CS60021: Scalable Data Mining

Sourangshu Bhattacharya

COURSE BACKGROUND

What is Data Mining?

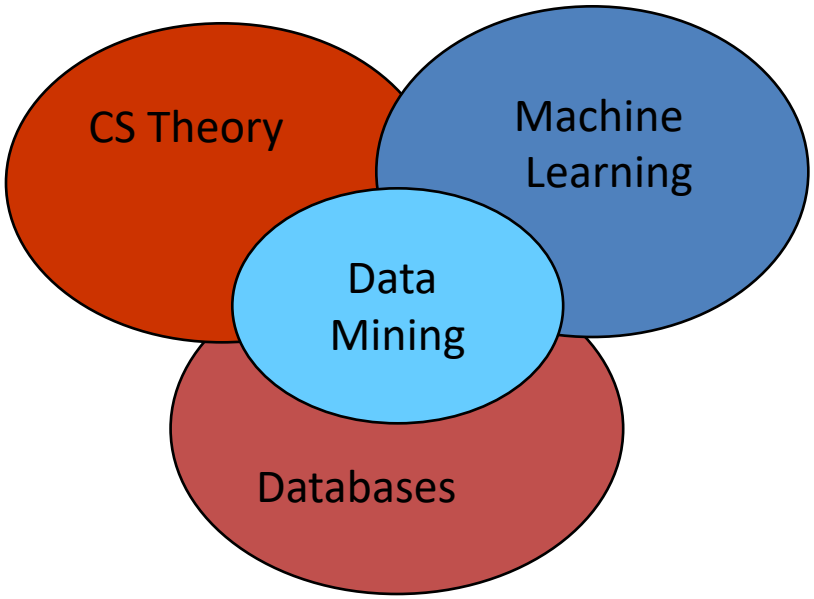
- **Given lots of data**
- **Discover patterns and models that are:**
  - **Valid:** should hold on new data with some certainty
  - **Useful:** should be possible to act on the item
  - **Unexpected:** non-obvious to the system
  - **Understandable:** humans should be able to interpret the pattern

  - A lot of the Data Mining Techniques are borrowed from Machine Learning / Deep Learning techniques.

Data Mining: Cultures

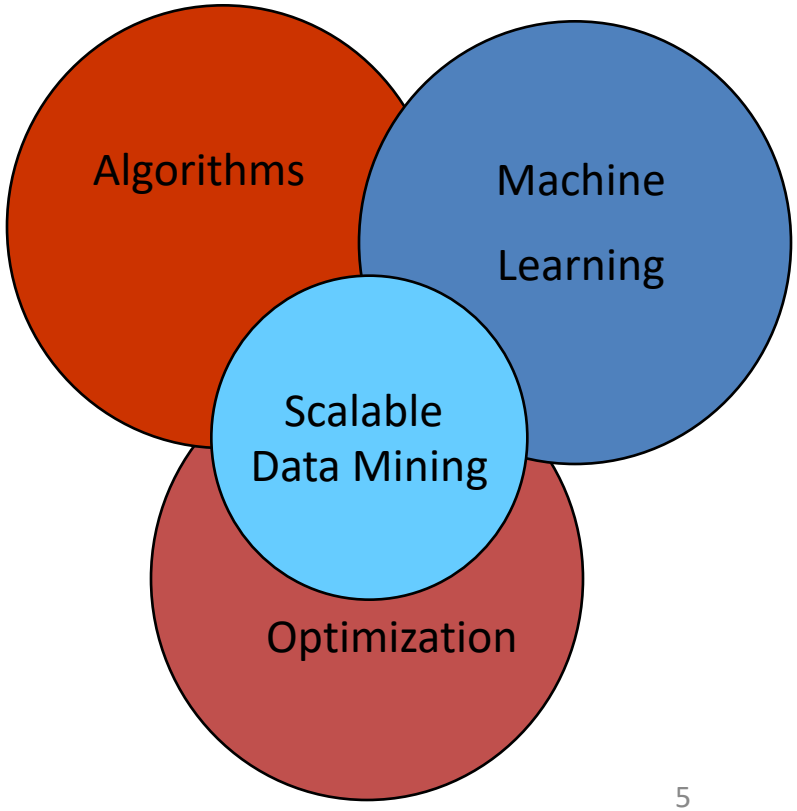
- **Data mining overlaps with:**
  - **Databases:** Large-scale data, simple queries
  - **Machine learning:** Small data, Complex models
  - **CS Theory:** (Randomized) Algorithms

- **In this course, we will explore scalable algorithms and systems for Data Mining.**



This Course

- **This class overlaps with machine learning, statistics, artificial intelligence, databases but more stress on**
  - **Algorithms**
    - **Online / Streaming**
    - **Optimization**
  - **Computing architectures**



Pre-requisites

- Algorithms.
- Machine Learning / Data Analytics / Information Retrieval.

- Linear Algebra
- Probability, statistics, calculus

EXAMPLE APPLICATIONS

Analytics: Word Count Distribution

- Motivation: Compute word-bigram count distribution for wikipedia corpus.
  - 5 million documents
  - 1.9 million unique words, ? bigrams
- Problem: Input, output and intermediate results are large.
  - Algorithm is simple. Implementation.
  - Spark: Map-reduce framework.
- Online version: Sketching algorithms for finding frequency of most frequent items:
  - Misra-Gries Sketching
  - Count-min and Count sketch.

Large Scale Machine Learning

- Train Massive deep learning models on massive datasets.

- Dataset too large:
  - Speed up train by speeding up optimization
  - Acceleration techniques
  - Distributed optimization.

- Model size too big:
  - Reduce redundant parameters using LSH.
  - LoRA parameter fine-tuning.

Nearest Neighbor Search (LSH)

- Active learning / Subset selection
  - Calculate pairwise similarity between examples
  - Select examples which provide highest improvement in loss function and are most similar to other non-selected examples.

- Compute similarity to all existing examples in dataset and pick the top ones.
  - Fast nearest neighbor seach.

SYLLABUS

Syllabus

- **Optimization and Machine learning algorithms:**
  - **Optimization algorithms**: Stochastic gradient descent, Variance reduction, Momentum algorithms, ADAM. Dual-coordinate descent algorithms.
  - **Algorithms for distributed optimization:** Stochastic gradient descent and related methods. ADMM and decomposition methods, Federated Learning.

Syllabus

- **Software paradigms:**
  - **Big Data Processing:** Motivation and Fundamentals. Map-reduce framework. Functional programming and Scala. Programming using map-reduce paradigm. Example programs.
  - **Deep Learning Frameworks (Pytorch):** Motivation, Computation graphs, Tensors, Autograd, Modules, Example programs.

Syllabus

- **Algorithmic techniques:**
  - **Subset Selection:** Formulations, Coresets, Submodular optimization, Orthogonal Matching Pursuit, Online Convex-optimization.
  - **Finding similar items:** Shingles, Minhashing, Locality Sensitive Hashing families.
  - **Stream processing:** Motivation, Sampling, Bloom filtering, Count-distinct using FM sketch, Estimating moments using AMS sketch.

Syllabus (New addition)

- **LLM finetuning:**
  - Transformer architecture, LORA updates., Flash attention.
  - Long context modelling.

COURSE DETAILS

Venue

- Classroom: NR - 121
- Slots:
  - Monday (11:00 – 11:55)
  - Tuesday (8:00 – 9:55)
- Website: https://cse.iitkgp.ac.in/~sourangshu/coursefiles/cs60021_2025a.html
- Moodle (for assignment submission): https://moodlecse.iitkgp.ac.in/moodle/

Teaching Assistants

- Saptarshi Mondal
- Suman Kumar Bera
- Vaishnovi Arun

Evaluation

- Grades:
  - Midsem, Endsem: 60 - 70
  - Class Test + Assignments: 30 – 40
  - Term Project (optional): 10
- Assignments: 2 – 3
- Both Term Project and assignment will require you to write code.

Tentative Schedule (Changeable)

Week	Dates	Topics	Class Test	Assignment
1	28/7, 29/7	Introduction to DM, ML, Stochastic gradient descent.		
2	4/8, 5/8	SGD convergence rate, Accelerated SGD		Assignment 1 (SGD experiment)
3	11/8, 12/8	SGD variance reduction		
4	18/8, 19/8	Distributed Optimization, ADMM	CT1 (SGD + Variants)	
5	25/8, 26/8	Map-reduce framework, Hadoop / Spark		Assignment 2 (Pyspark + Pytorch)
6	1/9, 2/9	Spark / DL frameworks (GPU)		
7	8/9, 9/9	DL frameworks, Subset selection	CT2 (Dist. Opt. + Hadoop/Spark)	
8	15/9, 16/9	Subset Selection		
		Mid-sem		
		Autumn Break		
9	6/10, 7/10	Approximate NNS - Locality Sensitive Hashing		Assignment 3 (ANNS)
10	13/10, 14/10	ANNS - HNSW		
11	20/10, 21/10	Streaming - Sampling, Set Membership, Distinct Count		
12	27/10, 28/10	Streaming - Frequency Counting	CT3 (ANNS + Sampling)	
13	3/11, 4/11	Transformer Models, LoRA Fine-tuning, Flash Attention		
14	10/11, 11/11	Long context modelling		

Attend classes regularly!

Don’t count on material that you can just “cover” before the exam and get a decent grade.

THANKS !