CIS 8000: AI SYSTEMS SEMINAR

# schedule

## presentation schedule

although this is a seminar, the first few weeks will include cross-cutting intro/thematic talks to help set the stage for the semester. because we expect students to have a specialization in a sub-area already, the intro lectures will mostly focus on broad ideas, establish common ground, and point to resources. students can use these independently to fill in the gaps based on their own interests, perhaps in consultation with instructors/their advisors.

the (student) seminar presentation schedule will comprise a subset of papers from the list of papers below, determined after students are matched with their topics + assigned seminar dates. students will present a seminar-style lecture on their assigned paper(s), and lead a class discussion following it.

please register your preferences for topics using the form on slack by 11:59pm on 2/23. we'll use this to match students with papers/topics. if you do not submit a preference, you will be assigned a topic based on availability. we can't guarantee that we will be able to accommodate all preferences, but we will do our best to match students with topics that are of interest to them. we reserve the right to make changes to the schedule and seminar assignments based on student preferences and availability.

the presentation schedule will be maintained on under the "resources" tab on slack.

## a tentative list of topics + readings

this is an evolving list of papers that we plan to cover, roughly covering a week of material per "block" - mostly for students to read ahead and get a general idea of the topics, or dive deeper into specific topics. please keep an eye out for updates!

| THEME | TOPICS | SELECTED PAPERS/READINGS/ SYSTEMS |
|---|---|---|
| **Analytical systems fundamentals** | DB architecture<br>Access paths, storage, encodings | **Architecture of a Database System** (Hellerstein+, '07)<br>**Red Book Ch. 4** (Stonebraker) |
| | Query planning & execution | **Access path selection in a relational database management system** (Selinger+, SIGMOD '79)<br>**A Deep Dive into Common Open Formats for Analytical DBMSs** (Liu+, VLDB '23)<br>**An Empirical Evaluation of Columnar Storage Formats** (Zeng+, VLDB '23) |
| | Indexing & search 1 (Exact search)<br>Optimizers | **The Cascades Framework for Query Optimization** (Graefe+, IEEE Data Eng. Bull. '95)<br>**Apache Calcite: A Foundational Framework for Optimized Query Processing Over Heterogeneous Data Sources** (Begoli+, SIGMOD '18) |
| **Modern systems** | Data warehouses<br>Streaming engines<br>Lakehouses | **The Snowflake Elastic Data Warehouse** (Dageville+, SIGMOD '16)<br>**The Stratosphere platform for big data analytics** (Alexandrov+, SIGMOD '14)<br>**Apache Flink** (various |

| THEME | TOPICS | SELECTED PAPERS/READINGS/ SYSTEMS |
|---|---|---|
| | | publications, documentation)<br>**Apache Spark: a unified engine for big data processing**<br>(Zaharia+, SIGMOD '16 - article)<br>**Spark SQL: Relational Data Processing in Spark**<br>(Ambrust+, SIGMOD '15)<br>**Delta Lake: High-Performance ACID Table Storage over Cloud Object Stores**<br>(Ambrust+, VLDB '20)<br>**Lakehouse: A New Generation of Open Platforms that Unify Data Warehousing and Advanced Analytics**<br>(Armburst+, CIDR '21)<br>**Photon: A Fast Query Engine for Lakehouse Systems**<br>(Behm+, SIGMOD '22) |
| | Big data systems | **Dremel: Interactive Analysis of Web-Scale Datasets**<br>(Melnik+, VLDB '10)<br>**Dremel: A Decade of Interactive SQL Analysis at Web Scale**<br>(Melnik+, VLDB '20)<br>**BigLake: BigQuery's Evolution toward a Multi-Cloud Lakehouse**<br>(Levandoski+, SIGMOD '24)<br>**Spanner: Google's Globally-Distributed Database**<br>(Corbett+, OSDI '12) |

| THEME | TOPICS | SELECTED PAPERS/READINGS/ SYSTEMS |
|---|---|---|
| | ML for systems<br>Optimization techniques: bandits, mathematical programming (linear, integer, dynamic)<br>RL (part 1): MDPs, Regret, Q error...<br>Misc.: Heuristic/ metaheuristic algorithms (MCTS + DRL, ...), weak supervision | **The Case for Learned Index Structures**<br>(Kraska+, SIGMOD '18)<br>**Neo: A Learned Query Optimizer**<br>(Marcus+, VLDB '19)<br>**SageDB: A Learned Database System**<br>(Kraska+, CIDR '19)* |
| **Vectors, Embeddings** | Vector search<br>Embedding models<br>Indexing & search 2<br>Index optimization: Quantization, parameter tuning, rebuilding indices, dim. reduction (PCA, t-SNE)<br>Hybrid, Multivector, Cross-modal retrieval<br>Retrieval Augmented Generation | **Efficient Estimation of Word Representations in Vector Space**<br>(Mikolov+, ICLR '13)<br>**Dense Passage Retrieval for Open-Domain Question Answering**<br>(Karpukhin+, EMNLP '20)<br>**ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT**<br>(Khattab+, SIGIR '20) |
| **AI/ML systems** | Computation and Numerics<br>ML compilation overview<br>Computational graphs<br>Automatic differentiation<br>GLO<br>Kernel fusion | **TASO: Optimizing Deep Learning Computation with Automatic Generation of Graph Substitutions**<br>(Jia+, SOSP '19) |

| THEME | TOPICS | SELECTED PAPERS/READINGS/ SYSTEMS |
|---|---|---|
| | Distributed ML<br>Data parallelism (Parameter servers, All-reduce)<br>Model, Pipeline parallelism | **GPipe: Efficient Training of Giant Neural Networks Using Pipeline Parallelism**<br>(Huang+, NIPS '19)<br>**Scaling Distributed Machine Learning with the Parameter Server**<br>(Li+, OSDI '14)<br>**Large Scale Distributed Deep Networks**<br>(Dean+, NIPS '12) |
| | Hardware for AI<br>Architecture-aware design<br>GPU and systems architecture intro | **QLORA: Efficient Finetuning of Quantized LLMs**<br>(Dettmers+, NIPS '23)<br>**FlashAttention: Fast and Memory-Efficient Exact Attention with IO-Awareness**<br>(Dao+, NIPS '22) |
| | Inference + serving at scale (1)<br>Part 1: | **DeepSpeed Inference: Enabling Efficient Inference of Transformer Models at Unprecedented Scale**<br>(MSR)<br>**AlpaServe: Statistical Multiplexing with Model Parallelism for Deep Learning Serving**<br>(Li+, OSDI '23)<br>**Orca: A Distributed Serving System for Transformer-Based Generative Models** |

| THEME | TOPICS | SELECTED PAPERS/READINGS/ SYSTEMS |
|---|---|---|
| | | (Yu+, OSDI '22)<br>**Ray: A Distributed Framework for Emerging AI Applications** |
| **Systems for Reasoning** | Reasoning frameworks and techniques:<br>CoT, ReAct, Tree of Thoughts, Self-reflection, RL for reasoning (Q*, ...)<br>Additional topics: Self and path consistency, Neuro-symbolic systems, Diffusion models | **Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer**<br>(Shazeer+, ICLR '17)<br>**Chain-of-Thought Prompting Elicits Reasoning in Large Language Models**<br>(Wei+, NIPS '22)<br>**ReAct: Synergizing Reasoning and Acting in Language Models**<br>(Yao+, ICLR '23)<br>**Tree of Thoughts: Deliberate Problem Solving with Large Language Models**<br>(Yao+, NIPS '23)<br>**Hierarchical Reasoning Model**<br>(Sapient Intelligence, '25)<br>**On the role of planning in model-based deep reinforcement learning**<br>(Hamrick+, '21) |
| **Compound AI systems** | Prompt optimization frameworks: GRPO, MIPRO, GEPA<br>Orchestrated systems: Deep research, ... | **The Shift from Models to Compound AI Systems**<br>(BAIR)<br>**AlphaEvolve: A coding agent for scientific and algorithmic discovery**<br>(Google DeepMind) |

| THEME | TOPICS | SELECTED PAPERS/READINGS/ SYSTEMS |
|---|---|---|
| | | DSPy: Compiling Declarative Language Model Calls into Self-Improving Pipelines (Khattab+, ICLR '24)<br>Automated Design of Agentic Systems (Hu+, ICLR '25) |
| | Inference + serving at scale (2) | vLLM: Efficient Memory Management for Large Language Model Serving with PagedAttention<br>CacheGen: KV Cache Compression and Streaming for Fast Large Language Model Serving<br>CacheBlend: Fast Large Language Model Serving for RAG with Cached Knowledge Fusion<br>Autellix: An Efficient Serving Engine for LLM Agents as General Programs (Luo+, '25) |
| **AI-native analytical systems** | CHASE-SQL: Multi-Path Reasoning and Preference Optimized Candidate Selection in Text-to-SQL<br>DocETL: Agentic Query Rewriting and Evaluation for Complex Document Processing (Shankar+, '24)<br>LOTUS: Enabling Semantic Queries with LLMs Over Tables of Unstructured and Structured Data (Patel+, '24)<br>A Declarative System for Optimizing AI Workloads (Liu+, CIDR | |

| THEME | TOPICS | SELECTED PAPERS/READINGS/ SYSTEMS |
| --- | --- | --- |
| | ’24) | |

## important dates

- project proposals: Mar 6
- project milestone: Apr 10
- project submission: Apr 29
- presentation preferences: Feb 23
- short summaries: before the start of class (1:45p) on the day of the seminar

CIS 8000: AI SYSTEMS SEMINAR

# policies

## grading

| COMPONENT | WEIGHT |
|---|---|
| short summaries | 10% |
| presentation + seminar | 20% |
| seminar report | 15% |
| project<br>• proposal: 5%<br>• milestone: 20%<br>• presentation: 5%<br>• final submission: 70% | 35% |
| attendance and participation | 20% |

## attendance and participation

all classes will be held in person barring extreme circumstances. since seminar courses are centered around discussion, the course will not be recorded so that students do not need to fear their questions or comments being immortalized.

as a result, (1) participation and discussion is highly encouraged, and we suggest you

bring a pen(cil) and paper to take notes and leave your electronics in your bag (still bring them – some lectures will have online interactive components), although this is not mandatory and you are welcome to take electronic notes if you prefer. (2) there is essentially no way to "make ud class, so i will expect students miss no more than 4 class periods throughout the semester.

*policy adopted from ryan's offering of cis6500.*

## short summaries

when it is *not* your turn to present, you will submit a ~500 word summary that thematically connects and discusses the papers chosen for the seminar. summaries are graded on completion. consider only selected "main" papers for the summary (marked with an asterisk * on the presentation schedule on slack).

your summary is due before the start of class (1:45p) on the day of the seminar. late summaries will receive no credit. *you can turn in your summaries for the first week of the seminar until 11:59pm on 2/23.*

the submission form can be found here.

## research, academic integrity and ai policies

due to the unique positioning of this class, which places it closely with publishable research undertakings, we take academic integrity very seriously. you are responsible for knowing penn's code of academic integrity. we will aggressively pursue all cases of suspected violations.

unless otherwise noted, all assignments are individual work.

**ai tools:** judicious ai tool use is acceptable and encouraged for learning concepts and literature review in assignments. ideally, ai tools should not be used as a substitute for thinking or completing assignments end-to-end. you can also try using it to automate boilerplate or simple code that is low stakes.

however, for any work you turn in or present towards a grade, we will assume a yardstick where you are the sole creator of your work. that is, you should both understand this

work thoroughly and be fully responsible for what you turn in, since it is ”your work”.

more concretely, you’re expected to be able to explain all important details of the work you turn in, except for only very minor boilerplate/setup parts. this ensures that you can engage in discussions and contribute to projects effectively, justifying and iterating on work. this also prevents you from potentially misleading your peers, advisors, and other stakeholders (even if unconsciously) about your progress, work, and decision-making.

moreover, this also means that should your work demonstrate signs of violations resulting from inadvertent careless ai tool use, you (and not the ai tool) would be held solely responsible. for example, if an experimental result is found to be fabricated, a claim of an ai tool implementing the code will still land you in grave danger.

finally, certain assignments may contain a section that requires you to disclose any ai tool use. you must answer this honestly and transparently.

you should also save your conversations with ai tools for future reference.