

INTELIGENCIA COMPUTACIONAL

Materia:	Inteligencia computacional
Clave y Grupo:	TC3023
Horario:	Jueves, 16:00 – 18:55 hrs.
Salón:	4509
Modalidad del Curso:	Curso con proyecto – POI (Project Oriented Learning) Curso en el que se recomienda incorporar el uso de tecnologías de información.
Profesor:	Octavio Navarro Hinojosa
Correo Electrónico:	octavio.navarro@tec.mx
Ubicación:	Aulas 3, Piso 0, Laboratorio de Realidad Virtual y Aumentada
Horario de Asesoría:	Lunes, Jueves, 10:00 – 12:00 hrs Martes, Miércoles, Viernes: 10:00-13 hrs

OBJETIVO GENERAL

Es un curso de nivel avanzado en el campo de estudio de la computación e intermedio para la concentración de sistemas inteligentes y computación visual. Como producto de aprendizaje de este curso se espera que los estudiantes sean capaces de modelar y resolver problemas utilizando aprendizaje profundo.

OBJETIVOS ESPECÍFICOS

- Que los alumnos sean capaces de analizar las necesidades específicas de un problema, y puedan implementar un marco de trabajo para solucionarlo con aprendizaje profundo.
- Que los alumnos entiendan la importancia de la experimentación, así como del papel del hardware y los datos para el aprendizaje profundo.
- Que los alumnos utilicen herramientas como Tensorflow y keras para implementar sistemas completos de aprendizaje profundo.
- Que los alumnos utilicen los conceptos del curso para desarrollar un sistema de visión para un vehículo autónomo.

CONTENIDO TEMÁTICO

1. Introducción al aprendizaje profundo
2. Redes neuronales
 - Fundamentos matemáticos
 - Representación de datos
 - Gradient descent
 - Backpropagation
3. Redes neuronales profundas
 - Tensorflow
 - Keras
 - Hyperparámetros
 - Transfer learning
4. Redes neuronales convolucionales
 - Capas convolucionales
 - Max pooling
 - Visualización de características
 - Faster R-CNN
 - YOLO
5. Deep reinforcement learning
 - Reinforcement learning
 - Q learning
 - OpenAI Gym
 - Deep Q-learning
 - Policy gradients

RECURSOS DIDÁCTICOS

→ Libros de Texto

1. Goodfellow, Ian, et al. Deep learning. Vol. 1. Cambridge: MIT press, 2016.
2. Chollet, Francois. Deep learning with python. Manning Publications Co., 2017.
3. Gulli, Antonio, and Sujit Pal. Deep Learning with Keras. Packt Publishing Ltd, 2017.
4. Lapan, Maxim. Deep Reinforcement Learning Hands-On: Apply modern RL methods, with deep Q-networks, value iteration, policy gradients, TRPO, AlphaGo Zero and more. Packt Publishing Ltd, 2018.

→ Plataforma Tecnológica

El material del curso, así como las actividades a desarrollar estarán disponibles en la plataforma Blackboard, y Github.

→ Calendario de Actividades

Semana	Temas	Tareas y/o actividades extra clase
1	Introducción al aprendizaje profundo	
2	Redes neuronales	Tarea 1
3	Redes neuronales	
4	Redes neuronales profundas	Tarea 2
5	Redes neuronales profundas	
6	Redes neuronales profundas	Tarea 3
7	Redes neuronales convolucionales	
8	Redes neuronales convolucionales	Tarea 4
9	Redes neuronales convolucionales	
10	Redes neuronales convolucionales	Tarea 5
11	Deep reinforcement learning	
12	Deep reinforcement learning	Tarea 6
13	Deep reinforcement learning	
14	Aplicaciones para autos autónomos	Tarea 7
15	Aplicaciones para autos autónomos	

→ Evaluaciones

Primer Parcial: TBD

Segundo Parcial: TBD

Final: TBD

PRÁCTICAS, Y PROYECTO FINAL

Serán desarrolladas de manera individual. Pueden haber tareas escritas, o de programación.

De cada programa:

- Deberá entregarse en el repositorio correspondiente en Github Classroom, acatando la fecha de entrega correspondiente.
- Se evaluará cada entrega en relación a una rúbrica que estará incluida con el readme de la misma.
- Los programas debe estar hechos en Python, utilizando jupyter notebooks, y solo hay que subir los archivos solicitados al repositorio.
- Solo subir al repositorio correspondiente el notebook donde trabajaron, tanto en formato pdf como en formato ipynb.
- Se considerará la estructura y presentación de cada notebook. El código y el contenido del mismo tiene que ser legible, utilizando comentarios, referencias, imágenes, y secciones relevantes. No es válido dejar todo el contenido del proyecto o tarea en una celda y usar comentarios solamente. Si no se sigue ésta indicación, se pierden 30 puntos.
- Si es entregada posterior a la hora de la clase indicada, será evaluada sobre 70.

De las tareas escritas:

- Subir al repositorio un reporte técnico en formato PDF, siguiendo el template proporcionado.
- Incluir las referencias e imágenes necesarias. Cualquier código, página web, libro, etc. que utilicen para solucionar las tareas tiene que ser referenciado en el documento.

Del proyecto final:

- Se desarrollará un proyecto entre dos personas como máximo.
- Para evaluar se va a seguir la rúbrica que será incluida en el mismo.

EVALUACIÓN DEL SEMESTRE

Parciales

Rúbrica	Porcentaje
Tareas y Prácticas	100%
Total	100%

Final

Rúbrica	Porcentaje
Calificaciones Parciales	60%
Proyecto Final	40%
Total	100%

OTRAS POLÍTICAS Y EL REGLAMENTO

- Este documento estará en el repositorio del curso, en la sección de Documentos. Lean este documento primero para cualquier duda con respecto a la forma de trabajar en el curso.
- Todos los materiales del curso estarán en el repositorio del curso.
- Todas las tareas se entregarán en Github.
- Para asesoría acudir en el horario indicado, o solicitar cita por correo electrónico. Se atenderá conforme vayan llegando. El tiempo dedicado no puede ser mayor a 30 minutos. Se debe asistir con dudas o preguntas específicas por resolver. En las asesorías las clases no se repiten ni se les resuelve la tarea. Si hay alguna duda del código y no pudieron tomar la asesoría, por favor envíen con tiempo de anticipación su código por email (al menos 4 días antes de la entrega para que pueda ser considerado a revisar en tiempos disponibles de horas de asesoría).
- NO SE REVISARA NINGUNA ACTIVIDAD ENVIADA POR DROPBOX, O SIMILARES.
- Si las tareas o proyectos están muy bien desarrollados o incluyen más allá de lo que se pidió pueden llegar a obtener un 10% extra de calificación (calificación máxima de 110).
- Las fechas y horas de entrega de tareas, proyectos y exámenes se respetaran sin excepción alguna.
- Las tareas deben estar completas y con buena presentación.
- La lista se pasa a más tardar a los 10 minutos de la hora de inicio de la clase vía sistema.
- No se justifican faltas a menos que el Director de Carrera lo indique por alguna razón fuera de lo normal.
- No falten! Cuiden sus faltas para emergencias, recuerden que se puede reprobar por faltas.
- Sus calificaciones se irán subiendo en los readme de las tareas y proyectos que se soliciten.
- Durante la clase, los celulares deberán ser apagados para no afectar el curso de la sesión. Al alumno que le suene el celular tendrá un punto menos en ese parcial.
- Recuerden que se debe seguir el reglamento del Tecnológico de Monterrey, en clase: NO se debe comer, NO se debe beber, NO se puede fumar, NO se puede oír música, NO se puede utilizar ningún chat, NO jugar videojuegos (a menos claro que sea un videojuego desarrollado por ustedes para la clase), NO entrar a facebook u otra red social, etc. Si se les sorprende infringiendo alguna de estas reglas se les restará un punto en ese parcial.
- Si se sorprende a alguien copiando tareas, prácticas o exámenes a otros compañeros tienen de calificación 0 (cero).
- Todos los estudiantes deben cumplir y hacer cumplir los reglamentos de la institución. Todo trabajo sometido por un estudiante en este curso debe ser fruto de su propio trabajo. Para este curso el trabajo colaborativo es permitido cuando la actividad así lo requiera.
- La calificación que obtengan en cada parcial y al final no se puede modificar ni en un sólo punto.
- NO hay trabajos extras para aumentar calificación en ningún momento del semestre. Los puntos medios suben (69.5 sube a 70, están reprobados con 69.49999 o menos).

DEL PROFESOR

Octavio Navarro Hinojosa

Educación:

2017-2018. Artificial Intelligence Nanodegree, Computer vision Nanodegree.

2012-2018. Doctorado en Ciencias Computacionales con especialidad en Gráficas y computo paralelo. Instituto Tecnológico de Estudios Superiores de Monterrey - Campus Estado de Mexico (ITESM-CEM).

2006 – 2010. Ingeniería en Sistemas Computacionales. Instituto Politécnico Nacional.

Experiencia:

2017 – Actual. Encargado del laboratorio de Realidad Virtual, supervisando el desarrollo de diversos proyectos. ITESM – CSF.

2014 – 2017. Profesor asistente. ITESM – CSF.

2015. Lider de desarrollo de un sistema de detección de manos utilizando el Kinect. ITESM – CSF.

2010 – 2012. Desarrollo de aplicaciones web. IBM México.

2007-2009. Asistente de investigación. IPN.