Syllabus for CSE 293:

Stream Processing Systems (Winter 2023)

(Last Revision: Jan 1, 2023)

Instructor Information

Instructor Name:	Prof. Liting Hu
Course Hours:	Tuesday / Thursday, 9:50 AM - 11:25 AM (5 Unit Course)
Course Location:	Social Sciences 2 Room 363
Office Hours:	Friday 2:00 PM - 3:00 PM on Zoom (or by appointment)
Email:	liting@ucsc.edu

General Information

Description

Welcome to advanced topics in computer science & engineering: stream processing systems! Stream processing is one of the most popular topics in the big data area. It is used to query continuous streams of data within a small time period varying from a few milliseconds to minutes. For that reason, stream processing technology has become a critical building block of many real-time applications, such as fraud detection, smart car, smart home, smart grid, traffic monitoring, e-commerce promotion, and advertising.

This seminar-style course will cover advanced topics in stream processing studies, including real-time event-based stream processing, batch and mini-batch processing, graph-based stream processing, and streaming SQL. Example platforms include Hadoop, MapReduce, Spark Streaming, Storm, GraphX, Kafka, Flink, and the like. This course will center around four basic entities: learning stream processing concepts, reading, critiquing, and discussing research papers, student presentations, and course projects.

This course will be highly beneficial to students who are interested in understanding how to handle large-scale data streams in real-time or with low latency expectations, to whose research interests are in large-scale data processing, real-time analysis, event processing, cloud computing, and big data methodology, and also to those who may need the first-hand knowledge of industry stream processing applications and platforms.

Pre- &/or Co-Requisites

You are expected to have familiarity with core computer systems and programming concepts.

- (Expected) CSE130: Principles of Computer Systems Design/ CSE231: Advanced Operating Systems or equivalent
- (Expected) CSE138: Distributed Systems/CSE232: Distributed Systems or equivalent
- (Helpful But Not Expected) CSE111: Advanced Programming/CSE117: Open Source Programming or equivalent

Course Goals and Learning Outcomes

The goals of this course are to expand your understanding of real-world systems and prepare you to:

- Understand the state-of-the-art in real-time event-based stream processing, batch and mini-batch processing, graph-based stream processing, and streaming SQL.
- Interpret and critically analyze research papers on solving problems with data-intensive workloads and applications.
- Apply stream processing concepts and build stream processing systems to solve high-velocity and high-variety data problems.

Course Requirements & Grading

As a seminar-style course, the grading will be based on reading and presenting the assigned paper readings, engaging with class discussion, and conducting and presenting a research-oriented final project.

Assignment	Weight	Description
Participation	10%	Attend and engage with class meetings (ask and answer questions, provide comments)
Discussion Lead	20%	For one class during the semester, prepare and present a 40-minute presentation summarizing the class's reading, and help lead the class discussion (20 minutes) For each class, 1 student will serve as the discussion lead
Paper Summaries **(Written)**	20%	For each paper, submit a brief paper summary, comments, and questions. In total, 2 summaries can be skipped during the semester. If you lead a discussion, you can skip an additional summary (for a total of 3 summary skips)
Final Project	50% Total	Semester-long research project (group of 2 students).
-Project Proposal	5%	Write and present a project proposal (due 2/10, subject to change).
-Project Presentation and Demo	20%	Present a talk and a demo on your final project (starting 3/7).
-Project Writeup	25%	Submit a research-style paper on your final project (due 3/24).

Extra credit will be given to students who finish the project's challenging tasks.

Grading Scale

The course will not be graded on a curve. Your final grade will be assigned as a letter grade according to the following scale:

A	90-100%
B	80-89%
C	70-79%
D	60-69%
F	0-59%

Course Materials

Course TextBook

There is no required course textbook. For optional supplemental or background reading, we recommend:

Streaming Systems: The What, Where, When, and How of Large-Scale Data Processing, Reuven Lax, Slava Chernyak, and Tyler Akidau,2018.

Course Website and Other Classroom Management Tools

Public Course Website: https://canvas.ucsc.edu/courses/60462

We will use Canvas for course organization as well as for discussion and classwide communication.

Course Expectations & Guidelines

Attendance and/or Participation

As this is a seminar-style course, attendance and engagement with class discussions are vital. While I will not be taking explicit attendance in each class, I will track those who speak and participate. Participation is 10% of your grade.

In addition, students will be expected to present presentations of 1-2 papers throughout the semester and lead the associated class's discussion. This discussion leading accounts for an additional 20% of your grade.

You may need to miss a class for legitimate reasons (e.g., sick, onsite interviews).

Collaboration & Group Work

Paper summaries should be written and submitted separately by each student, but discussion about papers in groups is allowed and encouraged within reason (e.g., students should still submit distinct paper comments).

Final projects can be done in groups of 2 people (depending on class size).

Extensions & Late Assignments

Assignments are due at the time listed in the schedule. There are no undocumented exceptions. If you have an emergency situation or a school sanctioned exception, please contact me before the due date so we can adjust your assignment deadlines (some documentation may be needed).

Student Use of Mobile Devices in the Classroom

As this course is heavily based on class discussion, I ask that you stay engaged and limit your mobile device usage during class.

Academic Integrity

The Baskin School of Engineering has a zero tolerance policy for any incident of academic dishonesty.

If cheating occurs, consequences may range from getting zero on a particular assignment to failing the course. In addition, every case of academic dishonesty is referred to the students' college Provost, who sets in motion an official disciplinary process. Cheating in any part of the course may lead to failing the course, suspension, or dismissal from the Baskin School of Engineering, or from UCSC.

What is cheating? In short, it is presenting someone else's work as your own. Examples would include copying written homework solutions from another student or allowing your own work to be copied. Sharing any kind of information on an exam would also be considered cheating. You may discuss your homework solutions with fellow students, but your collaboration must be at the level of ideas only. You may freely give and receive help with any example discussed in class, in the text, or in one of the handouts. However, you may not share in the act of writing your solutions to homework problems.

Please see the following links for the official UCSC policies on Academic Misconduct for

Graduate Students: https://www.ucsc.edu/academics/academic-integrity/

Undergraduate Students: https://www.ue.ucsc.edu/academic_misconduct

Paper Summaries

At the beginning of each class, each student (except for those that are presenting that day) will need to turn in a one-page review of the assigned papers readings for that day. Paper summaries will be submitted to Canvas (pdf). Use "Paper Name" + "Your Name" as the name of the pdf.

Note that this review will only need to cover the mandatory readings, but students are encouraged to peruse the supplemental readings. Be sure to include your name at the top of the paper.

Each review must include the following information:

- An overview of the main idea and contributions. (One paragraph)
- Three positive comments about the paper. (One sentence each)
- Three negative comments about the paper. (One sentence each)
- Technical/research discussion questions unanswered about the paper.

Make sure to have a copy of your reports handy during class to help guide class discussion and your participation.

Paper Presentations

Each student will choose one date from the course schedule and present the papers assigned on those days to the class. This talk is supposed to be an in-depth description and analysis of the papers. It should be 60 minutes long (approximately 40 minutes per paper) and then with 15-20 minutes remaining for questions. The format of the talk should be similar to a conference presentation. Because it is the responsibility of the presenter to teach the class about the papers, he/she will be expected to know and understand all the aspects of the material. Thus, it is important to be prepared. This may require you to do additional background reading.

On the day of the presentation, the presenter does not need to submit a reading report. The presenter should send his/her presentation ahead of time so it can be posted on the class Web page at Canvas. The presentation should include discussion points/questions to encourage in-class discussion/participation. If you have questions regarding the content of your assigned papers, you should arrange to meet with the instructor well in advance of your talk date.

WARNING: It is acceptable for students to use information and content (e.g., images and graphics) found on the Internet but the original source must be properly attributed/cited. No credit will be given for presentations without proper citations.

Course Schedule

Below is the current course schedule, which is subject to some change as the semester progresses. A paper presentation demo can be a recorded demo. A project's demo needs to be a live demo.

===== Week 1 =====

1/10: First Class - Introduction to Stream Processing Systems

No readings, will cover topics including stream processing motivation and stream processing models (streams, windows, operators).

1/12: Second Class - Introduction to Stream Processing Systems

No readings, will cover stream processing introduction including data stream management systems (DSMS), batch processing systems, micro-batch stream processing systems, stream processing systems with distributed dataflows, and stateful stream processing systems.

===== Week 2 =====

1/17: Bach Processing Systems

Reading: The Hadoop Distributed File System
https://storageconference.us/2010/Papers/MSST/Shvachko.pdf

The student who presents this paper is required to show a demo about HDFS (15 minutes for paper presentation and 25 minutes for demo).

1/19: Bach Processing Systems

Reading: Improving MapReduce Performance in Heterogeneous Environments
https://cs.stanford.edu/people/matei/papers/2008/osdi_late.pdf

The student who presents this paper is required to show a demo about MapReduce (15 minutes for paper presentation and 25 minutes for demo).

===== Week 3 =====

1/24: Micro-batch Stream Processing Systems

Reading: Discretized Streams: Fault-Tolerant Streaming Computation at Scale
https://cs.stanford.edu/people/matei/papers/2013/sosp_spark_streaming.pdf

The student who presents this paper is required to show a demo about Spark (15 minutes for paper presentation and 25 minutes for demo).

1/26: Event-based Stream Processing Systems

Reading: Kafka: a Distributed Messaging System for Log Processing
https://notes.stephenholiday.com/Kafka.pdf

The student who presents this paper is required to show a demo about Kafka (15 minutes for paper presentation and 25 minutes for demo).

===== Week 4 =====

1/31: Event-based Stream Processing Systems

Reading: Storm @Twitter
https://cs.brown.edu/courses/csci2270/archives/2015/papers/ss-storm.pdf

The student who presents this paper is required to show a demo about Storm (15 minutes for paper presentation and 25 minutes for demo).

2/2 Event-based Stream Processing Systems

Reading: ELF: Efficient Lightweight Fast Stream Processing at Scale
https://people.ucsc.edu/~lhu82/doc/elf.pdf

===== Week 5 =====

2/7 Stateful Stream Processing Systems

Reading: FP4S: Fragment-based Parallel State Recovery for Stateful Stream Applications
https://people.ucsc.edu/~lhu82/doc/fp4s.pdf

2/9 Stateful Stream Processing Systems

Reading: Meces: Latency-efficient Rescaling via Prioritized State Migration for Stateful Distributed Stream Processing Systems
https://www.usenix.org/system/files/atc22-gu-rong.pdf

===== Week 6 =====

2/14 Graph Stream Processing Systems

Reading: Pregel: A System for Large-Scale Graph Processing
https://15799.courses.cs.cmu.edu/fall2013/static/papers/p135-malewicz.pdf

The student who presents this paper is required to show a demo about Pregel (15 minute for paper presentation and 25 minutes for demo).

2/16 Graph Stream Processing Systems

Reading: GraphX: Graph Processing in a Distributed Dataflow Framework
https://www.usenix.org/system/files/conference/osdi14/osdi14-paper-gonzalez.pdf

The student who presents this paper is required to show a demo about GraphX (15 minutes for the paper presentation and 25 minutes for demo).

===== Week 7 =====

2/21 Graph Stream Processing Systems

Reading: GraphLab: A New Framework For Parallel Machine Learning
https://arxiv.org/ftp/arxiv/papers/1408/1408.2041.pdf

The student who presents this paper is required to show a demo about GraphLab (15 minutes for paper presentation and 25 minutes for demo).

2/23 Graph Stream Processing Systems

Reading: Low-Latency Graph Streaming using Compressed Purely-Functional Trees
https://arxiv.org/pdf/1904.08380.pdf

The student who presents this paper is required to show a demo about ASPEN (15 minutes for paper presentation and 25 minutes for demo).

===== Week 8 =====

2/28 Graph Stream Processing Systems

Reading: Practice of Streaming Processing of Dynamic Graphs: Concepts, Models, and Systems
https://arxiv.org/pdf/1912.12740.pdf

3/2 Edge Stream Processing Systems

Reading: Bladerunner: Stream Processing at Scale for a Live View of Backend Data Mutations at the Edge
https://dl.acm.org/doi/pdf/10.1145/3477132.3483572

===== Week 9 =====

3/7 Edge Stream Processing Systems

Reading: DART: A Scalable and Adaptive Edge Stream Processing Engine
https://people.ucsc.edu/~lhu82/doc/dart.pdf

3/9 Project Presentation

===== Week 10 =====

3/14 Project Presentation

3/16 Project Presentation