

## Proyecto de curso

**Curso:** Técnicas Avanzadas de Data Mining y Sistemas Inteligentes (INF659)  
**Docentes:** Dr. César Beltrán Castañón  
 Mag. César Olivares Poggi

El proyecto de curso permitirá que los estudiantes de postgrado apliquen su experiencia en el diseño, implementación y evaluación de algoritmos de aprendizaje profundo. Para su elaboración se conformará grupos de hasta cuatro alumnos cada uno. El objetivo principal de la actividad será la experimentación con metodologías del estado del arte en aprendizaje profundo aplicando buenas prácticas de investigación. Se requerirá reportar y reproducir a manera de línea base la experimentación con un modelo relevante del estado del arte, y luego realizar una aplicación propia a un problema de su elección aplicando las mejores prácticas en investigación.

El proyecto tendrá un peso del 50% de la nota final y constará de los siguientes entregables:

Actividad/Entregable	Fecha límite de entrega	Puntaje
1. Jupyter Notebook con ejecución comentada del modelo base y propuesta de proyecto de aplicación	4 de noviembre	4 puntos
2. Primera parte del informe escrito (Introducción, estado del arte y metodología), en formato IEEE, tamaño A4, máximo 3 páginas de extensión	11 de noviembre	3 puntos
3. Jupyter Notebook con ejecución comentada del proyecto de aplicación	25 de noviembre	3 puntos
4. Informe final, en formato IEEE, tamaño A4, máximo 8 páginas de extensión	2 de diciembre	7 puntos
5. Diapositivas y presentación en clase del proyecto de aplicación	9 de diciembre	3 puntos

### Elección del tema

La primera parte del trabajo consiste en entender a cabalidad y ejecutar el código de implementación de un artículo del estado del arte en aprendizaje profundo.

Cada grupo deberá elegir uno de los temas que se indica a continuación. Para todos ellos se indica como referencia la sección correspondiente del directorio Papers With Code en el cual podrán encontrar implementaciones en código abierto de artículos recientes del estado del arte: <https://paperswithcode.com/sota>

Cada tema podrá ser elegido por un máximo de 2 equipos, pero deberán tomar como base un artículo/modelo diferente.

- Visión computacional / Detección de objetos  
<https://paperswithcode.com/task/object-detection>
- Visión computacional / Segmentación semántica  
<https://paperswithcode.com/task/semantic-segmentation>
- Audio / Clasificación de audio  
<https://paperswithcode.com/task/audio-classification>
- Procesamiento de lenguaje natural / Análisis de sentimiento  
<https://paperswithcode.com/task/sentiment-analysis>
- Visión computacional / Generación de imágenes  
<https://paperswithcode.com/task/image-generation>

- Visión computacional / Transferencia de estilo  
<https://paperswithcode.com/task/style-transfer>
- Metodología / Aprendizaje one-shot  
<https://paperswithcode.com/task/one-shot-learning>
- Visión computacional / Clasificación de imágenes  
<https://paperswithcode.com/task/image-classification>
- Visión computacional / Estimación de pose  
<https://paperswithcode.com/task/pose-estimation>
- Procesamiento de lenguaje natural / Traducción automática  
<https://paperswithcode.com/task/machine-translation>
- Procesamiento de lenguaje natural / Modelamiento de lenguaje  
<https://paperswithcode.com/task/language-modelling>
- Metodología / Representación distribuida de palabras (Word Embeddings)  
<https://paperswithcode.com/task/word-embeddings>
- Metodología / Aprendizaje por transferencia  
<https://paperswithcode.com/task/transfer-learning>
- Reconocimiento del habla  
<https://paperswithcode.com/task/speech-recognition>

## Ejecución de modelo base

El primer entregable consistirá en la presentación de un Jupyter Notebook en el que se muestre haber ejecutado satisfactoriamente la implementación del modelo base en TensorFlow, Keras o PyTorch:

- **Explicación del modelo.** El notebook debe exponer por separado, comentando en celdas de *markdown*, las principales funciones y secciones del código implementado.
- **Entrenamiento.** Se debe mostrar el entrenamiento del modelo con al menos uno de los conjuntos de datos con los que fue experimentado en el artículo de referencia. Por motivos de capacidad computacional puede ser suficiente mostrar el entrenamiento del modelo por un número reducido de épocas.
- **Evaluación del aprendizaje.** Se debe mostrar métricas y una gráfica de la curva de aprendizaje del modelo tanto en el conjunto de entrenamiento como en el conjunto de validación. Por motivos de eficiencia computacional, las mediciones en el conjunto de validación pueden tomarse con menor granularidad que en el conjunto de entrenamiento.
- **Carga de pesos del modelo entrenado.** Diversas implementaciones vienen acompañadas de pesos pre-entrenados o *checkpoint*. De ser así, se debe mostrar haber cargado satisfactoriamente dichos pesos o *checkpoint* y haber utilizado el modelo pre-entrenado con muestras de ejemplo.

## Propuesta de proyecto

La propuesta de proyecto definirá el trabajo a ser realizado en la parte final del curso. Este trabajo consistirá en la aplicación del modelo base o una variación de éste a un problema de su elección aplicando las mejores prácticas en investigación. La propuesta de proyecto deberá incluir lo siguiente:

- Nombre del proyecto
- Objetivo del proyecto
- Conjunto(s) de datos a utilizar
- Artículos científicos relevantes
  - Referencia de al menos 4 artículos relevantes por su semejanza al problema a abordar.

## Informe del proyecto

El informe debe incluir la siguiente información:

- **Introducción**
  - Presentación del problema general sobre el que versará el trabajo, su dificultad y/o relevancia, y cómo se integra dentro del campo del aprendizaje automático
  - Objetivo del estudio
  - Organización del informe (secciones)
- **Estado del arte**
  - Breve síntesis del aporte que otros artículos científicos han realizado para este problema: cómo se ha abordado anteriormente este problema (antes y después de Deep learning), cuáles son las publicaciones con mejores resultados hasta la fecha y qué caracteriza a sus propuestas. Si existe una revisión sistemática o *survey* sobre el tema, indicarlo.
- **Metodología**
  - De ser el caso, qué se requiere en cuanto preprocesamiento de datos, selección y/o extracción de características.
  - Selección y justificación de la medida de calidad que se usará para evaluar el rendimiento del modelo.
  - Explicación y diagrama de la arquitectura del modelo que se propone.
  - Estrategia de validación a emplear para el ajuste de hiperparámetros (p.ej., búsqueda en grilla, búsqueda aleatoria, etc.).
  - Explicación y diagrama del proceso propuesto, desde la adquisición de los datos hasta la evaluación del rendimiento. Asegurarse de nunca emplear el conjunto de pruebas en tareas de entrenamiento, validación ni ajuste de hiperparámetros.
- **Experimentación y resultados**
  - Descripción del conjunto de datos
    - Origen del conjunto de datos
    - Según corresponda, número y tipo de características, dimensiones, resolución, profundidad, etc.
    - Número de muestras en los conjuntos de entrenamiento, validación y prueba. En caso aplique, número de muestras por clase.
  - Descripción del entorno de experimentación empleado (hardware y software).
  - Línea base: Reproducción de resultados reportados en una publicación anterior.
  - Reporte del entrenamiento del modelo con diversos hiperparámetros. Incluir curvas de aprendizaje de la función de pérdida en el conjunto de entrenamiento y de validación. Indicar configuración de los modelos, número de épocas de entrenamiento, tiempo de ejecución y evaluación del rendimiento usando la medida de calidad elegida.
- **Discusión**
  - Comparación de línea base y resultados propios.
  - Interpretación de los resultados obtenidos.
  - Identificación y visualización de ejemplos en los que tienen dificultad los modelos ensayados. ¿A qué se podría atribuir?
  - ¿Cómo podría ser mejorado su sistema?
- **Conclusiones y trabajos futuros**

## Presentación en clase del proyecto

En la última sesión del curso se realizará una presentación en clase del proyecto. La presentación tendrá una duración máxima de 25 minutos por grupo, y deberá ser acompañada de diapositivas que faciliten la comunicación de lo reportado en el informe del proyecto.

### Código y/o scripts (Jupyter Notebooks)

- El código de implementación deberá presentarse en Jupyter Notebooks, empleando Python con TensorFlow, Keras o PyTorch.
- Se recomienda usar GitHub como entorno de colaboración.
- Si el equipo desea trabajar en otro entorno, o en un lenguaje de programación diferente, deberá consultarlo previamente con el profesor.
- Se deberá asignar nombres representativos a los archivos, de manera que se pueda identificar su orden relativo y el propósito de cada uno.
- No se aceptará código simplemente copiado de otras fuentes.
- En caso de adaptar código existente, citar siempre la fuente.
- El código deberá estar mínimamente comentado.
- No se aceptará comentarios en otros idiomas.

### Recursos

- Buscador de literatura académica  
<https://scholar.google.com.pe/>
- Acceso a bases de datos electrónicas – PUCP  
<http://biblioteca.pucp.edu.pe/recursos-electronicos/bases-de-datos/>
- Formato IEEE (MS Word y LaTeX)  
[https://www.ieee.org/conferences\\_events/conferences/publishing/templates.html](https://www.ieee.org/conferences_events/conferences/publishing/templates.html)
- Editor colaborativo LaTeX en línea  
<https://www.overleaf.com/>
- Working with Jupyter Notebook files on GitHub  
<https://help.github.com/articles/working-with-jupyter-notebook-files-on-github/>