details:

Title: Scalable Data Mining
Credits: 3-0-0
Prerequisites: None
Offering Semester: Autumn
Course Level: PG elective

Motivation:

Consider the following problems:

- One is interested in computing summary statistics (word count distributions) for a set of words which occur in the same document in entire wikipedia collection (5 million documents). Naïve techniques, will run out of main memory on most computers.
- One needs an approximate count of the number of distinct IP addresses, for packets passing through a router. The algorithm that maintains a list of all distinct IP addresses will consume too much memory.
- One needs to train an SVM classifier for text categorization, with unigram features (typically ~10 million) for hundreds of classes. One would run out of main memory, if they store uncompressed model parameters in main memory.

In all the above situations, a simple data mining / machine learning task has been made more complicated due to large scale of input data, output results or both. In this course, we discuss *algorithmic techniques* as well as *software paradigms* which allow one to write scalable algorithms for the common data mining tasks.

Syllabus:

Big Data Processing: Motivation and Fundamentals. Map-reduce framework. Functional programming and Scala. Programming using map-reduce paradigm. Case studies: Finding similar items, Page rank, Matrix factorisation.

Stream processing: Motivation, Sampling, Bloom filtering, Count-distinct using FM sketch, Estimating moments using AMS sketch.

Finding similar items: Shingles, Minhashing, Locality Sensitive Hashing families.

Dimensionality reduction: Linear dimensionality reduction, PCA, SVD. Random projections, Johnson-Lindenstrauss lemma, JL transforms, sparse JL-transform. Random hashing, Clarkson-Woodruff algorithm.

Algorithms for distributed machine learning: Distributed supervised learning, distributed clustering, distributed recommendation. distributed optimization on Big data platforms**:** Gradient descent, spectral gradient descent, Stochastic gradient descent and related methods. ADMM and decomposition methods.

Overlap with other courses:

CS60017 SOCIAL COMPUTING
Nearest neighbor search problem. Shingling. Min-hashing.
Locality sensitive hashing, different distance measures.

CS61064 HIGH PERFORMANCE PARALLEL PROGRAMMING
Programming using map-reduce paradigm. System implementation details.

Interested Faculty:

Niloy Ganguly, Animesh Mukherjee, Pabitra Mitra, Pawan Goyal, Sourangshu Bhattacharya

Lecture Schedule:

1. **Big data paradigms and problems:** [6 lectures]
    a. Motivation and Fundamentals. Programming using map-reduce paradigm. System implementation details. [3 lectures]
    b. Case studies: Finding similar items, Page rank, Matrix factorisation. [3 lectures]
2. **Finding similar items: [4 lectures]**
    a. Nearest neighbor search problem. Shingling. Min-hashing. [2 lectures]
    b. Locality sensitive hashing, different distance measures [2 lectures]
3. **Stream computing: [6 lectures]**
    a. Introduction and examples. Sampling from stream. [2 lectures]
    b. Hashing and filtering, Bloom filter. [2 lectures]
    c. Counting distinct elements in a stream. FM sketch. Finding moments and AMS sketch. [2 lectures]
4. **Dimensionality reduction: [9 lectures]**
    a. Motivation and high-dimensional data. PCA and SVD. CUR decomposition. [3 lectures]
    b. Random projections, Johnson-lindenstrauss lemma, JL transforms. [3 lectures]
    c. Random hashing, Sparse-JL transform, Clarkson-Woodruff algorithm. [3 lectures]
5. **Algorithms for large scale machine learning: [14 lectures]**
    a. Large scale supervised learning problems, clustering, recommendation algorithms. [3 lectures]
    b. Large scale optimization, gradient based algorithms. Gradient descent, Spectral gradient descent. [3 lectures]
    c. Stochastic gradient descent, practical tricks, lock-free approach, stochastic averaged gradient. [5 lectures]

d. Constrained optimization, ADMM, consensus based distributed optimization. [3 lectures]

References:

1. **Mining of Massive Datasets.** 2nd edition. - *Jure Leskovec, Anand Rajaraman, Jeff Ullman.* Cambridge University Press. http://www.mmds.org/
2. **Data-Intensive Text Processing with MapReduce.** *Jimmy Lin and Chris Dyer.* Morgan and Claypool. http://lintool.github.io/MapReduceAlgorithms/index.html
3. **Hadoop: The definitive Guide.** *Tom White.* Oreilly Press.
4. **Distributed optimization and statistical learning via the alternating direction method of multipliers.** S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, 2011.
5. **Recent literature.**