DO NOT TYPE IN THIS BOX
Bulletin # : 2
Academic Year : 2023-24

FLORIDA INTERNATIONAL UNIVERSITY
UNIVERSITY CURRICULUM COMMITTEE
Proposal for a New Course

1. **School/College** College of Engineering and Computing
   Div./Dept. in Which Taught Knight Foundation School of Computing and Information Sciences

2. | CAP | 4 | | | 3 |
   |---|---|---|---|---|
   | **Alpha Prefix** | **1st Digit** | **Last 3 Digits** | **"C"-lec-lab "L"-Lab** | **Cr. Hrs.** |

   CIP Code ***(Leave this blank):*** ______

3. **Grading Method (select one):** ☒ Graded ☐ Pass/Fail

4a. **Course Title** Introduction to Deep Learning

b. **Abbreviated course Title** ***(for computer class schedules, transcripts)*** Intro to Deep Learning
   LIMITED TO 25 Characters *(including spaces)*

5. **Statewide Course Numbering Subject Matter Area** Computer Applications

6. **Catalog Description/Major Topics** *(not to exceed 200 characters including spaces)*
   College of Medicine and College of Law: Attach description not exceeding 1,000 characters including spaces.

   This course introduces the fundamentals of deep learning, covering feed forward neural networks, convolutional neural networks (CNNs), recurrent neural networks (RNNs), and other advanced models.

7. **Attach detailed syllabus course outline and course justification on separate page(s).**

8. **Prerequisite(s):** (STA 3033 or STA 2023 or STA 2122 or STA 4322) and (COP 3465 or COP3530)

9. **Corequisite(s):** ______

10. **Objective(s) of Course:**

    Identify the role of deep learning within the fields of AI and machine learning.
    Optimize neural networks design and parameters.
    Apply deep learning to solve a variety of real-world problems.

11. **Does this course duplicate/overlap other courses at FIU?** ☒ No ☐ Yes
    If yes, please explain: ______

12. **What other closely related department(s) have been consulted about this course?**
    ______

13. **Is this course used for the assessment of a program or a certificate (if yes, then send a notification to assessment@fiu.edu)?** ☑ No ☐ Yes

PROPOSAL REQUESTED BY:

Faculty Contact Kaoutar Ben Ahmed — [signature] — 10 / 11 / 2023
(Type name) **(Signature)**
kbenahme@fiu.edu — 305-348-7984
(Email address) **(Phone number)**

Chairperson (Dept./Div.) Jason Liu — [signature] — 10 / 13 / 2023
(Type name) **(Signature)**

Chairperson (Curr. Comm.) Alex Afanasyev — [signature] — 10 / 17 / 2023
(Type name) **(Signature)**

College/School Dean Mark Weiss, Assoc Dean — Mark Weiss — 10 / 20 / 2023
(Type name) **(Signature)**

Submit one original form. Attach one copy of the course justification and a draft of the course syllabus for this New Course Proposal. The syllabus should include the course description, objectives, learning outcomes, major topics, and textbooks.

Introduction to Deep Learning

New Course Justification

This course is designed to provide a comprehensive introduction to deep learning, one of the most promising and exciting fields in artificial intelligence. Deep learning has revolutionized the field of AI, allowing machines to learn from massive amounts of data and make decisions that were previously thought to be the exclusive domain of human experts. This course will cover the basics of deep learning, including neural networks, backpropagation, convolutional neural networks, recurrent neural networks, and transfer learning.
By the end of this course, students will have a deep understanding of the fundamental principles of deep learning, as well as practical experience in implementing and applying deep learning algorithms. They will be well-prepared to pursue further study in this exciting and rapidly growing field, as well as to apply deep learning techniques to a wide range of real-world problems.

FIU | Knight Foundation School of Computing and Information Sciences

September 8th, 2023

Subject: Memorandum of Understanding between the Knight Foundation School of Computing and Information Sciences and the Department of Mathematics & Statistics regarding a new BS in Data Science

To Whom It May Concern:

The Knight Foundation School of Computing and Information Sciences (KFSCIS) is proposing a Bachelor of Science in Data Science, and the KFSCIS committee in charge of that proposal has discussed this with relevant leadership within the Department of Mathematics & Statistics (DM&S). This Memorandum of Understanding is intended to capture the content of that discussion and agreement.

1. In general, both DM&S and KFSCIS express their sincere desire to maintain collaborative, productive, collegial, and friendly relations between the units in service of providing our students with as many of the highest quality and flexible educational options as possible.

2. In view of the above, DM&S has no objection to KFSCIS creating a Bachelor of Science in Data Science. The degree name was agreed to be "Bachelor of Science in Data Science" to clearly distinguish it from the DM&S's major in Mathematical Data Science, to show that it is an approach to Data Science that emphasizes computing and information sciences, and to distinguish it from a more mathematical course of study.

3. DM&S, in general, also has no objection to KFSCIS creating new Data-Science-related courses that overlap with existing DM&S offerings, as long as those courses are named and designed in such a way as to clearly indicate the computing and information sciences focus of the offering. For example, DM&S is supportive of KFSCIS offering the following courses in their new degree: "Introduction to Data Science" and "Advanced Data Science".

4. Regardless of the above, both units agree to continue to provide to the other unit's leadership, in accordance with the usual FIU processes, any other new course proposals that overlap with courses in the other unit, for their review and consent.

5. The new degree lists several required mathematics courses. DM&S is willing to serve KFSCIS students in these courses with the understanding that resources are available to DM&S to perform this service, such as: MAS 3105 - Linear Algebra (as an alternative to MAC 2313 - Calculus III), MAD 2104 - Discrete Mathematics (as an alternative to COT 3100), and STA 3163 - Statistical Methods I, STA 3164 - Statistical Methods II, STA 4234 - Introduction to Regression Analysis, MAD 3301 - Graph Theory, MAD 3401 - Numerical Analysis, and MAD 4203 - Combinatorics for a concentration in Statistical Modeling.

6. Finally, KFSCIS had no objection to DM&S proposing a new major in 2022 focused on Mathematical Data Science inside their existing B.S. in Mathematical Sciences, homed in DM&S. DM&S will provide the details of any further new proposed courses or major which overlap with Computer Science and consult with KFSCIS in accordance with the usual FIU processes.

Louis Tebou
Chair, Department of Mathematics & Statistics

Louis Tebou
September 8, 2023

Jason X. Liu
Director, Knight Foundation School of Computing and Information Sciences

[signature], 9/8/2023

Knight Foundation School of Computing and Information Sciences
11200 S.W. 8th St. • Modesto A. Maidique, CASE 352 • Miami, FL 33199 • Tel: 305-348-2744 • Fax: 305-348-3549 • www.cs.fiu.edu

Knight Foundation School of Computing and Information Sciences

Course Title: Introduction to Deep Learning **Date:** 10/16/2023

Course Number: CAP 4XXX

Number of Credits: 3

Subject Area: Artificial Intelligence	**Subject Area Coordinator:** Leonardo Bobadilla **Email:** bobadilla@cs.fiu.edu
Catalog Description: This course introduces the fundamentals of deep learning, covering feed forward neural networks, convolutional neural networks (CNNs), recurrent neural networks (RNNs), and other advanced models.	
Textbooks: " Introduction to Deep Learning" (2018) Sandro Skansi ISBN: 978-3-319-73004-2	
References:" Introduction to Deep Learning" (2018) Sandro Skansi ISBN: 978-3-319-73004-2	
Prerequisites Courses: (STA 3033 or STA 2023 or STA 2122 or STA 4322) and (COP 3465 or COP 3530)	
Corequisite Courses: None	

Type: Core Course for BS in Data Science; Elective for CS (Applications) and CY Majors.

Prerequisites Topics:
1. Basic statistics and probability concepts.
2. Data structures.
3. Programming languages.

Course Outcomes:
1. Identify the role of deep learning within the fields of AI and machine learning.
2. Describe the basic components of a neural network and how they are used to model complex relationships between input and output data.
3. Implement and train basic neural networks using popular deep learning frameworks such as TensorFlow.
4. Analyze and optimize deep neural network design and hyperparameters to enhance performance.
5. Solve a variety of real-world problems by applying deep learning techniques.
6. Evaluate the principles underlying advanced neural network architectures like convolutional neural networks (CNNs) and recurrent neural networks (RNNs).
7. Critically assess current research trends and developments in deep learning, and assess their implications for the future of artificial intelligence.
8. Collaborate with peers on hands-on projects and assignments to develop practical experience with deep learning concepts and techniques.

Knight Foundation School of Computing and Information Sciences
CAP 4XXX Introduction to Deep Learning

Association between Student Outcomes and Course Outcomes

BS in Computing: Student Outcomes	Course Outcomes
1) Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions.	5, 6, 7, 8
2) Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.	5, 6, 7, 8
3) Communicate effectively in a variety of professional contexts.	8
4) Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles.	8
5) Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline.	8

Program Specific Student Outcomes

6) Apply theory, techniques, and tools throughout the data science lifecycle and employ the resulting knowledge to satisfy stakeholders' needs.	5, 6, 7, 8

6) Provide students with a strong foundation in the mathematical and computational concepts used in data science.	1, 2, 3, 4

6) Apply the skills and knowledge they have gained to real-world problems and make data-driven decisions.	8

Assessment Plan for the Course and how Data in the Course are used to assess Student Outcomes

Student and Instructor Course Outcome Surveys are administered at the conclusion of each offering, and are evaluated as described in the School's Assessment Plan: https://abet.cis.fiu.edu/

Knight Foundation School of Computing and Information Sciences
CAP 4XXX Introduction to Deep Learning

Outline

Topic	No. of Lecture Hours	Course Outcomes
• Introduction to Supervised Learning & classification o Introduction to AI and machine learning o Supervised learning methods Classification Regression	3	1, 2
• Multilayer Perceptron (MLP) o The Perceptron o Feedforward neural network o Backpropagation o Activation functions o Design and Train an MLP	9	1, 2, 3
• Deep learning o What is deep learning? o History & Applications o Introduction to deep learning in Python and Keras	3	1, 2, 3, 7
• Optimization techniques o Stochastic gradient decent o Learning rate schedules o Regularization o Overfitting avoidance methods	6	2, 3, 4
• Convolutional Neural Network (CNN) o Motivation & advantages o Convolutional & Pooling layers o Design & train a CNN for image analysis	6	5, 6, 7, 8
• Autoencoder o Motivation & application	3	3, 5, 8
• Recurrent Neural Network (RNN) o Motivation & Advantage o Basic RNN & their limitation o LSTM networks	3	5, 6, 7, 8
• Advanced topics o Transfer learning o Ensemble learning o Shortcut learning o Generative adversarial networks & Deep fake o Deep reinforcement learning o Transformers	6	5, 6, 7, 8

Performance Measures for Evaluation

All assignments are assigned through the Canvas course site.

- Late Work Policy: Late submissions will receive a 10% automatic deduction for every day past the due date. The deduction will continue until 3 days past the due date. The assignment is automatically closed on the third day at midnight. The late work policy is non-negotiable.
- Policy Regarding Contesting a Grade: You will have one week (seven calendar days) following the posting of a grade to contest the grade. If the grade is not contested by 5 pm (Eastern) on the seventh day, then the grade posted will stand as final. If the seventh day falls on a holiday, then you will have until the next business day. For purposes of contesting a grade, an email to the email address listed above with a subject line of CONTESTING MY GRADE and a body with your name, the course, the assignment, and a brief explanation of why you are contesting the grade shall constitute notice of your intention to contest a grade.

Assignment	Total Points	Percentage of Final Grade
Homework (5)	100 each	15%
Exams (2)	100 each	35%
Projects (3)	100 each	25%
Final Project	100	25%
	TOTAL	100%

Letter Grade Distribution Table

Letter	Range%		Letter	Range%		Letter	Range%
A	95 or above		B	83 - 86		C	70 - 76
A-	90 - 94		B-	80 - 82		D	60 - 69
B+	87 - 89		C+	77 - 79		F	59 or less

Knight Foundation School of Computing and Information Sciences
CAP 4XXX Introduction to Deep Learning

Description of Possible Projects

Project 1: Implementing a Neural Network from Scratch

Criteria	Excellent (100)	Good (80)	Average (60)	Below Average (40)	Poor (20)
Neural Network Implementation (30%)	Code implements a neural network with correct forward and backward passes, weight initialization, and activation functions.	Code implements a neural network with mostly correct forward and backward passes, weight initialization, and activation functions.	Code implements a neural network with some errors in forward and backward passes, weight initialization, or activation functions.	Code implements a neural network with significant errors in forward and backward passes, weight initialization, or activation functions.	Code does not implement a neural network or has major errors.
Training and Convergence (25%)	Network trains successfully and converges to a good solution with clear evidence of correct hyperparameter tuning.	Network trains successfully and converges to a reasonable solution with some hyperparameter tuning.	Network trains but has difficulties converging or suboptimal hyperparameter choices.	Network struggles to train or does not converge, with poor hyperparameter choices.	Network does not train at all.
Performance Metrics (20%)	Achieves excellent performance metrics (e.g., accuracy, loss) on the synthetic dataset with insightful analysis.	Achieves good performance metrics on the synthetic dataset with adequate analysis.	Achieves acceptable performance metrics on the synthetic dataset with limited analysis.	Achieves poor performance metrics on the synthetic dataset with minimal analysis.	Fails to achieve any meaningful performance metrics.
Code Quality and Readability (25%)	Code is well-structured, well-documented, and follows best practices, making it easy to understand.	Code is mostly well-structured and documented but may lack some clarity.	Code is organized but may lack proper documentation and readability.	Code lacks structure, documentation, and readability, making it challenging to understand.	Code is disorganized and entirely lacking documentation

Knight Foundation School of Computing and Information Sciences
CAP 4XXX Introduction to Deep Learning

Project 2: Convolutional Neural Networks (CNNs) for Image Classification

Criteria	Excellent (100)	Good (80)	Average (60)	Below Average (40)	Poor (20)
CNN Architecture (30%)	Implements a CNN with a well-structured architecture, appropriate layers, and effective use of pre-trained models for transfer learning if applicable.	Implements a CNN with a sound architecture, suitable layers, and some use of pre-trained models for transfer learning if applicable.	Implements a CNN with a basic architecture, limited layer choice, or minimal use of pre-trained models for transfer learning if applicable.	Implements a CNN with a poor architecture, inadequate layer choice, or no use of pre-trained models for transfer learning if applicable.	Does not implement a CNN or has a severely flawed architecture.
Data Preprocessing (25%)	Preprocesses the image data effectively with data augmentation techniques where necessary, leading to improved model performance.	Preprocesses the image data adequately, with some data augmentation, contributing to reasonable model performance.	Performs basic data preprocessing with limited augmentation, resulting in modest model performance.	Performs minimal data preprocessing and lacks data augmentation, leading to subpar model performance.	Neglects data preprocessing and augmentation, severely impacting model performance.
Model Training and Evaluation (30%)	Successfully trains the CNN, achieves high accuracy, and provides detailed evaluation metrics and insightful analysis.	Trains the CNN with reasonable accuracy, offers evaluation metrics, and provides some analysis.	Manages to train the CNN with moderate accuracy, but evaluation metrics and analysis are limited.	Struggles to train the CNN with low accuracy, and evaluation metrics and analysis are lacking.	Fails to train the CNN effectively and does not provide any evaluation or analysis.
Code Quality and Readability (15%)	Code is well-structured, well-documented, follows best practices, and is easy to understand.	Code is mostly well-structured and documented but may lack some clarity.	Code is organized but may lack proper documentation and readability.	Code lacks structure, documentation, and readability, making it challenging to understand.	Code is disorganized and entirely lacking documentation.

Knight Foundation School of Computing and Information Sciences
CAP 4XXX Introduction to Deep Learning

Project 3: Sequence-to-Sequence Models for Natural Language Processing

Criteria	Excellent (100)	Good (80)	Average (60)	Below Average (40)	Poor (20)
Model Architecture (30%)	Implements a sequence-to-sequence model (RNN or Transformer) with a well-structured architecture, correct attention mechanisms, and effective embeddings.	Implements a sequence-to-sequence model with a sound architecture, attention mechanisms, and reasonable embeddings.	Implements a sequence-to-sequence model with a basic architecture, limited attention mechanisms, or suboptimal embeddings.	Implements a sequence-to-sequence model with a poor architecture, inadequate attention mechanisms, or incorrect embeddings.	Does not implement a sequence-to-sequence model or has a severely flawed architecture.
Data Preprocessing (25%)	Preprocesses the text data effectively, handles tokenization, padding, and attention masks properly, leading to improved model performance.	Preprocesses the text data adequately, handles tokenization, padding, and attention masks, contributing to reasonable model performance.	Performs basic data preprocessing with some issues in tokenization, padding, or attention masks, resulting in modest model performance.	Performs minimal data preprocessing and struggles with tokenization, padding, or attention masks, leading to subpar model performance.	Neglects data preprocessing, tokenization, padding, or attention masks, severely impacting model performance.
Model Training and Evaluation (30%)	Successfully trains the sequence-to-sequence model, achieves high performance (e.g., BLEU score, ROUGE score), and provides detailed evaluation metrics and insightful analysis.	Trains the sequence-to-sequence model with reasonable performance, offers evaluation metrics, and provides some analysis.	Manages to train the sequence-to-sequence model with moderate performance, but evaluation metrics and analysis are limited.	Struggles to train the sequence-to-sequence model with poor performance, and evaluation metrics and analysis are lacking.	Fails to train the sequence-to-sequence model effectively and does not provide any evaluation or analysis.
Code Quality and Readability (15%)	Code is well-structured, well-documented, follows best practices, and is easy to understand.	Code is mostly well-structured and documented but may lack some clarity.	Code is organized but may lack proper documentation and readability.	Code lacks structure, documentation, and readability, making it challenging to understand.	Code is disorganized and entirely lacking documentation.