82.18 Procesamiento de Lenguaje Natural

Franco Estévez, Gonzalo Manuel Beade, Gonzalo Rossin, Lucas Agustin Vittor

Instituto Tecnológico de Buenos Aires

{*festevez, grossin, gbeade, lvittor*}*@itba.edu.ar*

13 de noviembre de 2023

Estructura de la investigación

1. Análisis exploratorio del dataset
2. Sesgos del dataset
3. Desarrollo y comparación de modelos *discriminatorios*
4. Desarrollo y comparación de modelos *generativos*
5. Experimentos cruzados entre los dos tipos de modelos previos

Análisis exploratorio

Corpus

El corpus *(William Lifferth. Fake News. Kaggle, 2018.*[1]*)* cuenta con los siguientes campos:

- **id**: identificador único
- **title**: el título de un artículo
- **author**: autor del artículo
- **text**: el texto del artículo, podría estar incompleto.
- **label**: una etiqueta que clasifica el artículo como confiable (0) o no confiable (1).

En la competencia, ya nos dividen el dataset en *train.csv* y *test.csv* (igual al anterior pero sin la columna **label**).

[1] Lifferth 2018, `https://kaggle.com/competitions/fake-news`

Preprocesamiento

Se eliminaron:

- La columna **id** dado que no daba informacion
- Filas repetidas
- Filas nulas

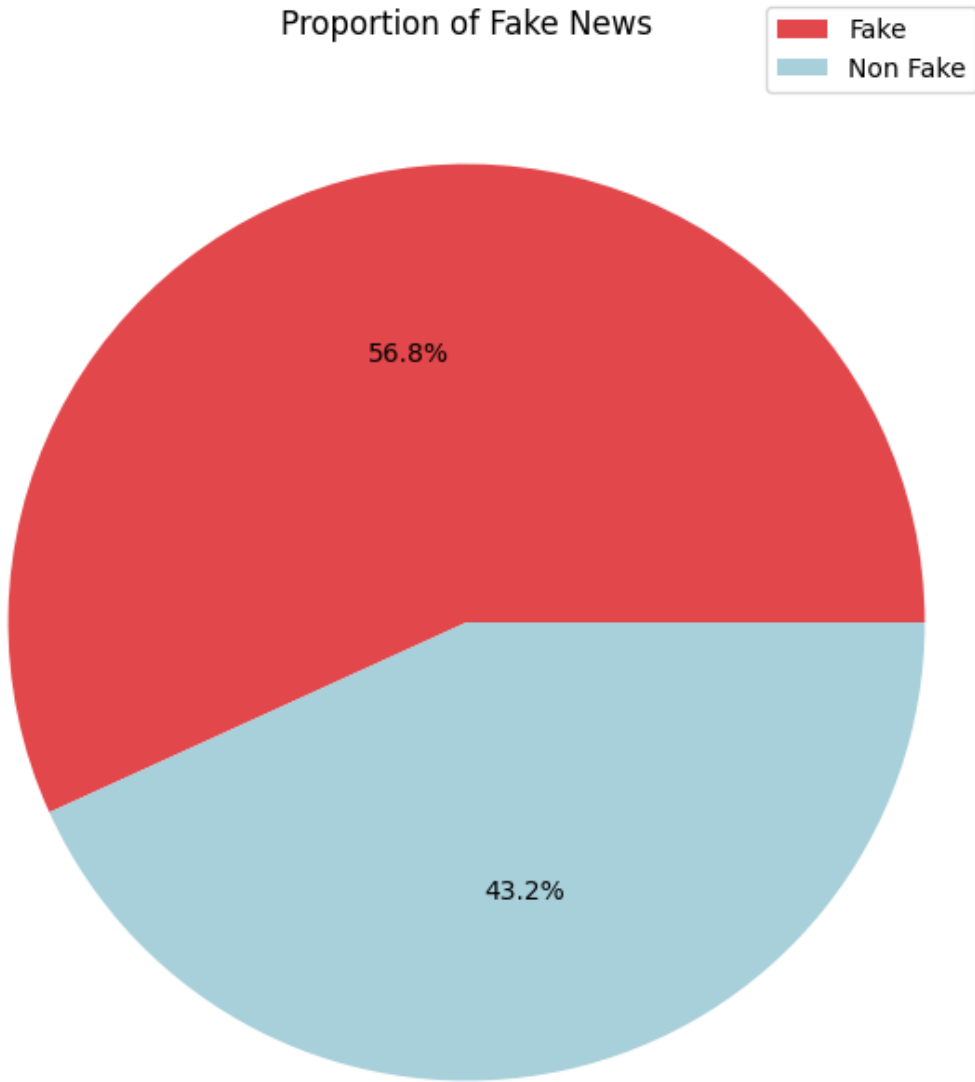
Preprocesamiento

Al dataset *train* se agregaron las columnas:

- **removed_punc**: Columna **text** sin signos de puntuación.[2]
- **tokens**: Columna **removed_punc** en tokens.[3]
- **filtered_tokens**: Columna **tokens** sin las palabras de menos de 3 caracteres.
- **clean_tokens**: Columna **filtered_tokens** sin stopwords.[4]
- **lemma_words**: Formas canónicas (lemmas) de los tokens en la columna **clean_tokens**.[5]
- **clean_text**: Texto resultante de concatenar los lemmas de la columna **lemma_words**.[6]

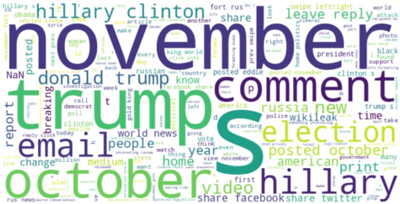
[2] usando *string.punctuation*
[3] usando *nltk.WhitespaceTokenizer()*
[4] usando *nltk.corpus.stopwords.words('english')*
[5] usando *nltk.WordNetLemmatizer()*
[6] usando *' '.join()*

Pie Chart



Word Cloud

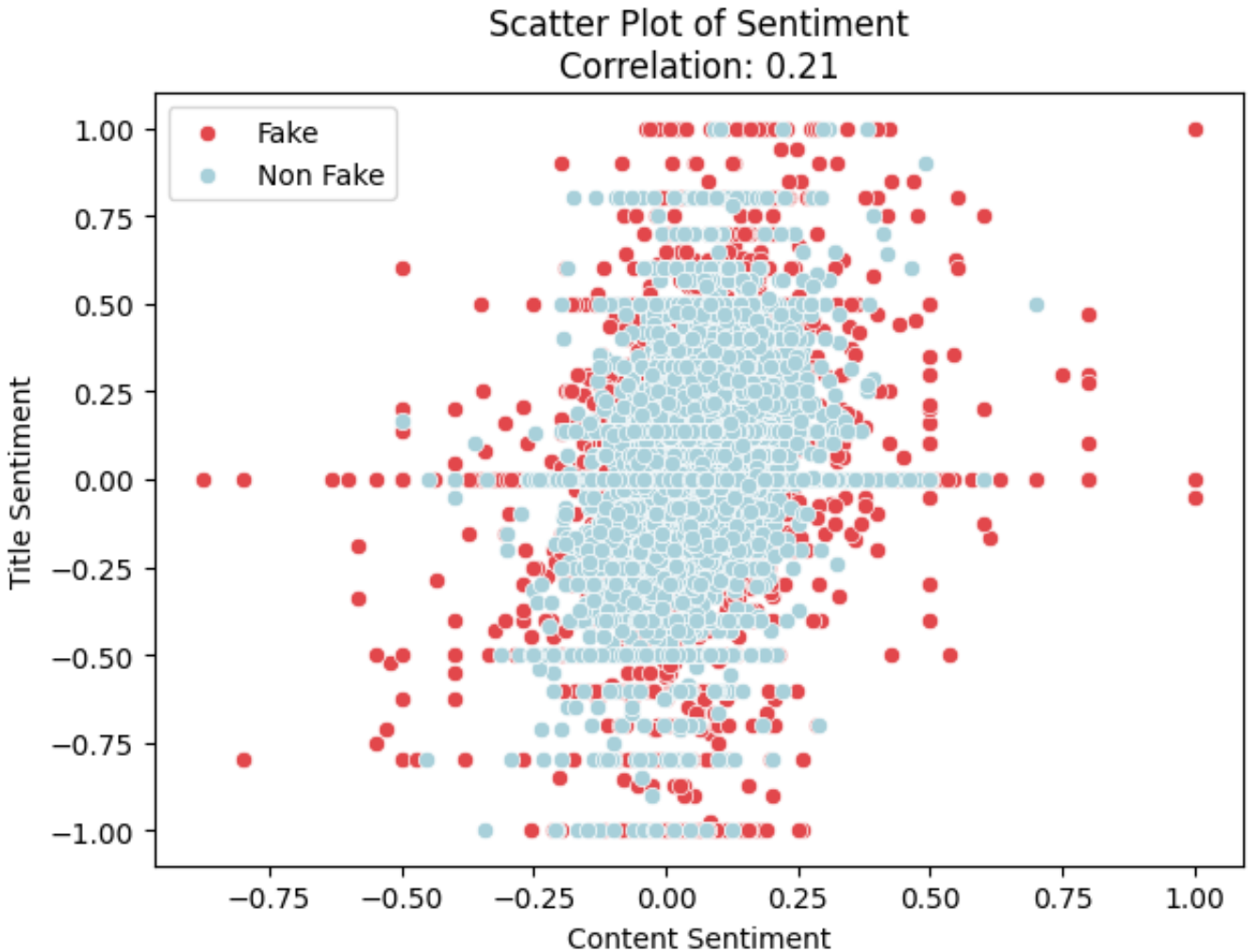
Word Cloud of Fake News Titles



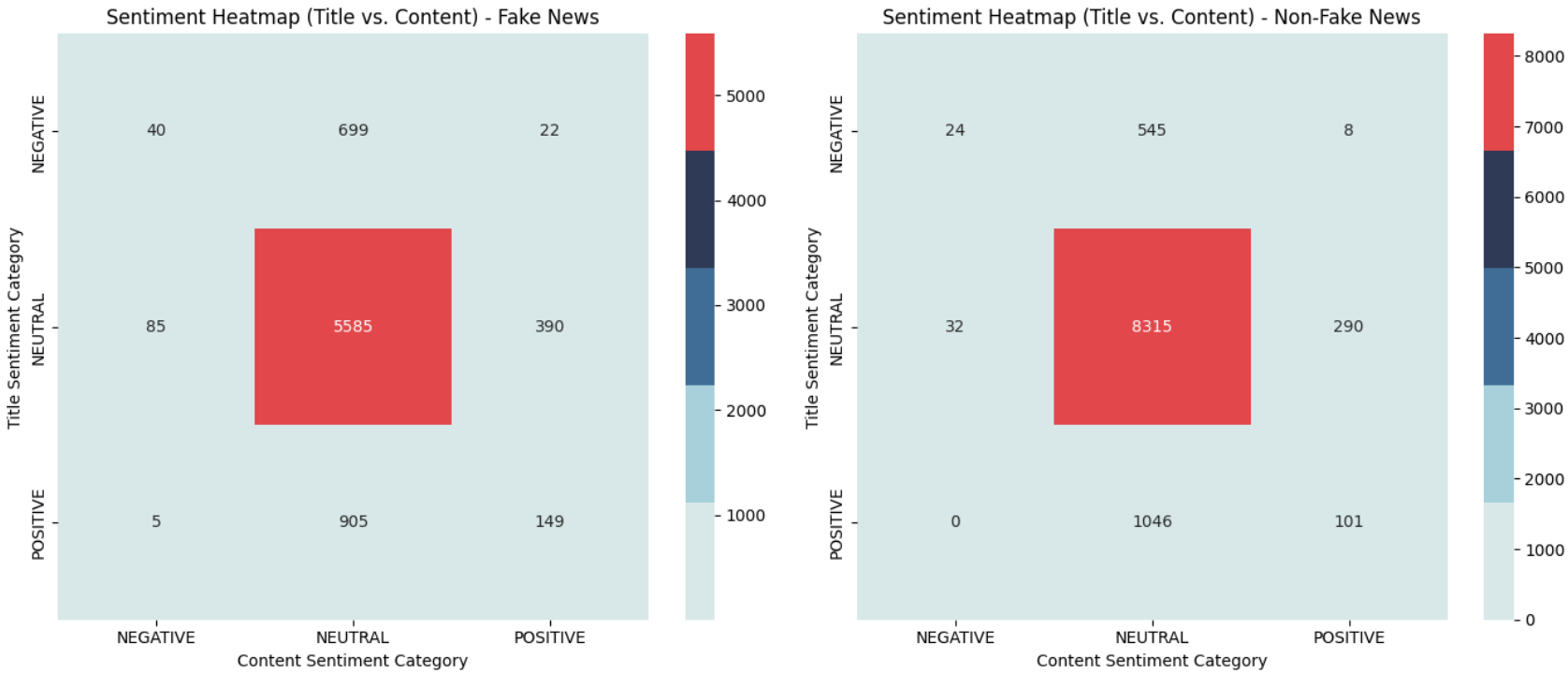
Word Cloud of Non-Fake News Titles

Scatter plot

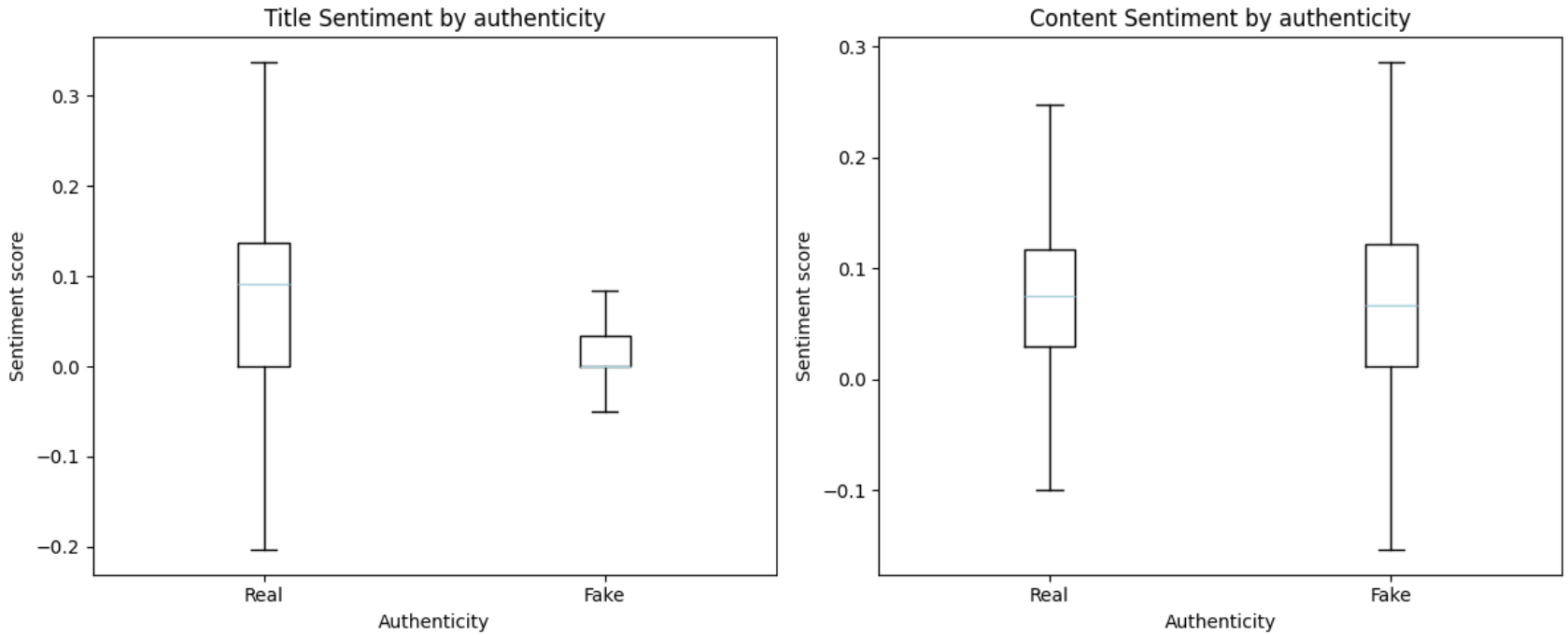
Scatter plot Merged



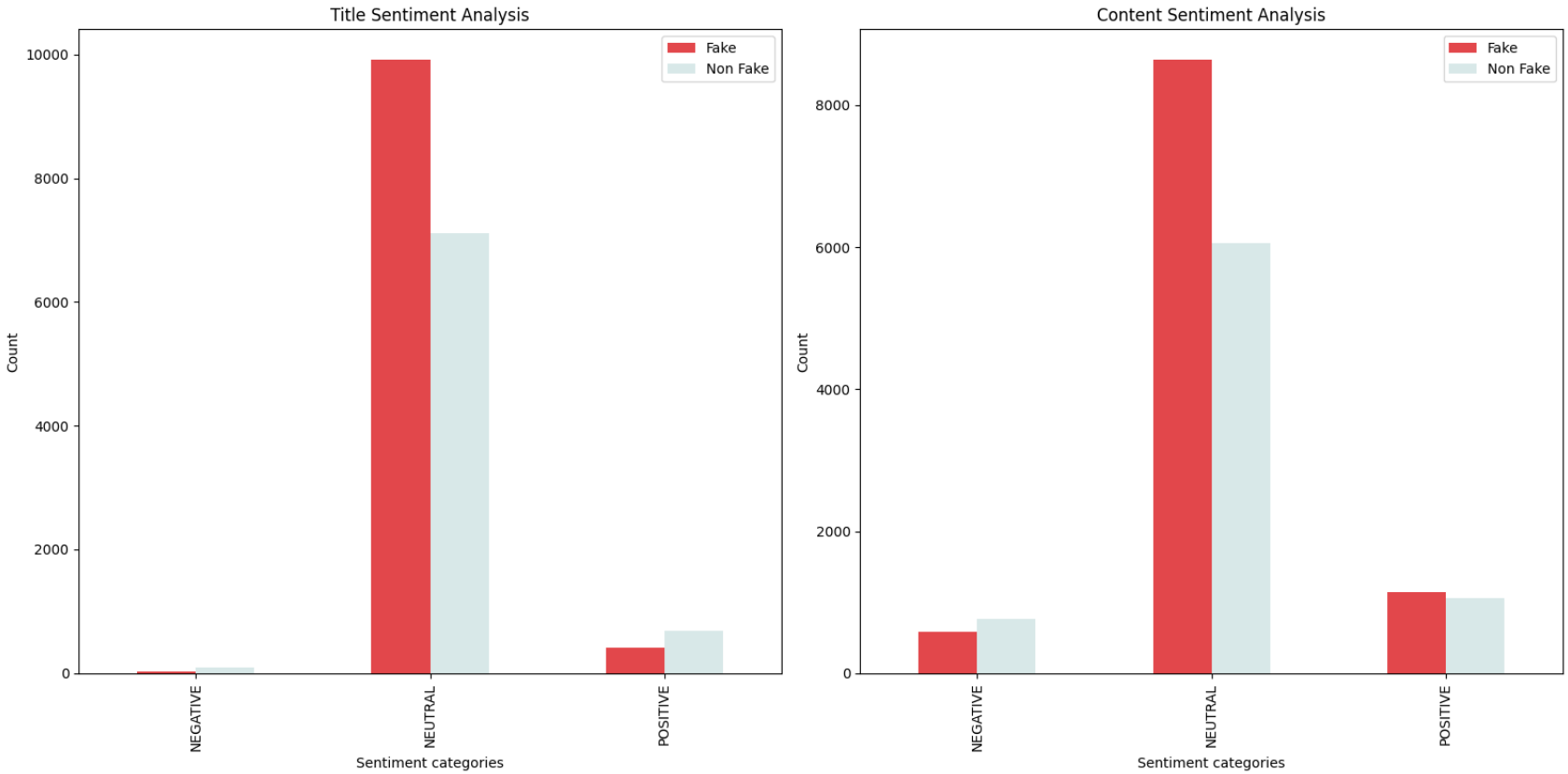
Sentiment HeapMap



Sentiment Autenticity Boxplot



Sentiment Barplot



Sesgos del dataset

Enunciado

Como vimos en clase, los sesgos pueden ser *de modelo* o *de dataset*. Vamos a trabajar con el segundo.

Buscamos poner en manifiesto como inspección inicial que el dataset de noticias que usamos está o no *sesgado* dependiendo de su veracidad.

Se supone que las noticias expositivas puras buscan presentar los hechos de manera no sesgada, mientras que las falsas buscan connotar relaciones semánticas de manera implícita.

No siempre queremos evitar los sesgos, queremos estudiarlos e identificarlos. Los recursos literarios *inducen* sesgos: metonimia, sinécdoque, kenning, sinestesia, juxtaposition...

Medir sesgos con word embeddings

La versión propuesta en clase consiste en entrenar word embeddings estáticos (WE) en el corpus y usar similitud coseno para medir las similitudes sim(A,C) y sim(B,C).

Definir grupos de palabras de contexto (A y B) y palabras objetivo (C)

$$Bias_{WE} = \text{mean}_{a \in A} \cos(\mathbf{w_c}, \mathbf{w_a}) - \text{mean}_{b \in B} \cos(\mathbf{w_c}, \mathbf{w_b})$$

El problema que nos encontramos es que no podemos usar encoders pre-entrenados porque podríamos estar trasladando el sesgo del modelo y hacerlo pasar como sesgo del dataset.

Medir sesgos con word embeddings - Sets

Garg, N., Schiebinger, L., Jurafsky, D. Zou, J. Word embeddings quantify 100 years of gender and ethnic stereotypes. PNAS 201720347 (2018). doi:10.1073/pnas.1720347115

Ponen a disposición sets de palabras para calcular sesgos. Nosotros elegimos dos tipos de sesgos a estudiar: sexo binario contra adjetivos de personalidad y sexo binario contra adjetivos de apariencia.

Medir sesgos con word embeddings - Sexo Vs. Personalidad

Medimos el sesgo puntual para cada palabra y luego lo agregamos calculando el desvío y la media. Para las noticias falsas:

Mean Bias: 0.146
Std. Bias: 0.247

Word in C	CStoA	CStoB	Bias
monstrous	-0.183	-0.570	0.387
monstrous	-0.183	-0.570	0.387
envious	-0.171	-0.558	0.387
devious	-0.218	-0.603	0.385
deceitful	-0.231	-0.613	0.382
erratic	-0.132	-0.513	0.381
deceptive	-0.073	-0.435	0.363
brutal	-0.018	-0.354	0.336
cruel	0.080	-0.188	0.268
frightening	-0.344	-0.586	0.242
aggressive	0.152	-0.048	0.200
bizarre	-0.336	-0.461	0.125
greedy	0.270	0.229	0.041
intolerant	0.278	0.252	0.026
forceful	0.294	0.300	-0.006
contemptible	0.319	0.385	-0.066
venomous	-0.234	-0.136	-0.098
calculating	0.345	0.534	-0.189
barbaric	0.341	0.600	-0.259
hateful	0.235	0.616	-0.381

Medir sesgos con word embeddings - Sexo Vs. Personalidad

Medimos el sesgo puntual para cada palabra y luego lo agregamos calculando el desvío y la media. Para las noticias verdaderas:

Mean Bias: -0.023
Std. Bias: 0.082

Word in C	CStoA	CStoB	Bias
envious	-0.067	-0.187	0.120
calculating	-0.093	-0.180	0.088
erratic	-0.095	-0.174	0.079
bizarre	-0.095	-0.171	0.076
intolerant	-0.096	-0.165	0.069
brutal	-0.096	-0.157	0.062
forceful	-0.089	-0.107	0.019
deceitful	0.091	0.118	-0.027
frightening	-0.069	-0.041	-0.028
barbaric	-0.064	-0.029	-0.036
cruel	-0.051	0.006	-0.056
monstrous	-0.034	0.042	-0.076
monstrous	-0.034	0.042	-0.076
hateful	-0.012	0.085	-0.097
aggressive	0.088	0.188	-0.099
devious	-0.004	0.098	-0.103
greedy	0.002	0.109	-0.107
deceptive	0.033	0.154	-0.121
venomous	0.057	0.180	-0.123

Medir sesgos con word embeddings - Sexo Vs. Apariencia

Medimos el sesgo puntual para cada palabra y luego lo agregamos calculando el desvío y la media. Para las noticias falsas:

Mean Bias: -0.019
Std. Bias: 0.292

Word in C	CStoA	CStoB	Bias
fat	-0.225	-0.608	0.384
fashionable	-0.062	-0.420	0.358
plump	-0.303	-0.643	0.340
alluring	-0.017	-0.353	0.336
muscular	-0.016	-0.352	0.335
weak	0.101	-0.149	0.251
handsome	0.116	-0.121	0.237
bald	0.124	-0.105	0.229
athletic	0.131	-0.091	0.222
blushing	0.220	0.102	0.118
strong	0.249	0.173	0.076
healthy	-0.299	-0.315	0.016
attractive	0.306	0.338	-0.033
feeble	0.326	0.415	-0.089
ugly	-0.177	-0.005	-0.173
slender	0.346	0.550	-0.204
gorgeous	-0.042	0.257	-0.299
stout	0.320	0.637	-0.317
voluptuous	0.295	0.643	-0.348
sensual	0.106	0.481	-0.375
thin	0.247	0.624	-0.378
slim	0.231	0.613	-0.382
pretty	0.226	0.609	-0.383
beautiful	0.207	0.593	-0.386

Medir sesgos con word embeddings - Sexo Vs. Apariencia

Medimos el sesgo puntual para cada palabra y luego lo agregamos calculando el desvío y la media. Para las noticias verdaderas:

Mean Bias: -0.025
Std. Bias: 0.069

Word in C	CStoA	CStoB	Bias
slim	-0.064	-0.185	0.121
handsome	-0.012	-0.125	0.113
fashionable	-0.094	-0.177	0.083
alluring	-0.096	-0.159	0.063
stout	0.050	-0.007	0.057
healthy	-0.086	-0.096	0.010
attractive	-0.084	-0.090	0.006
bald	0.080	0.076	0.004
strong	-0.082	-0.080	-0.001
thin	-0.080	-0.075	-0.005
ugly	-0.068	-0.038	-0.030
weak	-0.067	-0.035	-0.032
beautiful	-0.058	-0.013	-0.045
pretty	0.095	0.141	-0.047
fat	-0.052	0.001	-0.054
gorgeous	-0.048	0.011	-0.059
homely	-0.044	0.020	-0.064
muscular	-0.042	0.026	-0.068
sensual	-0.041	0.027	-0.068
slender	0.094	0.175	-0.080
voluptuous	-0.014	0.082	-0.096
feeble	-0.011	0.087	-0.098
blushing	-0.010	0.089	-0.099
athletic	0.081	0.191	-0.110
plump	0.033	0.154	-0.121

Modelo discriminatorio

Estrategia de clasificación

Como estrategia, vamos a clasificar a las noticias en base a la transformación TfidfVectorizer de su titulo y de su contenido por separado.

Luego, vamos a verificar si es mejor clasificar por titulo, o por contenido.

Modelos a estudiar

Los modelos a estudiar para clasificar las noticias son:

- *LogisticRegression*
- *PassiveAggressiveClassifier*
- *XGBClassifier*
- *LGBMClassifier*

LogisticRegression (**title**)

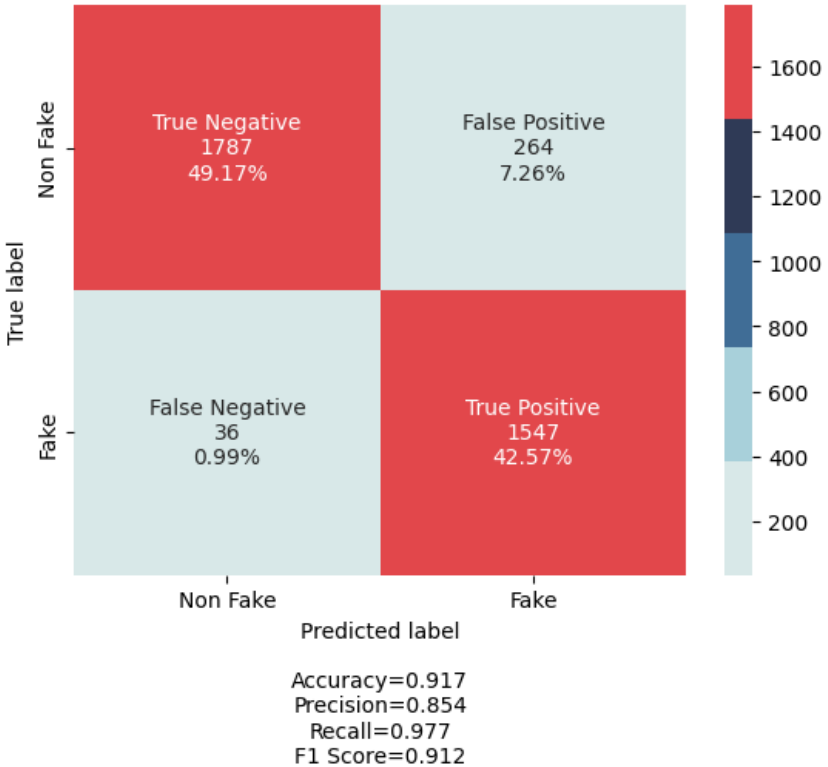


Figura: Matriz de Confusión

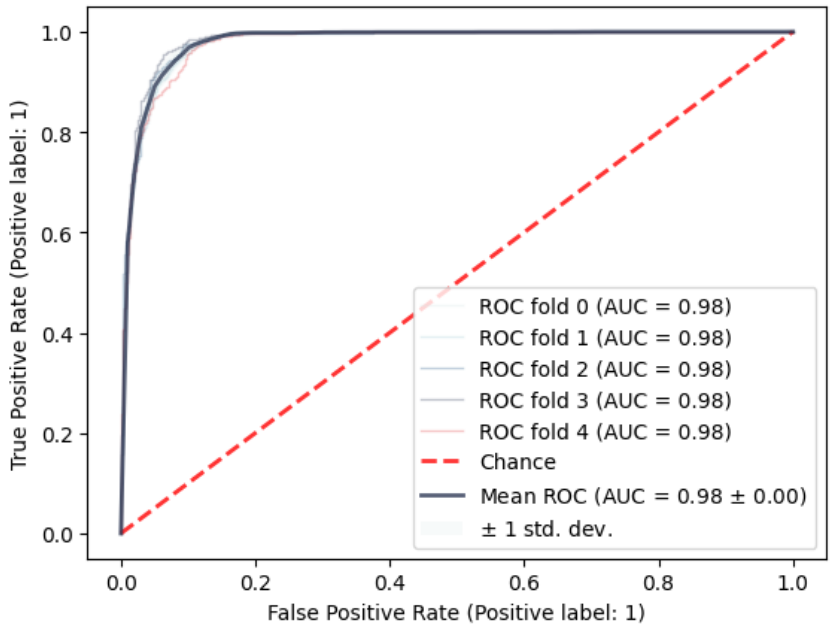


Figura: Curva ROC con Validación Cruzada de 5 folds

LogisticRegression (**clean_text**)

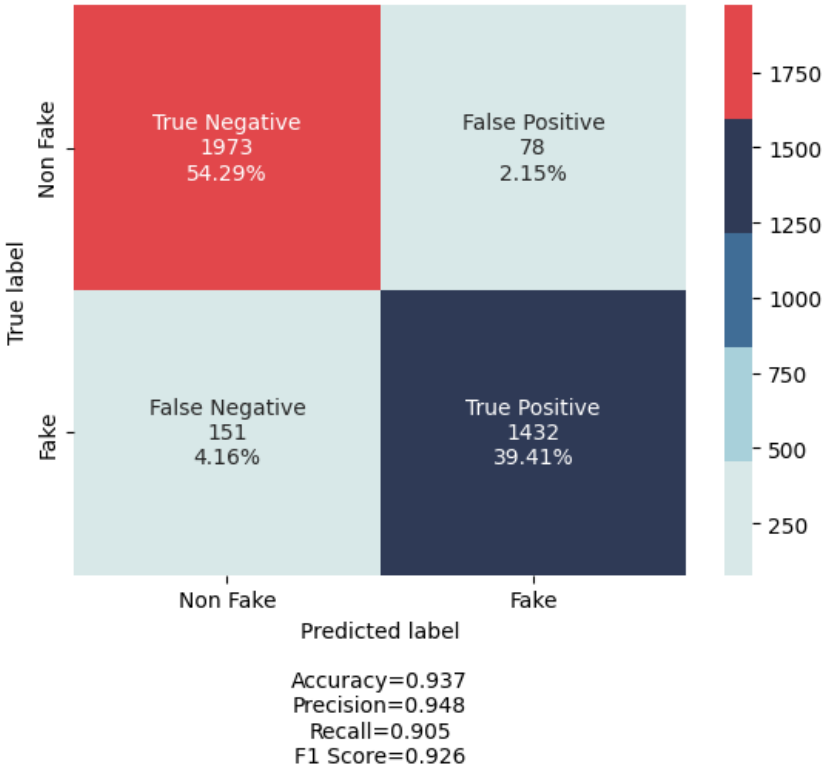


Figura: Matriz de Confusión

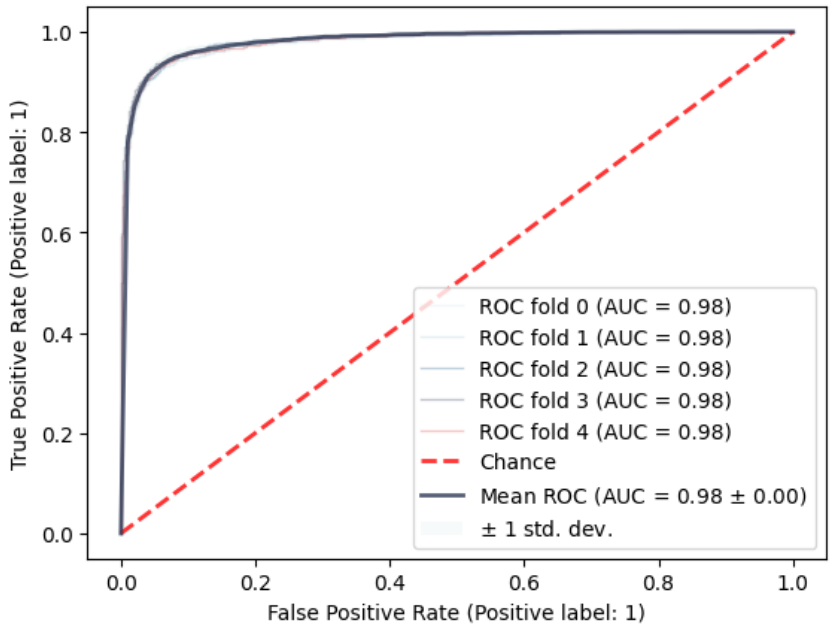


Figura: Curva ROC con Validación Cruzada de 5 folds

PassiveAggressiveClassifier (**title**)

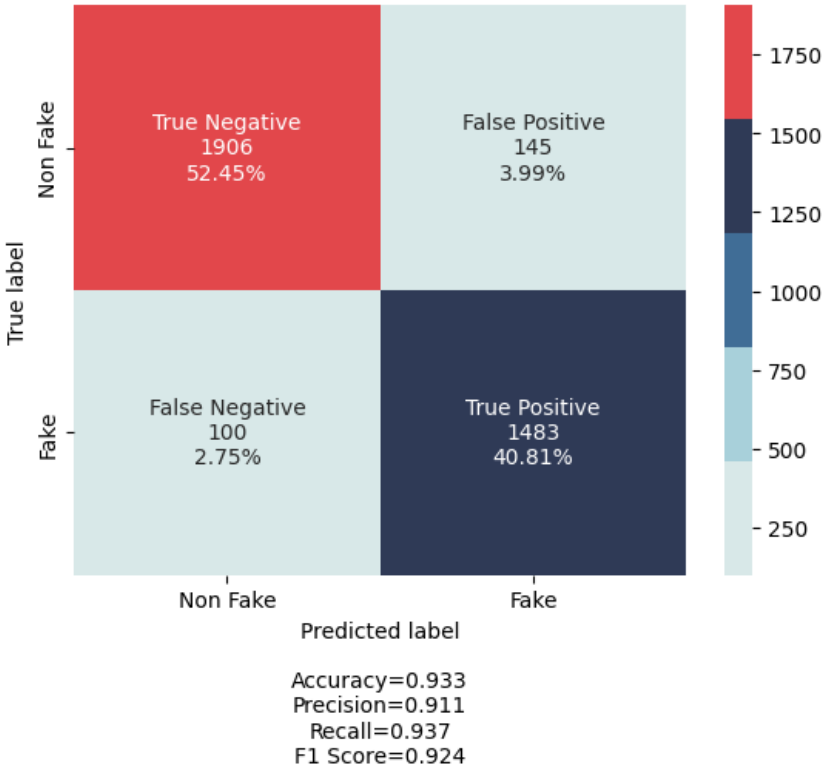


Figura: Matriz de Confusión

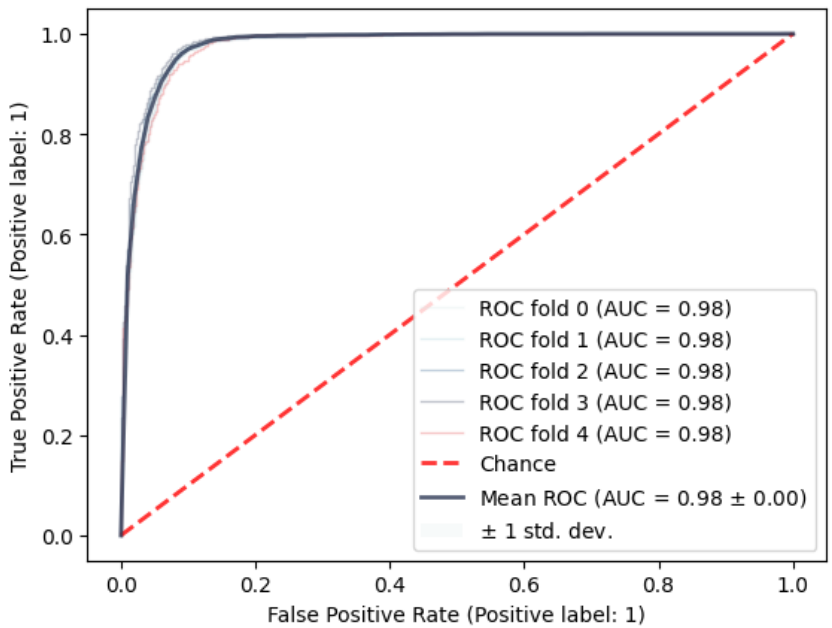


Figura: Curva ROC con Validación Cruzada de 5 folds

PassiveAggressiveClassifier (**clean_text**)

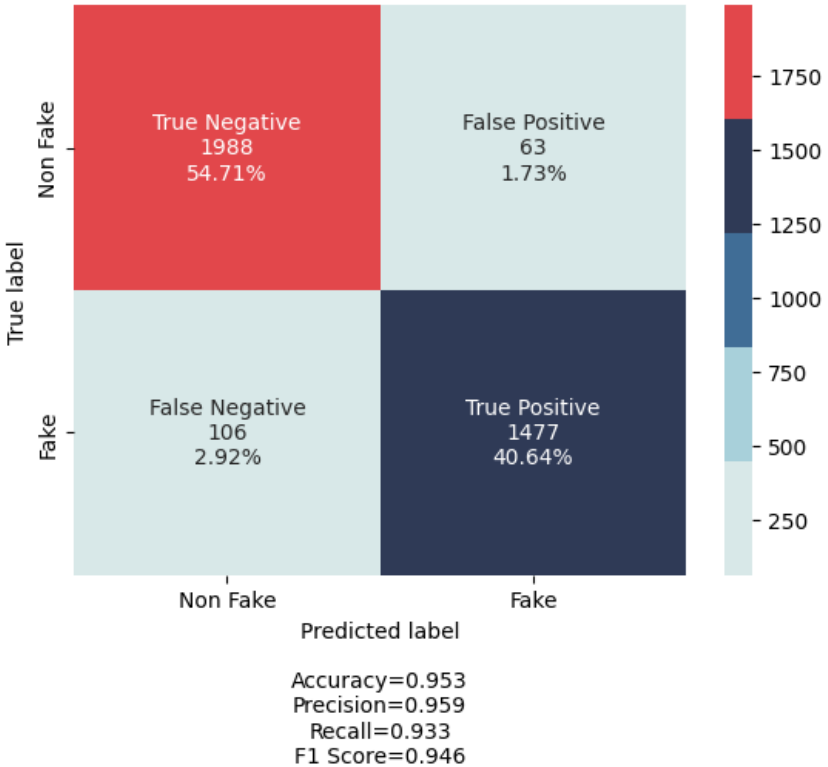


Figura: Matriz de Confusión

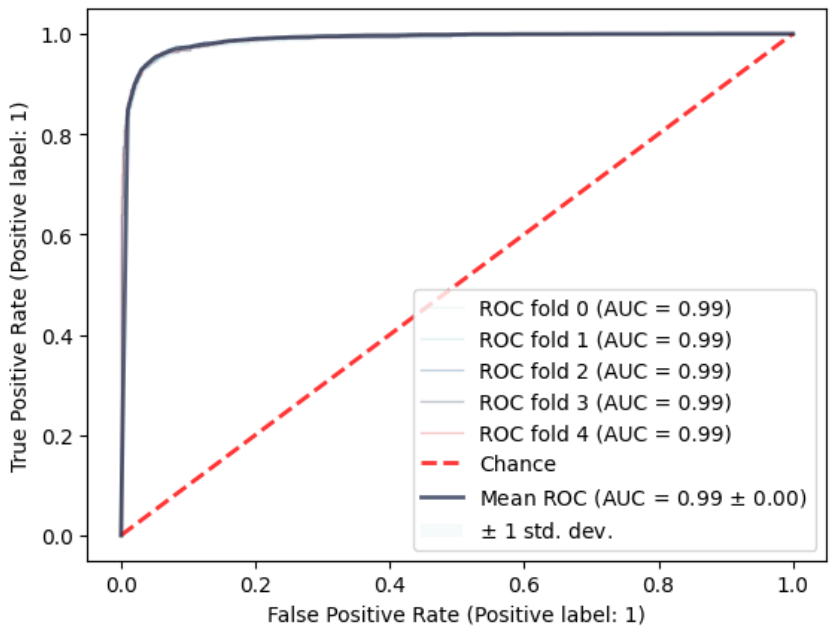


Figura: Curva ROC con Validación Cruzada de 5 folds

LGBMClassifier (**title**)

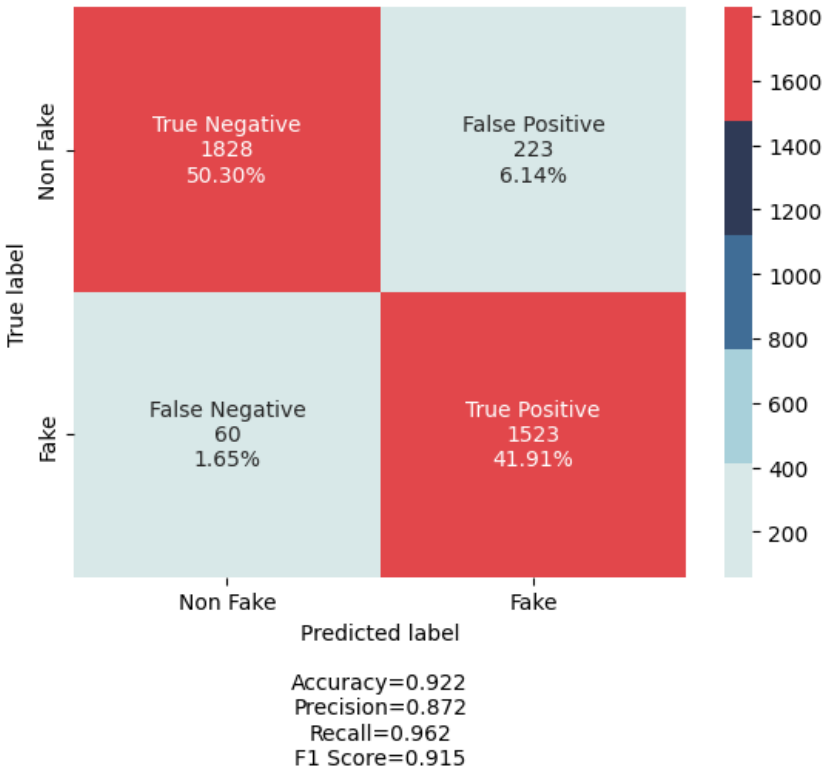


Figura: Matriz de Confusión

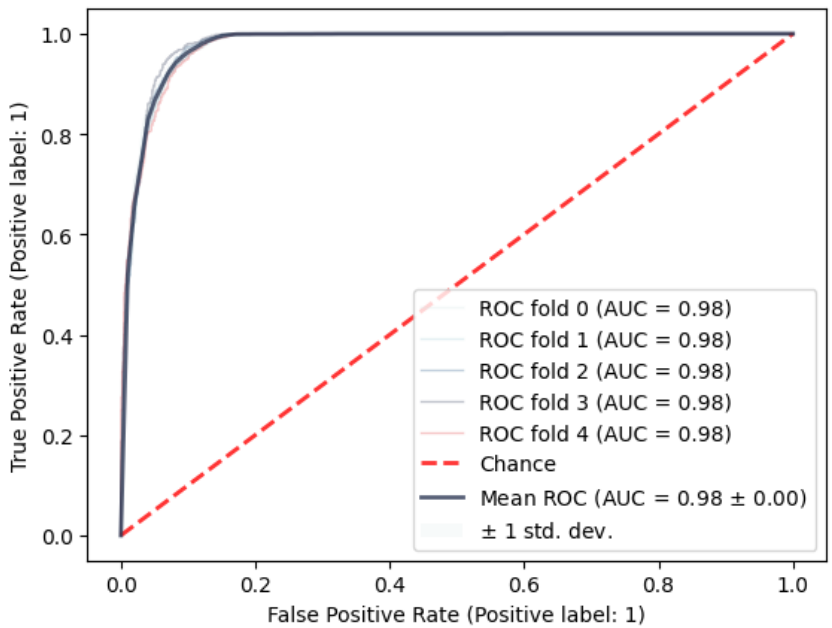


Figura: Curva ROC con Validación Cruzada de 5 folds

LGBMClassifier (**clean_text**)

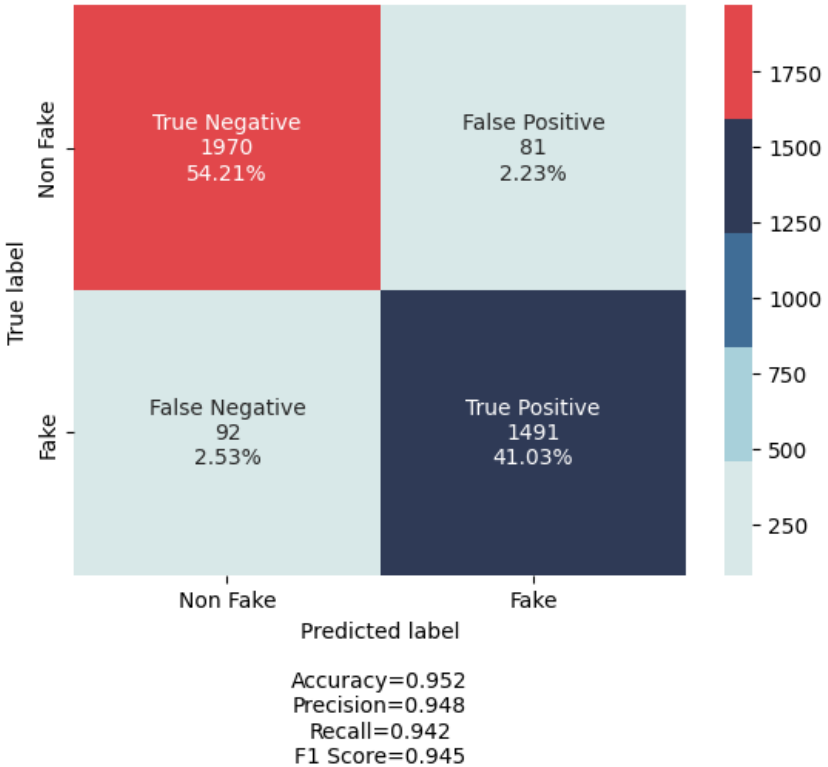


Figura: Matriz de Confusión

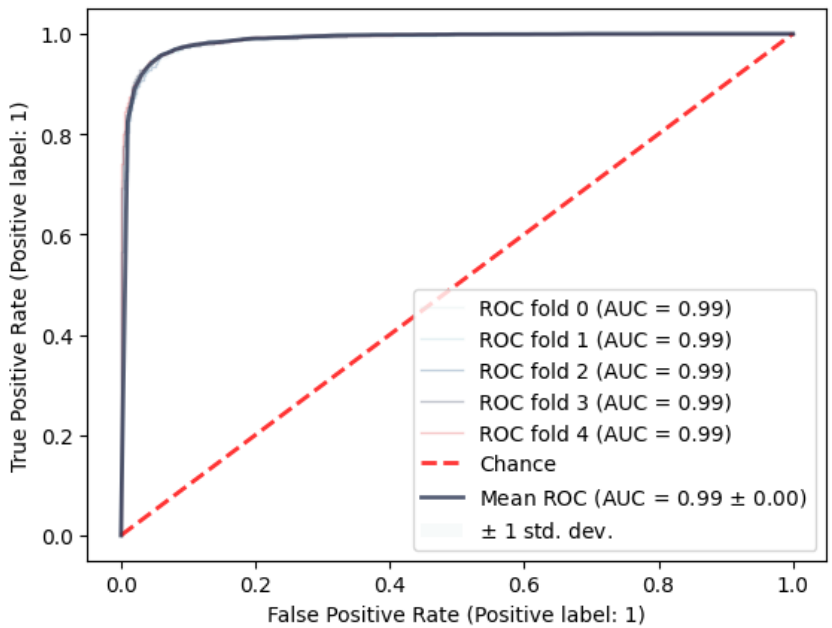


Figura: Curva ROC con Validación Cruzada de 5 folds

XGBClassifier (**title**)

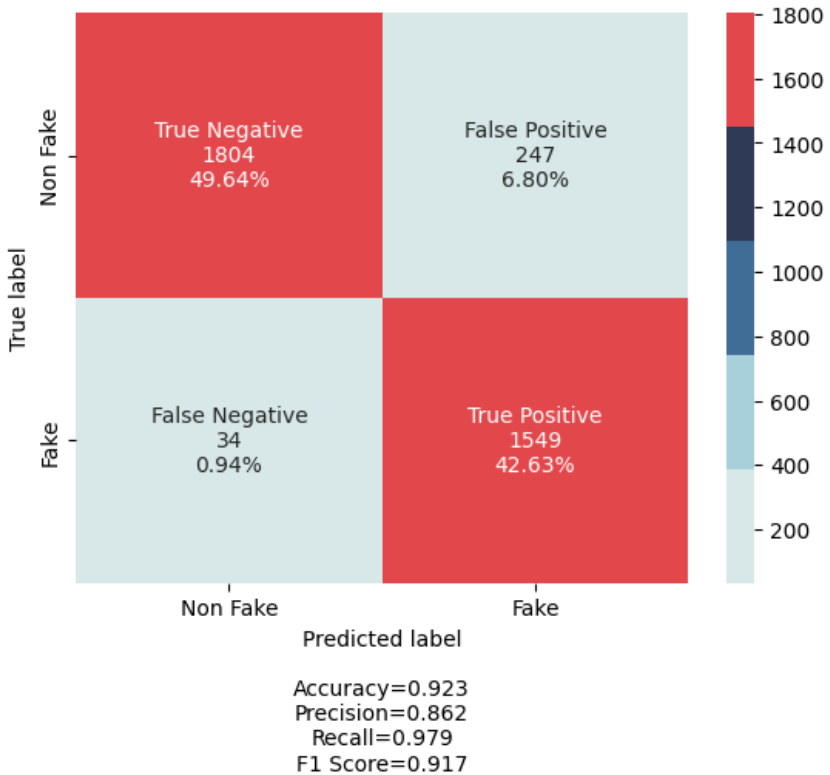


Figura: Matriz de Confusión

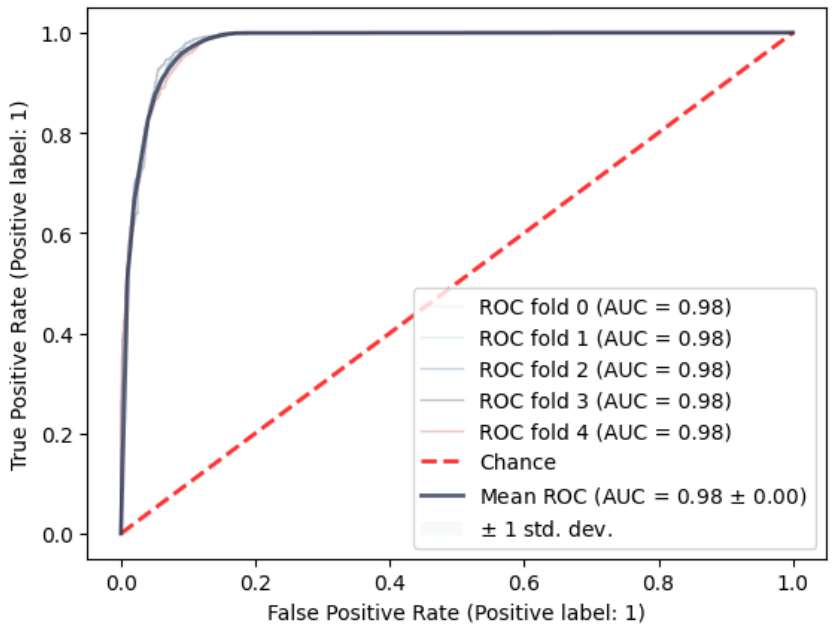


Figura: Curva ROC con Validación Cruzada de 5 folds

XGBClassifier (**clean_text**)

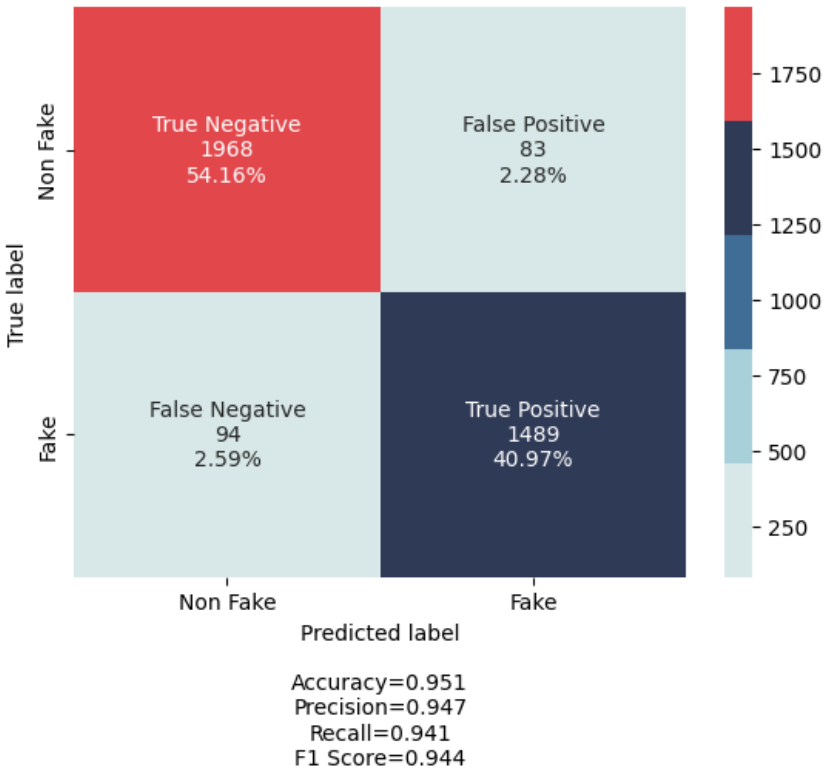


Figura: Matriz de Confusión

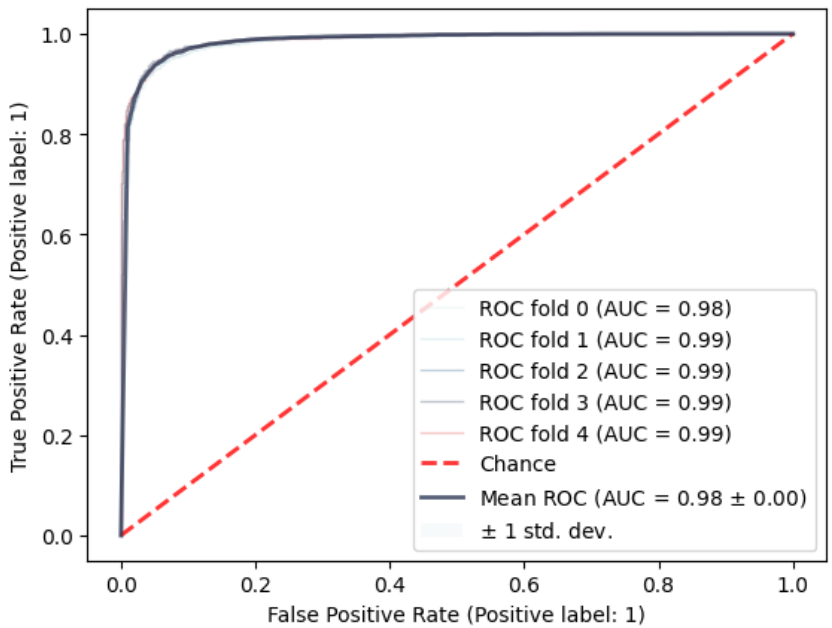
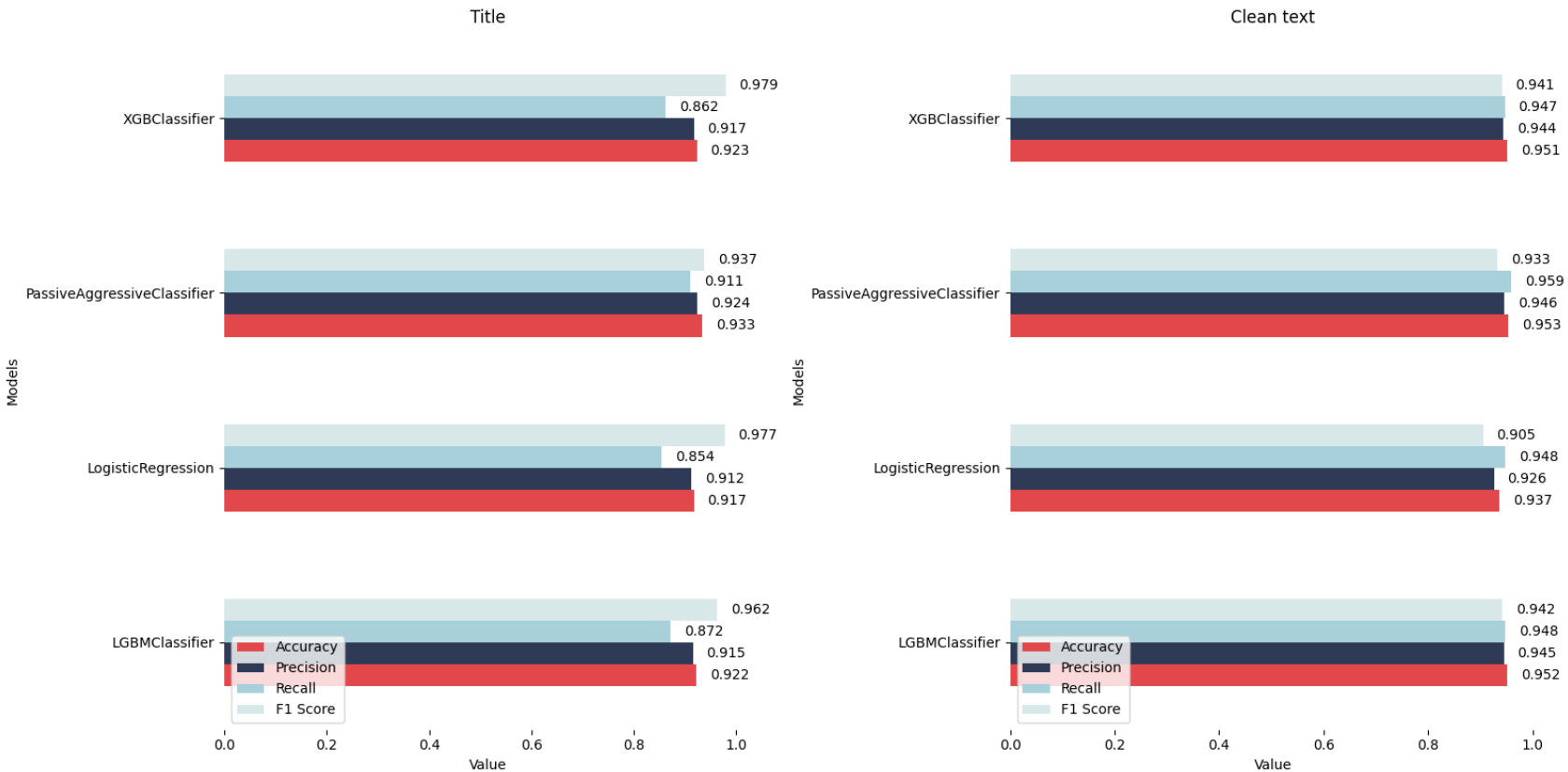


Figura: Curva ROC con Validación Cruzada de 5 folds

Comparación general de modelos y métricas



Modelo generativo

Enunciado

Implementamos dos tipos de modelos generativos con el subconjunto de datos de noticias falsas.

Compararemos los resultados de usar dos tipos de tranformers pre-entrenados y habiendo aplicado fine-tuning. Los modelos pre-entrenados utilizados son BERT y GPT2

¿Cómo podemos saber si un modelo generativo está performando *bien*?

Resultados de BERT

Donald Trump plans to finance the war in quentin ' amid

The nuclear attempt addition to the rock ' patrice suspension

Argentina imposes residual ' lethal revelation patrice suspension shea

Resultados de GPT2

Donald Trump plans to use the 1.5 trillion in federal tax breaks for corporations and individuals, including a massive increase of corporate taxes on middle-class families.

The nuclear attempt was a success. The first test, in the early 1950s and at least two more later tests were conducted by Japan's Atomic Energy Agency (AEO) on July 1-3 of 1954.

Argentina imposes a new rule on the country's economy, which is expected to grow by 2.5 percent this year and 4-6 in 2016 — an increase of about 1 percentage point from last time.

Metricas

La perplejidad es una medida comúnmente utilizada en procesamiento del lenguaje natural para evaluar qué tan bien una distribución de probabilidad o un modelo predictivo predice una muestra. Es una forma de cuantificar cuán sorprendido está el modelo en promedio al predecir la siguiente palabra en una secuencia.

GPT 2	1.4392
BERT	818

Conclusiones

Conclusiones

1. En la competencia de kaggle tuvimos 0.94065 de score privado y 0.94423 de score publico utilizando PassiveAggressiveClassifier con **clean_text**
2. Determinar la veracidad de un texto exclusivamente a partir del propio texto es un desafío complejo. Los modelos pueden proporcionar información como patrones lingüísticos, coherencia interna y contexto semántico, pero existen limitaciones importantes, ya que la veracidad a menudo depende de conocimientos externos.
3. El modelo generativo BERT tuvo un rendimiento mucho menor que GPT2 para tareas que involucran generación de texto.