

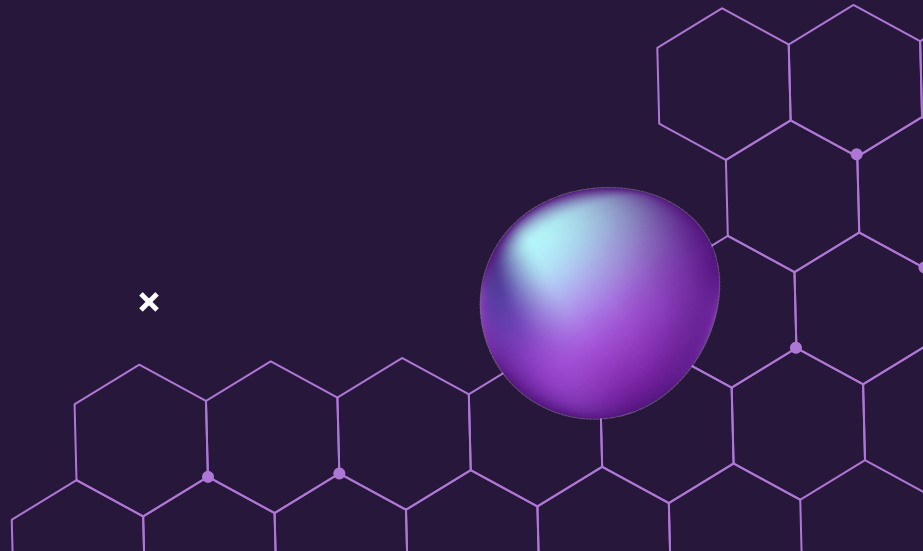


# AI-TEXTIFICATION

Identificación de autoría en textos

Sobre textos de IA vs Humanos

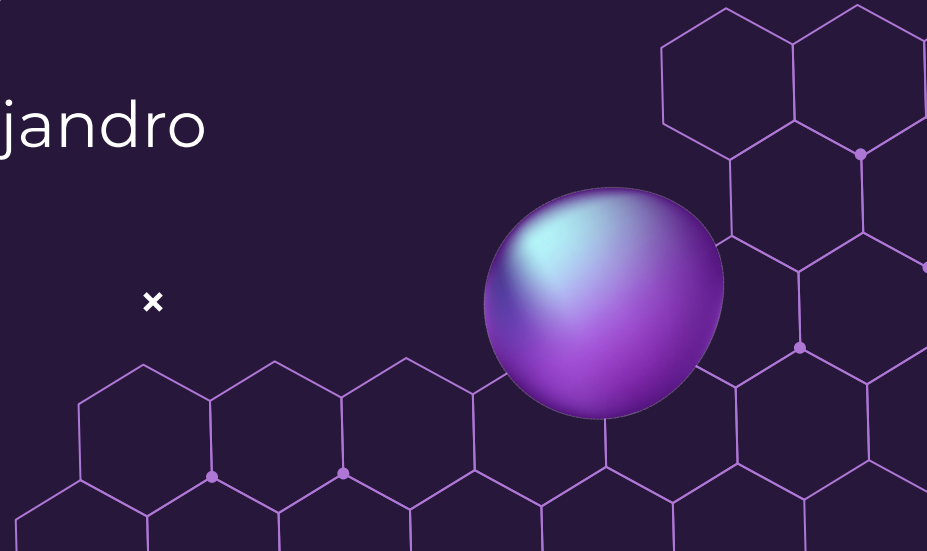
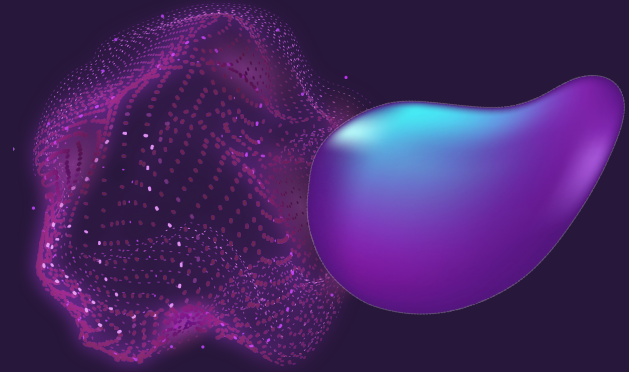
León Rosas Manuel Alejandro y  
Ramos Herrera Iván Alejandro





Autores:

- León Rosas Manuel Alejandro
- Ramos Herrera Iván Alejandro





# OBJETIVO

Desarrollar un sistema que identifique (con un considerable porcentaje de acierto) si un texto fue escrito por un ser humano o un Modelo Generativo de Lenguaje

# OBJETIVOS GENERALES

x

x



## TAREA A

Clasificar un texto en inglés bajo alguna de las dos etiquetas:

- Humano
- Máquina

x



## TAREA B

Si un texto en inglés es generado por máquina, darle una etiqueta correspondiente al modelo que lo generó



# FASES RELEVANTES

x



## PREPARACIÓN

Se aplicaron técnicas de pre-procesamiento para generar 4 conjuntos de datos distintos

+



## MODELOS

Se implementó:

1. Red Neuronal Recurrente LSTM
2. KNN Classifier



x

## EVALUACIÓN

Con Kfolds CV de 5 y las métricas:

1. Accuracy
2. F1-Micro
3. F1-Macro





01

# PREPARACIÓN

Aplicada sobre los datos de ejemplo

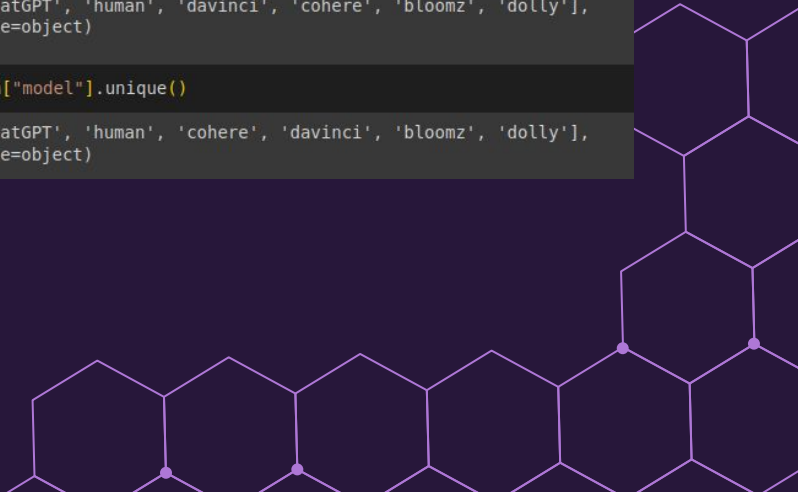


## TAREA A

```
[ ] # Verificando la cantidad de autores:  
    dataAdev["model"].unique()  
  
    array(['bloomz', 'human'], dtype=object)  
  
[ ] dataAtrain["model"].unique()  
  
    array(['chatGPT', 'cohere', 'davinci', 'dolly', 'human'])
```

## TAREA B

```
[ ] # Verificando la cantidad de autores:  
    dataBdev["model"].unique()  
  
    array(['chatGPT', 'human', 'davinci', 'cohere', 'bloomz', 'dolly'],  
          dtype=object)  
  
[ ] dataBtrain["model"].unique()  
  
    array(['chatGPT', 'human', 'cohere', 'davinci', 'bloomz', 'dolly'],  
          dtype=object)
```





# DATASETS GENERADOS

x

- A. Cleaned => Lemma => UNK => GENSIM Own Embbedings
- B. Cleaned => Lemma => UNK => CBOW OWN Embbedings
- C. Cleaned => UNK => GENSIM Own Embbedings
- D. Cleaned => UNK => CBOW OWN Embbedings





# NOTA

- A. PARA LA TAREA A SE EXPLORÓ LA DISTRIBUCIÓN DE LAS ETIQUETAS Y SE CONFIRMÓ QUE NO FUE NECESARIO UN RESAMPLEO:

```
Instancias de textos [HUMANOS][LABEL: 0] 63351  
Instancias de textos [MÁQUINA][LABEL: 1] 56406
```

- B. PARA LA TAREA B SE ENCONTRÓ DESBALANCE; POR ELLO LA MÉTRICA F1 SERÁ MÁS RELEVANTE QUE EL ACCURACY:

```
Instancias de textos [HUMANOS][LABEL: 1] 11995  
Instancias de textos [MÁQUINA][LABEL: 0, 2, 3, 4, 5] 59032
```



# Paso CLEANED

1. Se transforma el texto a minúsculas<sup>x</sup>
2. Se reemplazan algunos símbolos selectivos por palabras representativas (para no perder los símbolos en la tokenización)\*

```
# Símbolos que se reemplazarán por texto:
symbols_replacement = {
    "(": " xparenthesis ",
    ")": " parenthesisx ",
    ",": " xcomma ",
    ".": " xpoint ",
    ";": " xpointcomma ",
    "\"": " xdoublequote ",
    "'": " xsimplequote ",
    "-": " xdash ",
    "?": " xinterrogation ",
    "!": " xadmiration ",
    "&": " xand "
}
```

3. Se eliminan todo el resto de símbolos especiales que no sean letras o números



# Paso LEMMA

1. Se aplicó lemmatización con el WordNetLemmatizer de NLTK

# Paso UNK

1. A las palabras con frecuencia menor a 4 en todo el dataset se reemplazaron por la palabra “xunk”



# EMBEDDINGS

## A. EMBEDDINGS GENSIM:

### A. Embeddings de librería de Python:

- `from gensim.models import Word2Vec`

### B. Vectores de palabras de dimensión 50

### C. Window: 5

### D. Workers: 4

## B. EMBEDDINGS PROPIOS:

### A. Algoritmo CBOW:

- $h = W1 * X + b1$
- $a = \text{ReLU}(h)$
- $z = W2 * a + b2$
- $y = \text{Softmax}(z)$

### B. Vectores de palabras de dimensión 50



02

# MODELOS

Para la clasificación en ambas tareas

# OBJETIVOS GENERALES

x

x



## KNN-Coseno

Clasificador de Vecinos  
Más Cercanos con  
Distancia Coseno

x



## RNN LSTM

Red Neuronal Recurrente  
con la forma:  
LSTM => FeedForward





03

# EVALUACIÓN

Métricas para los experimentos

A

# Binaria

Texto de máquina vs Texto humano



+





# KNN D-COSENOS TAREA A 1-grama de oraciones

CON 70% ENTRENAMIENTO – 30% EN EL DATASET “TRAIN”

DATASET	ACCURACY	WEIGAVG	F1-MACRO
Clean => Lemma => Unk => Gensim Emb	83%	83%	82%
Clean => Lemma => Unk => Own Emb	77%	77%	77%
Clean => Unk => Gensim Emb	80%	79%	79%
Clean => Unk => Own Emb	73%	73%	73%

# KNN D-COSENOS TAREA A

## 1-grama de oraciones

PREDICCIÓN CON EL DATASET DE PRUEBA DEL EJERCICIO

DATASET	ACCURACY	F1-MICRO	F1-MACRO
Clean => Lemma => Unk => Gensim Emb	58%	55%	55%

	precision	recall	f1-score	support
0	0.55	0.83	0.66	2500
1	0.65	0.32	0.43	2500
accuracy			0.58	5000
macro avg	0.60	0.58	0.55	5000
weighted avg	0.60	0.58	0.55	5000



B

# Multiclase

Texto de máquina vs cuatro modelos generativos

# KNN D-COSENOS TAREA B

## 1-grama de oraciones

CON 70% ENTRENAMIENTO – 30% EN EL DATASET “TRAIN”

DATASET	ACCURACY	WEIGAVG	F1-MACRO
Clean => Lemma => Unk => Gensim Emb	52%	51%	51%
Clean => Lemma => Unk => Own Emb	50%	51%	50%
Clean => Unk => Gensim Emb	49%	49%	50%
Clean => Unk => Own Emb	46%	44%	46%

# KNN D-COSENOS TAREA B

## 1-grama de oraciones

PREDICCIÓN CON EL DATASET DE PRUEBA DEL EJERCICIO

DATASET	ACCURACY	F1-MICRO	F1-MACRO
Clean => Lemma => Unk => Gensim Emb	51%	50%	50%

	precision	recall	f1-score	support
0	0.48	0.68	0.56	2963
1	0.48	0.63	0.54	3016
2	0.51	0.47	0.49	2778
3	0.33	0.34	0.34	3027
4	0.84	0.73	0.78	2993
5	0.50	0.21	0.29	2980
accuracy			0.51	17757
macro avg	0.52	0.51	0.50	17757
weighted avg	0.52	0.51	0.50	17757

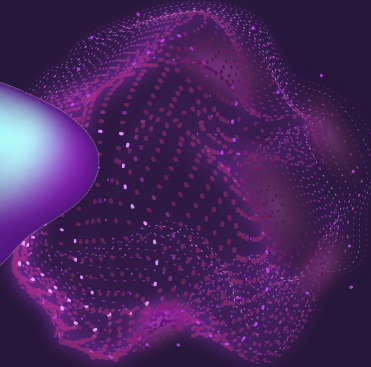
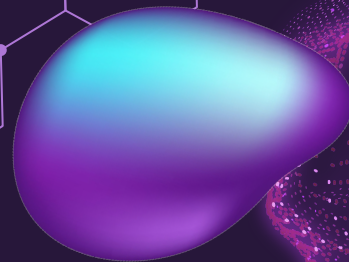
A

# Binaria

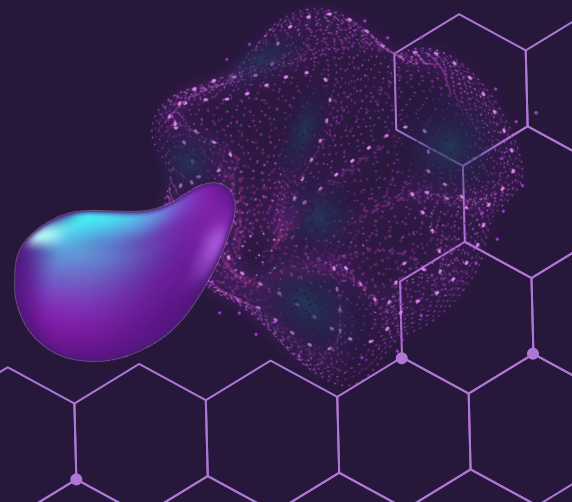
Texto de máquina vs Texto humano



+



x



# RNN LSTM TAREA A

0.01% del dataset  
3 épocas

CON K-FOLDS-CV K = 5

DATASET	ACCURACY	F1-MICRO	F1-MACRO
Clean => Lemma => Unk => Gensim Emb	58.33%	58.33%	36.84%
Clean => Lemma => Unk => Own Emb	-	-	-
Clean => Unk => Gensim Emb	-	-	-
Clean => Unk => Own Emb	-	-	-