Analysis and Classification of Websites Using Artificial Intelligence for Domain Registration Authorities

Néstor Adrián Balich[0009-0002-3868-1967], Berenice Lourdes Balich[0009-0007-2783-2842]

Laboratorio de Robótica Física e Inteligencia Artificial, CAETI - Centro de Altos Estudios en Tecnología Informática, Universidad Abierta Interamericana, Montes de Oca 745, Ciudad Autónoma de Buenos Aires, Argentina
{Nestor.Balich, BereniceLourdes.Balich}@uai.edu.ar

Abstract. Massive web data collection is a key task for research, cybersecurity, market analysis, and national domain registries such as NIC.ar in Argentina. However, traditional scraping techniques face increasing challenges due to dynamic websites using images, banners, and elements generated with JavaScript. This paper proposes a hybrid scraping model combining traditional static and dynamic scraping with text recognition (OCR) and object recognition powered by artificial intelligence. We implemented two softbots: one for OCR (Tesseract) and one for object recognition (YOLO) on screenshots of websites previously inaccessible via traditional methods. The system processed 50,000 domains and was able to recover information from 80% of the previously unprocessable cases. This lays the groundwork for the next stage involving supervised learning-based website classification.

Keywords: scraping, OCR, artificial intelligence, domain analysis, distributed processing.

Análisis y clasificación de páginas webs mediante inteligencia artificial para organismo de registro de dominios

Resumen. La recolección masiva de datos es una tarea crucial en ámbitos como la investigación, la seguridad y la regulación de dominios, especialmente en organismos nacionales como NIC.ar en Argentina. Sin embargo, el scraping tradicional enfrenta limitaciones ante sitios web dinámicos que presentan contenido como imágenes, banners o elementos generados por JavaScript. Este trabajo propone un modelo de scraping híbrido que complementa las técnicas estática y dinámica con reconocimiento de texto (OCR) y de objetos mediante inteligencia artificial. Se implementaron dos softbots: uno para OCR con Tesseract y otro para reconocimiento de objetos con YOLO. El sistema fue evaluado sobre un conjunto de 50.000 dominios, logrando recolectar información del 80% de los casos previamente inaccesibles. Este trabajo sienta las bases para

la siguiente etapa de análisis y clasificación automática mediante aprendizaje supervisado.

Palabras clave: scraping, OCR, inteligencia artificial, dominios web, procesamiento distribuido.

1 Introducción

El scraping de páginas web es una técnica utilizada para extraer información de sitios mediante navegación programática y procesamiento del contenido HTML. Existen dos enfoques comunes: el scraping estático, que actúa sobre contenido fijo en el código fuente, y el scraping dinámico, que emplea herramientas como Selenium o Puppeteer para acceder a contenido generado por JavaScript.
Sin embargo, cada vez más sitios web dificultan la recolección de información presentando su contenido principal como imágenes, banners o animaciones dinámicas, impidiendo que los scrapers convencionales detecten texto o enlaces relevantes. Este problema es particularmente significativo para organismos de registro de dominios, como NIC.ar, que deben monitorear grandes volúmenes de sitios registrados para verificar su actividad, detectar usos indebidos y garantizar el cumplimiento de las políticas nacionales sobre dominios. Cuando el contenido está oculto en imágenes o elementos dinámicos, estos organismos no pueden comprobar si un dominio está activo, si aloja contenido ilegal, o si es utilizado con fines fraudulentos como phishing.
La capacidad de recuperar información de sitios que eluden el scraping tradicional es clave para la investigación, la seguridad y la regulación de dominios. Permite a los organismos:

- Monitorear el cumplimiento de normativas de uso de dominios.
- Detectar sitios que promueven contenidos prohibidos o actividades ilegales.
- Generar estadísticas confiables sobre la actividad real de los dominios registrados, fundamentales para investigaciones en ciberseguridad y análisis de comportamientos web.

Este trabajo propone un enfoque mixto que complementa los modelos de scraping estático y dinámico con procesamiento de imágenes basado en inteligencia artificial. Se utilizan técnicas de reconocimiento óptico de caracteres (OCR) con Tesseract y detección de objetos con YOLO, para obtener información semántica de capturas de pantalla de sitios web que resultan inaccesibles mediante scraping tradicional. Este modelo híbrido sienta las bases para la posterior clasificación automática de sitios mediante aprendizaje supervisado, permitiendo un análisis más profundo y eficiente del ecosistema de dominios web.

2 Problemas detectados

Durante el desarrollo de proyectos previos en NIC.ar y en investigaciones apoyadas por el registro de direcciones de internet para América Latina y el Caribe (LACNIC), como el proyecto FRIDA (Frida, 2020), se identificaron diversos problemas que dificultan el scraping mediante técnicas convencionales:

- Sitios web que presentan solo imágenes sin texto HTML, lo que impide a los scrapers extraer contenido relevante.
- Texto incrustado en banners o imágenes, ilegible para scrapers basados en HTML.
- Navegación basada exclusivamente en botones o elementos visuales sin texto identificable en el código fuente.
- Cambios frecuentes en la estructura de los sitios, que invalidan los selectores de los scrapers.
- Bloqueos mediante CAPTCHAs, firewalls o restricciones de IP, que frenan o imposibilitan la automatización.
- Velocidad insuficiente en el procesamiento masivo de dominios debido a la complejidad de carga de los sitios.

Para NIC.ar, estos problemas son críticos porque dificultan el monitoreo de dominios registrados, impidiendo verificar que se encuentren activos y sean utilizados conforme a las políticas nacionales. Si el contenido del sitio está oculto tras imágenes o animaciones, se dificulta comprobar si un dominio es usado con fines legítimos o para actividades ilegales como fraude, phishing o distribución de contenido prohibido. Esta situación no solo obstaculiza la regulación efectiva, sino que también limita la generación de estadísticas confiables sobre la actividad y el uso de los dominios .ar.
En análisis realizados sobre 50.000 dominios registrados, se detectó que en 5.000 casos (10%) el scraping tradicional no permitió recuperar información del contenido de la página principal, lo que motivó la necesidad de implementar un modelo más robusto capaz de interpretar visualmente el contenido de las páginas y superar estas limitaciones.
Para resolver el problema del procesamiento y scraping de todas las páginas argentinas registradas en NIC (ver Fig. 1), se propuso un sistema de procesamiento distribuido que combinado con técnicas de virtualización, multiprocesamiento y procesamiento concurrente (Fujii, 2019; Wong, 2022), que fue altamente efectivo y que sentó las bases para el presente trabajo (Balich, N. Y Balich, F. 2024).

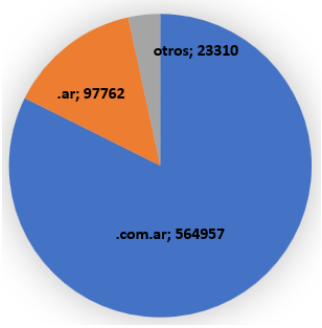

Fig. 1. Dominios registrados en NIC

3 Modelo propuesto

El OCR (Goei, 2022) es una técnica ampliamente probada en diversas áreas y, junto con el reconocimiento de objetos (OR), permite mejorar la extracción de datos tradicional basada en la estructura HTML (Selvy et al., 2022; Ruchitaa et al., 2023). Por otro lado, los recientes avances en clasificación mediante aprendizaje automático (ML) (Ertam, 2018) plantean un escenario propicio para la siguiente etapa de esta investigación: el análisis y clasificación automática de sitios. Pero para ello, primero es esencial poder recolectar datos de las páginas que eluden el scraping convencional, definiendo un ciclo de extracción, depuración y análisis de datos (EDA).

Para implementar el proceso de scraping mixto, se utilizaron herramientas de software libre como el lenguaje Python y librerías como Selenium (Selenium, 2025), BeautifulSoup (Beautiful Soup, 2025), PyQuery (PyQuery, 2017), YOLO (YOLO, 2025) y Tesseract (Tesseract, 2025). Las pruebas se realizaron en un servidor DELL PowerEdge 440 con un sistema de virtualización ESXi (VMware, 2025).

A. Arquitectura del sistema

Se diseñó una arquitectura distribuida basada en:

- Una API orquestadora desarrollada en Flask, desplegada con Waitress.
- Dos scrapers remotos con Selenium, Chrome headless y librerías de IA (softbots).
- Un sistema de multiprocesamiento y subprocesos para lanzar tareas en paralelo.
- Almacenamiento temporal de capturas y resultados en archivos CSV para su posterior análisis.

B. Tecnologías utilizadas

- Python como lenguaje principal.
- Selenium y BeautifulSoup para scraping estático y dinámico.
- Tesseract OCR como softbot 1 para detección de texto en capturas de pantalla.
- YOLOv8 como softbot 2 para detección de objetos en las imágenes de las páginas.
- OpenCV para el preprocesamiento de imágenes.
- Pandas para manejo y depuración de datos.
- VMware ESXi como entorno de virtualización.

C. Entrenamiento de YOLO

Para el reconocimiento de objetos se utilizó YOLOv8 con un modelo preentrenado en el conjunto COCO, que permite identificar elementos comunes como botones, íconos y banners presentes en la mayoría de sitios web. En esta etapa, no se realizó un ajuste fino (fine-tuning) con un dataset propio, ya que el objetivo principal fue validar la viabilidad del enfoque híbrido de scraping visual. En futuras etapas, se prevé entrenar YOLO con ejemplos específicos de dominios locales para mejorar la precisión de detección.

D. Pipeline del scraper

El pipeline implementado sigue estas etapas:

1. Intento de scraping estático usando BeautifulSoup.
2. Si falla, scraping dinámico con Selenium.
3. Captura de pantalla de la página principal.
4. Extracción de texto con OCR usando Tesseract.
5. Detección de objetos relevantes con YOLO.
6. Depuración y normalización de los datos.
7. Almacenamiento de resultados en archivos CSV para su posterior análisis.

4 Resultados

Se procesaron 50.000 dominios extraídos de NIC.ar. Se evaluaron tres configuraciones principales para optimizar el scraping:

A. **Threading:** 100 hilos por máquina virtual (VM), con alta velocidad pero sobrecarga en CPU durante el procesamiento de IA. Tiempo total: 110 horas.
B. **Multiprocesamiento:** 20 procesos por VM, mejorando la eficiencia en el uso de CPU. Tiempo total: 80 horas.
C. **Subprocesos + multiprocesamiento:** combinación que permitió ejecutar 100 procedimientos simultáneos. Tiempo total: 50 horas.

En la versión unificada (una sola VM con 40 hilos), el tiempo se redujo a 16 horas. Tras optimizar el modelo mixto de scraping, se logró procesar las 50.000 páginas en 12 horas, incluyendo el preprocesamiento y la depuración de los datos según los requisitos para la etapa de análisis con inteligencia artificial.

Gracias a la combinación de OCR y detección de objetos, se recuperó información en el 80% de los 5.000 sitios previamente inaccesibles mediante scraping convencional, lo que permite la construcción de un dataset para la siguiente etapa del proyecto. IA.

5 Análisis de resultados

El enfoque mixto demostró ser eficaz para superar las limitaciones del scraping tradicional. El uso de IA permitió extraer texto oculto en imágenes y reconocer patrones semánticos, permitiendo construir un dataset sentando las bases para la continuación de este trabajo, que consistente en la clasificación de las páginas mediante aprendizaje supervisado.
Se comprobó que la eficiencia del sistema depende de:

- La correcta asignación de tareas.
- El uso de procesamiento distribuido.
- La gestión eficiente de recursos (RAM y CPU).
- La creación de un softbot para el reconocimiento de OCR por IA.
- La creación de un softbot para el reconocimiento de Objetos por IA.

El sistema también incorporó tolerancia a fallas, control de tareas principalmente en recuperación y asignación de trabajo para los softbot encargados de realizar el procesamiento por IA.
Este enfoque mixto demostró ser eficaz para superar las limitaciones del scraping tradicional. El uso de IA permitió extraer texto oculto en imágenes y reconocer patrones semánticos, permitiendo construir un dataset para la siguiente etapa del proyecto, clasificación con aprendizaje supervisado.

Se comprobó la eficiencia del sistema:

- Logrando la recolección de información del 80% de las 5.000 páginas que no se pudieron procesar por el sistema tradicional de scraping.
- El uso de procesamiento distribuido permitió la gestión eficiente de recursos (RAM y CPU).

Sobre una primera clasificación manual de 100 páginas al azar sobre el 80% de ellas, se clasificaron en las categorías que pueden observarse en la Tabla 1.

Tabla 1. Clasificación manual de 100 páginas procesados con IA.

Categoría	Descripción	Número
1. Sitios con texto embebido en imagen	El contenido principal es una imagen con texto sin HTML legible.	35
2. Redireccionamiento externo visual	Redirige a otra página mediante un botón o imagen sin enlace HTML directo.	18
3. Página en blanco con logos	Página sin texto, solo muestra logos o imágenes institucionales.	6
4. Bloqueo por CAPTCHA visual	Muestra un CAPTCHA gráfico que impide el scraping automatizado.	2
5. Estructura animada sin texto	Sitios construidos con animaciones donde el contenido es inyectado posteriormente.	8
6. Ofertas de dominios en imagen	Páginas de venta del dominio donde el contenido está en imágenes.	14
7. Error o mantenimiento visualizado	Imágenes que informan error 404 o mantenimiento, sin código explícito.	8
8. Botón de acceso sin texto visible	Botones de acceso que no presentan texto legible por el scraper.	3
9. Páginas con contenido ilegal oculto	Páginas que presentan imágenes promocionando contenido prohibido.	4
10. Sitios sin contenido aparente	Capturas donde no se encuentra información visible o interpretable.	2

Este primer muestreo del 35% de las páginas principales que tenían imágenes, fotografías, o botones con significado para el ser humano sin texto reconocible, el 18%

eran páginas de redireccionamiento por lo general servicios de venta de hosting o a páginas maliciosas, un 14% se dedicaba a la venta de hosting, un 8% con problemas técnicos , en mantenimiento o sitio en construcción, 4% con contenido prohibido, y 3% no se pudo leer el texto que sumado al 2% hacen un 5% de páginas que no se pudieron clasificar de forma manual.

6 Conclusiones

Se logró implementar un sistema de scraping masivo capaz de interpretar contenido visual mediante inteligencia artificial, mejorando significativamente la capacidad de recolección de datos frente a sitios resistentes al modelo tradicional. El modelo híbrido propuesto demostró ser una alternativa viable para proyectos de vigilancia web, ciberseguridad y regulación de dominios, permitiendo recuperar información en el 80% de los casos previamente inaccesibles.

Si bien un 5% no pudo ser clasificado por el humano, se valido que en gran medida existen patrones visuales y de OCR promisorios para el entrenamiento del modelo de IA al escalar a las 5.000 páginas, encontramos que en la muestra aleatoria más de la mitad de las 100 páginas evidenciaban que el ocultamiento al scraping tradicional fue intencional..

Este trabajo evidencia que un porcentaje considerable de sitios utiliza técnicas que dificultan intencionalmente la indexación automatizada, como incrustar texto clave en imágenes o emplear redireccionamientos sin enlaces HTML. La combinación de scraping estático, dinámico, OCR y detección de objetos posibilitó la creación de un dataset inicial que será clave para futuras etapas de clasificación automática.

En próximas etapas se planea entrenar modelos supervisados para la clasificación automática de sitios web aprlicand modelos de aprendizaje supervisado y clustering, aplicando métricas como precisión, recall y F1-score para cuantificar su rendimiento. También es importante resaltar que, una vez entrenado el modelo, esperamos combinar los datos estáticos, dinámicos y de IA para poder clasificar las 50.000 páginas.

Se espera que este enfoque motive nuevas investigaciones en scraping avanzado, análisis de datos no estructurados y uso de IA en el monitoreo de sitios web con relevancia para investigación, seguridad y regulación de dominios.

Bibliografía

Balich, N., & Balich, F. (2024). Implementación y optimización de un sistema masivo de scraping basado en técnicas de procesamiento paralelo para dominios argentinos. 30º Congreso Argentino de Ciencias de la Computación (CACIC), La Plata, Buenos Aires, Argentina.

Beautiful Soup. (2025). Beautiful Soup Documentation. https://www.crummy.com/software/BeautifulSoup/bs4/doc 14/04/2025.

Ertam F. (2018). Deep learning-based text classification with Web Scraping methods. International Conference on Artificial Intelligence and Data Processing (IDAP), Malatya, Turkey, 2018, pp. 1-4, doi: 10.1109/IDAP.2018.8620790.

FRIDA (2020). Herramienta de Relevamiento Continuo. NIC.ar - Argentina https://programafrida.net/archivos/project/herramienta-relevamiento-continuo

Fujii, Y. (2019). Multi-thread and multi-process. Medium. https://yuta-san.medium.com/multi-thread-and-multi-process-5559ea5b19ba

Goei S. C. S. (2022). Image pre-processing for Tesseract OCR. Thesis, Universitas Katholik Soegijapranata Semarang.

Google Colab. (2025). FAQ. https://research.google.com/colaboratory/faq.html

PyQuery. (2017). pyquery 2.0.x documentation. https://pyquery.readthedocs.io/en/latest . 2012-2017.

Ruchitaa, R. N. R., Raj, N. S., & Vijayalakshmi, M. (2023). Web scrapping tools and techniques: A brief survey. 2023 4th International Conference on Innovative Trends in Information Technology (ICITIIT), 1-4.

Selvy, P. T., Anitha, M., Varthan, L. R. V., Sethupathi, P., & Adharsh, S. P. (2022). Intelligent web data extraction system for e-commerce. Journal of Algebraic Statistics, 13(3).

SeleniumLibrary. (2025). http://robotframework.org/SeleniumLibrary 14/04/2025.

Tesseract. (2025). Tesseract user manual. https://tesseract-ocr.github.io 14/04/2025.

Visual Studio Code. (2025). Documentation, getting started. https://code.visualstudio.com/docs 14/04/2025.

VMware. (2025). Documentación de VMware vSphere. https://docs.vmware.com/es/VMware-vSphere/index.html 2005-2025

Wong, K. J. (2022). Multithreading and multiprocessing in 10 minutes. Towards Data Science. https://towardsdatascience.com/multithreading-and-multiprocessing-in-10-minutes-20d9b3c6a867

YOLO. (2025). Introducing Ultralytics YOLOv8. https://docs.ultralytics.com 14/04/2025.