****

**School of Computer Science and Engineering**

Syllabus

**CS 618, Deep Learning**

**Late Spring 2023**

**Class meeting: Monday 6:30 to 9:15 Pm**

**Credits:** 3 credits

**Instructor:** Mostafa Omar

**Office hours:** Monday-Thursday, 5-10:00PMby appointment

**Phone:** (216) 9714139

**Email:** omarm2@sacredheart.edu

|  |
| --- |
|  |

**Course Description** :

This course provides an introduction to the field of deep learning, which is a subfield of machine learning that uses neural networks to model and solve complex problems. The course will cover the fundamental concepts and techniques of deep learning, including neural network architectures, optimization methods, regularization techniques, and deep learning frameworks.

Students will learn to design and implement deep learning models for a range of applications, such as image and speech recognition, natural language processing, and reinforcement learning. The course will include hands-on programming assignments using popular deep learning libraries such as TensorFlow and PyTorch, and students will gain practical experience training and evaluating deep neural networks

This course will help you:

* Gain a solid understanding of deep learning concepts and techniques: Deploy a structured lifecycle approach to data analytics problems
* Develop practical skills: A deep learning course will give you hands-on programming experience using popular deep learning libraries such as TensorFlow
* Overall, a deep learning course can help you develop the skills and knowledge needed to excel in a career in artificial intelligence, machine learning, and data science.

**Topics Covered:**

* Introduction to neural networks and deep learning
* Multilayer perceptrons and backpropagation
* Convolutional neural networks for image processing
* Recurrent neural networks for sequence modeling
* Autoencoders and generative models
* Reinforcement learning and policy gradients
* Optimization and regularization techniques
* Advanced topics in deep learning, such as attention mechanisms, transfer learning, and adversarial training
* Deep learning frameworks and tools, such as TensorFlow, PyTorch, and Keras
* Applications of deep learning, such as image classification, speech recognition, natural language processing, and game playing

**Course Prerequisites:**

* Basic knowledge of linear algebra, calculus, and probability theory
* Experience with a programming language such as Python
* Familiarity with machine learning concepts such as supervised and unsupervised learning, and overfitting and underfitting.

**Assessment:**

Assessment for this course may include regular programming assignments, and exams, where students will implement a deep learning model for application.

Bottom of Form

**Required Texts/Source Materials/Readings and References**:

* "Deep Learning" by Ian Goodfellow, Yoshua Bengio, and Aaron Courville:

This comprehensive book provides a detailed introduction to the field of deep learning, covering topics such as neural networks, optimization algorithms, and generative models. It is widely regarded as a must-read for anyone interested in deep learning.

**Publisher**:  The MIT-Press

**Print ISBN**-13: 978-0262035613

* "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" by Aurélien Géron:

This practical book provides a hands-on approach to learning deep learning using popular Python libraries such as Scikit-Learn, Keras, and TensorFlow. It covers topics such as convolutional neural networks, recurrent neural networks, and reinforcement learning.

**Publisher**: O'Reilly Media, Inc.

**Pub. Date**: March 30, 2017

**Print ISBN**-13: 978-1-4919-6229-9

* "Python Machine Learning" by Sebastian Raschka and Vahid Mirjalili:

This book covers a wide range of machine learning topics, including deep learning, using the Python programming language. It provides a practical, hands-on approach to learning deep learning, with examples and case studies throughout the book.

Publication date: December 2019

Publisher: Packt

ISBN:9781789955750

* "Deep Learning for Computer Vision" by Rajalingappaa Shanmugamani:

This book focuses specifically on deep learning for computer vision applications, covering topics such as image classification, object detection, and semantic segmentation. It provides practical examples and detailed explanations of the underlying concepts.

Publication date: January 2018

Publisher: Packt

ISBN: 9781788295628

* "Grokking Deep Learning" by Andrew Trask:

This book takes a unique approach to teaching deep learning, using a visual, intuitive approach to explain the underlying concepts. It covers topics such as backpropagation, convolutional neural networks, and recurrent neural networks.

Manning publications, 2019  
ISBN-13: 978-1617293702

**Course Web Page:** All announcement, email notifications, class material, and grading will be done through Blackboard. Check for announcements and updates daily. Use only your SHU email address.

**Attendance:**

Attendance is mandatory in person.

**Grading Policy:**

|  |  |
| --- | --- |
| **Assessment Criteria** | **Weight** |
| Programming Assignments, classwork and attendance | 40% |
| Midterm Exam | 30% |
| Final Project | 30% |

**Grading Schema:**

|  |  |
| --- | --- |
| 93-100 | A |
| 90-92 | A- |
| 87-89 | B+ |
| 83-86 | B |
| 80-82 | B- |
| 77-79 | C+ |
| 70-76 | C |
| Below 70 | F |

**Tentative Course Schedule**:

|  |  |
| --- | --- |
| NO | **Lecture Topic** |
| 1 | Introduction to Deep Learning   * What is deep learning * Why is deep learning important * Applications of deep learning * Overview of deep learning frameworks |
| 2 | Neural Networks   * Introduction to neural networks * Types of neural networks * Multilayer perceptrons * Activation functions * Forward and backward propagation |
| 3 | Convolutional Neural Networks   * Introduction to ConvNets * Convolutional layers * Pooling layers * Transfer learning with pre-trained ConvNets * Applications of ConvNets in computer vision |
| 4 | * Regression Models * Classification Models |
| 5 | Training Deep Neural Networks   * Overfitting and regularization * Loss functions * Optimization algorithms * Hyperparameter tuning |
| 6 | Midterm |
| 7 | Recurrent Neural Networks   * Introduction to RNNs * Unrolling RNNs * LSTMs and GRUs * Applications of RNNs in natural language processing |
| 8 | Image processing |
| 9 | Generative Models   * Introduction to generative models * Autoencoders * Variational autoencoders * Generative adversarial networks |
| 10 | Advanced Topics in Deep Learning   * Attention mechanisms * Transformers * Reinforcement learning * Generative models for 3D data |
| 11 | Analytical methods: Text Analysis   * Text Analysis steps, * collecting raw data, * Term frequency-Inverse Document frequency, * categorizing documents by topic, * Determining sentiment |
| 12 | Final Exam |

**Changes to the syllabus:** This syllabus and course outline is subject to change by the instructor during the course of the semester. Changes may be necessary because of students’ specific interest(s), the general class progression and emerging topics of interest. If such changes are implemented, they will be announced in class and posted to Blackboard if used in the course.

**Academic Integrity Policy:** The University has a standing policy in place with regard to academic integrity. As stated in the University policy, this requires on the part of students a commitment to the fundamental values of honesty, trust, fairness, respect and reasonability. All students are expected to familiarize themselves with the policy and be in compliance. This policy can be found online at <http://www.sacredheart.edu/officesservices/registrar/academicintegritypolicy/>. In addition to this policy, the Department of Computer Science and Information Technology has the following guidelines.

* At no time is a student permitted to turn in work belonging to someone else and represent it as his/her own. This includes material from another student, the internet, a book or any other source not your own.
* Any material “borrowed” from the internet, must be cited in a student’s assignment. This may take the form of images, graphic designs or other material allowed by a professor. Any time that material is used from the internet, it must be explicitly cited. For example, if a student uses images “borrowed” from the internet as part of a web page, somewhere in the assignment the student must state “All of my images were downloaded from the internet.” Some faculty may require a specific bibliography of information used on assignments and others may allow a more general statement of information used.
* If a student downloads material from the internet and modifies it in some way to fit the assignment, he/she must cite the original work in his/her assignment.
* At all times, students must be able to explain code they have written. If it is suspected that a student has cheated (as defined by university policy) or submitted another’s work as his/her own, a faculty member may question a student on his/her code/assignment. The student must be able to explain the details of the assignment and nature of the solution.
* A student must not submit the same assignment to two different classes (in the same or different semesters) unless previously discussed with the instructor.
* It is acceptable for a student to discuss the meaning of an assignment and how he/she will plan to go about solving the problem with other students, however, every assignment must be completed independently unless otherwise stated.
* A student must not create program output that is inconsistent with the actual output of submitted assignments
* A student must not knowingly allow another student to hand in his/her work represented as his/her own.

If at any point you are unsure of what is allowed, ask your professor for clarification.

Anyone found violating this policy will be penalized as described in the university policy. At a minimum, students guilty of cheating on an assignment will receive a zero for that grade. Anyone found cheating on an exam will receive a failing grade for the course.