Clasificación de Sexismo en tweets

MIARFID - ALC - Laboratorio 1

Shiyi Cheng - Pablo Segovia Martínez

27 de febrero de 2026

1. Introducción

Este proyecto aborda la tarea de detección automática de contenido sexista en publicaciones de redes sociales, utilizando el dataset de la competición EXIST 2025. El objetivo es desarrollar y evaluar diferentes aproximaciones de clasificación binaria (sexista/no sexista) mediante modelos clásicos de machine learning y modelos de lenguaje pre-entrenados con fine-tuning.

2. Metodología

2.1. Preprocesamiento de Datos

Se han implementado dos estrategias de preprocesamiento:

- **Tweet Original**: Texto sin modificaciones, manteniendo URLs, menciones y emojis.
- **Text Clean**: Limpieza avanzada eliminando URLs, menciones, hashtags, y normalizando el texto.

2.2. Modelos Evaluados

Se ha experimentado con tres familias de modelos:

2.2.1. Modelos Clásicos

Modelos de machine learning tradicionales utilizando vectorización TF-IDF y Bag-of-Words:

- Regresión Logística
- Support Vector Machines (LinearSVC, SVC-RBF)
- Árboles de Decisión y Random Forest
- Gradient Boosting y AdaBoost
- Naive Bayes (Multinomial, Complement, Bernoulli)
- K-Nearest Neighbors
- Ensambles (Voting, Bagging, Stacking)

2.2.2. Modelos de Lenguaje Fine-tuned

Modelos transformer pre-entrenados ajustados con LoRA (Low-Rank Adaptation):

- **F2LLM-4B**: Modelo de 4B parámetros especializado en español
- **KaLM**: Modelo multilingüe con soporte para español

2.2.3. Modelos Generativos

Modelos grandes de lenguaje evaluados en modo zero-shot e inference con fine-tuning:

- **Ministral-3-8B-Instruct**: Modelo instruccional de 8B parámetros con cuantización FP8

3. Resultados

3.1. Evolución entre Versiones

Se realizaron dos iteraciones completas del proyecto (V1 y V2), con mejoras en el preprocesamiento y ajustes en los hiperparámetros de entrenamiento. La Tabla 1 muestra la evolución del rendimiento en los dos mejores modelos.

Modelo	Versión	Accuracy	Recall	F1-Score
F2LLM-4B (tweet)	V1	0.8604	0.8642	0.8464
F2LLM-4B (tweet)	V2	**0.8593**	**0.9185**	**0.8532**
KaLM (tweet)	V1	0.8429	0.8346	0.8254
KaLM (tweet)	V2	0.8429	0.8346	0.8254

Tabla 1: Comparativa V1 vs V2 (Mejores Modelos)

Mejoras en V2: F2LLM-4B mejoró su recall en +5.4 puntos porcentuales y su F1-Score en +0.68 puntos, logrando un mejor equilibrio entre detección de casos positivos y precisión.

3.2. Métricas Finales en Conjunto de Validación (V2)

La Tabla 2 muestra los resultados de los modelos de lenguaje fine-tuned, mientras que la Tabla 3 presenta los mejores modelos clásicos.

Modelo	Accuracy	Precision	Recall	F1-Score
F2LLM-4B (tweet)	**0.8593**	0.7966	**0.9185**	**0.8532**
F2LLM-4B (clean)	0.8418	0.7867	0.8765	0.8317
KaLM (tweet)	0.8429	0.8164	0.8346	0.8254
KaLM (clean)	0.8363	0.8137	0.8198	0.8167
Ministral-3-8B (ZS)	0.8264	0.7892	0.8321	0.8073
Ministral-3-8B (FT)	0.5725	0.9000	0.0444	0.0847

Tabla 2: Resultados de Modelos de Lenguaje V2 (DEV)

Nota importante: El fine-tuning de Ministral-3-8B con LoRA empeoró drásticamente el rendimiento (F1: 0.0847), sufriendo un colapso en el recall. Esto indica problemas en el proceso de fine-tuning que requieren investigación adicional.

Modelo	Accuracy	Precision	Recall	F1-Score
Stacking	**0.7824**	0.7867	0.7012	**0.7415**
Gradient Boosting	0.7736	**0.8373**	0.6099	0.7057
LinearSVC	0.7736	0.7901	0.6691	0.7246
Logistic Regression	0.7725	0.7845	0.6741	0.7251
Bagging (LR)	0.7703	0.7849	0.6667	0.7210

Tabla 3: Mejores Modelos Clásicos (DEV) - TF-IDF

3.3. Análisis Comparativo

3.3.1. Rendimiento por Familia de Modelos

- **Modelos de Lenguaje**: Los modelos transformer fine-tuned superan significativamente a los modelos clásicos, con mejoras de hasta **+11.2 puntos** en F1-Score (F2LLM-4B vs Stacking). El modelo F2LLM-4B alcanza el mayor recall (0.9185) manteniendo un F1 competitivo.
- **Modelos Clásicos**: El ensamble mediante Stacking alcanza el mejor rendimiento (F1=0.7415), demostrando que la combinación de múltiples clasificadores mejora los resultados individuales en aproximadamente +1.6 puntos sobre la regresión logística simple.
- **Impacto del Preprocesamiento**:
  - Los textos originales (tweet) obtienen mejor equilibrio precision-recall
  - Los textos limpios (clean) mejoran ligeramente el recall pero reducen precision
  - La diferencia entre ambas estrategias es menor en V2 que en V1
- **Ensamble de Top 5 Modelos**: Se evaluó un ensamble por votación mayoritaria combinando F2LLM-4B (tweet y clean), KaLM (tweet), Ministral-3B y Regresión Logística. El resultado obtuvo **F1=0.8532**, idéntico al mejor modelo individual (F2LLM-4B tweet), sin aportar mejora debido a la dominancia de este último.

3.3.2. Mejores Configuraciones (V2)

1. **Mayor F1-Score**: F2LLM-4B con tweet original **(0.8532)** — Mejor modelo general
2. **Mayor Recall**: F2LLM-4B con tweet original (0.9185) — Maximiza detección de casos positivos
3. **Mayor Precision**: Ministral-3-8B zero-shot (0.7892) — Entre los modelos viables
4. **Mejor modelo clásico**: Stacking con TF-IDF (0.7415) — Mejor alternativa computacionalmente eficiente
5. **Ensemble Top 5**: Votación mayoritaria (0.8532) — No mejora sobre F2LLM-4B individual

Recomendación final: Para la competición EXIST 2025, el modelo **F2LLM-4B con texto original** (F1=0.8532) es la mejor opción, ya que el ensemble no aporta mejora adicional y añade complejidad innecesaria.

4. Conclusiones

1. Los modelos de lenguaje pre-entrenados con fine-tuning superan ampliamente a los métodos clásicos de ML, con **mejoras de hasta +11.2 puntos** en F1-Score, especialmente en tareas de comprensión contextual como la detección de sexismo.

2. El modelo **F2LLM-4B con texto original (V2)** alcanza el mejor rendimiento global (**F1=0.8532, Recall=0.9185**), mejorando +0.68 puntos sobre V1 y constituyendo la mejor solución para esta tarea.

3. El preprocesamiento en V2 logró mejorar el recall sin sacrificar demasiado la precision, demostrando que el texto original (sin limpieza agresiva) mantiene información contextual valiosa para la clasificación.

4. Los ensambles de modelos clásicos (Stacking, F1=0.7415) son competitivos y computacionalmente más eficientes que los LLMs. Sin embargo, el ensemble de LLMs (votación top 5) **no mejoró** sobre el mejor modelo individual, indicando la dominancia de F2LLM-4B.

5. El fine-tuning con LoRA permite adaptar modelos grandes (4B parámetros) con recursos limitados, pero requiere **cuidadosa supervisión**: Ministral-3-8B colapsó tras el fine-tuning (F1: 0.0847), sugiriendo problemas con la tasa de aprendizaje o datos de entrenamiento.

6. La iteración V1→V2 validó la importancia de refinar hiperparámetros y estrategias de preprocesamiento para maximizar el rendimiento de modelos transformer.

4.1. Trabajo Futuro

- **Investigar el fallo de Ministral-3-8B fine-tuning:** Analizar por qué el recall colapsó a 0.0444 y probar ajustes en learning rate, warmup steps o arquitectura LoRA.
- **Análisis de errores cualitativo:** Identificar patrones lingüísticos específicos (ironía, sarcasmo, referencias culturales) donde los modelos fallan sistemáticamente.
- **Explorar ensambles avanzados:** Probar estrategias de combinación ponderada basadas en confianza de predicción o stacking con meta-aprendizaje, en lugar de votación simple.
- **Optimización de hiperparámetros:** Búsqueda sistemática (grid/random search) para LoRA rank, alpha, learning rate y dropout.
- **Evaluar modelos más recientes:** Probar arquitecturas como Llama 3, Mixtral 8x7B o modelos específicos de español como BERTIN o RoBERTa-es.
- **Augmentación de datos:** Generar ejemplos sintéticos con LLMs o técnicas de back-translation para balancear clases y mejorar generalización.