TC3003B

DETECTOR DE PLAGIO DE SOFTWARE

Rafael Hinojosa López
Enrique Santos Fraire
Keyuan Zhao

Profesores

Dr. Benjamín Valdés Aguirre
Dr. Pedro Óscar Pérez Murrueta
Mtro. Manuel Iván Casillas del Llano

Según el General Report on Software Piracy publicado en BSA (Business Software Alliance), el plagio de software causó pérdidas económicas cerca de 46.3 mil millones de dólares en todo el mundo en 2019.

PROBLEMA

El plagio de software se considera como utilizar el código fuente de otra persona sin su autorización y adjudicarse como propio.

Principales consecuencias en el plagio de software:

- Viola los principios éticos de honestidad, integridad y respecto por la propiedad intelectual
- Mala calidad y seguridad del software
- No hay competencia justa para los desarrolladores
- Pérdida económica

ESTADO DEL ARTE

Autores	Hechos relevantes	Precisión
Omi, Hossain, Islam y Mittra (2021)	DNN, SVM y LSTM	96.7%
Bandara y Wijayarathna (2011)	Naïve Bayes Classifier, KNN y AdaBoost Meta-learning	(75% - 86)%
Ullah, Wang, Farhan, Habib y Khalid (2021)	PCA y MLR	(73% - 86)%

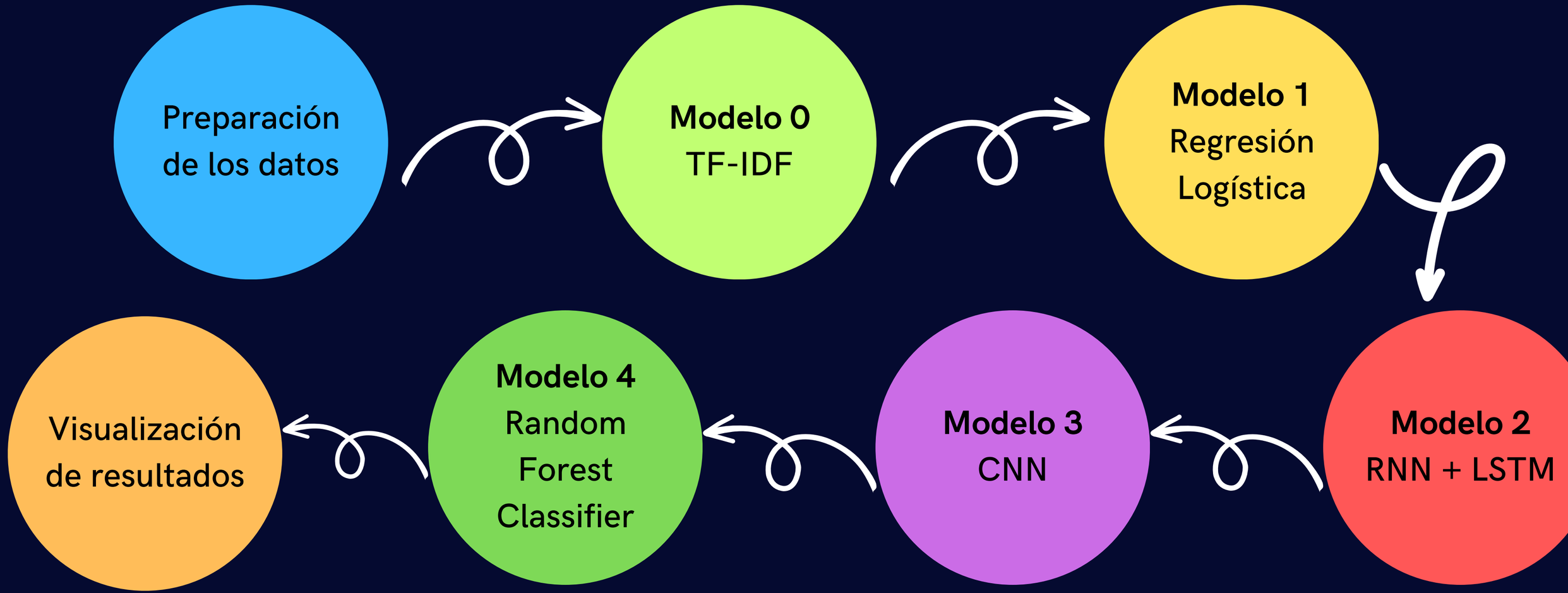
OBJETIVO

Determinar si dos códigos fuente escritos en Java contienen plagio al determinar su similitud utilizando métodos matemáticos como primera aproximación, y posteriormente refinar dicha predicción con el uso de Machine Learning y Deep Learning para obtener resultados más precisos.

OBJETIVO ESPECÍFICO

- Determinar el porcentaje de plagio que contiene un código de Java y posteriormente dictaminar un veredicto, evitando caer en falsos positivos (detectado como PLAGIO cuando NO lo es) al elevar el porcentaje de similitud a un 85% para considerar el par de códigos como plagiados.
- Identificar los segmentos plagiados de los códigos.

FLUJO DE TRABAJO

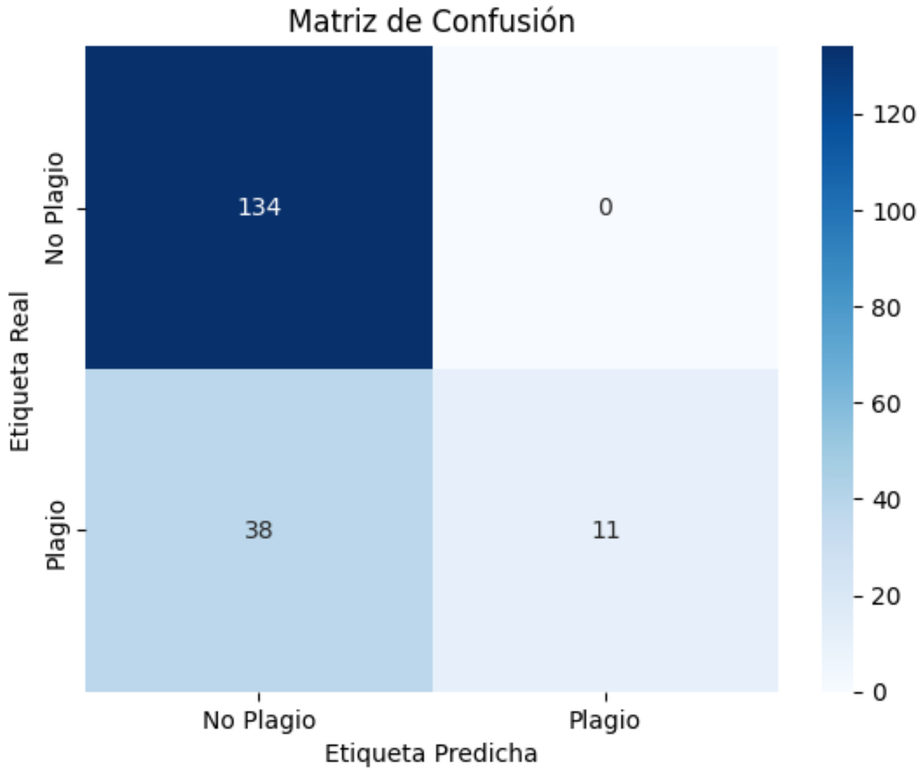


M0: TFD-ID

Técnica que calcula los valores de frecuencia para cada término y compara los vectores resultantes utilizando la distancia del coseno.

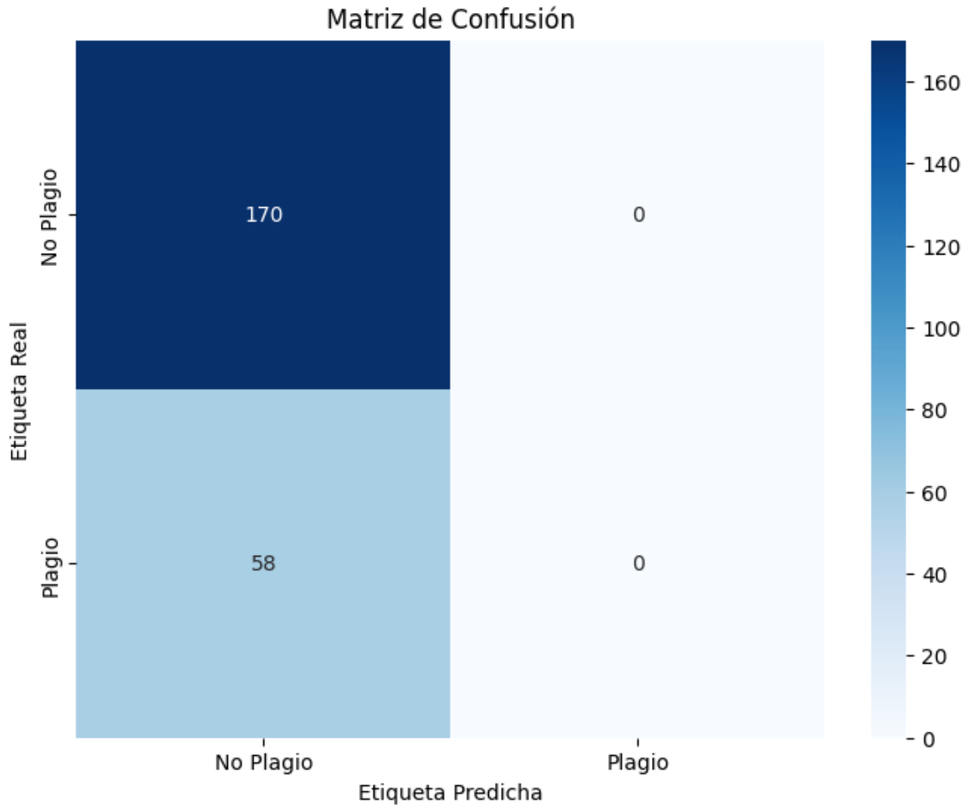
	Name1	Name2	Similitud	verdict
0	0017d438	9852706b	0.659862	1
1	0017d438	ac180326	0.582237	0
2	0048a372	0adb1ee5	0.437342	0
3	00af3420	5449d33c	0.529478	0
4	00af3420	86102d81	0.570880	0
...	...	...	...	...
906	eea69e7f	f6ca6fc8	0.494697	0
907	f229aa7f	fcc7e8fa	0.338990	0
908	f28b8cb4	ff3283cf	0.588601	0
909	fadc1365	fdd85afb	0.588751	0
910	fc7dfa16	fe94ee2f	1.000000	1

M1: REGRESIÓN LOGÍSTICA



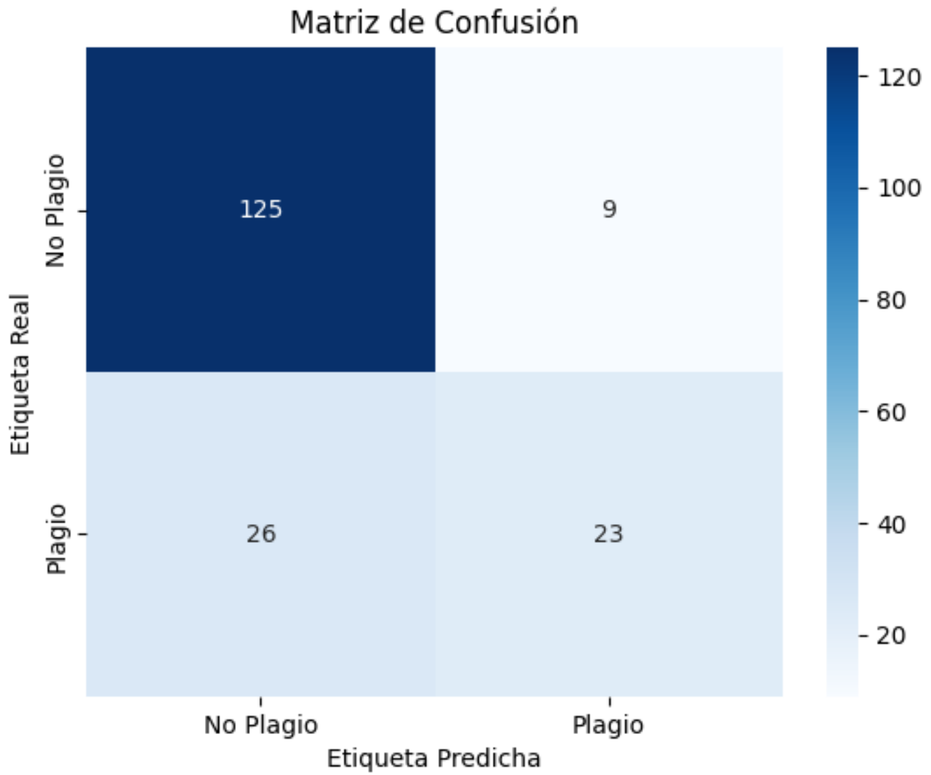
Precisión	79.82%

M2: RNN + LSTM



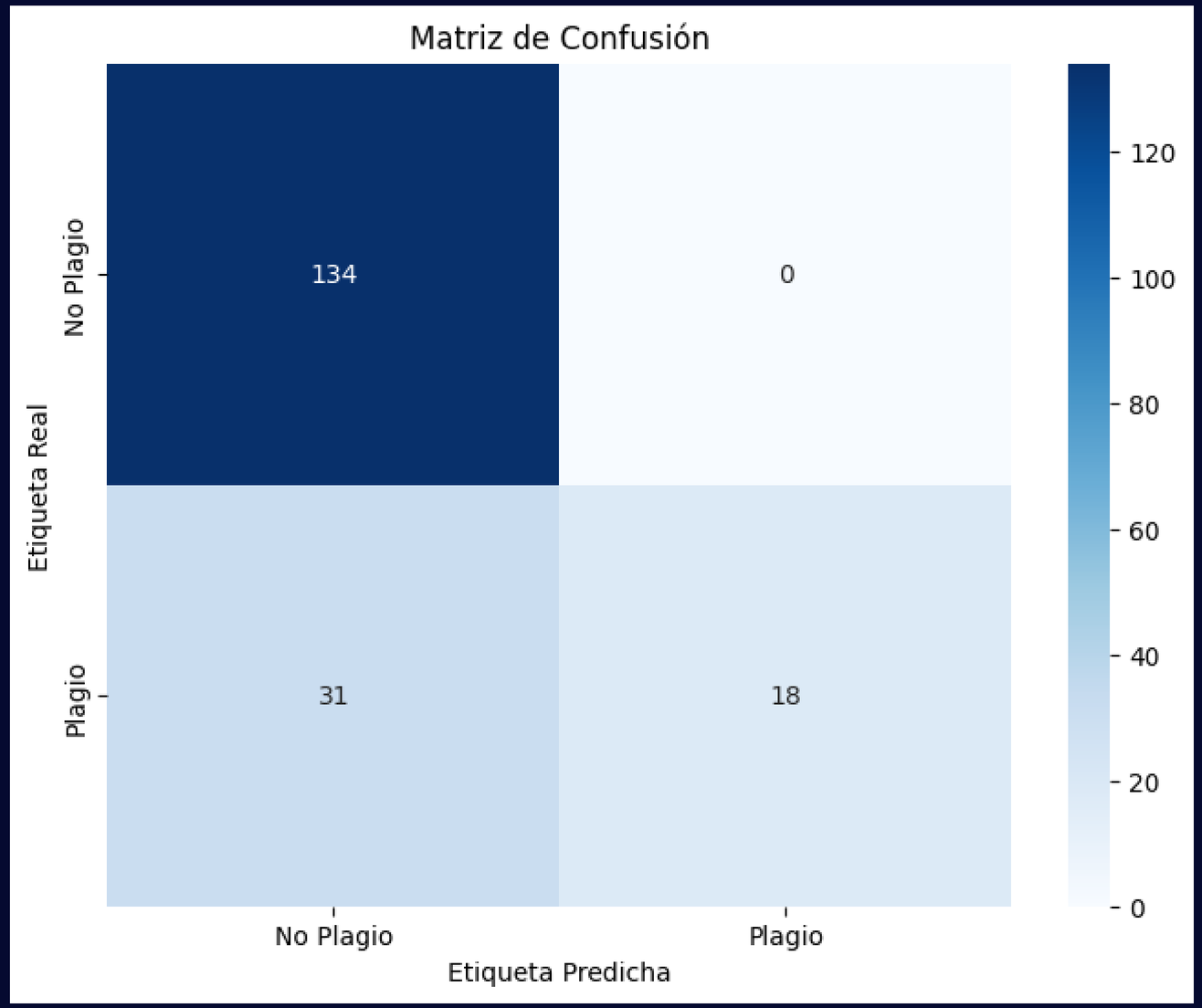
Precisión	70.61%

M3: CNN



Precisión	84.15%

M4: RANDOM FOREST CLASSIFIER

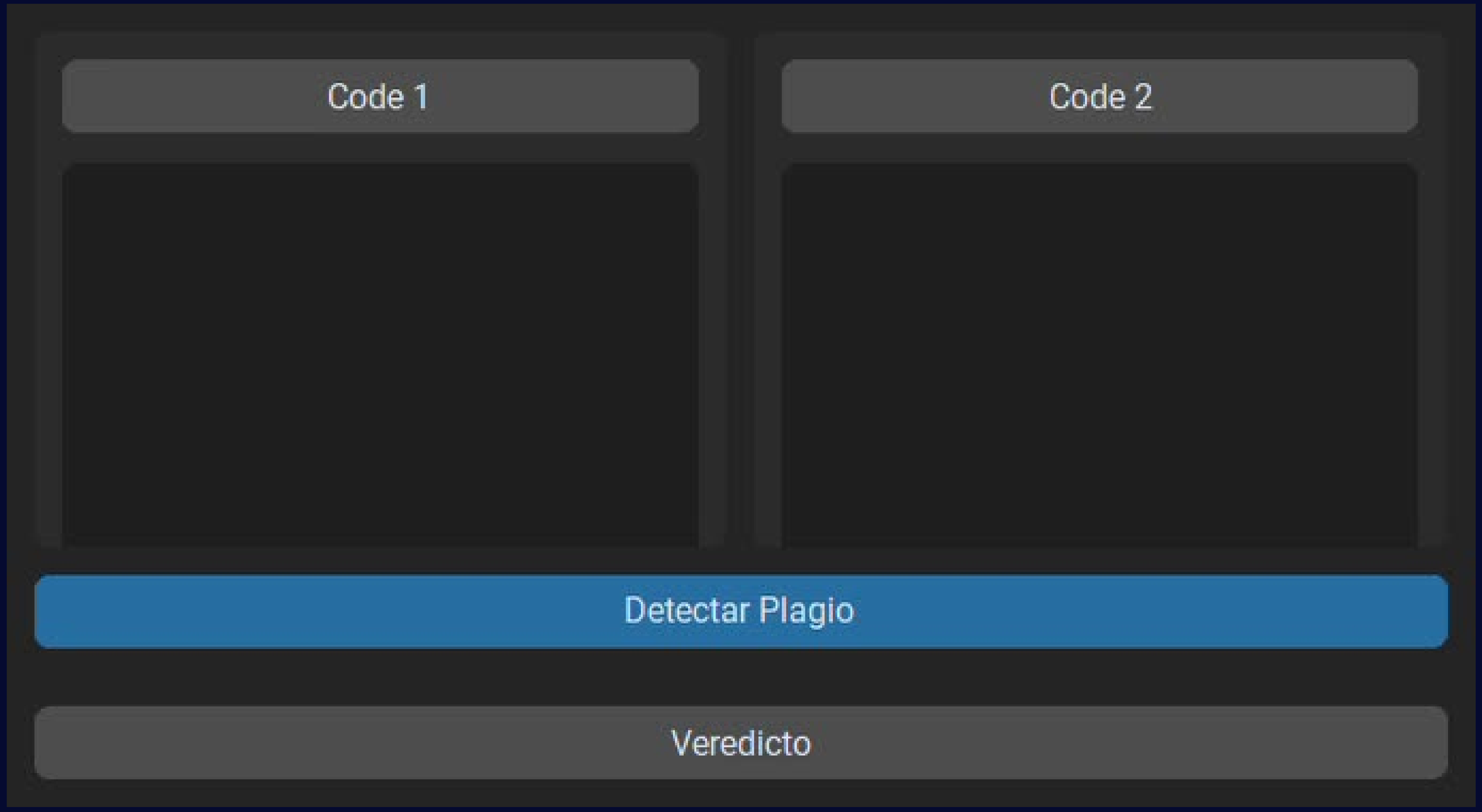


Métrica	Score
Exactitud	83.06%
Precisión	100%
Recuperación	36.73%
Puntuación F1	53.73%

COMPARACIÓN DE MODELOS

Modelo	Accuracy
CNN	84.15%
Random Forest Classifier	83.03%
Regresión Logística	79.82%
RNN + LSTM	70.61%

DEMOSTRACIÓN



Conclusiones y trabajo futuro

REFERENCIAS

F. Ullah, J. Wang, M. Farhan, M. Habib and S. Khalid, “Software plagiarism detection in multiprogramming languages using machine learning approach” Concurrency Computat Pract Exper, vol. 33, no. 4, Feb 2021.

U. Bandara and G. Wijayarathna, “A Machine Learning Based Tool for Source Code Plagiarism Detection”, *International Journal of Machine Learning and Computing*, vol. 1, no. 4, Oct 2011.

A. M. Omi, M. Hossain, M. N. Islam and T. Mittra, "Multiple Authors Identification from Source Code Using Deep Learning Model," *2021 International Conference on Electronics, Communications and Information Technology (ICECIT)*, pp. 1-4, Dec 2021.

"Homepage | BSA | The Software Alliance" [Online]. Homepage | BSA | The Software Alliance. Available: https://www.bsa.org/

Gracias