



SÍLABO

I. INFORMACIÓN GENERAL

Curso	: Procesamiento de Lenguaje Natural
Código	: CC0C2
Pre-requisito	: CC421
Dpto. Académico	: Ciencia de la Computación
Condición	: Electivo
Ciclo Académico	: 2025-2
Créditos	: 4
Horas teóricas	: 2 horas semanales
Horas prácticas	: 4 horas semanales
Sistema de Evaluación	: G

II. SUMILLA

Este curso ofrece una exploración exhaustiva y rigurosa del Procesamiento del Lenguaje Natural (NLP) y los Modelos de Lenguaje de Gran Escala (LLMs), dotando a los estudiantes de fundamentos teóricos, habilidades prácticas y técnicas de vanguardia para desarrollar soluciones basadas en inteligencia artificial. Inspirado en la investigación y prácticas de la industria de primer nivel, el curso cubre conceptos fundamentales, arquitecturas de modelos, metodologías de evaluación y técnicas avanzadas como Generación Aumentada por Recuperación (RAG), sistemas basados en agentes e ingeniería de prompts.

Los estudiantes participarán en proyectos prácticos utilizando herramientas del mundo real (por ejemplo, Hugging Face, PyTorch, LangChain) y aprenderán a aplicar técnicas de NLP y LLMs para construir aplicaciones innovadoras para startups, empresas o flujos de trabajo integrados con IA en diversos dominios como programación, educación y automatización.

III. COMPETENCIAS

- Diseñar, implementar y optimizar sistemas de NLP y LLMs utilizando frameworks y herramientas modernas.
- Evaluar y comparar modelos fundacionales según rendimiento, costo y aplicabilidad específica al dominio.
- Desarrollar soluciones impulsadas por IA para problemas del mundo real, integrando técnicas de NLP/LLMs en aplicaciones empresariales o de startups.
- Aplicar metodologías de investigación para avanzar en técnicas de NLP/LLMs y proponer soluciones novedosas.
- Presentar conceptos técnicos y resultados de proyectos de manera efectiva en entornos grupales e individuales.

IV. LOGROS DE APRENDIZAJE

Al finalizar el curso, los estudiantes podrán:

- Comprender la evolución, arquitectura y entrenamiento de modelos fundacionales, incluyendo LLMs.
- Implementar y ajustar modelos de NLP utilizando bibliotecas como Hugging Face y PyTorch.

- Diseñar pipelines de evaluación para medir el rendimiento y la robustez de los modelos.
- Aplicar técnicas avanzadas como RAG, ingeniería de prompts y sistemas basados en agentes a casos de uso del mundo real.
- Desarrollar y presentar proyectos impulsados por IA adecuados para emprendimientos o integración empresarial.
- Colaborar eficazmente en equipos para resolver desafíos complejos de NLP y comunicar hallazgos a audiencias técnicas y no técnicas.

V. UNIDADES DE APRENDIZAJE

El curso está dividido en cinco unidades, cada una cubriendo aspectos clave de NLP y LLMs. Las clases se imparten los jueves (2 horas, basadas en conferencias) y sábados (4 horas, laboratorios prácticos y discusiones), excepto en fechas de exposición y feriados. Cada unidad incluye fundamentos teóricos, programación práctica y aprendizaje basado en proyectos.

Unidad 1: Fundamentos de NLP e ingeniería de IA

Objetivo: Comprender los fundamentos de NLP, LLMs y la pila de ingeniería de IA.

Semanas: 1-4 (25 de agosto -20 de septiembre)

Temas:

- Introducción a NLP, LLMs y modelos fundacionales.
- Pila de ingeniería de IA: capas, casos de uso y planificación.
- Técnicas básicas de NLP: tokenización, normalización y n-gramas.
- Herramientas prácticas: bibliotecas de Python, Hugging Face y cargadores de datos.

Unidad 2: Técnicas centrales de NLP y arquitecturas de modelos

Objetivo: Explorar algoritmos centrales de NLP, arquitecturas de modelos y representaciones vectoriales.

Semanas: 5-8 (25 de septiembre -18 de octubre)

Temas:

- Representaciones de palabras, vectores semánticos y modelos de secuencia.
- Redes Neuronales Recurrentes (RNNs) y modelos de secuencia a secuencia.
- Introducción a Transformers y mecanismos de atención.
- Preparación para el examen parcial y planificación de proyectos.

Unidad 3: Arquitecturas avanzadas de LLMs y evaluación

Objetivo: Dominar modelos basados en Transformers, métricas de evaluación y técnicas de optimización.

Semanas: 9-12 (23 de octubre -15 de noviembre)

Temas:

- Mecanismos de atención, codificación posicional y Transformers para traducción y clasificación
- BERT, modelos de lenguaje causales y estrategias de ajuste fino (LoRA, adaptadores).
- Metodologías de evaluación: perplejidad, entropía cruzada, IA como juez.
- Optimización de modelos: inferencia, cuantización y cómputo en tiempo de prueba.

Unidad 4: Aplicaciones prácticas de LLMs

Objetivo: Aplicar LLMs a casos de uso reales usando RAG, agentes e ingeniería de prompts.

Semanas: 13-16 (20 de noviembre -6 de diciembre)

Temas:

- Generación Aumentada por Recuperación (RAG) y LlamaIndex.
- Agentes de IA: herramientas, planificación y modos de fallo.
- Ingeniería de prompts: mejores prácticas, estrategias defensivas e iteración.
- Ingeniería de datos: curación, aumento y síntesis.

Unidad 5: Proyecto final y síntesis

Objetivo: Sintetizar conocimientos mediante un proyecto final y prepararse para la integración de IA en el mundo real.

Semanas: 16-17 (11 de diciembre -20 de diciembre)

Temas:

- Aprendizaje por Refuerzo con Retroalimentación Humana (RLHF) y Optimización Directa de Preferencias (DPO).
- Optimización de inferencia.
- Presentaciones de proyectos finales y cierre del curso.

CRONOGRAMA SEMANAL

Semana 1

- *28 de agosto (Jueves):* Introducción a NLP, LLMs y Pila de Ingeniería de IA.
- *30 de agosto (Sábado): Feriado - sin clases.*

Semana 2

- *4 de septiembre (Jueves):* Carga de datos y bibliotecas básicas de NLP.
- *6 de septiembre (Sábado):* Bibliotecas y cargadores de datos.

Semana 3

- *11 de septiembre (Jueves):* Tokenización y algoritmo BPE.
- *13 de septiembre (Sábado):* Normalización, lematización y segmentación de palabras.

Semana 4

- *18 de septiembre (Jueves):* Modelos de lenguaje y n-gramas.
- *20 de septiembre (Sábado): Exposición 1*

Semana 5

- *25 de septiembre (Jueves):* Regularización y evaluación de modelos.
- *27 de septiembre (Sábado):* Representaciones de texto y embeddings.

Semana 6

- *2 de octubre (Jueves):* Modelos de lenguaje neuronales y RNNs.
- *4 de octubre (Sábado):* Modelos secuencia a secuencia y técnicas de decodificación.

Semana 7

- *9 de octubre (Jueves):* Retropropagación a través del tiempo (BPTT).
- *11 de octubre (Sábado): Exposición 2*

Semana 8

- *18 de octubre (Sábado): Examen Parcial.*

Semana 9

- *23 de octubre (Jueves)*: Mecanismos de atención y codificación posicional.
- *25 de octubre (Sábado)*: Transformers para traducción y clasificación.

Semana 10

- *30 de octubre (Jueves)*: Preprocesamiento y preentrenamiento de BERT.
- *1 de noviembre (Sábado)*: **Feriado - sin clases.**

Semana 11

- *6 de noviembre (Jueves)*: Modelos de lenguaje causales y tutorial de Hugging Face.
- *8 de noviembre (Sábado)*: Ajuste fino de Transformers con PyTorch.

Semana 12

- *13 de noviembre (Jueves)*: Ajuste fino eficiente en parámetros (LoRA, adaptadores).
- *15 de noviembre (Sábado)*: **Exposición 3**

Semana 13

- *20 de noviembre (Jueves)*: RAG y LlamaIndex.
- *22 de noviembre (Sábado)*: Agentes de IA y planificación.

Semana 14

- *27 de noviembre (Jueves)*: Mejores prácticas de ingeniería de prompts.
- *29 de noviembre (Sábado)*: Ingeniería de datos: curación, aumento y síntesis.

Semana 15

- *4 de diciembre (Jueves)*: RLHF y PPO.
- *6 de diciembre (Sábado)*: **Exposición 4**

Semana 16

- *11 de diciembre (Jueves)*: Optimización directa de preferencias (DPO) y optimización de inferencia.
- *13 de diciembre (Sábado)*: **Exposición 5**

Semana 17

- *20 de diciembre (Sábado)*: **Examen Final.**
- *27 de diciembre (Sábado)*: **Examen Sustitutorio.**

VI. METODOLOGÍA

- **2h/semana de teoría (jueves):**
Sesiones magistrales orientadas a explicar fundamentos, arquitecturas y técnicas avanzadas de NLP y LLMs. Se combinan con lecturas recomendadas (papers y capítulos de Jurafsky & Martin, Hugging Face docs, etc.).
- **4h/semana de práctica (sábado):**
Laboratorios con Jupyter notebooks, donde se implementan algoritmos y se experimenta con modelos en Hugging Face, PyTorch y LangChain.
Incluyen discusiones guiadas y *mini-proyectos* para consolidar los contenidos vistos en teoría.

- **Exposiciones intermedias:**
Los estudiantes presentan en grupos o individualmente avances de prácticas y proyectos (2 exposiciones antes del parcial y 3 después).
Se busca reforzar la comunicación científica, el *peer feedback* y la capacidad de defender propuestas técnicas.
- **Trabajo en equipo y aprendizaje basado en proyectos (PBL):**
Se organizan grupos pequeños que aplican las técnicas aprendidas a problemas reales (traducción, RAG para educación, etc.).

VII. SISTEMA DE EVALUACIÓN

Sistema de evaluación G:

- 1) Son 5 prácticas (P1, P2, P3, P4, P5) y exámenes: parcial (EP), final (EF) y opcionalmente sustitutorio (ES). Se califica sobre 20 puntos.
- 2) PP = Elimina la práctica de menor nota y promedia las otras 4.
- 3) Promedio del curso (PC):

$$\begin{array}{ll} \text{PC} = [(PP + EP + EF) / 3]. & \text{Si no se rinde ES} \\ \text{PC} = [(PP + \text{máxima nota (EP, EF)} + ES) / 3] & \text{Si se rinde ES.} \end{array}$$

Entregables y evaluación:

1. **Cuadernos Jupyter (Prácticas, P1-P5)**
 - a. Cada semana de laboratorio se trabaja con notebooks.
 - b. Incluyen código, visualizaciones, experimentos y breves reflexiones escritas.
 - c. Se califican como **prácticas (P1-P5)**.
 - d. Se elimina la nota más baja para el promedio (PP).
 2. **Exposiciones (5 en total)**
 - a. **Exposición 1 (semana 4, grupal):** Casos de uso de IA y fundamentos de NLP.
 - b. **Exposición 2 (semana 7, individual):** Arquitecturas y modelos de secuencias.
 - c. **Exposición 3 (semana 9, grupal):** Aplicaciones con Transformers.
 - d. **Exposición 4 (semana 12, individual):** Técnicas de ajuste fino.
 - e. **Exposición 5 (semana 15-16, grupal):** Presentación de proyecto final.
- Aunque las exposiciones no se califican como notas separadas, sus entregables alimentan las prácticas (P1-P5) y forman parte de la evaluación continua.
3. **Proyecto de clase (entregables parciales)**
 - a. Grupos pequeños desarrollan mini-proyectos con datasets o problemas reales (ej. un chatbot para un dominio específico).
 - b. Se presentan avances en las exposiciones intermedias y se documentan en los notebooks.
 4. **Proyecto final (Startup AI prototype)**
 - a. Desarrollo de una aplicación inspirada en una **idea de startup**.
 - b. Incluye:
 - **Implementación técnica** con NLP/LLMs.
 - **Documentación en Jupyter + README** (pipeline, métricas, justificación de arquitectura).
 - **Pitch final** en Exposición 5, defendiendo la aplicación como si fuera ante inversionistas o comité técnico.

VIII. BIBLIOGRAFÍA

- Philipp Koehn, Statistical Machine Translation 2010, Cambridge University Press, New York.
- Speech and Language Processing An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition, 2020. Borrador: https://web.stanford.edu/~jurafsky/slp3/ed3book_dec302020.pdf.
- Automated Grammatical Error Detection for Language Learners, Second Edition, Claudia Leacock, Martin Chodorow, Michael Gamon, and Joel Tetreault 2014 Morgan & Claypool
- Kyunghyun Cho, Natural Language Understanding with Distributed Representation 2015. Courant Institute of Mathematical Sciences and Center for Data Science, New York University.
- Yoav Goldberg, Neural Network Methods for Natural Language Processing 2017, Morgan & Claypool Publisher.
- Sowmya Vajjala, Bodhisattwa Majumder, Anuj Gupta, Harshit Surana, Practical Natural Language Processing: A Comprehensive Guide to Building Real-World NLP Systems 2020 O'Reilly
- Embeddings in Natural Language Processing: Theory and Advances in Vector Representations of Meaning Mohammad Taher Pilehvar and José Camacho-Collados Morgan & Claypool 2020.
- Deep Learning Approaches to Text Production Shashi Narayan and Claire Gardent Morgan & Claypool 2020.
- Transfer Learning, Qiang Yang, Yu Zhang, Wenyuan Dai y Sinno Jialin Pan Cambridge University Press 2020.
- Machine Learning Q and AI: 30 Essential Questions and Answers on Machine Learning and AI de Sebastián Raschka No Starch Press 2024.
- Generative AI with LangChain: Build large language model (LLM) apps with Python, ChatGPT and other LLMs de Ben Auffarth. Packt Publishing (22 diciembre 2023).
- Lean AI: How Innovative Startups Use Artificial Intelligence to Grow de Lomit Patel. O'Reilly Media; 1er edición 2020.