



## PROYECTO DE CURSO

---

### 1 Introducción

Muchach@s, tal como adelantamos en clase, el proyecto de curso es una de las piezas angulares del curso y se centrará en investigar, usar y/o desarrollar, técnicas de IA asociadas a los tópicos que discutimos en clases. Como conversamos, el proyecto de curso será individual y estará centrado en una temática común para todos. Específicamente, este semestre nos centraremos en la temática de *prompting*.

### 2 Prompting

En el icónico trabajo detrás del popular modelo de language GPT-3 [1], los autores destacan la capacidad que emerge en este modelo para resolver nuevas tareas, no vistas durante entrenamiento, mediante la interacción con pocos nuevos ejemplos (few shot learning or FSL). Esto es altamente destacable pues el modelo no fue entrenado para realizar tareas de FSL, sino que esta propiedad emerge al entrenar un modelo a gran escala usando billones de parámetros y palabras (tokens). Más aún, los autores destacan la capacidad de GPT-3 para realizar esta tarea no solo a través del típico approach de refinamiento supervisado (finetuning), sino que también a través de mostrar al modelo ejemplos de la nueva tarea a través de un *prompt* en el input, o incluso entregar en este *prompt* una descripción de la tarea en language natural. Este iluminador resultado ha abierto una amplia agenda de investigación dedicada a tópicos como: i) Estudiar que tipos de prompt son útiles [1, 4], ii) Aprender prompts en forma automática [3, 2], o iii) Entender cuales son los elementos claves de un buen *prompt* [6, 5], entre otros. En una de las sesiones de lecturas de las próximas semanas nos dedicaremos a estudiar algunas de estas técnicas.

### 3 Desafío de Investigación

El desafío de investigación de este semestre consiste en investigar sobre prompting, ya sea en el ámbito de modelos para análisis de textos, imágenes, o un escenario multimodal. Específicamente, a partir de esta investigación desarrollar un modelo que utilice prompting de acuerdo a algunos de los siguientes niveles de complejidad:

1. Estudiar y evaluar algún método existente para el cual **si** exista código público (ver OBS1) y que ya haya sido aplicado al problema propuesto. En su reporte debe incluir un análisis de rendimiento bajo distintas condiciones, estudios de sensibilidad (ablation), creación de casos de prueba que permitan apreciar las principales fortalezas y falencias, y una comparación respecto a modelos alternativos descritos en la literatura.
2. Implementar y probar algún método existente para el cual **no** exista código público y que ya haya sido aplicado al problema propuesto. En su reporte debe incluir un análisis de rendimiento bajo distintas condiciones, estudios de sensibilidad (ablation) y creación de casos de prueba que permitan apreciar las principales fortalezas y falencias.
3. Implementar y probar alguna de las técnicas vistas en clases o alguna que haya encontrado en su revisión bibliográfica. En este caso, debe ser una técnica atingente al problema y que no haya sido probada previamente para este problema. Para este caso puede desarrollar su propia implementación o adaptar algún código público.
4. Proponer, implementar y evaluar alguna mejora incremental a algún método existente. En este caso, la comparación de rendimiento sería con respecto al algoritmo base al cual se agregó la mejora.

5. Realizar una innovación mayor a un método existente o desarrollar un nuevo modelo.

Lograr un buen nivel en cualquiera de los puntos anteriores garantizará una buena calificación. En todos los casos, especialmente en los puntos I) y II), será fundamental respaldar el estudio con un adecuado análisis de los resultados obtenidos, en términos de análisis de sensibilidad, visualizaciones, creación de casos de prueba y comparación con métodos alternativos.

El proyecto es una de las partes más importantes del curso, tanto en términos de la metodología del curso como de su sistema de evaluación. IIC-3692 es un curso de 10 créditos semanales donde se espera que parte sustancial de ese tiempo quede plasmada en la calidad del proyecto de curso. Así, la recomendación es a no dejar el trabajo del proyecto para último minuto, sino trabajar en forma metódica durante todo el semestre.

**OBS1:** el uso y evaluación de métodos con código público es bienvenido y puede ser usado como baseline en sus trabajos. Sin embargo, el trabajo del proyecto de curso requiere una fase de implementación. Esta implementación puede ser el desarrollo de mejoras a métodos existentes o nuevos métodos, también puede ser el desarrollo de códigos para estudiar la operación de métodos ya existentes, incluyendo visualizaciones relevantes. Lo importante a tener en cuenta es que para efectos del proyecto de curso **no basta con sólo ejecutar implementaciones públicas**.

## 4 Entregas

Con objeto de ir chequeando el avance del proyecto se fijan los siguientes hitos:

- **Entrega 1, Fecha Octubre 15 (5%):** Informe escrito con propuesta inicial. Como elemento mínimo requerido debe incluir revisión bibliográfica y propuesta tentativa de investigación.
- **Entrega 2, Fecha Noviembre 15 (10%):** Informe escrito de avance. Como elemento mínimo requerido, debe incluir implementación inicial del trabajo propuesto, incluyendo evaluación experimental.
- **Entrega 3, presentación final, TBD de 08:30 a 11:30 (35%):** Entrega final en formato paper y presentación oral de poster en jornada de cierre del curso. Se requiere asistencia presencial salvo casos especiales de fuerza mayor.

## 5 Entrega Final

El resultado final de su trabajo debe ser documentado mediante una presentación en formato poster y un documento en formato paper. Para el caso del poster, éste irá acompañado de una presentación oral a ser realizada en jornada presencial de cierre del curso. Para el paper, la extensión máxima es de 7 páginas en formato doble columna, sin contar las páginas con referencias. En la página web del curso se publicarán templates para el paper en latex y MS-Word. El documento final debe incluir las siguientes secciones:

- Breve introducción describiendo las aristas principales de su proyecto y su relevancia.
- Revisión bibliográfica que incluya al menos 2 trabajos relevantes a su proyecto.
- Descripción del proyecto propuesto, incluyendo un diagrama de bloques con sus aspectos principales y una breve descripción de cada bloque.
- Resultados experimentales en base a alguno o algunos de los frameworks utilizados para el proyecto.
- Descripción de las principales conclusiones de su trabajo.
- Detalle de las referencias en formato paper.

**Atrasos:** Las entregas 1 y 2 pueden ser entregadas con atraso. El descuento es de 1.0 punto por cada día de atraso. En el caso de la entrega final no es posible entrega con atraso.

## References

- [1] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. Language models are few-shot learners. In *NEURIPS*, 2020.
- [2] B. Lester, R. Al-Rfou, and N. Constant. The power of scale for parameter-efficient prompt tuning. In *ACL*, 2021.
- [3] X.L. Li and P. Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *ACL*, 2021.
- [4] J. Liu, A. Liu, X. Lu, S. Welleck, P. West, R.L. Bras, Y. Choi, and H. Hajishirzi. Generated knowledge prompting for commonsense reasoning. In *ACL*, 2022.
- [5] Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. In *ACL*, 2022.
- [6] Albert Webson and Ellie Pavlick. Do prompt-based models really understand the meaning of their prompts? *ArXiv*, abs/2109.01247, 2022.