# Computation and Language Courses at Georgetown University

Nathan Schneider, April 2025

1. The CL Landscape at Georgetown	2
Course Conventions at Georgetown	2
Departments and Programs	3
Linguistics department	3
Computer Science department	5
Other graduate programs	6
2. Course Offerings	6
Computational Grammar Formalisms	6
LING-4411   Construction Grammar	6
LING-4419   Grammar Formalisms for Computational Research	7
Computational Corpus Linguistics	7
LING-4427   Computational Corpus Linguistics	7
LING-4449   Analyzing language data with R	8
LING-4424   All About Prepositions	8
Natural Language Processing	8
LING-2040/4400   Computational Language Processing	
(previous title: Introduction to Natural Language Processing)	8
LING-4401/DSAN-5400   Computational Linguistics with Advanced Python	8
LING-4466   Multilingual NLP	9
COSC/LING-5402   Empirical Methods in Natural Language Processing	9
DSAN-5800   Advanced NLP	9
COSC-6422/LING-8422   Advanced Semantic Representation	10
LING-8415   Computational Discourse Models	10
COSC-8405   Doctoral Seminar: NLP	10
Information Retrieval, Information Extraction, and Data Mining	11
COSC-4550   Information Retrieval	11
COSC-5540   Text Mining & Analysis	11
COSC-8530   Doctoral Seminar: Search and Mining of Textual Data	11
Spoken and Interactive Language Processing	11
COSC-4467/LING-4467   Speech & Audio Processing with Deep Neural Networks	11
COSC-4463/LING-4463   Dialogue Systems	12
Other CL Applications	12
LING-4464   Social Factors in Computational Linguistics and Al	12
Machine Learning and Artificial Intelligence	13
COSC-3450   Artificial Intelligence	13
COSC-3440   Deep Reinforcement Learning	13
COSC-3470   Deep Learning	13
LING-5444   Machine Learning for Linguistics	13
COSC-5450   Foundations of Machine Learning	14
COSC-5455   Introduction to Deep Learning	14
COSC-5480   Large Language Models	14
COSC-6440   Deep Reinforcement Learning	15
COSC-6480   Experimental Al	15
Data Structures and Algorithms	15
COSC-5200   Algorithms	15

# 1. The CL Landscape at Georgetown

Reflecting the interdisciplinary nature of the field of Computational Linguistics (CL) and the variety of reasons for students and scholars to come into contact with it, CL education and research at Georgetown is spread across multiple departments and degree programs. The <u>GUCL</u> group exists to publicize and foster interdisciplinary *research* at Georgetown via scientific talks, discussions, and a mailing list. Focusing instead on *courses*, this document consolidates the various opportunities to give a unified perspective and serve as a reference for prospective students, current students, and faculty.

This document is descriptive, not prescriptive; departmental program handbooks and directors of undergraduate/graduate studies are authoritative with respect to policies and degree requirements. Note the date above: we will make every effort to keep this document up to date, but cannot guarantee that all policy changes or new course offerings will be immediately reflected here.

## Course Conventions at Georgetown

Georgetown is on the semester system. CL courses are generally not offered during the summer. Thus, for purposes of this document, the academic year consists of a fall semester and a spring semester.

CL courses are taught by a mix of full-time faculty and adjunct faculty. A full course load for graduate students is 9 credits per semester. **Most courses at Georgetown consist of 3 credits, with 2.5 hours of class time per week.** The only exceptions pertinent to this document are (a) independent study credit and (b) Computer Science doctoral seminars, some of which are only 2 credits and are intended primarily for students who have completed their regular coursework. Descriptions below will therefore assume 3 credits per course unless stated otherwise.

Class sizes are small: 20 students is usually considered robust enrollment for an advanced or graduate course; enrollments of 10 (or less) for specialized courses are not uncommon. Enrollment limits vary depending on the curricular demands of the course. It is not likely that any CL course would allow enrollment to exceed 30.

Courses numbered below 4000 are for undergraduates<sup>1</sup>; those numbered between 4000 and 5999 may be **over/under** courses, open to juniors and seniors as well as graduate students. Some of the 5000+ courses, and all of the 6000+ courses, are limited to graduate students.

<sup>&</sup>lt;sup>1</sup> Undergraduate courses cannot be applied toward a graduate degree except in conjunction with tutorial (independent study) credit by special arrangement with the instructor.

## Departments and Programs

It should not be a surprise that the departments most directly relevant to CL are **Linguistics** and **Computer Science** (CS), both within the College of Arts and Sciences. Most of the CL course offerings are from these departments (individually or cross-listed between the two), and most of the students specializing in CL are pursuing a major, minor, or graduate degree in Linguistics, Computational Linguistics, or Computer Science.

Linguistics, CS, and allied departments/programs are summarized below. Linguistics and CS both offer undergraduate majors and minors, 2-year master's degrees, and Ph.D. degrees. There is also the Accelerated Master's option by which undergraduates can receive a combined bachelor's+master's degree in 5 years by double-counting courses toward the 2 degrees.

#### Linguistics department

Courses in the department belong to one of four *concentrations*: Theoretical Linguistics (TLI), Sociolinguistics (SLI), Applied Linguistics (ALI), and Computational Linguistics (CLI). Graduate students specialize in one of these areas, which define specific curricular requirements.<sup>2</sup> Henceforth, "CLI students" refers to Linguistics MS and Ph.D. students specializing in Computational Linguistics.

The main CLI courses are: **4427 Computational Corpus Linguistics** ("Corpus"), **4449 Analyzing Language Data with R** ("R"), **5444 Machine Learning for Linguistics** ("ML"), and the following NLP courses: **2040/4400 Computational Language Processing** ("CLP", formerly known as Introduction to NLP); **4401 Computational Linguistics with Advanced Python**; and **5402 Empirical Methods in NLP** ("ENLP"). Those numbered 4xxx are **over/under** courses, i.e. open to juniors and seniors as well as graduate students. CLP is the gateway course for those without computational experience, and is offered every semester at two levels: 2040 to fulfill a major core requirement, or 4400 to fulfill an advanced undergraduate elective or graduate requirement. The NLP courses all use Python. Corpus focuses on corpus annotation and corpus linguistics using existing tools; it does not require or teach programming skills beyond regular expressions.

The reason for several NLP courses is that CLP serves students who are new to programming—it teaches Python; ENLP serves graduate students who are experienced programmers; and CL with Advanced Python bolsters programming skills as a bridge between CLP and ENLP. In addition to Python, CLI students may benefit from additional languages, especially R (taught in the R course) and Java (not taught in Linguistics, but used in the machine learning course in CS).

<sup>&</sup>lt;sup>2</sup> As of 2025, graduate programs in Computational Linguistics are officially distinct programs rather than concentrations within a broader Linguistics degree, though all are offered within the Linguistics Department.

A variety of other courses offered in the Linguistics Department cover complementary or more advanced topics such as speech processing, meaning representations/parsing, discourse representations/parsing, and multilingual NLP.

Beginning in Fall 2025, we expect to generally offer courses on the schedule described below, though some semesters will vary:

#### Corpus+NLP Core

- LING-4427 Computational Corpus Linguistics (Fall, *Zeldes*)
- LING-2040/4400 Computational Language Processing (Fall, *Zeldes* + Spring, *Wilcox*)
- LING-4401/DSAN-5400 Computational Linguistics with Advanced Python (Fall, *Trevor Adriaanse*) (also offered in Spring as DSAN-5400 only)
- LING-5402/COSC-5402 Empirical Methods in NLP (Spring, *Schneider*)

#### **Applications**

- LING-4463/COSC-4463 Dialogue Systems (Fall: even years, Claire Bonial)
- LING-4466 Multilingual NLP (Spring: even years, *Kenton Murray*)
- LING-4464 Social Factors in Comp Ling and AI (Spring: odd years, Shabnam Tafreshi)
- LING-4467/COSC-4467 Speech & Audio Processing with Deep Neural Networks (Fall: odd years, *Joe Garman*)

#### Methods

- LING-4480 Computational Linguistics Research Methods (Fall, repeatable, rotating faculty)
- LING-4449 Analyzing Language Data with R (Spring: odd years, *Zeldes*)
- LING-5444 Machine Learning for Linguistics (Spring: even years, *Zeldes*)
- LING-44xx Computational Linguistics with LLMs (Spring: even years—new in Spring 2026, *Wilcox*)

#### Theories

- LING-4411 Construction Grammar (occasional, *Bonial*)
- LING-4419 Grammar Formalisms for Computational Research (occasional, *Portner*)
- LING-4428 Computational Approaches to Historical Linguistics (Spring: odd years, *Zeldes*)
- LING-4424 All About Prepositions (Fall: odd years, *Schneider*)
- LING-xxxx Computational Psycholinguistics (Spring: odd years—new in Spring 2027, *Wilcox*)

#### Seminars

- LING-8415 Discourse Models (Spring: even years, *Zeldes*)
- LING-8422/COSC-6422 Advanced Semantic Representation (Spring: odd years, *Schneider*)
- LING-8430 Information, Structure, and Language (Fall: even years, *Wilcox*)

Additional courses may be offered on an ad hoc basis. Because CLI courses offer practical skills, many non-CLI students enroll in them, sometimes with little or no prior programming experience. CLI students are required to become proficient programmers; all take Corpus as well as INLP and/or ENLP, and many take CS courses. There are also department-wide distribution requirements for courses in Sound (phonology), Form (syntax), and Meaning (semantics/pragmatics). CLI students are required to take at least one seminar (typically numbered above 600). Graduate course loads of full-time students in the Linguistics Department are as follows:

- **Ph.D.** students take a full course load (3 courses) throughout their first 3 years.
  - A Ph.D. student with a master's degree from another institution may be eligible for Advanced Standing, which reduces their Georgetown course requirements by up to 6 courses (1 year's worth).
- **CLI MS** students pursuing the **thesis** option take 8 courses in 2 years: e.g. 3+3+1+1 + thesis research in year 2.
- **CLI MS** students pursuing the **MRP** (master's research paper—as opposed to thesis) option take 10 courses in 2 years: 3+3+3+1.
- **Non-computational MS** students take a full course load for 2 years: 3+3+3+3, regardless of whether they choose the thesis or MRP option.

#### For more information: <u>https://linguistics.georgetown.edu/graduate/handbook</u>

#### Computer Science department

One of the major areas of research in the department is <u>Data-Centric Computing and Artificial</u> <u>Intelligence</u>. Faculty in this area teach courses in artificial intelligence (AI), machine learning, natural language processing (NLP), information retrieval (IR), text mining, and data science. Students do not formally specialize in an area. CS course loads are as follows:

- **CS MS & Ph.D. students** take 10 regular courses (30 credits) in their first 2 years (typically 3+3+3+1—this "front-loaded" schedule is required for international students).
  - **CS MS students** pursuing the **thesis** option enroll in COSC-9999 in place of 2 regular elective courses.
  - A **Ph.D. student** with a master's degree from another institution may be eligible for **Advanced Standing**, which reduces their Georgetown course requirements.
- CS Ph.D. students are additionally required to complete
  - 6 credits of doctoral seminars (2 or 3 credits each, pass/fail), typically in years 3–5, and
  - 2 workshops in the Apprenticeship in Teaching program, typically in years 4–5 (these are non-credit-bearing).

For more information: <u>https://cs.georgetown.edu/graduate-program-handbook/</u>

#### Other graduate programs

The Department of Spanish and Portuguese offers master's and Ph.D. degrees in <u>Spanish</u> <u>Linguistics</u>. Students in the program typically take several courses in Linguistics. Some core linguistics topics are cross-listed between the Spanish and Portuguese department and Linguistics.

An interdisciplinary doctoral concentration in <u>Cognitive Science</u>, housed within the Graduate School, is an alternate degree path available to Ph.D. students in relevant departments (Computer Science, Linguistics, Spanish Linguistics, Psychology, Philosophy, Neuroscience, Biology). Doctoral students pursuing the CogSci concentration take the usual amount of coursework for students in their home departments but with 4 of those courses from other departments and two 3-credit doctoral CogSci seminars.

The <u>MS in Data Science and Analytics</u> program, housed within the Graduate School, offers students training in "computational, mathematical, and statistical methods to prepare them for careers in data science and analytics." Most of the core Analytics courses are taught by faculty in the CS and Math/Statistics departments. Programming proficiency is required, though many of the students have stronger backgrounds in mathematics/statistics than CS. Analytics students are often interested in taking CL courses as electives.

**CCT:** "<u>Communication, Culture and Technology</u> is an interdisciplinary Master of Arts Program focusing on challenges posed by new technologies in a range of fields, including research, government, politics, arts, media, communication, business, health, and medicine." Technology is addressed by disciplines ranging from art, media, and design to culture, society, policy, and business. The program is directed by David Lightfoot (a member of the Linguistics department); Evan Barba, one of the CCT faculty, has an affiliation with the CS Department. Some CCT students have expressed interest in CL courses.</u>

# 2. Course Offerings

Below are descriptions of the courses mentioned above, organized thematically for convenience. The CL courses are also listed by semester on the GUCL website (<u>http://gucl.georgetown.edu/</u>).

## **Computational Grammar Formalisms**

#### LING-4411 | Construction Grammar

Many theories of compositionality posit strict separation between the lexicon on the one hand, and abstract procedural rules for grammatical combination on the other. In contrast, Construction Grammar cohesively accounts for the forms and meanings of morphemes, words, sentence patterns, and even "fringe" phenomena such as the Caused-Motion construction: *She blinked the snow off of her eyelashes*.

This course provides an overview of several variant approaches to Construction Grammar. The course will begin with a survey of readings representing differing Construction Grammar approaches, including work from Fillmore, Lambrecht, Michaelis, Goldberg, Croft, and Bergen. As we examine the different flavors of Construction Grammar, we will explore the validity, advantages, and disadvantages of Construction Grammar. In this exploration, you will be asked to consider and debate questions at the crux of grammar, such as: How does syntax fit into the big picture of language in general? How are generalizations over utterances represented? How are an indefinite variety of utterances producible from a finite system of grammatical knowledge? Finally, you will select and apply a construction grammar approach of your choice to a theoretical problem (e.g., how are light verb constructions extended, and what makes a novel combination acceptable or unacceptable?) or application area (e.g., language learning or multi-word expression detection) of interest to you in any language you are studying, comparing with other approaches to this problem.

#### LING-4419 | Grammar Formalisms for Computational Research

Linguists have developed a large number of formally precise syntactic theories, and many of them have been important tools for computational research. In this course, we will study five such systems with the goal of understanding both their perspective on syntax and its relation to parsing, production, and semantics, and will work to gain sufficient skill in using the formal systems to make them useful for computational work. The five systems we will discuss, along with classic early references, are the following:

- 1. HPSG (Head-driven Phrase-structure Grammar: Pollard and Sag 1994; Sag, Wasow, and Bender 1999)
- 2. CCG (Combinatory Categorial Grammar: Steedman 2000)
- 3. LFG (Lexical Functional Grammar: Kaplan and Bresnan 1982, Dalrymple 2001)
- 4. TAG (Tree Adjoining Grammar: Joshi 1987)
- 5. Minimalist Grammars (Stabler 2001)

We will spend most of our time on HPSG (with its semantic theory Minimal Recursion Semantics, MRS) and CCG. HPSG is is both widely used in computational research and influential as a framework for studying syntax. CCG is an important modern version of the classical framework of categorial grammar and supports a direct syntax-semantics interface. We will also do brief one-week overviews of LFG and TAG, and will take a look at Minimalist Grammars because they represent a formalization of the Minimalist syntax familiar to many Linguists.

Prerequisite: Syntax 1 or a course which includes basic formal language theory

## **Computational Corpus Linguistics**

#### LING-4427 | Computational Corpus Linguistics

Digital linguistic corpora, i.e. electronic collections of written, spoken or multimodal language data, have become an increasingly important source of empirical information for theoretical and applied linguistics in recent years. This course is meant as a theoretically founded, practical introduction to corpus work with a broad selection of data, including non-standardized varieties such as language on the Internet, learner corpora and historical corpora. We will discuss issues of corpus design, annotation and evaluation using quantitative methods and both manual and automatic annotation tools for different levels of linguistic

analysis, from parts-of-speech, through syntax to discourse annotation. Students in this course participate in building the corpus described here: <u>https://corpling.uis.georgetown.edu/gum/</u>

#### LING-4449 | Analyzing language data with R

This course will teach statistical analysis of language data with a focus on corpus materials, using the freely available statistics software 'R'. The course will begin with foundational notions and methods for statistical evaluation, hypothesis testing and visualization of linguistic data which are necessary for both the practice and the understanding of current quantitative research. As we progress we will learn exploratory methods to chart out meaningful structures in language data, such as agglomerative clustering, principal component analysis and multifactorial regression analysis. The course assumes basic mathematical skills and familiarity with linguistic methodology, but does not require a background in statistics or R.

#### LING-4424 | All About Prepositions

This course will take on the grammatical category of prepositions, which are hands-down some of the most intriguing and beguiling words once you get to know them. (How many prepositions are there in the previous sentence? The answer may surprise you!) We will look at their syntactic and semantic versatility in English and how they vary across languages. We will explore how they denote relations in space and time, as well as many other kinds of meanings. We will see why they are so hard to learn in a second language, and why they are difficult to define in dictionaries and teach to computers. The course will be project-based, including a significant project on a language other than English.

Prerequisites: Some background in syntactic description, e.g. satisfied by LING-2020, LING-4427, or LING-5127

## Natural Language Processing

# LING-2040/4400 | Computational Language Processing (previous title: Introduction to Natural Language Processing)

This course will introduce students to the basics of Natural Language Processing (NLP), a field that combines linguistics and computer science to produce applications, such as generative AI, that are profoundly impacting our society. We will cover a range of topics that form the basis of these exciting technological advances and will provide students with a platform for future study and research in this area. We will learn to implement simple representations such as finite-state techniques, n-gram models, and topic models in the Python programming language. Previous knowledge of Python is not required, but students should be prepared to invest the necessary time and effort to become proficient over the semester. Students who take this course will gain a thorough understanding of the fundamental methods used in natural language understanding, along with an ability to assess the strengths and weaknesses of natural language technologies based on these methods.

#### LING-4401/DSAN-5400 | Computational Linguistics with Advanced Python

This course teaches advanced topics in programming for linguistic data analysis and processing using the Python language. A series of assignments will give students hands-on practice implementing core algorithms for linguistic tasks. By the end of the course, students will be able to transform pseudocode into well-written code for algorithms that make sense of textual data, and to evaluate the algorithms

quantitatively and qualitatively. Linguistic tasks will include edit distance, semantic similarity, authorship detection, and named entity recognition. Python topics will include the appropriate use of data structures; mathematical objects in numpy; exception handling; object-oriented programming; and software development practices such as code documentation and version control.

*Prerequisites:* Basic Python programming skills are required (for example satisfied by LING-362, Intro to NLP)

#### LING-4466 | Multilingual NLP

This is a project based course focusing on the design and implementation of systems that scale Natural Language Processing methods beyond English. The course will cover both multilingual and cross-lingual methods with an emphasis on zero-shot and few-shot approaches, as well as 'silver' dataset creation. Modules will include Cross-Lingual Information Extraction & Semantics, Cross-Language Information Retrieval, Multilingual Question Answering, Multilingual Structured Prediction, Multilingual Automatic Speech Recognition, Typology, Morphology, as well as other non-English centric NLP methods. Students will be expected to work in small groups and pick from one of the modules to create a model based on state-of-the-art methods covered in the class. The course will be roughly three-quarters lecture based and one-quarter students presenting project updates periodically throughout the semester.

#### COSC/LING-5402 | Empirical Methods in Natural Language Processing

Systems of communication that come naturally to humans are thoroughly unnatural for computers. For truly robust information technologies, we need to teach computers to unpack our language. Natural language processing (NLP) technologies facilitate semi-intelligent artificial processing of human language text. In particular, techniques for analyzing the grammar and meaning of words and sentences can be used as components within applications such as web search, question answering, and machine translation.

This course introduces fundamental NLP concepts and algorithms, emphasizing the marriage of linguistic corpus resources with statistical and machine learning methods. As such, the course combines elements of linguistics, computer science, and data science. Coursework will consist of lectures, programming assignments (in Python), and a final team project. The course is intended for students who are already comfortable with programming and have some familiarity with probability theory.

*Prerequisite:* A data structures course, Computational Language Processing (Advanced Python recommended), or equivalent experience

#### DSAN-5800 | Advanced NLP

This course provides a formalism for understanding the statistical machine learning methods that have come to dominate natural language processing. Divided into three core modules, the course explores (i) how language understanding is framed as a tractable statistical inference problem, (ii) a formal yet practical treatment of the DNN architectures and learning algorithms used in NLP, and (iii) how these components are leveraged in modern AI systems such as information retrieval, recommender systems, and conversational agents. In exploring these topics, the course exposes students to the foundational math, practical applications, current research directions, and software design that is critical to gaining proficiency as an NLP/ML practitioner. The course culminates in a capstone project, conducted over its final six weeks, in which students apply NLP to an interesting problem of their choosing. In past semesters students have built chatbots, code completion tools, stock trading algorithms, just to name a

few. This course assumes a basic understanding of linear algebra, probability theory, first order optimization methods, and proficiency in Python.

This is an advanced course. Suggested prerequisites are DSAN 5000, DSAN 5100 and DSAN 5400. However, first-year students with the necessary math, statistics, and deep learning background will be considered.

#### COSC-6422/LING-8422 | Advanced Semantic Representation

Natural language is an imperfect vehicle for meaning. On the one hand, some expressions can be interpreted in multiple ways; on the other hand, there are often many superficially divergent ways to express very similar meanings. Semantic representations attempt to disentangle these two effects by exposing similarities and differences in how a word or sentence is interpreted. Such representations, and algorithms for working with them, constitute a major research area in natural language processing.

This course will examine semantic representations for natural language from a computational/NLP perspective. Through readings, presentations, discussions, and hands-on exercises, we will put a semantic representation under the microscope to assess its strengths and weaknesses. For each representation we will confront questions such as: What aspects of meaning are and are not captured? How well does the representation scale to the large vocabulary of a language? What assumptions does it make about grammar? How language-specific is it? In what ways does it facilitate manual annotation and automatic analysis? What datasets and algorithms have been developed for the representation? What has it been used for? Representations covered in depth will include Abstract Meaning Representation (http://amr.isi.edu/). Term projects will consist of (i) innovating on the representation's design, datasets, or analysis algorithms, or (ii) applying it to questions in linguistics or downstream NLP tasks.

Prerequisites: Corpus Linguistics or Empirical Methods in NLP

#### LING-8415 | Computational Discourse Models

Recent years have seen an explosion of computational work on higher level discourse representations, such as entity recognition, mention and coreference resolution and shallow discourse parsing. At the same time, the theoretical status of the underlying categories is not well understood, and despite progress, these tasks remain very much unsolved in practice. This graduate level seminar will concentrate on theoretical and practical models representing how referring expressions, such as mentions of people, things and events, are coded during language processing. We will begin by exploring the literature on human discourse processing in terms of information structure, discourse coherence and theories about anaphora, such as Centering Theory and Alternative Semantics. We will then look at computational linguistics implementations of systems for entity recognition and coreference resolution and explore their relationship with linguistic theory. Over the course of the semester, participants will implement their own coding project exploring some phenomenon within the domain of entity recognition, coreference, discourse modeling or a related area.

#### COSC-8405 | Doctoral Seminar: NLP

This course will expose students to current research in natural language processing and computational linguistics. Class meetings will consist primarily of student-led reading discussions, supplemented occasionally by lectures or hands-on activities. The subtopics and reading list will be determined at the start of the semester; readings will consist of research papers, advanced tutorials, and/or dissertations.

*Prerequisites:* Familiarity with NLP using machine learning methods (for example satisfied by Empirical Methods in NLP)

## Information Retrieval, Information Extraction, and Data Mining

#### COSC-4550 | Information Retrieval

Information retrieval is the identification of textual components, be them web pages, blogs, microblogs, documents, medical transcriptions, mobile data, or other big data elements, relevant to the needs of the user. Relevancy is determined either as a global absolute or within a given context or view point. Practical, but yet theoretically grounded, foundational and advanced algorithms needed to identify such relevant components are taught.

The Information-retrieval techniques and theory, covering both effectiveness and run-time performance of information-retrieval systems are covered. The focus is on algorithms and heuristics used to find textual components relevant to the user request and to find them fast. The course covers the architecture and components of the search engines such as parser, index builder, and query processor. In doing this, various retrieval models, relevance ranking, evaluation methodologies, and efficiency considerations will be covered. The students learn the material by building a prototype of such a search engine. These approaches are in daily use by all search and social media companies.

#### COSC-5540 | Text Mining & Analysis

This course covers various concepts and research areas in text search and mining. The structure of the course is a combination of lectures & students' presentations. The lectures will cover various search technologies, classification, text summarization, opinion and sentiment mining, covering applications on varying domains and formats, including scientific, health, and social media. The students are assigned a related topic in the field for further study, implementation, experimentation and presentation in the class.

#### **COSC-8530** | Doctoral Seminar: Search and Mining of Textual Data

In this doctoral seminar, doctoral students read, present, and discuss research papers on search and mining methodologies to process textual data of any form: short or long, general or domain specific, formal scientific text or some informal social media text. Student groups are assigned projects towards the aim of developing research insights.

## Spoken and Interactive Language Processing

# COSC-4467/LING-4467 | Speech & Audio Processing with Deep Neural

#### Networks

This course covers modern deep learning approaches for speech recognition, synthesis, and audio processing. Students learn PyTorch implementation of neural architectures, from foundational networks to state-of-the-art transformer models. Topics include basic text processing, audio feature extraction, automatic speech recognition, text-to-speech synthesis, and audio/music generation. The course emphasizes hands-on experience through weekly programming assignments using PyTorch. Prior programming experience in Python required; no previous signal processing or deep learning experience

assumed. Designed for computational linguistics and computer science graduate students or advanced undergraduates.

#### COSC-4463/LING-4463 | Dialogue Systems

Nearly all of us interact with dialogue systems -- from calling up banks and hotels, to talking with intelligent assistants like Siri, Alexa, or Cortana, dialogue systems enable people to get tasks done with software agents using language. Since the interaction is bi-directional, we must consider the fundamentals of how people engage in conversation so as to manage users' expectations and track how information is exchanged in dialogue. Dialogue systems require an array of technologies to come together for them to work well, including speech recognition, natural language understanding, dialogue management, natural language generation, and speech synthesis. This course will explore what makes dialogue systems effective in commercial and research applications (ranging from personal assistants and chatbots to embodied conversational agents and language-directed robots) and how this contrasts with everyday human-human dialogue.

This course will introduce students to the fundamentals of dialogue systems, expanding on technologies and algorithms that are used in today's dialogue systems and chatbots. There will also be emphasis on the psycholinguistic properties of human conversation (turn-taking, grounding) so as to prepare students for designing effective, user-friendly dialogue systems. The course will also include examining datasets and dialogue annotations used to train dialogue systems with machine learning algorithms. Coursework will consist of lectures, writing and programming assignments, and student-led presentations on special topics in dialogue. A final project will give students a chance to build their own dialogue system using open source and freely available software. This course is intended for students that are already comfortable with limited amounts of programming (in Python).

## Other CL Applications

#### LING-4464 | Social Factors in Computational Linguistics and AI

Advances in technologies for processing human languages have increasingly brought computational linguistics into contact with people. As such, what language reveals about people—and how AI algorithms make decisions affecting people based on their language—is of paramount concern. At the same time, contemporary algorithms for processing language offer powerful new tools for studying people and society on a large scale. Designed for students with grounding in computational linguistics, this course will examine the intersection of people, language, and algorithms with technical precision as well as an appreciation for human context. Topics will include: computational models of conversational interaction and power dynamics; emotions, sentiment, subjectivity, and politeness; toxic language; sociolinguistic variation; detection of attributes such as race and gender; issues of privacy, ethics, bias, and fairness, with special attention to minoritized speakers, languages, and dialects; and the use of large-scale language data for studying political framing and social movements like #MeToo and Black Lives Matter.

*Requirements:* Basic Python programming skills are required (for example satisfied by LING-362, Intro to NLP)

## Machine Learning and Artificial Intelligence

#### COSC-3450 | Artificial Intelligence

Artificial Intelligence (AI) is the branch of computer science that studies how to program computers to reason, learn, see, and understand. The lecture portion of this class surveys basic and advanced concepts and techniques of artificial intelligence, including search, knowledge representation, automated reasoning, uncertain reasoning, and machine learning. Additional topics include the Lisp programming language, theorem proving, game playing, rule-based systems, and philosophical issues. Applications of artificial intelligence will also be discussed and will include domains such as medicine, computer security, and face detection. Students must complete midterm and final exams, and five projects using the Lisp programming language.

Prerequisite: Data Structures

#### COSC-3440 | Deep Reinforcement Learning

Deep reinforcement learning is a machine learning area that learns how to make optimal decisions from interacting with an environment using deep neural networks. An intelligent agent observes the consequences of its action from the environment and alters its behavior to maximize the expected return. We study algorithms and applications in deep reinforcement learning. Topics include Deep neural networks, Markov decision processes, policy gradient methods, Q-Learning (DQN), Actor-Critic, Imitation Learning, and other advanced topics. The course has lectures, readings, programming assignments, and exams.

#### COSC-3470 | Deep Learning

This course will focus on building state-of-the-art systems in the intersection of deep learning and computer vision. Student will be introduced to deep architectures and learning algorithms for various discriminative and generative computer vision tasks. The course will demonstrate how such tasks are main building blocks in processing images and videos for applications such as self-driving cars, healthcare, surveillance, and human-computer interfaces.

#### LING-5444 | Machine Learning for Linguistics

In the past few years, the advent of abundant computing power and data has catapulted machine learning to the forefront of a number of fields of research, including Linguistics and especially Natural Language Processing. At the same time, general machine learning toolkits and tutorials make handling 'default cases' relatively easy, but are much less useful in handling non-standard data, less studied languages, low-resource scenarios and the need for interpretability that is essential for drawing robust inferences from data. This course gives a broad overview of the machine learning techniques most used for text processing and linguistic research. The course is taught in Python, covering both general statistical ML algorithms, such as linear models, SVMs, decision trees and ensembles, and current deep learning models, such as deep neural net classifiers, recurrent networks and contextualized continuous meaning representations. The course assumes good command of Python (ability to implement a program from pseudo-code) but does not require previous experience with machine learning.

Requirements: Intermediate Python (courses such as LING-472: Computational Linguistics with Advanced Python provide a good preparation)

#### COSC-5450 | Foundations of Machine Learning

This course provides a comprehensive introduction to the core principles and methodologies of machine learning. The course is designed to cover essential topics such as probability theories, common distributions, point estimation, sampling, model selection, gradient optimization, and evaluation, ensuring a comprehensive understanding of the theoretical and algorithmic aspects of machine learning and providing fundamental concepts to support further study in supervised learning, unsupervised learning, and reinforcement learning. It will also cover cutting-edge topics such as shallow learning vs. deep learning, self-supervised learning, and high-dimensional learning, focusing on enabling students to understand the theories and principles behind the latest advancements in the field. By the end of this course, students will not only grasp the fundamental concepts of machine learning but also cultivate a mindset and skill set that are adaptable to the dynamic nature of technological progress, keeping pace with the rapidly evolving technological landscape in the field of machine learning. The class will have lectures, mathematical homework, and exams.

#### COSC-5455 | Introduction to Deep Learning

Recent advances in hardware have made deep learning with neural networks practical for real-world problems. Neural networks are a powerful tool that have shown benefit in a wide range of fields. Deep learning involves creating artificial neural networks with greater layer depth or deep neural nets (DNN) for short. These DNNs can find patterns in complex data, and are useful in a wide variety of situations. In numerous fields, state-of-the-art solutions have been accomplished with DNNs and DNN systems dominate head-to-head competitions. This course will introduce the student to neural networks, explain different neural network architectures, and then demonstrate the use of these neural networks on a wide array of tasks.

#### COSC-5480 | Large Language Models

This course delves deep into the intricacies of Large Language Models (LLMs), offering students an understanding of their design, implementation, and applications. Beginning with the foundational architectures such as transformers and attention mechanisms, students will journey through the evolution from the fundamental models to contemporary marvels like GPT-3, ChatGPT, and GPT-4. The course aims to provide a comprehensive overview of the historical and current state of LLMs, equipping students with the knowledge to design, train, and fine-tune LLMs for custom applications. It will also encourage critical discussions on the ethical, societal, and technical challenges associated with LLMs. Key topics covered in the course include (1) Foundations: Review of RNNs, LSTMs, Attention Mechanisms, and Transformers. (2) Architectural Deep Dive: Behind the design of GPT-3, BERT, and other leading models. (3) Training Paradigms: Techniques and challenges in training massive models. (4) Applications: chatbots, content generation, recommendation systems, and beyond. (5) Societal Impact: Ethical considerations, fairness, and bias in LLMs. (6) Technical Challenges: Model explainability, controllability, and safety concerns. (7) Future Directions: Where LLMs are headed and emerging research areas. The course assessments consist of monthly assignments involving practical implementations and model evaluations, exams covering theoretical and applied concepts, and one optional final project focusing on designing a custom application utilizing LLMs. Class participation and critical discussion sessions are also important components in student assessments.

#### COSC-6440 | Deep Reinforcement Learning

Deep Reinforcement learning is an area of machine learning that learns how to make optimal decisions from interacting with an environment. From the environment, an agent observes the consequence of its action and alters its behavior to maximize the amount of rewards received in the long term. Reinforcement learning has developed strong mathematical foundations and impressive applications in diverse disciplines such as psychology, control theory, artificial intelligence, and neuroscience. An example is the winning of AlphaGo, developed using Monte Carlo tree search and deep neural networks, over world-class human Go players. The overall problem of learning from interaction to achieve goals is still far from being solved, but our understanding of it has improved significantly. In this course, we study fundamentals, algorithms, and applications in deep reinforcement learning. Topics include Markov Decision Processes, Multi-armed Bandits, Monte Carlo Methods, Temporal Difference Learning, Function Approximation, Deep Neural Networks, Actor-Critic, Deep Q-Learning, Policy Gradient Methods, and connections to Psychology and to Neuroscience. The course has lectures, mathematical and programming assignments, and exams.

#### COSC-6480 | Experimental AI

This course offers opportunities for students to have an in-depth understanding and hands-on experience with practical AI systems for state-of-the-art evaluation campaigns. It includes seminar-style classroom presentations and a significant project component. Students will be guided to go through the design and implementation of AI systems in different domains. The course will review recent AI and Machine Learning publications and lead students to work in small groups to build systems. Students are expected to have strong programming skills and previous experience in machine learning, deep learning, and/or AI.

## Data Structures and Algorithms

#### **COSC-5200 | Algorithms**

This introductory graduate course explores the design and analysis of efficient algorithms. Topics covered include divide-and-conquer, greedy algorithms, and dynamic programming as well as more advanced topics like amortization, randomization, linear programming, memory-efficient algorithms, online algorithms, and approximation algorithms. Specific problem domains are subject to change but may include string algorithms, computational geometry, graph algorithms, combinatorial optimization, number theory, and computational sciences. Although there is some limited overlap of material, this course is not a replacement for an undergraduate algorithms course and hence assumes some basic understanding of algorithm analysis and discrete mathematics.