

## Contexto

Este trabajo está parcialmente financiado por la UNCo, en el marco del nuevo proyecto de investigación Tecnologías Semánticas para el desarrollo de Agentes Inteligentes. El proyecto de investigación tiene una duración de cuatro años y ha comenzado en 2022 y se desarrolla en forma colaborativa con docentes-investigadores de la UNS.

## Línea de Investigación

La generación de comentarios de código es la tarea de generar una descripción en lenguaje natural de alto nivel para un método o función de código determinado.

Para lograr este objetivo, se hizo uso de técnicas de aprendizaje automático que permiten entrenar un modelo a partir de un corpus adecuado. Se emplea una técnica de transferencia de conocimiento en un modelo basado en atención, que permite aprovechar el aprendizaje previo de otros modelos con pequeños conjuntos de datos.

El modelo de lenguaje utilizado en esta investigación inicial se basa en la arquitectura de transformadores de codificación-decodificación de CodeTrans.

Se realiza una transferencia de aprendizaje utilizando tres tamaños del modelo T5: small, base y large. Esta consiste de dos etapas. Una etapa inicial de entrenamiento auto-supervisado donde se utilizan datos sin etiquetar y una segunda etapa, conocida como fine-tuning donde el modelo se entrena para una tarea específica utilizando datos etiquetados.

Otra técnica utilizada en CodeTrans es el entrenamiento multi-tarea, la cual consiste en entrenar un modelo en múltiples tareas, utilizando datos etiquetados y sin etiquetar. Esta metodología permite, además, realizar un fine-tuning posterior a través de la transferencia de aprendizaje.

En una primera instancia en este trabajo tomamos la arquitectura T5 ya entrenada con los datos de CodeTrans, tomando los modelos realizados por transferencia de aprendizaje (TF) y multi-tarea (MT) para los tres tamaños considerados: small, base y large, y realizamos fine-tuning sobre cada uno de ellos, utilizando sets de datos propios para realizar diversos experimentos.

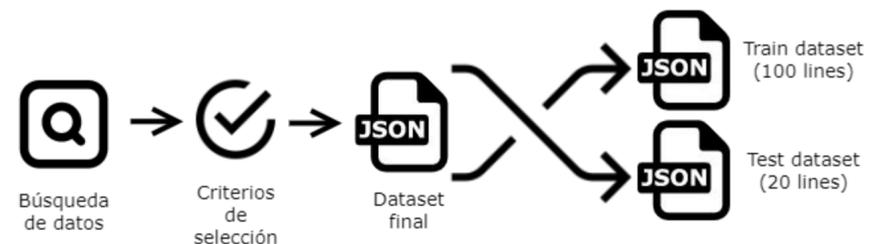
Estos conjuntos de datos propios están formados por dos corpus construidos: uno relacionado al lenguaje de programación GO y otro al lenguaje de programación PROLOG.

	Nuestra salida	CodeTrans
GO-TF-SMALL	<b>3.53</b>	2.75
GO-TF-BASE	<b>13.96</b>	11.34
GO-TF-LARGE	<b>20</b>	12.55
GO-MT-TF-SMALL	12.77	16.44
GO-MT-TF-BASE	11.19	11.69
GO-MT-TF-LARGE	<b>17.56</b>	9.95

TABLA 1: Resultados al comparar nuestros modelos con los de CodeTrans. La medida utilizada fue BLEU-4.

## Dataset

Cada dataset contiene 100 líneas para el entrenamiento y 20 líneas para el test, seleccionados de forma aleatoria. Para el dataset de Go se excluyeron construcciones fuera del espacio global de nombres y solo se utilizaron funciones, con independencia de si utilizaban o no elementos externos para realizar su tarea, incluidas otras funciones.



Una vez seleccionada una función para ser incorporada en el dataset, se eliminaron sus saltos de líneas y espacios no necesarios para facilitar el procesamiento.

En el dataset de PROLOG se evitó utilizar predicados para la incorporación de módulos de terceros, también se evitó usar predicados que contuvieran sintaxis de extensiones del lenguaje, como gramáticas de cláusulas definidas (DCG) o programación basada en restricciones (CLP).

## Resultados

Los experimentos relacionados al lenguaje GO se realizaron para comparar cómo se comportaba nuestro finetune con respecto a los modelos de CodeTrans utilizando nuestro dataset-test. El desarrollo del Fine-tuning sobre nuestro corpus brindó buenos resultados en la tarea de TF y solo en el MT-TF-LARGE se logró una mejora. Los resultados pueden observarse en la Tabla 1.

Por otro lado, se hicieron evaluaciones de estas mismas tareas en los modelos small, base, y large en el lenguaje PROLOG. Cabe destacar que CodeTrans no cuenta con este lenguaje. El objetivo fue probar cómo se comportaban los modelos para un lenguaje no contemplado previamente, siendo el mejor resultado para el TF de tamaño large. Los resultados, presentados en la Tabla 2, podrían deberse a las características propias de los comentarios para este tipo de lenguajes.

MEDIDA	PROLOG TF-SMALL	PROLOG TF-BASE	PROLOG TF-LARGE
ROUGE-L	24.45	28.26	<b>31.44</b>
BLEU-1	18.79	20.56	20.56
BLEU-2	10.71	12.21	13.43
BLEU-3	5.74	6.27	9.07
BLEU-4	0	0	<b>6.05</b>
	PROLOG MT-TF-SMALL	PROLOG MT-TF-BASE	PROLOG MT-TF-LARGE
ROUGE-L	27.52	29.06	28.57
BLEU-1	20.56	19.85	17.02
BLEU-2	13.43	11.6	9.53
BLEU-3	9.07	7.33	5.72
BLEU-4	5.1	4.34	0

TABLA 2: Resultados al comparar nuestros modelos de Prolog contra referencia humana.