

NLP en el Análisis de OFERTAS DE TRABAJO

GRUPO 3

Alejandro Dionis Ros, Adrián Lizzadro Plá,
Sergio Sebastiá García e Ilia Zhigarev



VNIVERSITAT
DE VALÈNCIA

Solicitar puesto



Guardar oferta



Contenido

- Introducción
- Extracción de Datos (Web Scraping)
- Preprocesamiento
 - Limpieza de Datos
 - Análisis Exploratorio del conjunto
- Modelos
 - Modelos Propios
 - Modelos Preentrenados
- Resultados
- Personalización de Stopwords
- Conclusiones

Introducción

El PLN es una rama de la informática y de la inteligencia artificial que emplea técnicas de aprendizaje automático para que los ordenadores puedan comprender, interpretar y generar lenguaje humano de forma eficiente.

Actualmente, el PLN está presente en muchos aspectos de la vida cotidiana.

Dado que nos encontramos en la fase final del curso e inmersos en la búsqueda de oportunidades laborales, se decidió tomar como tema del trabajo final de la asignatura el análisis de ofertas de empleo.



El **objetivo** principal de este proyecto es analizar ofertas de empleo con el fin de estimar el salario ofrecido a partir del contenido textual de las mismas.



Extracción de datos

Extracción de datos



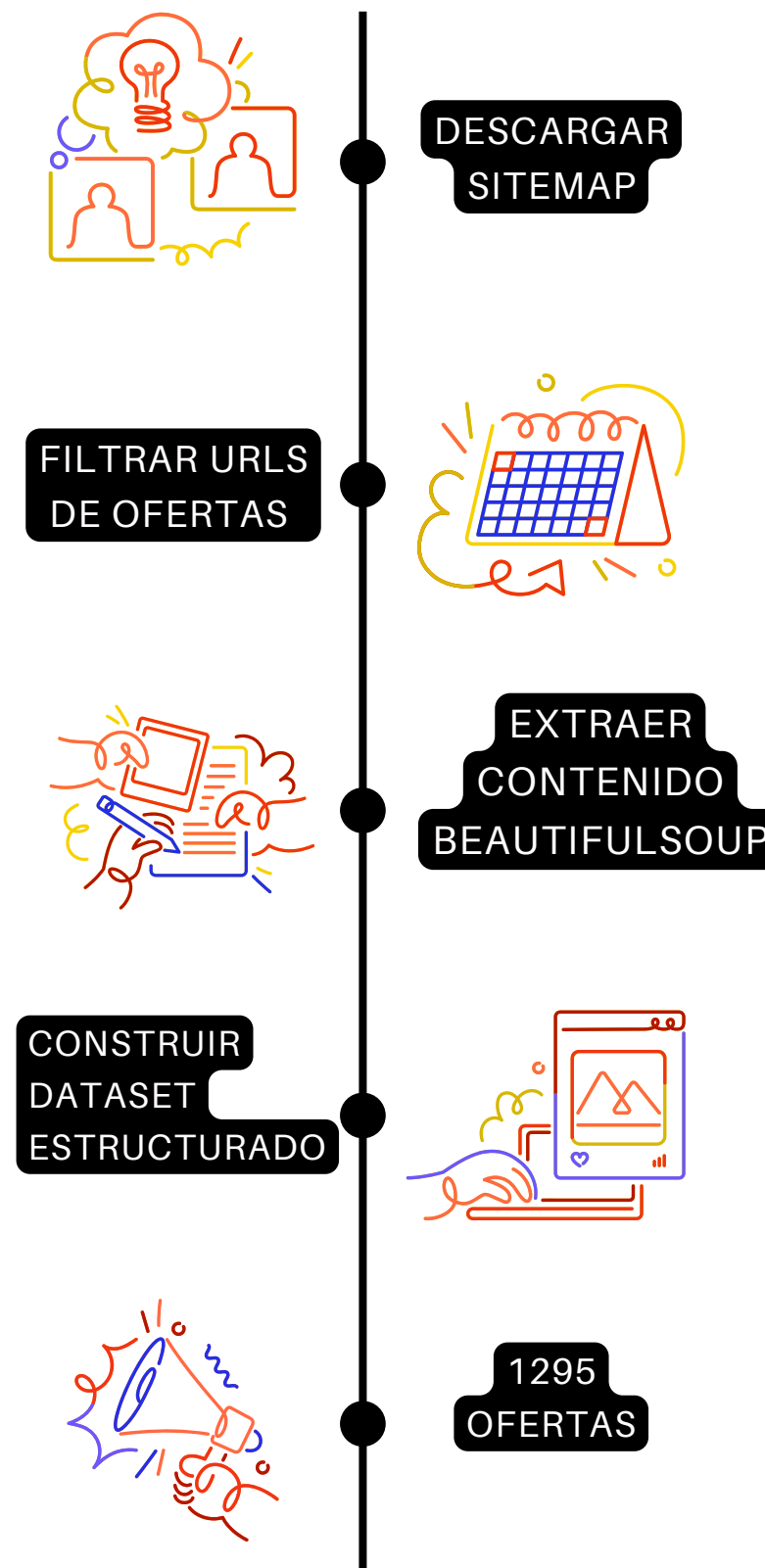
- escasa información salarial
- scraping bloqueado por robots.txt



- pocas ofertas con salario
- API no disponible y scraping limitado



- plataforma española
- scraping friendly



Campos extraídos por HTML:

Fuente / Campo

`<h1>`

Título del puesto

`<h3>` con "Salario" → ``

Salario

`<div id="skills">`

Skills

`<div id="requisitos">`

Descripción

`<div id="donde">`

Ubicación

Texto "Oferta cerrada" en el HTML

Estado

Preprocesamiento y limpieza de datos



Procesamiento y limpieza de datos

Conversión a minúsculas

Estandarización de los textos (`.lower_`)

Eliminación:

- Signos de puntuación
Mediante expresiones regulares
- Acentos
Reemplazo de las vocales acentuadas
- Emojis
Eliminar cualquier símbolo no textual (librería *emoji*)
- Stopwords
Modelo `es_core_news_lg`

Lematización

Modelo `es_core_news_lg`

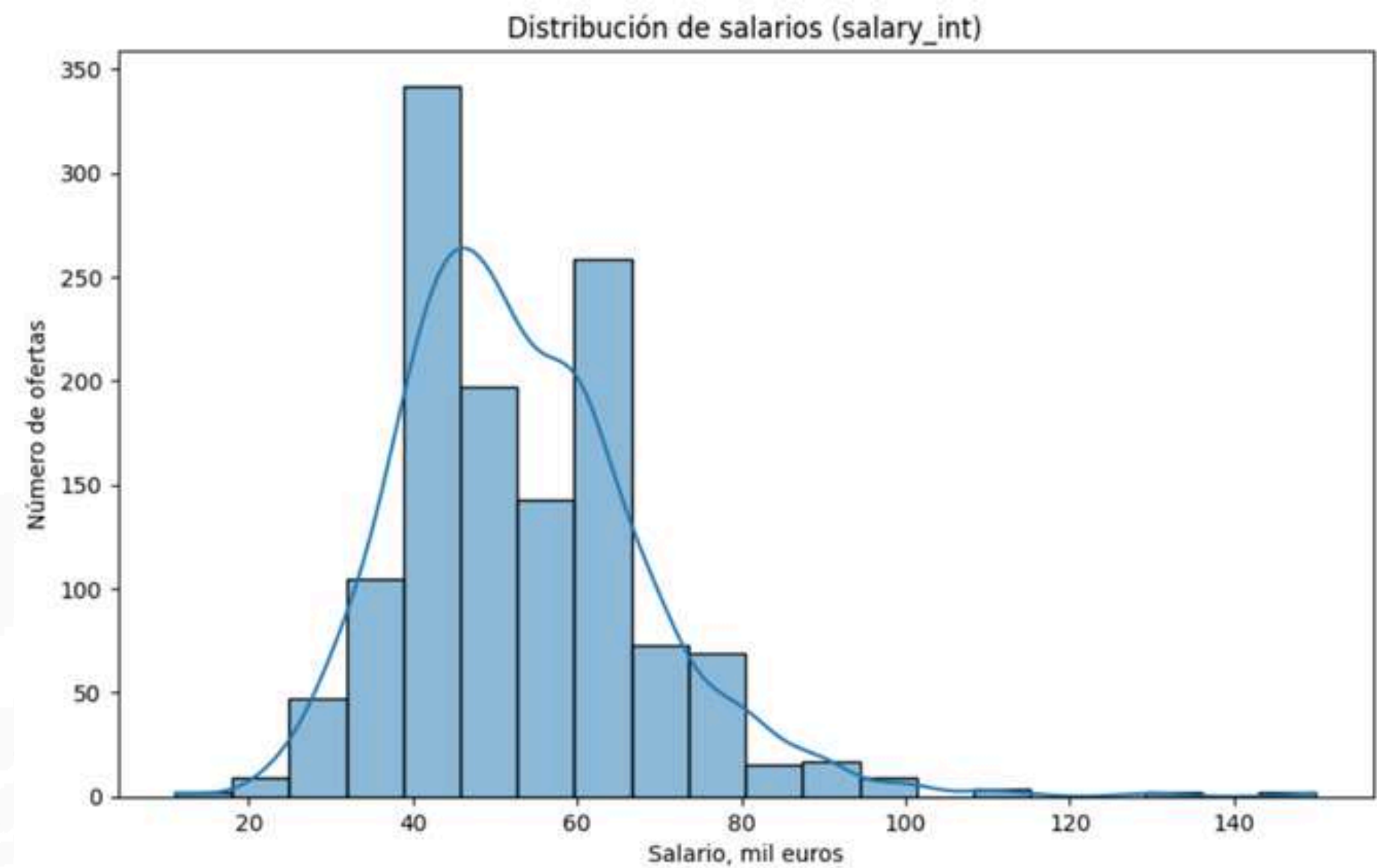


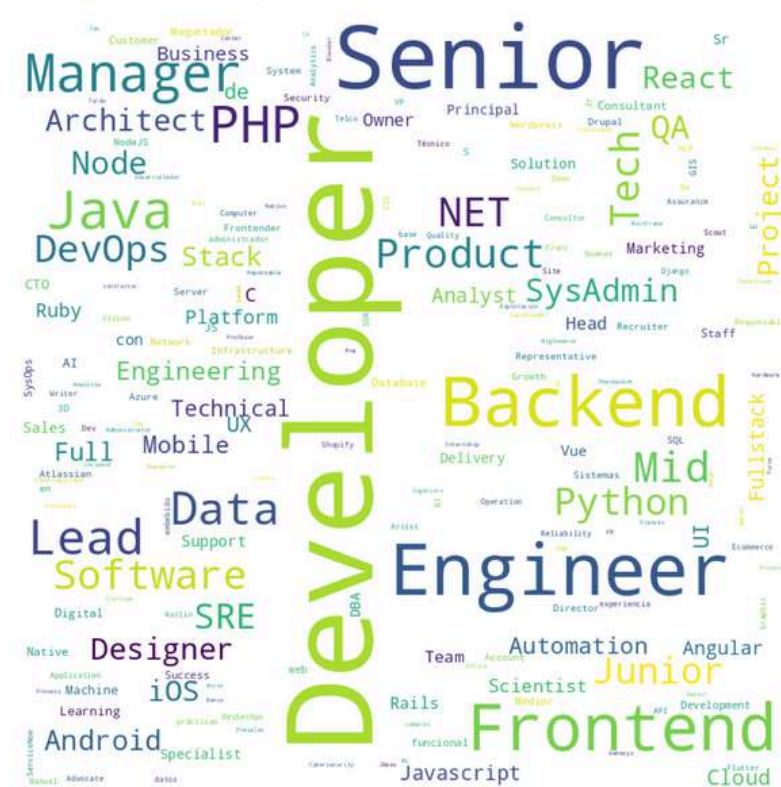
Análisis Exploratorio de Datos

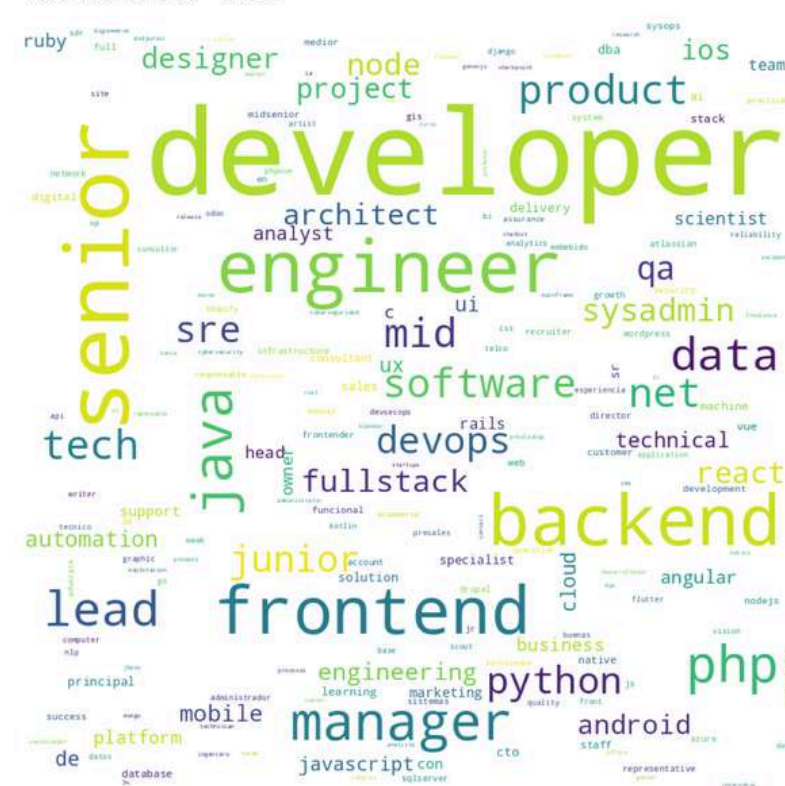
- 1250 ofertas en español
- Alta diversidad de títulos (587 perfiles únicos)
- Profundidad temporal (5 años)
- Histórico de ofertas (97.6% cerradas)

Análisis Exploratorio de Datos

Statistic	Value (miles)
Mean	52,7712
Std Dev	15,0386
Min	11
25% (Q1)	42
50% (Median)	50
75% (Q3)	60
Max	150







Modelos

Modelos Propios



Vectorización del texto

- Modelos de vectores dispersos (sparse):
 - Bag of Words (BoW)
 - Term Frequency-Inverse Document Frequency (TF-IDF)
- Modelos de word embeddings:
 - Word2Vec
 - FastText
 - Doc2Vec

Variable full_description

Variable objetivo salary_int

Modelos de Regresión

- Ridge
- Lasso
- RandomForest
- XGBoost


Métricas

- Error cuadrático medio (MSE)
- Error absoluto medio (MAE)
- coeficiente de determinación (R^2)



Modelos Preentrenados




 Zero-Shot Classification

MoritzLaurer/mDeBERTa

Multi -lenguaje

Large Size (> 200M pesos)



 Zero-Shot Classification

Recognai/bert

Español

Medium Size (100M - 200M pesos)

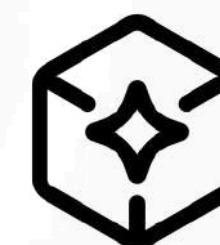


 Text Classification

papluca/xlm-roberta

Multi -lenguaje

Large Size (> 200M pesos)



 Feature Extraction

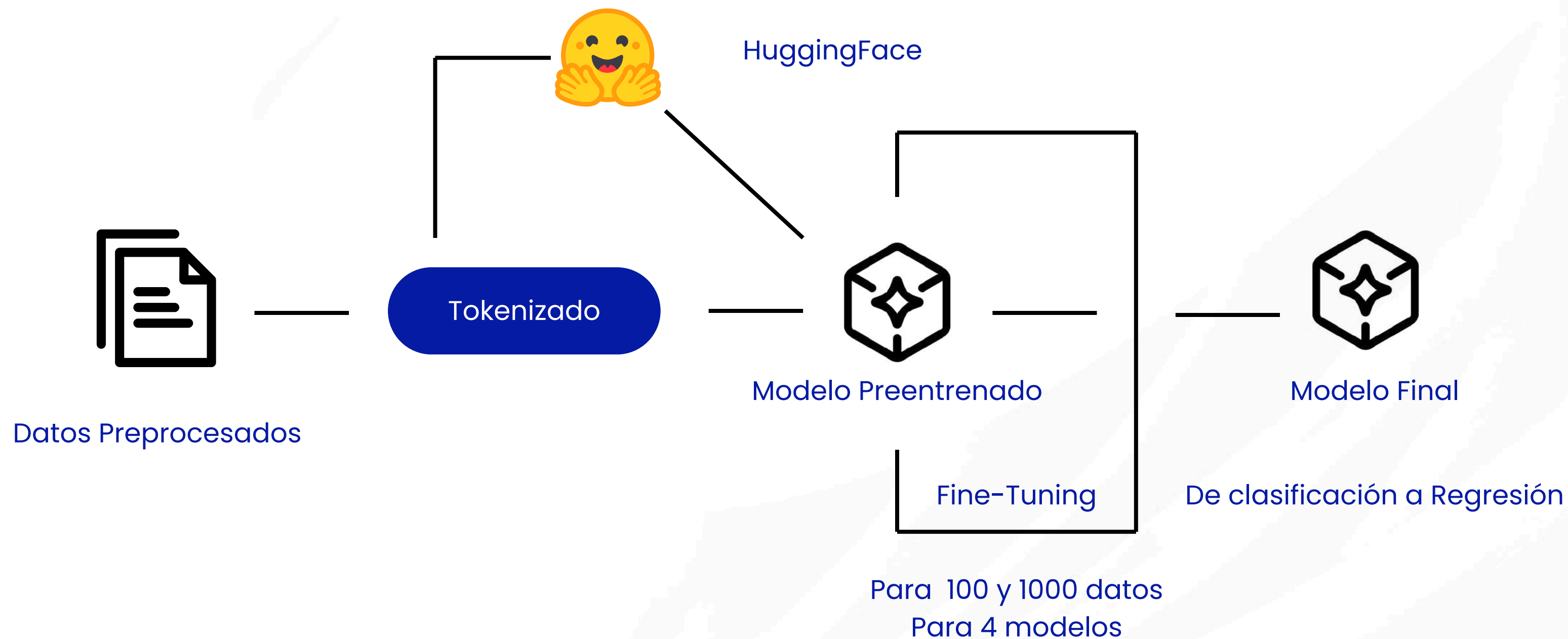
Unbabel/xlm-roberta

Multi -lenguaje

Small Size



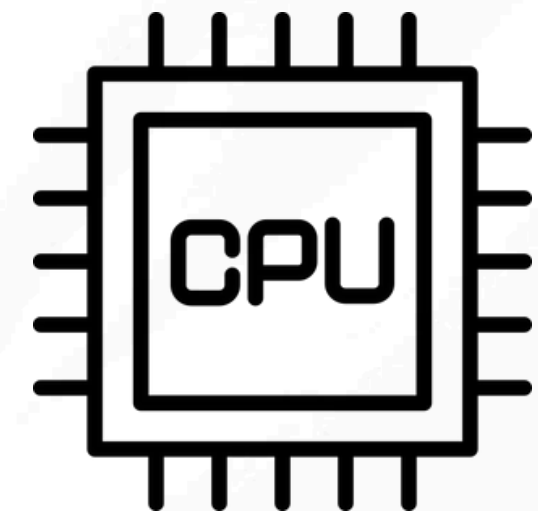
Modelos Preentrenados



Modelos Preentrenados

Limitaciones

- Límite de uso de CPU (Modelos muy pesados)
Batch size máximo de entrenamiento: 4
- Dataset reducido (1200 ofertas de empleo)
- Entrenamiento muy lento (250 iteraciones = 4 horas)



Resultados



Resultados

Modelos Propios

- Resultados bajos, siendo el mejor R^2 de apenas 0.569.
- Modelos con vectorizaciones más simples tienden a un mejor desempeño.
- No se beneficia significativamente de representaciones semánticas como los embeddings.
 - Naturaleza o al tamaño limitado del conjunto de datos.
 - Ausencia de modelos que utilicen representaciones contextuales del texto.

Model	Vectorization	R^2	MSE	MAE
Lasso	bow	0.569105	107.120354	7.289144
XGBoost	bow	0.503047	123.542330	7.407947
Ridge	tfidf	0.499786	124.353056	7.242946
XGBoost	tfidf	0.441463	138.851975	7.791703
Ridge	bow	0.436428	140.103802	8.114663
RandomForest	bow	0.435464	140.343370	7.847564
RandomForest	tfidf	0.421821	143.735008	7.866725
Ridge	fasttext	0.323746	168.116528	9.162820
RandomForest	fasttext	0.287029	177.244395	9.295514
Lasso	fasttext	0.264787	182.773607	9.677204
RandomForest	doc2vec	0.203552	197.996618	9.936000
XGBoost	fasttext	0.199874	198.911006	9.980428
Ridge	word2vec	0.177621	204.443157	10.088231
XGBoost	doc2vec	0.173001	205.591721	10.103736
RandomForest	word2vec	0.158187	209.274412	10.116769
Lasso	doc2vec	0.151876	210.843272	10.282277
Ridge	doc2vec	0.151118	211.031775	10.311420
XGBoost	word2vec	0.129343	216.444907	10.218765
Lasso	word2vec	0.068840	231.485961	11.268457
Lasso	tfidf	0.032042	240.634031	11.459257



Resultados

Modelos Preentrenados

Model	MSE	MAE	R2
MoritzLaurer 100	2483.5674	47.1785	-8.6354
Papluca 100	2908.6538	51.4848	-10.2845
Recognai 100	2893.9773	51.3430	-10.2276
Unbabel 100	3310.9653	55.2558	-11.8454
MoritzLaurer 1000	2192.8357	43.9974	-7.5074
Recognai1000	2169.5671	43.7373	-7.4171





Personalización de stopwords

Personalización de stopwords

Idea: Reducción del ruido léxico

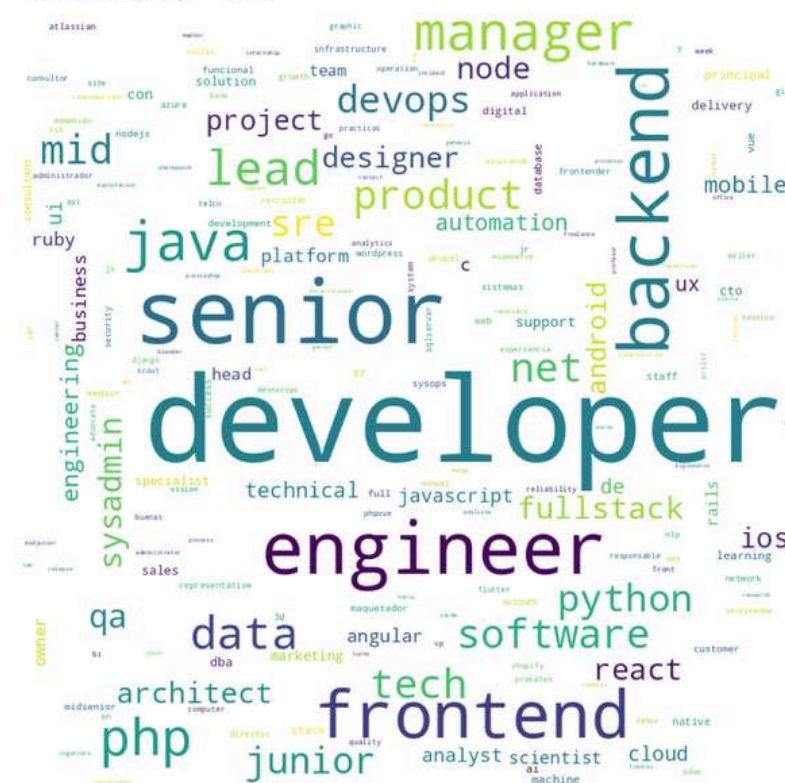
Filtrado estadístico

- Bag-of-Words



Filtrado por importancia predictiva

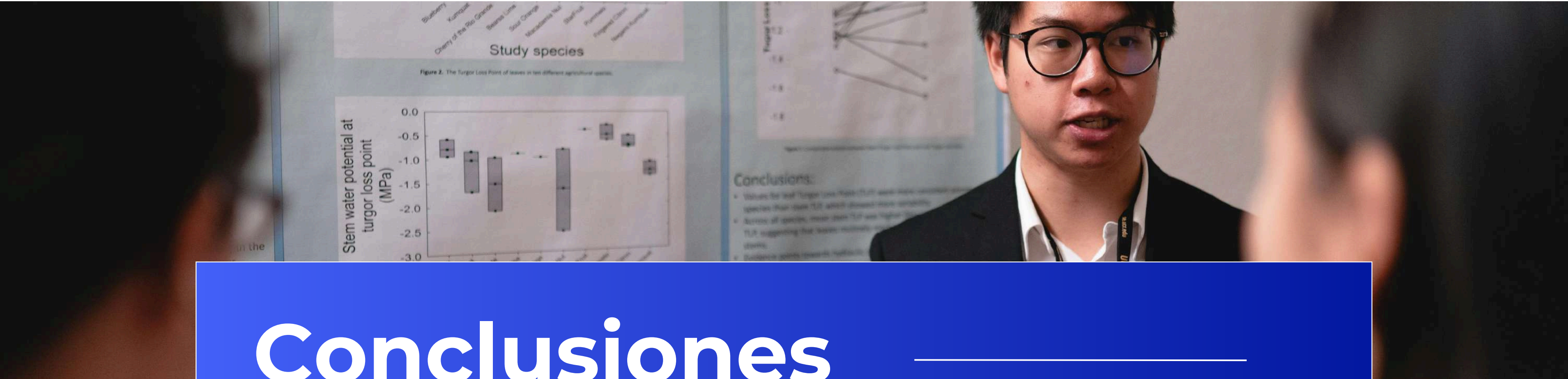
- TF-IDF
- Lasso model



Resultados

Model	Vectorization	R ²	MSE	MAE
Ridge	bow	0,7	83,36	6,34
Lasso	bow	0,62	107,52	6,74
XGBoost	bow	0,59	114,21	6,92
Ridge	tfidf	0,58	117,51	6,74
XGBoost	tfidf	0,58	118,47	7,15
RandomForest	bow	0,52	135,04	7,05
RandomForest	tfidf	0,51	138,26	7,32
Ridge	fasttext	0,41	163,97	8,4
Ridge	doc2vec	0,41	165,95	8,93
RandomForest	fasttext	0,41	166,26	8,39
XGBoost	doc2vec	0,4	167,92	8,77
XGBoost	fasttext	0,38	173,53	8,72
RandomForest	doc2vec	0,37	175,31	9,01
RandomForest	word2vec	0,33	187,29	9,19
XGBoost	word2vec	0,32	189,94	9,08
Lasso	fasttext	0,3	194,77	9,38
Ridge	word2vec	0,25	209,68	9,72
Lasso	doc2vec	0,24	213,62	10,09
Lasso	tfidf	0,23	216,21	9,35
Lasso	word2vec	0,16	235,35	10,46





Conclusiones

Conclusiones

- Aumento del corpus: recopilación y ampliación de datos estructurados a partir de fuentes como Kaggle.
- Extracción de características: identificación de entidades y palabras clave específicas (tecnologías, skills, experiencia...).
- Modelado: uso de modelos de representación contextual del texto para mejorar la comprensión semántica.
- Estudio de la evolución y tendencias a lo largo del tiempo, dada la disponibilidad de la variable temporal.

¡Muchas Gracias!

GRUPO 3

Alejandro Dionis Ros, Adrián Lizzadro Plá,
Sergio Sebastiá García e Ilia Zhigarev



VNIVERSITAT
DE VALÈNCIA

Solicitar puesto



Guardar oferta

