Course Prefix and Number: CS7652

Course Name: Natural Language

Section Number: A

Credit Hours: 3

Semester and Academic Year: Fall 2026

Instructor Information

Instructor: Alan Ritter

Email: alan.ritter@cc.gatech.edu

General Course Information

Description

Pre-Trained Language Models have led to a paradigm shift in the field of Natural Language Processing. These models are leading to new applications, state-of-the-art results on almost every existing language-related task, in addition to results that are surprisingly robust across tasks, domains, and data that is constantly evolving over time, without requiring task-specific labels.

This is an advanced graduate-level class that will cover recent developments in LLMs. Students will present and discuss recent research papers, and carry out a research project. Students are expected to have taken a prior course on Natural Language Processing or Deep Learning, and to be familiar with deep learning models, such as Transformers.

By the end of the course, students should expect to be up to date on recent advances in LLMs. Students will also learn how to read and understand cutting-edge research on LLMs, and improve their technical presentation skills.

Pre-Requisites

This is an advanced AI course on Large Language Models. Modern LLMs are heavily based on Machine Learning and Deep Learning. To succeed in this class, you will need a very strong math and programming background that can sufficiently make you feel at ease in the machine learning class (CS 4641/764). **Students should have completed CS**

4641/7641 in a prior semester; taking it concurrently is not sufficient preparation. The course assumes mastery of mathematical concepts in probability, linear algebra, and multivariable calculus. You should also be comfortable working on medium-size software projects, learning and using new libraries (in particular, PyTorch) quickly, and **debugging complex code when error messages are incomplete or absent, including writing your own unit tests, inspecting intermediate variables, and systematically isolating and diagnosing issues, etc.**

Course Learning Outcomes

- Understand the core ideas behind modern large language models, including pretraining objectives, scaling, Transformer-based architectures, instruction tuning, alignment, and inference-time prompting.
- Ability to critically read, analyze, and synthesize recent research papers on large language models, including identifying key technical contributions, assumptions, limitations, and connections to prior work.
- Design and carry out empirical evaluations of LLMs, including selecting appropriate benchmarks, baselines, prompting or adaptation methods, and metrics for analyzing model behavior, capabilities, and risks.
- Implement and experiment with modern methods for adapting and using LLMs, such as prompting, retrieval-augmented generation, fine-tuning, preference optimization, or agent-based workflows, using contemporary toolkits.
- Communicate advanced technical ideas about LLMs effectively through paper discussion, oral presentation, and written project reports, and develop a research project that engages with current open problems in the field.

Required Course Materials

There is no official textbook for the course, as this is a rapidly evolving topic. Instead we will read recent research papers covering the latest developments. However, the following resources, which are freely available online, will be helpful for you to independently review, in order to refresh your understanding of important background concepts:

Speech and Language Processing (Jurafsky and Martn)

Natural Language Processing (Eisenstein)

The Annotated Transformer

The Illustrated Transformer

Grading Policy:

40% - In-Class Participation (see below)

20% - In-Class Student Presentation

20% - Programming Homework

20% - Course Project

All graded work will be rescaled proportionally into the final numerical grade, which will then be mapped to letter grade according to a cutoff based on the overall class grade distribution. The standard cutoff is 90/80/70% for A/B/C, but we may curve up (never down), i.e., use lower cutoffs than these. The cutoffs will only be determined after we grade the final project at the end of the semester. Students must obtain at least 50% overall grades to pass the class; students who choose the pass/fail option shall not participate in the final group project.

Description of Graded Components:

40% In-Class Participation - We will read and discuss two papers per class. Before class, each student should read the assigned paper and write a short critique answering a set of provided questions. These critiques should not be simple summaries or criticisms of the paper, but discuss positive aspects of the paper and limitations, and engage in discussion about why the paper is a significant contribution that merits publication in a top-tier ML/NLP conference, in addition to how the work fits into the existing literature and to how it might be improved or extended. Students should also ask questions regarding what points about the paper were difficult to understand so we can discuss in class, and work to gain a deeper understanding. The point of these reviews is to help spark discussion during class. During the last 20 minutes of class, we will go over questions raised by students in the paper critiques. All students are expected to attend each class and make an effort to contribute to this discussion. Please notify the instructor if you are not able to attend class due to an illness or emergency. To help students in the class improve their technical presentation skills, you are also expected to provide feedback on two student presentations. This feedback should discuss the presentation's clarity, presentation style, level of technical depth, adequate coverage of background material, and provide concrete suggestions for improvement. Submit your feedback on Canvas after the presentation. During the first week we will conduct a two-minute-madness session in which each student selects a recent paper relating to Large Language Models from the proceedings of a specified set of recent conferences, creates a few slides in a shared presentation, and gives a two minute presentation in class (during the first week of the semester) discussing why we should consider reading the paper. These papers will be considered for selection in the papers that we read throughout the semester. The participation component of your grade will be broken down into the following parts:

Attendance and participation during class discussion (10%)

Written paper critiques (15%) - you can skip writing critiques for three papers throughout the semester, without penalty. You are also not required to submit a paper critique for a paper you are presenting.

Participation in Two-minute madness on January 10 (5%)

Feedback on student presentations (10%) - do this twice throughout the semester.

20% In-Class Presentations - For each class, two students will be assigned to work together and deliver a 30 minute lecture covering one paper on a specific topic. The goal of this presentation is to educate students in the class, and facilitate discussion. Think about how to design your slides to clearly communicate the ideas in the paper, and be prepared to answer questions. Also think about what background/prior work may be needed to understand the most important ideas and their significance, and how you can summarize this concisely, as part of your presentation, in order to facilitate a deeper understanding. You can expect to present 1-2 times throughout the semester.

Students are required to submit slides on Canvas at least two days ahead of the assigned presentation (by the end of the day on Monday for lectures on Wednesday, or by the end of day Friday for Monday lectures).

Presentation slots, student presentation teams, and papers selected for presentation will be assigned in advance and can not be changed unless you have an excused absence verified by the dean of students.

20% Programming Assignment - We plan to assign one programming project for everyone to complete. The assignment will be in Python, and make use of Numpy and Pytorch. It will require non-trivial computation to complete; we recommend using PACE/ICE, which will be requested for students in the class to use or Google's Colab platform which provides easy access to GPUs. Completing these projects will require waiting for your models to train (this can range from about 30 minutes to hours depending on the efficiency of your implementation), so we strongly recommend starting work on these programming assignments well in advance of the deadline. If you start working on an assignment the day before it is due, it is unlikely you will be able to complete it on time.

20% Course Project - As part of the course, students are required to conduct a course project related to LLMs, and submit a report at the end of the semester. Projects should be conducted in teams of 2 or 3 students. Example projects could include fine-tuning an LLM for a specific task, or evaluating LLMs to better understand their capabilities, limitations or risks. The project can not be the same as a project used in another class.

The grading rubric for the final project is as follows:

- Clarity (1-5) For the reasonably well-prepared reader, is it clear what was done and why? Is the report well-written and well structured?
- Originality / Innovativeness (1-5) How original is the approach? Does this project break new ground in topic, methodology, or content? How exciting and innovative is the work that it describes?
- Soundness / Correctness (1-5) First, is the technical approach sound and well-chosen? Second, can one trust the claims of the report – are they supported by proper experiments, proofs, or other argumentation?
- Meaningful Comparison (1-5) Does the author make clear where the problems and methods sit with respect to existing literature? Are any experimental results meaningfully compared with the best prior approaches?
- Substance (1-5) Does this project have enough substance, or would it benefit from more ideas or results? Note that this question mainly concerns the amount of work; its quality is evaluated in other categories.
- Overall (1-5) This will be the final score based on the overall quality of the project. This is not the sum of the above aspect-based scores.

Attendance, In-Class Presentation, and Participation

The class requires in-person attendance. Attendance will be taken on randomly selected class days. Students may miss up to two such sessions without penalty; additional absences will result in a grade deduction, except for official approved absence from the Office of the Dean of Students. Attendance may be recorded using various methods, including roll call, sign-in sheets, in-class quizzes, or student card readers.

Depending on class size and scheduling, each student may be asked to give a short in-class presentation on a scheduled date. Presentation dates will be announced at least one month in advance.

Students will also receive credit for asking and answering thoughtful questions related to the course content on Piazza, engaging in discussion in class and generally for participating in the class. There are many ways to show participation. Asking a question that is marked as a “good question” by an instructor on Piazza, or having an answer that is marked as an “endorsed answer” is one example. Asking insightful questions, and engaging in discussion during class is another example. Please be polite and respectful towards TAs and other students in the class.

Course Policies

Attendance and/or Participation

This course requires in-person attendance. Attendance will be taken on randomly selected class days. For absences due to illness or personal emergencies, students shall submit an absence request and obtain approval through the Office of the Dean of Students.

Academic Integrity

Georgia Tech aims to cultivate a community based on trust, academic integrity, and honor. Students are expected to act according to the highest ethical standards. Review Georgia Tech’s Honor Code and the student Code of Conduct.

Any student suspected of cheating or plagiarism on a quiz, exam, or assignment will be reported to the Office of Student Integrity, who will investigate the incident and identify the appropriate penalty for violations.

Core IMPACTS

Core IMPACTS is the University System of Georgia's General Education curriculum. If you are teaching a course that counts towards Core IMPACTS, you should include a syllabus statement about the Core area and associated career competencies. This resource developed by the Center for Excellence in Teaching and Learning and Online Education at Georgia State University includes template syllabus statements for each of the Core IMPACTS areas that you may adapt for your course.

Accommodations for Students with Disabilities

If you are a student with learning needs that require special accommodation, contact the Office of Disability Services (404-894-2563) as soon as possible to make an appointment to discuss your special needs and to obtain an accommodations letter. Please also e-mail me as soon as possible in order to set up a time to discuss your learning needs.

Student-Faculty Expectations Agreement

At Georgia Tech, we believe that it is important to strive for an atmosphere of mutual respect, acknowledgement, and responsibility between faculty members and the student body. The Student-Faculty Expectations articulate some basic expectations that you can have of me and that I have of you. In the end, simple respect for knowledge, hard work, and cordial interactions will help build the environment we seek. Therefore, I encourage you to remain committed to the ideals of Georgia Tech while in this class.