SYLLABUS

BIOL 1595 / 2595: Artificial Intelligence in Biomedicine

Spring 2019

Day, Time, Location: TTh 9:00 - 10:20, CIT 241 (Swig Room)

Instructors: Carsten Eickhoff, PhD
233 Richmond Street, Rm 209
401-863-9665
carsten@brown.edu

Office Hours: Tuesdays and Thursdays, 10:20 - 11:20, Location TBA
Additional hours by appointment

Course Summary

This course will teach the fundamental theory and methods of artificial intelligence (AI) alongside their application to the biomedical domain. It will give a representative overview of traditional methods as well as modern developments in the areas of (deep) machine learning, natural language processing and information retrieval.

The course is designed to be accessible to non-computer science audiences and will not require extensive prior programming experience. A brief primer on Python programming will be covered as part of the practical exercises, and tutorials in Julia will be made available.

The course will be accompanied by practical assignments applying the discussed techniques in a biomedical context. Understanding of formal theoretical knowledge will be assessed in a final exam.

Prerequisites

The course is designed for students concentrating in domains such as (Computational) Biology, Applied Mathematics, or Neuroscience who have completed a course in introductory statistics (e.g., BIOL 0495). Understanding of linear algebra (e.g., MATH 0520) as well as informatics and data science fundamentals and programming (e.g., BIOL 1555) are useful but not required. Other students may enroll with instructor permission.

Enrollment

Course enrollment is limited to 24 students (50% undergraduate and 50% graduate). Additional auditors are permitted.

Goals & Learning Objectives

The *goal* of this course is for students to gain a formal understanding of the underlying principles and capabilities of traditional and modern artificial intelligence techniques. By the end of the course, students will have gained the needed knowledge to understand, develop and use AI methods in biomedical environments as well as to make informed decisions about the benefits and risks of popular industrial solutions.

Specific *learning objectives* are thus to:

1. Understand general principles of artificial intelligence

2. Demonstrate competency in artificial intelligence techniques through the implementation and adaptation of existing algorithms in biomedical settings

Primary Competencies

1. Acquire knowledge and skills in research methodologies to collaborate with substantive investigators
2. Understand the underlying principles of modern AI systems
3. Attain proficiency in the evaluation of AI system quality in biomedical settings

Refresher Competencies

1. Identify and implement analytic techniques and models for analysis of data
2. Effectively function in an interdisciplinary, collaborative environment
3. Review and evaluate the use of analytic methods in biomedicine
4. Build productive collaborations across the spectrum of biomedicine

Required Textbook / Material

Recommended Texts:

Bishop, Christopher M. *Pattern Recognition and Machine Learning*. Springer (2011). ISBN: 978-0-387-31073-2

Jurafsky, Daniel and Martin, James H. *Speech and Language Processing*. Pearson (2008).
ISBN: 978-0-131-87321-6
Freely available from the authors at:
https://web.stanford.edu/~jurafsky/slp3/ed3book.pdf

Manning, Christopher and Raghavan, Prabhakar and Schütze, Hinrich. *Introduction to Information Retrieval*. Cambridge University Press (2008).
ISBN: 978-0-521-86571-5
Freely available from the authors at:
https://nlp.stanford.edu/IR-book/information-retrieval-book.html

Goodfellow, Ian and Bengio, Yoshua and Courville Aaron. *Deep Learning*. MIT Press (2016).
ISBN: 978-0-262-03561-3
Freely available from the authors at:
https://www.deeplearningbook.org/

An introduction to the python and julia programming languages can be found at https://docs.python.org/3/tutorial/
BIDSS Manual

Assignments, course communication, Q&A, as well as feedback will be handled via banner (class link tba).

Computing
Students will be required to access a computer that can connect to the Internet. Any version of MacOS, Linux, as well as Windows with PowerShell 5.0 can be used for this course. This is not a requirement during class but necessary at least to complete individual assignments.

Grading Criteria / Attendance Policy

The graded elements of the course and their contribution to the final grade are:

Assignments	60% (divided evenly across three components, see below)
Final Exam	40%

Specific criteria details:

- Assignments will be issued towards the middle of each component (machine learning, natural language processing, information retrieval) and must be turned in on specified due dates (3 weeks later) accompanied by an oral presentation. Late submissions will not be accepted.
    - **Undergraduate Students** will receive a basic version of the assignment with 2-3 tasks each that will require programming work as well as a brief report detailing the formal approach taken to solve the assignment.
    - **Graduate Students** will receive an extended version of the same assignment including an additional task each.
- Programming work for assignments can be completed in a programming language of the student's choice.
- See below for a detailed overview of the assignment topics, grading criteria etc.
- The final exam will test formal understanding of concepts covered in class. During the exam, no programming (on paper or computers) will be required.

Classroom Environment Expectations

All lectures will be structured following an active learning format in which periods of formal didactic teaching are interspersed with periods of student activity and discussion. Students are expected to contribute to discussions and actively participate in exercises during class (see Grading Criteria/Attendance Policy above). In addition to the three hours of weekly class time, students will be expected to spend an average of 6-7 hours a week outside of class on assignments and recommended reading.

As a guide, the minimum number of hours a student should expect to spend on each major task for this course is directly proportional to their grade contribution is summarized below:

Task	Hours
Class Time	40
Assignments / Readings	84
Exam Preparation	56
Total	**180**

Schedule of Topics

Each lecture will consist of “Theory” (didactic) and “Practice” (interactive) components.

Class	date	topic	component	reading
1	01/24	Intro, Features, Principles	ml	Bishop (Ch 1)
2	01/29	Regression		Bishop (Ch 3)
3	01/31	Classification		Bishop (Ch 4)
4	02/05	Unsupervised Learning		Bishop (Ch 9)
5	02/07	Model Generality		Bishop (Ch 1)
6	02/12	Ensembles and Mixtures		Bishop (Ch 14)
7	02/14	Deep Learning		Goodfellow (Ch 1)
-	02/19	**No Class (Long Weekend)**		-
8	02/21	Evaluation		-
9	02/26	Assignment 1		-
10	02/28	Intro, Overview	nlp	Jurafsky (Ch 1)
11	03/05	Language Modelling		Jurafsky (Ch 4)
12	03/07	State and Sequence		Jurafsky (Ch 6)
13	03/12	Text Representation		Jurafsky (Ch 25)
14	03/14	Machine Translation		Jurafsky (Ch 23)
15	03/19	Conversational Agents		Jurafsky (Ch 24)
16	03/21	Evaluation		-
-	03/26	**No Class (Spring Recess)**		-
-	03/28	**No Class (Spring Recess)**		-
17	04/02	Assignment 2		-
18	04/04	Intro, Concepts	ir	Manning (Ch 1)
19	04/09	Web Crawling		Manning (Ch 20)
20	04/11	Indexing		Manning (Ch 4, 5)
21	04/16	Statistical Ranking		Manning (Ch 6, 12)
22	04/18	Learning to Rank		Manning (Ch 15)
23	04/23	Link Analysis		Manning (Ch 21)
24	04/25	Recommender Systems		Manning (Ch 18)
25	04/30	Evaluation		Manning (Ch 8)
26	05/02	Assignment 3		-
27	05/07	Recap		-

Detailed Overview of Topics

ML: Introduction, Features, Principles (01/24)
This class will set the scene for the remainder of the course. We will begin with an overview of the artificial intelligence landscape and its historical evolution over the past decades. We will discuss the essential components of machine learning systems and research papers and introduce the notion of casting inference problems in terms of features of various types. Finally, we will discuss all relevant course logistics (content division into modules, communication channels, office hours, materials, recommended reading, assignments, final exam and grading).

ML: Regression (01/29)
Regression is one of the most basic techniques in the machine learning toolbox that is used to infer continuous values (e.g., a person's weight) based on input data (e.g., that same person's height). We will formally introduce linear regression and subsequently expand the concept to the linear basis function model which allows for non-linearity in the treatment of input features.

ML: Classification (01/31)
In many use cases, our target labels will be discrete rather than continuous (e.g., does this tissue sample originate from malignant or benign tumor?). This setting is dubbed classification and comes in a number of variations (e.g., multi-label, multi-class). We will begin our discussion with statistical methods such as the k-nearest neighbor algorithm or decision trees before moving forward to an introduction to Bayesian inference. We will close with an overview of more advanced methods such as kernel-based learning and stochastic processes.

ML: Unsupervised Learning (02/05)
The previous lectures discussed supervised learning techniques from the classification and regression families in which we use previously annotated ground-truth labels as the prediction target. This class will turn to unsupervised methods in which no such labels are needed. Instead, we will explore the distribution of data points in our collection and attempt to discover clusters and other structures of interest using k-means and hierarchical clustering, density-based spatial clustering and anomaly detection. Towards the end of this class we will discuss the first assignment (see below for details).

ML: Model Generality (02/07)
Some machine learning methods are so good at memorizing training data points, that they eventually lack generality and may perform poorly on unseen samples that behave differently from the training data. This effect is referred to as overfitting. In this class, we will discuss the separation of data into folds, cross-validation, stratification, early stopping, regularization and considerations regarding the number of available samples vs. the complexity of specific machine learning schemes.

ML: Ensembles and Mixtures (02/12)

As we have seen in previous classes, machine learning systems are not perfect and will make occasional (or even frequent) mistakes. Instead of directly relying on a classifier's output, we will now train multiple different machine learning systems and have each of them cast a vote as to the true label of a data point and rely on the resulting consensus. Concretely, we will study model ensembles, bagging, information fusion and the notion of late vs. early binding.

ML: Deep Learning (02/14)
Deep learning is a recently popular machine learning paradigm based on multi-layer methods that have revolutionized a wide range of applications from recognizing lung cancer to playing super mario. In this class we will introduce the formal framework of neural networks and discuss a number of frequently encountered network architectures such as feed-forward, convolutional or recurrent neural networks.

ML: Evaluation of Machine Learning Models (02/21)
Performance evaluation is a crucial step in the design of machine learning systems. The right choice of datasets, metrics and evaluation parameters can critically affect the outcome and credibility of an evaluation campaign. In this class we will discuss different types of errors and their respective cost, accuracy, and the use of ROC and PR curves for evaluation.

ML: Assignment 1 - Re-Admission Prediction (02/26)
In this class we will have 10-minute presentations from each group, showcasing their solution to the machine learning assignment (see below for details).

NLP: Introduction and Overview (02/28)
The previous module introduced basic machine learning concepts often working with generic feature representations of various types input data such as measurements, time series or images. These data types correspond to entities in the physical world and tend to follow a range of basic scientific rules. Natural language, on the other hand, is a human cultural artifact that does not necessarily abide to such rules and tends to be much more challenging to handle. In this class, we will discuss a historic overview of the field and visit a number of concrete applications and challenges.

NLP: Language Modelling (03/05)
Language models offer a stochastic view on understanding samples of natural language. They allow us to describe how likely a stretch of text or an utterance are given what we already learned about language. We will discuss the notion of n-grams, their incorporation into frequency-based language models and the concept of language model smoothing.

NLP: State and Sequence (03/07)
Much of the complexity of natural language stems from the sequential and context-dependent way in which information is encoded and the sometimes considerable difference between structural and semantic ordering (think of relative

clauses that only much later in the sentence quantify something we read before). This class will introduce part-of-speech tagging, dependency parsing, Markov chains and hidden Markov models.

NLP: Text Representation (03/12)
Many natural language processing tasks centrally revolve around finding the right machine-processable representation for textual or spoken word documents (e.g., a single note within an electronic health record or a biomedical paper listed on Pubmed). We will introduce the traditional bag-of-words representation before moving into more advanced concepts such as topic models and word embeddings.
As a conclusion to this class we will discuss the second assignment (see below for details).

NLP: Machine Translation (03/14)
As an additional level of natural-language specific complexity, human language has developed in many parallel branches all across the Globe. Before being able to get at the information encoded in a document, we need to first learn the language it was written in (regardless of whether we are a flesh-and-blood reader or a machine). In this class we will discuss machine translation techniques that are able to translate text from one language to another without the need for human intervention. We will begin with traditional statistical models and interlinguas before moving on to more modern neural network based alternatives.

NLP: Conversational Agents (03/19)
With the proliferation of small-screen (tablets, smart phones, watches, etc.) or no-screen (Alexa, Nest, etc.) devices, and in situations where the user's hands are otherwise occupied (e.g., in surgery) there is a need for non-textual interaction paradigms to perform information searches or transactions without using a keyboard and instead by relying on spoken word interaction only. This class will discuss the technical foundations underlying Siri, Alexa, Cortana and the like. Specifically addressing query generation and response formulation stages of the pipeline.

NLP: Evaluation of Natural Language Processing Models (03/21)
While the evaluation of many natural language processing tasks relies on generic machine learning evaluation metrics (introduced in lecture 8). There are a range of concepts that are specific to the language domain. This class class will introduce the related notions of cross-entropy and perplexity and discuss the machine translation metrics BLEU and COMET.

NLP: Assignment 2 - Automatic Translation of Medical Text (04/02)
In this class we will have 10-minute presentations from each group, showcasing their solution to the natural language processing assignment (see below for details).

IR: Introduction and Key Concepts (04/04)
Information retrieval enables the efficient search for a small set of relevant items within

a vast collection of non-relevant material. Web search is a typical example. The user types (or utters) a keyword query and expects the top 10 most relevant websites (out of an overall collection of about 1 trillion pages available on the Internet) to be found in no more than 20 milliseconds. This proverbial needle-in-the-haystack setting introduces unparalleled demands towards efficiency and accuracy. In the opening class of this module, we will give a historic overview of key developments and trends in the fields, discuss typical search engine architectures and introduce standing challenges that will be encountered throughout the module.

IR: Web Crawling (04/09)
The first step to setting up a search engine that searches through hundreds of billions of Web pages or products is content discovery. The Internet is constantly being traversed by multitudes of Web crawlers (also known as spiders or bots) that try to find all the available content by following hyperlinks much like a human user would, and subsequently harvesting the information encountered there. In this class, we will analyze the mechanics of Web crawlers, and discuss content freshness, near-duplicates locality sensitive hashing.

IR: Indexing (04/11)
To make near-instantaneous response times possible, search engines do not simply save the information discovered by their Web crawlers in standard human-readable text files but instead rely on an advanced data structure, the so-called index. In this class we will discuss advanced indexing techniques, the merit of inverted, positional and compressed indices as well as the use of sharding to distribute large indices over multiple physical computers.
We will conclude this class by discussing the third assignment (see below for details).

IR: Statistical Ranking (04/16)
Now that we are aware of the available content thanks to our Web crawler, and have stored it in the beneficial format of our index, we need to locate the exact material that the searcher was looking for and bring it into an ordering from most to least promising. There are a multitude of statistical and heuristic frameworks that estimate the usefulness (the so-called relevance) of a document to the user's search query. This class will discuss term-frequency-based retrieval models, vector space models and fully stochastic rankers.

IR: Learning to Rank (04/18)
As an alternative to the traditional probabilistic retrieval models discussed in the previous class, more modern retrieval systems increasingly rely on machine learning techniques to approximate complex non-linear ranking functions. such techniques additionally allow search engines to autonomously learn from user interactions. This class will cover the notion of log file analysis, click-through data and machine-learning based rankers such as ranking SVMs or neural networks.

IR: Link Analysis (04/23)

Large portions of humanity's knowledge is captured in the form of highly interconnected graph structures (think of the graphs induced by billions of websites linking to each other or millions of scholarly research papers citing each other). We can gain lots of information about the content without ever reading any of the documents by merely studying the connectivity structure of the graph. In this class, we will introduce the discipline of network analysis and discuss methods such as the pagerank and HITS algorithms that help us explore such graphs in an unsupervised manner.

IR: Recommender Systems (04/25)
So far, this module has been concerned with query-driven searches in which the user (more or less) explicitly tells the system what they are looking for. There is an alternative paradigm under which the system receives no such input and is merely supposed to recommend material that the user will generally enjoy. Typical examples of such recommender system scenarios are product recommendation in e-commerce settings, or generation of song or movie playlists on multimedia platforms. This class will introduce the problem of item recommendation and discuss matrix factorization techniques for learning from previous interactions both by the user at hand and by others with similar preferences.

IR: Evaluation of Information Retrieval Systems (04/30)
Due to the unique problem setting and the typically massively skewed distribution of relevance across information retrieval collections, system evaluation relies on a somewhat different range of metrics. We will discuss the concepts of precision, recall and the F-measure and derive an evolution of information retrieval specific measures of system quality ranging from average precision to cumulative gain and reciprocal rank metrics.

IR: Assignment 3 - Clinical Decision Support (05/02)
In this class we will have 10-minute presentations from each group, showcasing their solution to the information retrieval assignment (see below for details).

Recap (05/07)
The final session will begin with a brief recapitulation of the material covered in this course. There will be a brief (20-minute) mock exam in the same style as the real one. We will close by discussing the mock exam's model solution and any other open questions.

Graded Assignments

There will be three graded assignments, collected in lectures 9, 17, and 26 as the capstone to the respective thematic module. Assignments will have a teamwork component in which groups of three students will **jointly** develop a practical solution to a biomedical AI problem (see below for more details on those problems), and an individual component, in which each student will **individually** described their team's technical solution and address a number of questions related to the assignment's

problem domain. Graduate students taking this class will receive an additional question in this individual part of the assignment.

In the lecture of the assignment due date, teams will present their technical solution to the instructor and their peers in the form of a jointly developed 10-minute oral presentation. Teams will consist of three students and there will be three assignments throughout the semester. Every student will be expected to give one of these presentations.

Each assignment will constitute 20% of the course's overall grade, composed of four equally weighted components:

- **Presentation (25%)** - How clearly is the presentation structured and presented? Is all relevant information covered? Are questions answered proficiently? (team-wide grade)
- **Code quality (25%)** - Is the program code understandable, well documented and correct? (team-wide grade)
- **Report (25%)** - Is the report complete, correct and understandable? Are the general questions answered correctly? (individual grade)
- **System performance (25%)** - As an incentive for building competitive systems, the teams' systems will be pitted against each other and the final grade component will reflect how well the team oes in comparison to their peers in terms of a known performance metric. (team-wide grade)

The three assignments are:

1. **Readmission Prediction from ICU Records (due lecture 9).** Readmission after hospital discharge is a major burden to the healthcare system that can be avoided via correct initial treatment or delayed discharge. Students will be given a broad dataset of patient properties such as demographics, chief complaints, lab values and vital parameters. The task will be to develop a machine learning system that predicts whether or not the same patient will soon be readmitted to the hospital with the same chief complaints.
2. **Automatic Translation of Medical Text (due lecture 17).** The global biomedical publishing system becomes more and more locally fragmented with significant portions of original publications appearing in non-English outlets. The students will receive a parallel corpus of medical scholarly publications and will be tasked to develop a machine translation system that is capable of automatically translating Bulgarian research articles into English.
3. **Clinical Decision Support (due lecture 26).** The amount of clinically relevant published material is rapidly growing with more than 1 million new scholarly articles appearing every year. Clinical decision support systems help physicians by selecting those few publications that are most closely related to the case at hand. The students will receive a collection of 26 million research articles and will

be asked to develop a search engine that retrieves those articles from the pool that are related to a number of artificial patient descriptions.

Final Exam

The final exam will constitute 40% of the overall course grade. It will be held in the form of a multiple-choice open-book test. No programming will be needed for this exam. In the final lecture of the course, the instructor will provide a mock exam that will be representative of the final exam and for which an example solution will be provided and discussed.

Students with Special Needs

Brown University is committed to full inclusion of all students. Students who, by nature of a documented disability, require academic accommodations should contact the professor during office hours. Students may also speak with Student and Employee Accessibility Services at 401-863-9588 to discuss the process for requesting accommodations.

Diversity Statement

This course is designed to support an inclusive learning environment where diverse perspectives are recognized, respected and seen as a source of strength. It is our intent to provide materials and activities that are respectful of various levels of diversity: mathematical background, previous computing skills, gender, sexuality, disability, age, socioeconomic status, ethnicity, race, and culture.

Expenses and Financial Concerns

Undergraduates with concerns about the non-tuition cost(s) of this course, may apply to the Dean of the College Academic Emergency (E-Gap) Fund to determine options for financing these costs (while ensuring their privacy). For information: https://www.brown.edu/academics/college/advising/financial-advising/e-gap-funds