CS795/895 Special Topics: Large Language Model Architectures and Applications
Spring 2026

Instructor

Dr. Mahmoud Nazzal, Assistant Professor, CS Department
Office: Room 3210, ENGR & COMP SCI BLDG, Norfolk, VA 23529, USA
Email: mnazzal@odu.edu
Personal site: fs.wp.odu.edu/mnazzal
Research profile: mahmoudkanazzal.github.io
Phone: 757-683-7760
Class times: 3 credit hours. W: 07:25 pm-10:05 pm
Office Hours: W: 05:00 pm-06:00 pm and/or by appointment
Class location: ECSB 2120

Prerequisites

Linear algebra, basic probability theory, basic data structures and algorithms, computer programming (Python), basic familiarity with machine learning and deep learning concepts

Textbook and References

- Primary Course Material: **Lecture slides and instructor-prepared notes**.
- Textbook (assumed):
  Afshine Amidi and Shervine Amidi, Super Study Guide: Transformers & Large Language Models, 2024. ISBN: 979-8836693312

Optional References (for depth and additional reading)

- ***Lewis Tunstall, Leandro von Werra, and Thomas Wolf****, Natural Language Processing with Transformers, O'Reilly Media, 2022.*
  ISBN: ***9781098103248***
- **Sebastian Raschka**, *Build a Large Language Model (From Scratch)*, Manning Publications, 2024.
  ISBN: **9781633437166**

Brief Course Description

This course provides a principled, systems-level introduction to large language models, tracing their evolution from general learning machines to full-fledged computational systems whose behavior emerges from architectural, training, and inference choices. The course is organized into four blocks that progressively address what transformer-based language models are, how they are trained and adapted at scale, why their behavior is constrained and sometimes unreliable, and how they function as components within larger agentic systems. Students begin by revisiting foundational learning concepts and examining the architectural shift from sequential modeling to attention, developing a deep understanding of transformer structure, tokenization, context, and decoding behavior. The course then explores pretraining objectives, fine-tuning strategies, scaling limits, and system-level constraints that shape real-world deployment. Building on this foundation, students analyze grounding, alignment, and reasoning, with emphasis on hallucination, retrieval-augmented generation, and test-time computation. The final block focuses on agentic language-model systems, evaluation methodologies, and fundamental limitations, equipping students with a rigorous conceptual framework for understanding, analyzing, and designing reliable large language model–based systems.

Upon successful completion of this course, learners will be able to:

1. Explain the learning principles and architectural motivations behind transformer-based language models.
2. Describe Transformer components and information flow, including tokenization, embeddings, self-attention, and encoder–decoder variants.
3. Analyze how context length, tokenization, and decoding strategies shape model behavior, variability, and hallucination.
4. Explain pretraining, fine-tuning, and parameter-efficient adaptation methods, along with their scaling trade-offs.
5. Evaluate system constraints affecting LLM deployment, including compute, memory, latency, and context limits.
6. Identify causes of grounding and hallucination failures and assess retrieval-augmented generation designs.
7. Explain alignment and preference-optimization methods and their trade-offs.
8. Compare reasoning approaches and test-time scaling strategies such as chain-of-thought and self-consistency.
9. Analyze agentic LLM systems, including tool use, orchestration, and cascading failure modes.
10. Critically assess evaluation methods for LLMs and LLM-based systems, including robustness and cost-aware metrics.
11. Articulate key limitations, safety challenges, and open research problems in large language models.

Grading Policy

Item	Grade Allocated
Quizzes	10
Final Exam	30
Homework Assignments	25
Term Project*	35

*The project grade entails the grade of paper and proposal presentation, project presentation, and project report.

Late Submission Policy

Homework assignments submitted after the deadline will incur a 10% grade deduction per late day. Submissions more than 3 days late will receive a grade of zero, unless prior arrangements have been approved by the instructor.

Weekly Topic Organization

Block I (Weeks 1–3): Foundations of Transformer-Based Language Modeling
Focus: ***What is the machine, and why did this architecture prevail?***

Week	Date (Wed)	Topics	Related Chapters / Sections (Omidi & Omidi)
Week 1	Jan 21, 2026	***Foundations of Machine Learning for Language Modeling (Why modeling natural language is fundamentally hard)*** **Introduction & Course Logistics:** course scope & objectives, structure, assessment, project overview. **ML & DL Basic Revision:** inputs → parameters → outputs; conceptual training loop (loss, optimization, generalization); evaluation fundamentals. **Sequence Modeling & Generative Learning:** sequence data; autoregressive factorization (chain rule intuition); likelihood vs truth; why language generation is difficult. **Why Classical Sequence Models Fail:** recurrence bottlenecks; long-range dependency limitations; motivation for non-recurrent/global-context architectures.	**Ch. 1** (1.1–1.3 Foundations); **Ch. 2.3.2** (RNN limitations); **Ch. 4.1.2** (AR chain rule)
Week 2	Jan 28, 2026	***Core Components of Transformer-Based Language Models (How Transformers represent text and capture global context)*** **Text → Tokens → Embeddings Pipeline:** tokenization and subword units; token embeddings; positional encodings; representations as learned vectors. **Transformer Architecture:** self-attention (Q/K/V), multi-head attention, feed-forward networks, residual connections, normalization, end-to-end information flow. **Model Structures:** encoder-only, decoder-only, encoder–decoder architectures and task alignment.	**Ch. 2.1–2.2** (Tokenization, token embeddings); **Ch. 3.1–3.4** (Self-attention & Transformer architecture)
Week 3	Feb 4, 2026	***Architecture Variants, Decoding, & Prompting in LLMs (What really happens during generation time)*** **Overview of LLM fundamentals and configurations**: autoregressive and masked objectives. **Major Transformer-based architecture variation**s, **Mixture-of-Experts (MoE)** models. **Decoding in LLMs**: greedy search, beam search, top-k sampling, top-p sampling, temperature. **Introduction to advanced prompting techniques**: In-Context Learning (ICL), Chain-of-Thought (CoT), and related methods.	Ch. 4.1, 4.2.2; Ch. 3.4.1–3.4.3 Ch. 3.4.1–3.4.4 (MoE = 3.4.4) Ch. 4.1.2 (all decoding algorithms) Ch. 4.1.2, Ch. 3.1, Ch. 3.5 Ch. 4.3.1–4.3.3

Block II (Weeks 4–7): Training, Adaptation, and Scaling Constraints
Focus: ***How large language models are built, adapted, and constrained in practice.***

Week	Date (Wed)	Topics / Activities	Related Chapters / Sections (Omidi & Omidi + Xiao & Zhu)
Week 4	Feb 11, 2026	**LLM Pretraining: Objectives, Data, and Scale (How LLMs are pre-trained from the ground up using massive datasets)** LLM Pretraining: Objectives, Data, and Scale (What pretraining actually does) Pretraining objective (next-token prediction) and its effect on model behavior. Core data preparation steps: mixtures, deduplication, filtering, and contamination issues. Basic scaling behavior: compute–data–model trade-offs and scaling laws. High-level consequences of likelihood training (generalization vs memorization). Overview of capacity scaling using Mixture-of-Experts.	Omidi & Omidi: Ch. 4.2 (Pretraining), Ch. 3.4.4 (MoE). Xiao & Zhu: Ch. 1.1–1.2 (Pre-training tasks), Ch. 2.1.2 (Training LLMs), Ch. 2.2.4 (Scaling laws).
Week 5	Feb 18, 2026	**Base Paper & Project Proposal Presentations (*Presenting a base paper and mapping out the project roadmap*)** Student teams present an approved base paper and their project proposal, including motivation, related work positioning, and planned methodology.	*Project-specific papers*
Week 6	Feb 25, 2026	**Fine-Tuning, Parameter-Efficient Adaptation, Intro to Preference tuning (*How we adapt powerful base models without retraining them from scratch*)**	Ch. 4.4 (Fine-tuning: §§4.4.1–4.4.2); Ch. 4.5 (Preference tuning, *conceptual contrast*).

Week	Date (Wed)	Topics / Activities	Related Chapters / Sections (Omidi & Omidi + Xiao & Zhu)
		Supervised fine-tuning fundamentals. Parameter-efficient adaptation methods: Low-Rank Adaptation (LoRA), adapters, and prefix tuning. Trade-offs involving capacity, stability, and forgetting. Conditions under which fine-tuning improves or degrades model performance. Preference tuning intro (details will follow on week 9)	
Week 7	Mar 4, 2026	**LLM Inference: Scaling and Hands-On Details** **(How LLMs are enabled to scale and a practical guide to their usage)** **Inference Computation Path: prefill vs decode phases, parallel prefill vs sequential decoding, inference vs training computation differences.** **KV Cache: what keys/values store, reducing per-token computation, cache growth with sequence length, sliding-window attention and cache eviction.** **Context-Length Scaling: quadratic prefill attention cost, linear decode cost with KV cache, latency impact of long contexts, context-window feasibility limits** **Hands-On LLM Usage:** loading small open-source models in HF/Colab, running inference pipelines, configuring inference parameters (max tokens, context), inspecting token usage and latency, preparing a small dataset, applying lightweight fine-tuning, and evaluating model outputs before and after adaptation	Ch. 3.3 (Computational improvements); Ch. 3.5 (Attention computation speedup); Ch. 4.6 (Model compression: distillation and quantization).

Block III (Weeks 8–10): Grounding, Alignment, and Multimodality
Focus: ***How language model behavior is grounded, aligned with human preferences, and extended beyond text.***

Week	Date (Wed)	Topics	Related Chapters / Sections (Omidi & Omidi)
Week 8	Mar 11, 2026	**Grounding and Retrieval-Augmented Generation** ***(Why LLMs hallucinate and how retrieval changes model behavior)*** Structural causes of hallucination. Retrieval-augmented generation architectures. Indexing, embedding search, retrieval brittleness. Error propagation in grounded systems. Data contamination and retrieval failure cases.	Ch. 5.3.3 (RAG); Ch. 4.1 (hallucination context); Ch. 2.4 (Embedding operations and retrieval).
—	Mar 18, 2026	**NO CLASS** – Spring Holiday	—
Week 9	Mar 25, 2026	**Alignment and Preference Optimization** **(How LLMs learn human preferences and avoid harmful behavior)** Motivation for alignment. Human preference modeling. Reinforcement Learning from Human Feedback (conceptual). Direct Preference Optimization. Alignment trade-offs and emergent behaviors. GRPO and modern RLHF scaling variants (high-level)	Ch. 4.5 (Preference tuning: §§4.5.1–4.5.3); Ch. 4.4 (contrast with finetuning).
Week 10	Apr 1, 2026	**Vision–Language Models and Multimodal LLMs** ***(How language models are extended to understand and reason over visual inputs)*** **Motivation for multimodal modeling**: limits of text-only LLMs and tasks requiring joint visual–text understanding. **Vision representations**: CNNs and vision transformers for encoding images as tokens. **Multimodal representation learning**: contrastive image–text pretraining and alignment of visual and textual embeddings (e.g., CLIP). **Architectural patterns**: dual encoders and vision encoder + LLM integration. **Instruction-tuned multimodal systems**: adapting pretrained vision–language models for	Lecture Notes

Week	Date (Wed)	Topics	Related Chapters / Sections (Omidi & Omidi)
		conversational and reasoning tasks (e.g., LLaVA). **Capabilities and tasks**: visual question answering, captioning, retrieval, and visual grounding. **Failure modes**: multimodal hallucinations and grounding errors.	

Block IV (Weeks 11–14): Agentic Systems, Evaluation, Safety and Limitations
Focus: ***When language models become systems and where they break down.***

Week	Date (Wed)	Topics / Activities	Related Chapters / Sections (Omidi & Omidi)
Week 11	Apr 8, 2026	**Agentic Language Models and Tool-Using Systems** ***(How LLMs reason, plan, act, and call tools)*** **LLM reasoning:** meaning of reasoning, chain-of-thought prompting, self-consistency, process supervision, and verification-based reasoning. **Agentic LLMs:** function calling, tool use, ReAct-style loops (plan–act–observe), memory, state, orchestration, and emerging tool integration standards (e.g., Model Context Protocol). **Failure modes:** cascading errors in agentic workflows.	Ch. 4.3 (prompting); Ch. 4.1.2 (generation); Ch. 5.1 (system-level evaluation insight); Ch. 5.3 (task composition); Supplemental ReAct/tool-use papers.
Week 12	Apr 15, 2026	**Evaluation of LLMs and LLM-Based Systems** **(How to measure quality, robustness, and cost)** Task-based evaluation (pass at k, constraint satisfaction). Rule-based metrics (brief). LLM-as-a-Judge. Robustness under distribution shift. Cost-aware evaluation: quality, latency, compute. Common benchmarks.	Ch. 5.1 (evaluation); Ch. 5.2–5.3 (tasks and benchmarks).
Week 13	Apr 22, 2026	**Limitations, Safety Perspectives, and Open Problems** ***(Technical limits, safety risks, and future research questions in LLMs)*** ***Limitations*****: hallucination mitigation limits, system brittleness, over-trust, calibration and uncertainty, evaluation gaps.** ***Risks*****: safety risks, alignment failures, cascading errors in real deployments.** ***Research areas*****: emerging non-Transformer architectures (e.g., state-space models, Mamba) and early diffusion-based LLM research as future directions.**	Ch. 4.1 (limitations context); Ch. 4.3.3 (prompting risks); Ch. 5.1.2 (evaluation limitations); Supplemental safety papers.
Week 14	Apr 29, 2026	**Course Project Presentations:** Student teams present final project outcomes, evaluation results, limitations, and lessons learned.	—

Final Exam (Official Exam Week)

Period	Dates	Activity
Final Exam Week	May 6–13, 2026	**Final Comprehensive Exam** (scheduled by the Registrar; not held during a regular Wednesday session).

Student Mental Health and Wellbeing

ODU is committed to supporting your mental health. The Office of Counseling Services offers free, confidential support including virtual and in-person counseling, group sessions, and crisis services. Same-day or next-day appointments can be scheduled at odu.edu/counseling services. In case of a mental health emergency—such as thoughts of self-harm, harming others, or recent assault—call 911, or contact a crisis counsellor 24/7 at 757-683-4401 (press Option 2). You may also call the Suicide & Crisis Lifeline at 988 or text "HOME" to 741741.

Academic Integrity

Old Dominion University expects honesty in all academic work. In this course, your work and conduct must comply with the **Code of Student Conduct** (odu.edu/oscai). Academic dishonesty includes:

- **Cheating:** using unauthorized assistance, materials, study aids, or information.
- **Plagiarism:** using others' language or ideas without proper acknowledgment.
- **Fabrication:** inventing, altering, or falsifying data, citations, or information.
- **Facilitation:** helping another student commit a violation or failing to report suspected violations.

Classroom disruptions that undermine instruction are also prohibited. **Suspected violations will be reported to the Office of Student Conduct & Academic Integrity and may result in sanctions up to and including expulsion.**

Honor Code

Students must follow the ODU Honor Code for all assignments and exams. Collaboration on ideas is encouraged, but **all submitted work (code, text, analysis) must be your own**. Limited use of tools (e.g., ChatGPT) for brainstorming is acceptable; **do not submit AI-generated text/code as your own**. Suspected violations will be handled under University policy.

Drop Policy

As per University guidelines. See the University Calendar for drop dates.

Accessibility & Accommodations

ODU provides reasonable accommodations under the ADA. If you need accommodations, **obtain an accommodation letter from the Office of Educational Accessibility (OEA)** and share it with me early so we can implement the approved arrangements. If you anticipate barriers but don't yet have a letter, **contact OEA** to discuss eligibility.

- **OEA:** 1021 Student Success Center, (757) 683-4655, odu.edu/educationalaccessibility
- Accommodations begin **once** I receive your official OEA letter.

Attendance Policy

Students are expected to attend classes regularly.