--Name of Journal---------
Vol(issue), PP.
.atu.ac.ir
DOI:

ATU PRESS

Original Research / Review / ...
Received:
Accepted:
ISSN:
eISSN:

Using dynamic selectors and artificial intelligence to harden web data retrieval codes

Farnaz Taghizadeh kourayem — Department of Information Technology Management, Central Tehran Branch, Islamic Azad University, Tehran, Iran.

Mohammadreza Kabaranzad Ghadim* — Department of Industrial Management, Central Tehran Branch, Islamic Azad University, Tehran, Iran.

Seyed Abdollah Amin Mousavi — Department of Information Technology Management, Central Tehran Branch, Islamic Azad University, Tehran, Iran.

Abstract

Today, the sustainability and resilience of web data extraction codes are one of the main challenges in software engineering and data mining, particularly when target websites use dynamic structures. selenium, as one of the most widely used libraries for web scraping, is utilized in many industrial and research projects; however, its high sensitivity to changes in web page elements often results in errors, system interruptions, and the need for frequent code modifications. This research presents a novel method that combines “dynamic selectors” with an AI-based decision-making layer to provide a functional, reliable and flexible approach for strengthening selenium-based code. In this study, data was collected from the Digikala website over a two-year period with weekly intervals. The proposed method analyzes the behavior of page elements, automatically replaces the appropriate selectors, and applies multiple intelligent fallback strategies in case of

* Corresponding Author: kabaranzad@yahoo.com

This article is taken from the doctoral thesis of Information Technology Management Department, Central Tehran Branch, Islamic Azad University, Tehran, Iran.

How to cite: Taghizadeh Kourayem, Farnaz, Kabaranzad Ghamid, Mohammadreza, Mousavi, Seyed Abdollah Amin. (2025). Using dynamic selectors and artificial intelligence to harden web data retrieval codes. Quarterly Journal of Knowledge Retrieval and Semantic Systems, Year (Issue), pp. start-end.

failure. This approach increases the stability and reliability of selenium execution against website changes and can serve as an efficient model for web data collection systems operating in dynamic environments. Experimental results showed that when the proposed method was used for automatic and intelligent selection of HTML elements, all data extraction operations were completed without errors and without requiring manual code modifications. However, when the method was not applied, 67 extraction attempts required direct selector corrections by the developer. This performance difference demonstrates that the presented model significantly reduces maintenance costs, development time, and the risk of extraction process failure.

Keywords: Artificial Intelligence, HTML Code Hardening, HTML Structure of Web Pages, Selenium Library, Web Data Retrieval.

--نام مجله---------
دوره ؟، شماره ؟، نام فصل، سال، ص ص
.atu.ac.ir
DOI:

مقاله پژوهشی/مروری/سخن سردبیر
تاریخ دریافت: ۰۰/۰۰/۰۰؛ تاریخ پذیرش: ۰۰/۰۰/۰۰
eISSN: ISSN:

استفاده از انتخابگرهای پویا و هوش مصنوعی برای مقاوم سازی کدهای بازیابی اطلاعات وب

فرناز تقی زاده کورایم — گروه مدیریت فناوری اطلاعات، واحد تهران مرکزی، دانشگاه آزاد اسلامی، تهران، ایران.

محمدرضا کاباران زاد قدیم * — گروه مدیریت صنعتی، واحد تهران مرکزی، دانشگاه آزاد اسلامی، تهران، ایران.

سیدعبداله امین موسوی — گروه مدیریت فناوری اطلاعات، واحد تهران مرکزی، دانشگاه آزاد اسلامی، تهران، ایران.

چکیده

امروزه پایداری و تاب‌آوری کدهای برداشت اطلاعات از وب یکی از چالش‌های اصلی در حوزه مهندسی نرم‌افزار و داده‌کاوی است، به‌ویژه زمانی که وب‌سایت‌های هدف از ساختارهای پویا استفاده می‌کنند. کتابخانه سلنیوم به‌عنوان یکی از پرکاربردترین ابزارهای برداشت اطلاعات وب، در بسیاری از پروژه‌های صنعتی و پژوهشی مورد استفاده قرار می‌گیرد، اما حساسیت بالای آن نسبت به تغییر عناصر صفحه اغلب منجر به بروز خطا، توقف سیستم و نیاز به اصلاحات مکرر کد می‌شود. این پژوهش با ارائه روش نوین مبتنی بر استفاده هم‌زمان از «انتخابگرهای پویا» و یک لایه تصمیم‌گیری مبتنی بر هوش مصنوعی، رویکردی کاربردی، قابل اعتماد و منعطف برای مقاوم‌سازی کدهای سلنیوم ارائه می‌دهد. در این مطالعه، برداشت اطلاعات از وب‌سایت دیجی‌کالا در بازه زمانی دو ساله، با تناوب هفتگی انجام شد. روش پیشنهادی با تحلیل رفتار المان‌های صفحه، به‌طور خودکار انتخابگرهای مناسب را جایگزین و در صورت خطا از چندین استراتژی هوشمند استفاده می‌کند. این رویکرد پایداری و اطمینان اجرای سلنیوم را در برابر تغییرات سایت افزایش داده و می‌تواند الگویی کارآمد برای سامانه‌های جمع‌آوری داده از وب در محیط‌های پویا باشد. نتایج تجربی نشان داد هنگامی که از روش پیشنهادی جهت انتخاب خودکار و هوشمند المان‌های HTML استفاده شد،

* نویسنده مسئول: kabaranzad@yahoo.com

مقاله حاضر برگرفته از رساله دکتری رشته مدیریت فناوری اطلاعات دانشگاه آزاد اسلامی، واحد تهران مرکزی است.

استناد به این مقاله: تقی زاده کورایم، فرناز، کاباران زاد قدیم، محمدرضا، امین موسوی، سیدعبداله. (۲۰۲۵). استفاده از انتخابگرهای پویا و هوش مصنوعی برای مقاوم سازی کدهای بازیابی اطلاعات وب. فصلنامه بازیابی دانش و نظام های معنایی، سال(شماره)، ص آغاز-ص پایان.

تمامی برداشت‌ها بدون خطا و بدون نیاز به تغییر دستی کد انجام شدند. اما هنگام عدم استفاده از این روش، ۶۷ برداشت نیازمند اصلاح مستقیم انتخابگرهای کد توسط توسعه‌دهنده بودند. این اختلاف عملکرد نشان می دهد که استفاده از مدل ارائه‌شده باعث کاهش چشمگیر هزینه نگه‌داری، زمان توسعه و ریسک شکست اجرای عملیات استخراج داده می‌شود.

کلیدواژه‌ها: بازیابی اطلاعات از بستر وب، ساختار HTML صفحات وب، کتابخانه سلنیوم، مقاوم سازی کدهای HTML، هوش مصنوعی.

مقدمه

رشد روزافزون داده‌های موجود در وب و نیاز سازمان‌ها به بازیابی آنها، تحلیل مستمر و بلادرنگ اطلاعات موجود در این داده ها و درنهایت بازیابی دانش و تصمیم‌سازی معنایی براساس آنها، اهمیت فرایند «برداشت داده از وب» یا وب اسکریپینگ[1] را بیش از پیش آشکار کرده است. در این فرایند، خزنده‌های وب به‌عنوان برنامه‌های خودکار، صفحات اینترنتی را پیمایش کرده، داده‌های مورد نیاز را شناسایی و استخراج می‌کنند. این سامانه‌ها امروز بخش جدایی‌ناپذیر از ربات‌های هوشمند بازیابی و ارائه دانش نظیر چت جی پی تی[2] و سیستم‌های مختلفی از جمله سیستم‌های تست نرم‌افزاری تحت وب و گزارش خطا، تحلیل کسب‌وکار، پایش قیمت، رصد رفتار مشتریان، جمع‌آوری اطلاعات رقبا، پایش بازارهای مالی و بسیاری از خدمات مبتنی بر داده محسوب می‌شوند (تقی‌زاده و همکاران، ۲۰۲۴).

در صنایع کاربردی و محیط‌های پژوهشی، موفقیت سامانه‌های برداشت اطلاعات از وب زمانی تضمین می‌شود که این سیستم‌ها بتوانند در طول زمان عملکرد پایدار داشته و در برابر تغییرات ساختار صفحات وب مقاوم باشند. این مسئله به‌دلیل آن اهمیت دارد که امروزه اغلب وب‌سایت‌ها از ساختارهای پویای زبان نشانه گذاری ابرمتنی[3] و تغییرات مداوم در تگ[4]ها و کلاس‌ها استفاده می‌کنند. چنین پویایی موجب می‌شود روش‌های کلاسیک استخراج داده با کوچک‌ترین تغییر در ساختار صفحه دچار خطا شده و نیازمند اصلاح و نگه‌داری مداوم باشند؛ مشکلی که هزینه‌های عملیاتی را افزایش داده و اعتبار سیستم‌های جمع‌آوری داده را تهدید می‌کند (دگکی و همکاران، ۲۰۲۲).

در میان ابزارهای موجود، کتابخانه سلنیوم[5] یکی از پرکاربردترین فناوری‌ها برای برداشت اطلاعات از وب و شبیه‌سازی رفتار انسان در مرورگر است. قابلیت تعامل با عناصر پویا، پشتیبانی از مرورگرهای واقعی و کاربرد گسترده در تست نرم‌افزار باعث شده است سلنیوم انتخاب اصلی بسیاری از پژوهشگران و شرکت‌ها باشد. با این حال نقطه‌ضعف بنیادی سلنیوم وابستگی مطلق آن به انتخابگرهای ثابت است. تغییر در ساختار صفحه – حتی در سطح یک کلاس یا تگ – می‌تواند موجب شکست کامل اجرای برنامه و توقف فرآیند استخراج داده

---

1 Web Scraping
2 ChatGPT
3 HTML
4 Tag
5 Selenium Library

شود. این مسئله به‌ویژه در سناریوهایی که برداشت داده باید مستمر، بلندمدت و بدون مداخله دستی باشد به چالش مهمی تبدیل شده است (نس و همکاران، ۲۰۲۳).

در سال‌های اخیر روش‌های متنوعی برای افزایش پایداری استخراج داده از وب ارائه شده است. برخی از این روش‌ها بر انتخابگرهای سنتی و سلسله‌مراتبی تکیه دارند که در تغییرات جزئی عملکرد مناسبی دارند اما در تغییرات اساسی نیازمند اصلاح دستی هستند. گروهی دیگر از پردازش تصویر برای تشخیص الگوهای بصری استفاده کرده‌اند که انعطاف‌پذیرند اما زمان‌بر بوده و برای المان‌های ریز ناکارآمد هستند. دسته سوم با استفاده از ویژگی‌های داخلی تگ‌ها مانند نام کلاس تلاش به افزایش پایداری کرده‌اند، اما این ویژگی ها به شدت ناپایدار بوده و تغییرات طراحی رابط کاربری موجب توقف استخراج داده توسط آنها می‌شود (ساراسی و همکاران، ۲۰۲۴).

این وضعیت نشان می‌دهد که شکافی مهم در دانش موجود وجود دارد: نبود رویکردهای مقاوم و خودترمیم‌شونده که در زمان تغییر ساختار صفحات وب بتوانند بدون افزایش قابل توجه هزینه پردازش، به‌صورت خودکار، دقیق ترین انتخابگرهای جایگزین معتبر را شناسایی و استفاده کنند. به عبارت دیگر، سیستمی که به صورت خودکار و بدون افزایش قابل توجه در هزینه‌های محاسباتی و زمانی، بتواند داده‌ها و دانش موجود بین آنها را از بستر وب بازیابی کند و یک نظام معنایی جامع و خودترمیم‌شونده را ایجاد کند، وجود ندارد. با توجه به نیاز صنعت، کسب‌وکارها و پژوهشگران به سیستم‌های پایدار، رویکردهایی مبتنی بر هوش مصنوعی و انتخابگرهای پویا می‌توانند تحول قابل‌توجهی در این حوزه ایجاد کنند.

تحقیق حاضر با هدف پر کردن این شکاف، یک رویکرد خودکار مقاوم‌سازی کدهای سلنیوم مبتنی بر تحلیل رفتار عناصر، گزینش پویا و هوشمند انتخابگرهای جایگزین و تصمیم‌گیری مبتنی بر الگوریتم‌های هوش مصنوعی به منظور بازیابی خودکار دانش و ایجاد یک نظام معنایی جامع و خودترمیم‌شونده از داده‌های موجود در بستر وب ارائه می‌دهد. این روش به سیستم اجازه می‌دهد بدون نیاز به تغییر دستی کد در کوتاه ترین زمان ممکن، در هنگام ایجاد خطا، مجموعه‌ای از راهکارهای جایگزین را آزموده و انتخاب معتبرترین گزینه را انجام دهد. چنین قابلیتی زمینه ایجاد نسل جدیدی از خزنده‌های وب پایدار، هوشمند و کم‌هزینه را فراهم کرده و می‌تواند نقش مهمی در توسعه سامانه‌های جمع‌آوری داده سازمانی و پژوهشی ایفا کند.

پیشینه پژوهش

برای توسعه سامانه‌ای مقاوم و هوشمند جهت برداشت داده از وب با استفاده از سلنیوم، لازم است وضعیت موجود دانش و پژوهش‌های مرتبط به‌صورت نظام‌مند بررسی شود. ازاین‌رو در این بخش، پیشینه پژوهش در دو حوزه اصلی شامل «مقاوم‌سازی وب‌اسکریپت[1]ها و انتخابگرهای پویا» و «کاربرد هوش مصنوعی در افزایش پایداری فرایندهای استخراج داده از وب» مرور شده است.

پژوهش‌های دو دهه اخیر نشان می‌دهند که یکی از چالش‌های اساسی در وب‌اسکرپینگ، وابستگی شدید انتخابگر تگ ها به ساختار صفحات وب است که با کوچک‌ترین تغییر در ویژگی تگ‌ها یا کلاس‌ها، باعث شکست کامل یا جزئی اجرای اسکریپت‌ها می‌شود. این مسئله در کاربردهای واقعی نظیر پایش قیمت، تحلیل رقبا، خزنده‌های وب و سیستم‌های هوشمند جمع‌آوری داده، به اتلاف زمان، افزایش هزینه نگه‌داری و نیاز به اصلاح مستمر کد منجر می‌شود. به همین دلیل، محققان رویکردهایی مانند یادگیری ویژگی‌های پایدار صفحات، تولید خودکار انتخابگرها، تحلیل رفتاری المان‌ها و الگوریتم‌های خودترمیمی را پیشنهاد کرده‌اند (کلوگه و استوکو، ۲۰۲۵).

برای دستیابی به تصویری جامع و دقیق، ۱۱ مقاله معتبر و به‌روز از سال‌های اخیر بررسی شده‌اند. این مطالعات با معیارهایی نظیر هدف پژوهش، روش‌شناسی، متریک‌های ارزیابی، نوع داده‌ها و در نهایت شکاف یا محدودیت اعلام‌شده توسط نویسندگان تحلیل شده‌اند. مرور این مطالعات نشان می‌دهد که اگرچه پژوهش‌های خوبی در حوزه بهبود انتخابگرها و مدیریت خطا ارائه شده است، اما بیشتر آن‌ها در محیط‌های آزمایشگاهی یا داده‌های مصنوعی انجام شده‌اند و تعداد اندکی از آن‌ها به اجرای طولانی‌مدت در شرایط واقعی وب، آن هم در مقیاس زمانی چندساله، توجه کرده‌اند.

از سوی دیگر، اکثر پژوهش‌ها تنها به اصلاح انتخابگر پس از وقوع خطا پرداخته‌اند و کمتر به مسئله «یادگیری تدریجی و کاربرد تجارب اجرای قبلی» برای افزایش تاب‌آوری بلندمدت سیستم اشاره کرده‌اند. این شکاف پژوهشی، ضرورت توسعه روشی را مطرح می‌کند که ضمن اجرای واقعی

[1] Script

در محیط وب پویا، قادر باشد انتخابگرهای پایدارتر تولید کرده، تجربه اجرای پیشین را در تصمیم‌گیری لحاظ کند و بدون نیاز به مداخله انسانی پایداری اسکریپت را تضمین نماید.

بررسی شکاف‌های شناسایی‌شده در پژوهش‌های گذشته، ضرورت انجام مطالعه حاضر را تقویت می‌کند؛ زیرا پژوهش جاری با استفاده از هوش مصنوعی و انتخابگرهای پویا، در یک بازه زمانی طولانی دو ساله و کاملاً واقعی، به ارزیابی عملی پایداری و قابلیت خودترمیمی سیستم پرداخته است. در ادامه ابتدا تعریف برخی از مفاهیم استفاده شده در این پژوهش و سپس هدف، روش، متریک، نتایج، داده ها، و چالش و شکاف های تحقیقاتی موجود در پژوهش های پیشین آمده است (جدول ۱).

۱) المان هدف در صفحه سایت: المان هدف، به یک جزء مشخص از ساختار صفحه وب اطلاق می‌شود که از نظر پژوهشگر یا الگوریتم، مورد تحلیل، استخراج، شناسایی یا پردازش قرار می‌گیرد. این المان می‌تواند شامل نام یک محصول، قیمت آن، یک برچسب ، یک بخش متنی، تصویر، لینک یا هر عنصر دیگری باشد که در لایه ارائه صفحه وب حضور داشته و توسط مرورگر قابل نمایش به کاربر نهایی است.

۲) تگ: تگ یا برچسب، کوچک‌ترین واحد نشانه گذاری در یک سند "زبان نشانه گذاری ابرمتنی"[1] است که برای تعریف ماهیت، نقش و رفتار یک جزء در صفحه وب مورد استفاده قرار می‌گیرد. هر تگ معمولاً از یک تگ باز و یک تگ بسته تشکیل شده و می‌تواند متن، المان‌های فرزند یا هر داده دیگری را دربرگیرد. تگ‌ها ساختار سلسله‌مراتبی سند زبان نشانه گذاری ابرمتنی را شکل داده و اساس مدل شیء‌گرای سند را تشکیل می‌دهند. مثلا تگ p یک پاراگراف در صفحه نشان می دهد و به صورت <p>سلام</p> نوشته می شود و می تواند تگ های زیادی را به عنوان فرزند در خود نگه دارد یا خود فرزند تگی دیگر باشد. به عنوان مثال در <div><p>hi</p></div> تگ p فرزند تگ div می باشد.

[1] HTML

۳) ساختار "زبان نشانه گذاری ابرمتنی" صفحه وب: ساختار "زبان نشانه گذاری ابرمتنی"، بیانگر سازمان‌دهی سلسله‌مراتبی عناصر در یک سند وب است که در قالب مجموعه‌ای از تگ‌ها و المان‌های تو در تو تعریف می‌شود. این ساختار تعیین می‌کند که چگونه داده‌ها در صفحه وب توصیف، گروه‌بندی و نمایش داده شوند و مبنای ایجاد درختواره نحوه نمایش در مرورگر است. ساختار "زبان نشانه گذاری ابرمتنی" نه‌تنها بیانگر توالی و ارتباط والد–فرزندی میان عناصر است، بلکه بیان‌کننده نقش معنایی هر بخش از محتوا نیز می‌باشد.

٤) ویژگی‌های تگ: ویژگی‌های یک تگ، مجموعه‌ای از داده‌های کمکی هستند که در درون تگ باز تعریف شده و به آن معنا، رفتار، شناسه یا ویژگی‌های کنترلی اضافه می‌کنند. ویژگی‌هایی مانند نام کلاس، لینک، اندازه و موارد مشابه، از مهم‌ترین انواع ویژگی ها در "زبان نشانه گذاری ابرمتنی" به شمار می‌روند. این ویژگی‌ها می‌توانند توسط مرورگر، موتور جاوااسکریپت و سایر ابزارها برای کنترل نمایش، تعامل، شناسایی و پردازش محتوا مورد استفاده قرار گیرند.

۵) انتخابگر: انتخابگر، الگوی نحوی مورد استفاده برای شناسایی و ارجاع به المان‌های خاص در یک سند "زبان نشانه گذاری ابرمتنی" است. انتخابگرها در زبان‌های نشانه‌گذاری و طراحی و همچنین در ابزارهای پیمایش یک سند "زبان نشانه گذاری ابرمتنی" مانند مسیر اختصاصی[1] مورد استفاده قرار می‌گیرند. یک انتخابگر می‌تواند براساس نام تگ، شناسه، کلاس، ویژگی‌ها، روابط سلسله‌مراتبی والد–فرزند یا ترکیبی از این موارد، یک یا چند المان را در ساختار سند انتخاب و هدف‌گذاری کند و یا آن را بازگرداند.

٦) وب اسکرپینگ: یک نرم‌افزار خودکار است که با پیمایش سیستماتیک صفحات وب، داده‌ها و محتوای موجود در اینترنت را بازیابی و ذخیره‌سازی می‌کند. خزنده‌ها معمولاً از یک یا چند نقطه شروع آغاز کرده و با تحلیل ساختارهای موجود در صفحات، به صورت بازگشتی اطلاعات صفحات را کشف و برداشت می‌کنند. این ابزارها نقش بنیادی در موتورهای جستجو، پایگاه‌های داده محتوای وب، سیستم‌های پایش تغییرات وب و سامانه‌های استخراج داده ایفا می

[1] XPath

کنند. استفاده از خزنده‌های وب امکان جمع آوری سازمان‌یافته داده از منابع گسترده اینترنت را فراهم کرده و مبنای بسیاری از فرآیندهای تحلیل داده و بازیابی اطلاعات در پژوهش‌های علمی و کاربردهای صنعتی محسوب می‌شود.

جدول ۱. هدف، روش، متریک، نتایج، داده ها، و چالش و شکاف های تحقیقاتی موجود در پژوهش های پیشین

ردیف	نویسندگان و سال	هدف	روش	متریک ها و نتایج	داده ها	چالش/شکاف
۱	Kirinuki et al., (2019)	توصیه انتخابگر جایگزین با استفاده از سرنخ‌های متنوع (ویژگی های تگ، متن، تصویر، موقعیت) برای تعمیر اسکریپت‌های شکست‌خورده	ترکیب سرنخ‌های ساختاری و تصویری و وزن‌دهی آن‌ها برای رتبه‌بندی کاندیدها و توصیه انتخابگرهای محتمل	در برنامه های کاربردی تحت وب مختلف دقت بین ۷۷ تا ۹۳ درصد گزارش شده است	چند پروژه متن-باز و مقایسه بین نسخه‌ها	خوب برای تغییرات ساختاری نه چندان شدید. در تغییرات ریشه‌ای ظاهر/سیاستگذاری های جدید یا در صفحات کاملاً بازطراحی شده، محدودیت دارد
۲	Degaki et al., (2022)	بررسی اینکه	ابتدا با بینایی ماشین المان‌های	precision/ recall visual detection	نمونه‌های صنعتی و	هماهنگ‌سازی بین فضای

		ترکیب سرنخ‌های بصری (هوش مصنوعی و پردازش تصویر) و تگ های صفحه تا چه حد می‌تواند پایداری را بالا ببرد.	بصری را تشخیص می‌دهند و سپس با تگ های صفحه اعتبارسنجی می‌کنند	و robustness در تشخیص تصویر؛ گزارش بهبود نسبت به هر روش تکی (فقط بینایی ماشین و فقط تگ در کتابخانه سلنیوم)	مطالعات موردی از وب سایت ها	تصویر (پیکسل) و فضای تگ ها (عنصر ساختاری) پیچیده است؛ همچنین هزینه های محاسباتی و تاخیر برای کاربردهای بلادرنگ یک مشکل عملی است.
۳	Khaliq et al., (2023)	کاربرد یادگیری عمیق برای کمک به شناسایی المان‌ها از تصویر/وی ژگی های تگ	چارچوب مبتنی بر شبکه‌های کانولوشنال/encoder برای شناسایی المان از اسکرین‌شات و تولید/تکمیل اسکریپت‌های سلنیوم	دقت شناسایی المان، repair-rate، coverage تست؛ گزارش بهبود نسبت به روش‌های سنتی	مجموعه صفحات و تصاویر محصول/ سبد خرید از اپلیکیشن‌ه ای تجارت الکترونی ک	نیاز به دیتاست‌های برچسب‌خور ده زیاد برای آموزش؛ مشکلات در صفحات با چینش‌های متنوع و واکنش‌گرا؛ ادغام تصویر و ویژگی های تگ برای تعادل،

						دقت و پایداری هنوز حل نشده است
۴	Nass et al., (2023)	بازیابی المان هدف پس از تغییرات صفحه با محاسبه امتیاز تشابه بین چند ویژگی تگ. نام مدل طراحی شده Similo می باشد	ویژگی های یک تگ مثل نام، id، کلاس، متن، موقعیت، اندازه، و ... را با پارامترهای همان عنصر در نسخه قدیمی مقایسه می کند و به آن امتیاز می دهد و بهترین کاندید را انتخاب می کند	۷۲ شکست از ۵۹۸ مورد بازیابی اطلاعات	داده های قابل بازیابی ۴۰ وب‌سایت پرکاربرد و مقایسه بین نسخه‌های بازیابی مختلف در زمان های متفاوت	امکان دارد چند تگ نام، کلاس یا ویژگی های کاملا یکسانی داشته باشند یا در طول زمان ویژگی های تگ به صورت کامل در طول زمان تغییر کند.
۵	تقی زاده و همکاران (۲۰۲۴)	شناسایی موقعیت هایی که نه خطا بازمی گردد، نه کد شکست می خورد و نه کد مربوط به	استفاده از اسکریت های نوشته شده به زبان پایتون	مقایسه تعداد خطا ها هنگام استفاده از اسکریپت و عدم استفاده از اسکریپت. هنگام استفاده از اسکریپت هیچ خطایی اعلام نشده و	مطالعه سایت دیجی کالا	فقط مقادیر دارای خطا و محل خطا را اعلام می کند. همچنان نیاز به دستکاری دستی کدها وجود دارد

		تگ اجرا می شود و همچنین مدیریت زمان در اسکریپت های سلنیوم		اطلاعات به صورت دقیق در جای خود نشسته اند		
۶	Danesh var & Wang, (2024)	ارزیابی مدل‌های مدرن object-detecti on در تشخیص عناصر با استفاده از اسکرین‌ش ات	استفاده از کتابخانه تشخیص تصاویر یولو روی تصاویر سایت	map^ precision / recall	دیتاستی از هزاران اسکرین شات برچسب‌خ ورده از برنامه های کاربردی تحت وب	عملکرد خوب در تشخیص اما مشکلِ همپوشانی، آیتم‌های ریز و چگالی بالای عناصر؛ ترکیب با تگ های صفحه برای کاهش false positive s توصیه شده است. سرعت بسیار پایین دارد چرا که برای یک تگ کوچک و

						مشخص و ثابت هم از هوش مصنوعی استفاده می کند که احتمال خطا وجود دارد.
۷	Nass et al., (2024)	بازیابی المان هدف پس از تغییرات صفحه با محاسبه امتیازِ تشابه بین چند ویژگی و تگ و استفاده از یک مدل زبانی بزرگ GPT-4 برای بازیابی دقیقتر	ویژگی های یک تگ مثل نام، id، کلاس، متن، موقعیت، اندازه، و ... را با پارامترهای همان عنصر در نسخه قدیمی مقایسه می کند و به آن امتیاز می دهد و بهترین کاندیدها را انتخاب می کند. سپس یک مدل زبان بزرگ استفاده می‌شود تا از میان این کاندیدها، بهترین گزینه را انتخاب کند	۴۰ شکست از ۸۰۴ مورد بازیابی اطلاعات	داده های قابل بازیابی ۴۸ وب‌سایتِ پرکاربرد و مقایسه بینِ نسخه‌های بازیابی مختلف در زمان های متفاوت	استفاده از یک مدل زبانی بزرگ کمک‌کننده است ولی هزینه ها بالا و پاسخ‌ها گاهی غیردقیق و نیاز به کنترل اعتبارسنجی انسانی دارد؛ همچنین مسائل حریم‌خصوصی هم مطرحند
۸	Saarathy et al., (2024)	جمع‌بندی وضعیت	مرور نظام‌مند مقالات و	جمع‌بندی گزارش‌های	جمیع نتایج	توافق در متریک‌ها و

		مقالات و مطالعات موجود در زمینه پایداری کدهای کتابخانه سلنیوم	پروژه‌های صنعتی، طبقه‌بندی روش‌ها و تحلیل مزایا/معایب	نرخ موفقیت، میزان کاهش اصلاح کدها به صورت دستی، هزینه محاسباتی، تعداد شکست های برطرف شده.	مطالعات تجربی، مخازن کدهای متن-باز	مقادیر اندازه گیری کم است؛ فقدان استاندارد آزمایشی و قابل‌تکرار برای ارزیابی جامع دیده می‌شود.
۹	Coppola et al., (2025)	بررسی الگوریتم‌های امتیازدهی برای انتخاب مناسب‌ترین کاندید از لیست کاندیدها (مثل Similo)	استخراج ویژگی های تگ و مقادیر تگ و آموزش مدل‌های رتبه‌بندی برای بهینه‌کردن k انتخاب برتر.	دقت پیشنهاد	برنامه های توسعه یافت برای similo	یادگیری بر اساس رتبه نیاز به داده های برچسب‌خور ده زیاد دارد و مناسب سازی کد برای همه سایت‌ها هنوز چالش است؛ همچنین تاخیر در محیط عملیاتی مسئله‌ساز است
۱۰	Healenium, (2025)	ایجاد لایه میانی که	اسکریپت های سلنیوم اجرا می	نرخ شکست از ۳۰ درصد به ۵	نمونه‌های صنعتی،	عملکرد عملی دارد

		شکست اسکریپ های سلنیوم را کشف و به طور خودکار تلاش به ترمیم/جای گزینی آنها کند	شود و اگر خطایی وجود نداشت، کدهای مربوط به المان هدف در دیتابیس ذخیره می شوند تا اگر بازیابی های بعدی با خطا مواجه شد از آنها استفاده گردد	درصد کاهش یافته	دموها و مخازن نمونه شرکت؛ پیاده‌ساز ی عملی در پروژه‌های شرکت	اما نیاز به تنظیم دستی، مانیتورینگ، دارد و اغلب روی سناریوهای خاص تنظیم می‌شود؛ نیاز به ارزیابی گسترده‌تر دارد
۱۱	Kluge & Stocco, (2025)	بازتولید و مقایسه الگوریتم‌ها ی موجود (ازجمله Similo و ...) در سایت های بزرگ‌تر و تکرار الگوریتم ها	پیاده سازی و ارزیابی الگوریتم های موجود	...	سایت های بزرگ با هزاران المان هدف	نشان می‌دهد هیچ راه‌حل واحدی برای همه موارد وجود ندارد و نیاز به روش‌های ترکیبی و معیارهای اعتبارسنجی خودکار میباشد

در ادامه به ارتباط بین هر یک از پژوهش های موجود در جدول ۱ و مساله مورد بررسی در این مقاله و شکافی که مقاله حاضر در تلاش برای برطرف نمودن آن است، پرداخته شده. مقاله‌های کیرینوکی و همکاران (۲۰۱۹)، نس و همکاران (۲۰۲۳) و کوپلا و همکاران (۲۰۲۵) با استفاده از سرنخ‌های مختلف، انتخابگرهای جایگزین را برای مقام‌سازی کدهای

برداشت اطلاعات از بستر وب پیشنهاد می دهند. با این حال، در این روش‌ها خطای پیشنهادهای ارائه شده، نسبتا بالا می‌باشد و برای صفحاتی با تغییرات ساختاری شدید و دارای تگ‌های با ویژگی‌های یکسان، مناسب نمی باشد. همچنین باوجود اینکه استفاده از یک مدل زبانی بزرگ توسط نس و همکاران (۲۰۲۴) به منظور بهبود این روش‌ها ارائه شده که گاهی خطا را کاهش می دهد، اما هم مسائل حریم خصوصی ایجاد می کند و هم هزینه محاسباتی را افزایش می‌دهد. در مقابل، مقاله حاضر در تلاش است تا روشی ارائه دهد که نه تنها خطا را کاهش می‌دهد، بلکه برای صفحاتی با تغییرات ساختاری شدید نیز مناسب باشد و مسائل حریم خصوصی و هزینه محاسباتی نیز ایجاد نکند. مقاله‌های دگکی و همکاران (۲۰۲۲)، خلیق و همکاران (۲۰۲۳) و دانشور و ونگ (۲۰۲۴) استفاده از بینایی ماشین را برای پیشبینی المان‌های موجود در صفحات وب پیشنهاد داده‌اند. با این حال، پردازش تصویر و بینایی ماشین، هنگامی که اطلاعات موجود در صفحه بسیار ریز باشند، دقت بسیار پایینی در تشخیص و پیشبینی دارند و همچنین هزینه محاسباتی و زمانی این روش‌ها بسیار بالا می‌باشد. در مقابل مقاله حاضر، روشی ارائه داده که هم بتواند مقدار دقیق اطلاعات را بدون هیچ خطایی بازگرداند و هم هزینه محاسباتی و زمانی آن حداقل باشد. مقاله تقی‌زاده و همکاران (۲۰۲۴) به ارائه سازوکاری برای مدیریت زمان و افزایش دقت اطلاعات برداشت شده از بستر وب با کمک کتابخانه سلنیوم پرداخته است. با این حال، در این مقاله فقط محل خطا مشخص می گردد و مقادیر دارای خطا با مقدار تهی بازمی‌گردند، همچنین در این مقاله، راه حلی برای مقاوم‌سازی کدها ارائه نشده است. در مقابل، مقاله حاضر ضمن استفاده از روش پیشنهاد شده توسط تقی‌زاده و همکاران (۲۰۲۴) به عنوان کدهای پایه برای برداشت اطلاعات از بستر وب، راه حلی برای مقاوم‌سازی کدها و برداشت خودکار مقادیر مورد نظر از بستر وب با بالاترین دقت ممکن ارائه می‌دهد. ساراسی و همکاران (۲۰۲۴) به مرور نظام‌مند مقالات موجود در زمینه پایداری کدهای کتابخانه سلنیوم پرداخته اند، اما مدلی برای مقاوم‌سازی کدهای برداشت اطلاعات از بستر وب ارائه ننموده اند. در مقابل، مقاله حاضر

ضمن بررسی مقالات و مدل‌های موجود، مدلی را برای مقاوم‌سازی کدهای کتابخانه سلنیوم ارائه کرده است. هیلنیوم (۲۰۲۵)، مدلی ارائه داده که در آن ابتدا اسکریپت‌های سلنیوم اجرا می شوند و اگر خطایی وجود نداشت، کدهای مربوط به المان هدف در دیتابیس ذخیره می شوند تا اگر بازیابی های بعدی با خطا مواجه شد از آنها استفاده گردد. با این حال، اگر اسکریپت‌ها با خطا مواجه شوند، نیاز به تنظیمات دستی می‌باشد. درمقابل مقاله حاضر در تلاش است تا مدلی ارائه دهد که درصورت بروز خطا، اسکریپت جایگزین با کمک الگوریتم‌های هوش‌مصنوعی در کمترین زمان، شناسایی شوند. کلوگه و استوکو (۲۰۲۵) به مقایسه و بازتولید الگوریتم‌های موجود پرداخته و عنوان نموده‌اند که برای مقاوم‌سازی کدهای برداشت اطلاعات از بستر وب، هیچ راه‌حل واحدی وجود ندارد و نیاز به روش‌های ترکیبی و معیارهای اعتبارسنجی خودکار میباشد. با این حال، این مقاله هیچ مدلی ارائه نداده و تنها به ترکیب، مقایسه و بررسی مدل‌های موجود پرداخته است. در مقابل مقاله حاضر، مدلی ارائه نموده که بتواند کاستی‌های مدل‌های پیشین را کاهش دهد.

روش شناسی پژوهش

نوع و رویکرد پژوهش: این پژوهش از نظر هدف، کاربردی و از نظر ماهیت و روش اجرا، ترکیبی از روش‌های تحلیل داده‌های وب، توسعه الگوریتم‌های مقاوم‌سازی، و تحلیل کمی-تجربی است. با توجه به ماهیت داده‌ها، مطالعه حاضر مبتنی بر جمع‌آوری داده‌های واقعی از سایت دیجی‌کالا و استخراج اطلاعات محصولات آن از طریق وب‌اسکرپینگ انجام شده است. سایت دیجی‌کالا به این دلیل انتخاب گردید که شامل داده‌ها و مشتریان واقعی است و با توجه به مشتریان و محصولات زیاد، ترافیک و سرعت آن برای یک مطالعه پژوهشی واقعی می‌باشد. همچنین سایت دیجی‌کالا، یکی از سایت‌هایی می‌باشد که برنامه آن همواره در حال به روزرسانی و بهبود است، درنتیجه ساختار صفحات وب آن در طول زمان به دلایل مختلف تغییر می‌کند، درنتیجه می‌تواند برای مقاله حاضر، نمونه مطالعاتی

مناسبی باشد. همچنین رویکرد پژوهش، استقرایی–تحلیلی بوده و تلاش دارد از داده‌های مشاهده‌شده، الگوها و شاخص‌های عملکردی در برداشت داده‌ها و پایداری اسکریپت‌ها استخراج و تبیین کند.

جامعه آماری و محدوده زمانی پژوهش: جامعه آماری شامل تمامی محصولات موجود در وب‌سایت دیجی‌کالا (دسته‌بندی داروهای مکمل) است که اطلاعات آن‌ها در بازه زمانی مهر ۱۴۰۲ تا مهر ۱۴۰۴ در دسترس بوده است. داده‌ها از نظر حجم و تنوع، توانایی ارزیابی پایداری اسکریپت‌های سلنیوم و عملکرد الگوریتم‌های انتخابگر پویا را دارند. برداشت داده‌ها به صورت هفته‌ای یک بار و در مجموع ۱۰۰ برداشت کامل انجام شده است.

روایی محتوایی: شاخص‌های اولیه شامل اطلاعات ساختاری و محتوایی صفحات محصول (عنوان، قیمت، ویژگی‌ها، کد کالا و سایر داده‌های متنی) استخراج شدند و با تحلیل مفهومی و دسته‌بندی منطقی بررسی گردیدند تا تنها متغیرهای مرتبط با پایداری اسکریپت‌ها و موفقیت برداشت داده در مدل باقی بمانند.

روایی سازه: ساختار نهایی شاخص‌ها با استفاده از الگوریتم‌های هوش مصنوعی و یادگیری ماشین شامل سری‌های و تحلیل ویژگی‌ها، اعتبارسنجی شد و با چارچوب نظری سیستم‌های خودترمیمی و مقاوم‌سازی اسکریپت‌های سلنیوم همراستا است. از جمله الگوریتم‌های سری زمانی استفاده شده شامل الگوریتم‌های LSTM، ANN، SVR و XGBoost بودند که برای ارزیابی دقت آنها از متریک ضریب تعیین R^2 استفاده شد. ورودی الگوریتم از جدول تگ‌های HTML با ستون‌های تاریخ برداشت اطلاعات، مسیر اختصاصی المان و ستون "استفاده شد" می‌باشد و خروجی الگوریتم پیش‌بینی مقدار ستون "استفاده شد" برای سطرهایی است که مقادیر ستون تاریخ برداشت اطلاعات آنها با تاریخ روز یکسان باشد. در پایان هر برداشت نیز مقدار ستون "استفاده شد" در جدول تگ‌های

HTML به روز می‌گردد و هر بار تقریبا ۲۰ هزار سطر به جدول اضافه می‌شود تا برای برداشت‌های بعدی و افزایش دقت الگوریتم‌های سری زمانی، مورد استفاده قرار گیرد.

پایایی داده‌ها: فرایند استخراج داده‌ها توسط اسکریپت‌های پایتون و کتابخانه سلنیوم چندین بار اجرا شد تا از ثبات و دقت نتایج اطمینان حاصل گردد و خطاهای احتمالی وبجاسکرپینگ به حداقل برسد.

پایایی تحلیل‌ها: مدل‌ها و الگوریتم‌های هوش مصنوعی برای تولید و انتخاب انتخابگرهای پویا با استفاده از معیارهای کمی شامل درصد موفقیت، نرخ بازیابی خودکار خطا، و تعداد برداشت موفق ارزیابی شدند تا نتایج در اجرای مجدد از ثبات کافی برخوردار باشند.

شبه کد روش پیشنهادی: در ادامه مراحل مختلف روش پیشنهادی به منظور ارائه نمای کلی از روش پیشنهادی ارائه شده است:

۱. ایجاد جداول مورد نیاز:
   i. جدول اطلاعات محصول
   ii. جدول تگ‌های HTML
   iii. جدول شمارش تگ‌ها، جهت تشخیص انتخابگرهای محتمل
۲. مراحل اجرای اسکریپت‌های پایتون
   i. اسکریپت برداشت اطلاعات از سایت دیجی‌کالا
   ii. در صورت عدم بروز خطا و برداشت کامل اطلاعات، شامل اطلاعات محصولات و تگ‌ها آنها در جداول مربوطه ذخیره می شوند و در صورت بروز خطا، اسکریپت‌های مراحل بعد اجرا می گردند.
   iii. استخراج تمام تگ‌ها و ساختارهای HTML محصولات و پاکسازی آنها
   iv. استخراج Xpathها

۷. اجرای اسکریپت‌های مربوط به هوش‌مصنوعی به منظور پیشبینی و تعیین Xpathهای صحیح و پس از آن اجرای مجدد اسکریپت برداشت اطلاعات از بستر وب (به عبارت دیگر، رفتن به مرحله i از اجرای اسکریپت‌های پایتون)

جمع‌بندی: روش‌شناسی حاضر به گونه‌ای طراحی شده است که پایداری و قابلیت اعتماد اسکریپت‌های سلنیوم در برداشت داده‌های واقعی از وب بررسی و تقویت شود و امکان تحلیل دقیق اثر الگوریتم‌های انتخابگر پویا و هوش مصنوعی بر موفقیت استخراج داده‌ها فراهم گردد.

روش[1]

این پژوهش با هدف افزایش پایداری و دقت کدهای برداشت اطلاعات از وب‌سایت دیجی کالا، چارچوبی ترکیبی شامل پایگاه داده SQL، اسکریپت‌های سلنیوم و پایتون و الگوریتم‌های هوش مصنوعی طراحی کرده که هر کدام از اینها به تفصیل در ادامه آمده اند. این رویکرد یک سامانه خودکار و مقاوم برای برداشت داده از وب ارائه می‌دهد که تغییرات ساختار صفحات را مدیریت کرده و نیاز به اصلاح دستی را کاهش می‌دهد.

۱. طراحی پایگاه داده

در گام نخست به‌منظور ذخیره‌سازی اطلاعات محصولات سایت دیجی‌کالا و همچنین داده‌های مربوط به تگ‌های HTML و انتخابگرها، چند جدول در پایگاه داده sql ایجاد شد. این جداول شامل:

۱-۱. جدول اطلاعات محصولات: این جدول مشخصاتی از قبیل " نام محصول، قیمت اصلی، درصد تخفیف، قیمت پس از تخفیف، امتیاز کاربران، لینک تصویر، برچسب محصول و تاریخ روز" را برای هر محصول در بازه‌های زمانی هفتگی ذخیره می کند.

1. method

۱-۲. جدول تگ‌های HTML دیجی کالا: این جدول تمام تگ‌های موجود در صفحه HTML ۲۰ محصول اول دیجی کالا را به صورت ساختارمند و والد–فرزندی ذخیره می‌کند. هر سطر شامل "نام تگ، xpath یا مسیر اختصاصی تگ (به صورت شماره‌گذاری شده در صورت تکرار چندباره مثل [div[3]/a[2])، تاریخ روز یا همان تاریخ برداشت اطلاعات و ستون "استفاده شد" که با کلمه های "بله" و "خیر" پرمی شود، است. نحوه برداشت کل تگ ها و محاسبه مسیر تگ ها یا همان xpath آنها در بخش اسکریپت های سلنیوم و پایتون به تفصیل و با ارائه کد نوشته شده است. درصورتی که اسکریپت سلنیوم برداشت اطلاعات از وب بدون خطا و با اطلاعات کامل، اجرا شود؛ پس از اتمام اجرا فقط یک xpath برای هر یک از مقادیر نام محصول، قیمت اصلی، درصد تخفیف، قیمت پس از تخفیف، امتیاز کاربران، لینک تصویر، برچسب محصول به جدول افزوده شده و ستون "استفاده شد" مقدار بله را می گیرد. اما درصورتی که اسکریپت سلنیوم با خطا و مشکل روبرو شوند، تمام تگ های html دیجی کالا به جدول اضافه شده و ستون "استفاده شد" با مقدار پیشفرض تهی پرمی شود و پس اجرای تمام اسکریپت های بعدی و مشخص شدن مقدار درست xpath برای هر کدام از المان های صفحه نظیر نام محصول و ...، این ستون با مقدار بله که به معنی xpath موجود و درست برای المان های مورد نظر نظیر نام محصول و ... پر می شود و بقیه سطرها مقدار خیر را می گیرند.

۱-۳. جدول شمارش تگ‌ها جهت تشخیص انتخابگرهای محتمل: بعد از استخراج کامل HTML و تکمیل جدول تگ‌های HTML دیجی کالا ، این جدول ساخته می شود. این جدول شامل ستون های "مسیر اختصاصی تگ بدون تکرار، فراوانی ستون مسیر اختصاصی تگ از جدول تگ‌های HTML دیجی کالا برای هر مسیر اختصاصی و ستون تاریخ ثبت مسیر اختصاصی" می باشد. برای تکمیل این جدول، تگ‌های مشابه یا همان مسیرهای اختصاصی مشابه، شمارش شده است. به این دلیل این جدول تهیه می شود که اکثر

سایت های فروشگاهی تعداد محدودی محصول را در یک صفحه نشان می دهند که این عدد ثابت می باشد. به عنوان مثال، سایت دیجی کالا در هر صفحه از صفحات محصول خود تنها ۲۰ محصول را نشان می دهد، مثلا در یک صفحه ما ۲۰ عنوان کالا، ۲۰ قیمت، ۲۰ تصویر و ... داریم. با توجه به ثابت بودن این عدد، وجود آن در این جدول به الگوریتم های هوش مصنوعی و اسکریپت های برداشت اطلاعات می تواند کمک قابل توجهی کند. درنتیجه انتخابگرهایی تعداد ۲۰ یا کمتر دارند، احتمال بیشتری دارد که بتوانند اطلاعات مورد نظر ما را برداشت کنند. یک نکته مهم اینکه، این جدول در واقع یک view ساخته شده در دیتابیس sql می باشد که از روی جدول تگ‌های HTML دیجی کالا ساخته شده است و به دلیل راحتی در جمله بندی و بیان صریح تر از کلمه جدول استفاده شده است. ساخت view باعث افزایش سرعت بسیار خوبی در اجرای کد شده و پیچیدگی های ساختاری برنامه را به شدت کاهش می دهد و به واسطه آن نیازی به اسکریپت نویسی و اتصال فرانت اند و بک اند و اتصال اسکریپت به بک اند برای ثبت اطلاعات در دیتابیس وجود ندارد و جدول از طریق view با کسری از ثانیه تولید شده و به راحتی می توان از آن در اسکریپت های مختلف استفاده نمود.

۲. اسکریپت های پایتون و سلنیوم

۲-۱. اسکریپت برداشت اطلاعات محصول از سایت دیجی کالا: این بخش براساس کدهای موجود در پژوهش انجام شده توسط تقی زاده و همکاران (۲۰۲۴) نوشته شد تا با کمک آن بتوان مقادیر بدون خطا را دقیق در جای مناسب خود دریافت نمود و همچنین محل خطا را نیز شناسایی نمود. این کار باعث سرعت بخشیدن به برنامه می شود، چرا که مقادیر فقط دارای خطا از طریق اسکریپت های بعدی شناسایی و اصلاح می شوند. البته کدهای نوشته شده توسط تقی زاده و همکاران (۲۰۲۴) برای مدیریت زمان و نامتقارن بودن و تاخیر استفاده شده اند و در این پژوهش برای خطای ایجاد شده در تغییرات ساختار

html صفحه استفاده نموده ایم. این تفاوت تنها در ارائه خروجی ها قابل نمایش است، چرا که در خروجی موضوع مطرح شده توسط تقی زاده و همکاران (۲۰۲۴) تعدادی از اطلاعات موجود در یک ستون مثلا قیمت خالی نمایش داده می شود ولی در خروجی این پژوهش با توجه به اینکه ساختار html صفحه تغییر یافته است، کل ستون مربوط به قیمت خالی نمایش داده می شود که این ستون توسط اسکریپت های بعدی اصلاح و پرخواهد شد.

۲-۲. استخراج تمام تگ ها و ساختار HTML صفحه اول محصولات دیجی کالا (یا به عبارت دیگر ۲۰ محصول اول): با توجه به تحقیقات گذشته و مطالعات انجام شده و همچنین اصول طراحی وبسایت با استفاده از جاوا اسکریپت، ری اکت و سایر زبان های برنامه نویسی، ساختار تمام محصولات در یک سایت فروشگاهی مشابه هم هستند ، اما ممکن است برخی محصولات تگ های بیشتر یا کمتری داشته باشند. به عبارت دیگر، ساختار والد فرزندی قیمت محصول در تمام محصولات یکسان می باشد و تنها تفاوت در یکی از تگ هاست که آن هم مربوط به والد اصلی هر محصول است که هر محصول از طریق آن با محصول دیگر متفاوت می شود. به همین دلیل در این بخش، تنها تگ ها و ساختار HTML فقط صفحه اول محصولات برداشت می شود. در ادامه اسکریپت مربوط به این بخش آمده است (شکل ۱). این اسکریپت پس از جمع آوری تمام تگ ها و ساختارهای HTML صفحه اول آن را در متغیری به نام html_content می ریزد که با استفاده از این متغیر و دستورات regex که در بخش بعد آمده است، می توان XPATH یا همان مسیر اختصاصی المان های صفحه یا همان اشیاء صفحه را استخراج نمود (شکل ۱).

شکل ۱. اسکریپت مربوط به استخراج تمام تگ ها و ساختار HTML صفحه اول محصولات دیجی کالا (یا به عبارت دیگر ۲۰ محصول اول)

```
from selenium import webdriver
from webdriver_manager.firefox import GeckoDriverManager
from selenium.webdriver.firefox.service import Service
from selenium.webdriver.firefox.options import Options
from bs4 import BeautifulSoup
import time
URL = "https://www.digikala.com/search/category-nutritional-supplements/?page=1"
def fetch_full_html_with_divs(url):
    options = Options()
    options.add_argument("--headless")
    options.add_argument("--disable-gpu")
    service = Service(GeckoDriverManager().install())
    driver = webdriver.Firefox(service=service, options=options)
    try:
        driver.get(url)
        time.sleep(5)
        last_height = driver.execute_script("return document.body.scrollHeight")
        while True:
            driver.execute_script("window.scrollTo(0, document.body.scrollHeight);")
            time.sleep(5)
            new_height = driver.execute_script("return document.body.scrollHeight")
            if new_height == last_height:
                break
            last_height = new_height
        html = driver.page_source
        soup = BeautifulSoup(html, "html.parser")
        div_count = len(soup.find_all("div"))
        print(f"تعداد کل divها در صفحه: {div_count}")
        return html
    finally:
        driver.quit()
html_content = fetch_full_html_with_divs(URL)
```

۲-۳. اسکریپت استخراج Xpath تمامی تگ‌ها: در این مرحله تگ‌ها به ترتیب والد–فرزند با یک الگوریتم پیمایشی یا همان رجکس استخراج می‌شوند. در ادامه کد مربوط به این اطلاعات و آمده است. این کد لیستی از مسیرهای اختصاصی هر تگ به نام paths را باز می گرداند. در این روش از متد رجکس که با re مشخص شده و همچنین

حلقه های while و for تو در تو استفاده شده است و ساختار html را به مسیر اختصاصی هر تگ براساس والد و فرزندی آن تبدیل می کند. البته قبل از اجرای این اسکریپت، اسکریپت های مربوط به پاکسازی ساختار انجام شد و بخش هایی مانند هدر و فوتر و برخی بخش های ثابت از ساختار html حذف شدند تا متغیر html_content حجم سبک تری از داده را در خود نگهداری نماید(شکل ۲).

شکل ۲. اسکریپت استخراج Xpath تمامی تگ‌ها

```
import re
from collections import defaultdict
tag_pattern = re.compile(r'<(/?)(\w+)[^>]*>')

paths = []  # خروجی نهایی مسیرها
stack = []  # مسیر فعلی
child_counters_stack = []  # شمارنده فرزندان هر سطح

pos = 0
while pos < len(html_content):
    match = tag_pattern.search(html_content, pos)
    if not match:
        break
    is_closing, tag_name = match.groups()
    pos = match.end()

    if not is_closing:  # تگ باز
        stack.append(tag_name)
        child_counters_stack.append(defaultdict(int))
    else:  # تگ بسته
        if stack:
            # شمارنده فرزند مشابه در والد
            parent_counter = child_counters_stack[-1]
            parent_counter[tag_name] += 1
            index = '' if parent_counter[tag_name] == 1 else f'[{parent_counter[tag_name]}]'

            # ساخت مسیر کامل از والد تا فرزند
            full_path_parts = []
            for i, t in enumerate(stack):
                if i < len(stack)-1:
                    # بررسی تعداد فرزندان مشابه در سطح والد
                    cnt = child_counters_stack[i][t]
                    part = t if cnt <= 1 else f"{t}[{cnt}]"
                else:
                    # خود تگ بسته شده
                    part = t + index
                full_path_parts.append(part)
            full_path = '/'.join(full_path_parts)
            paths.append(full_path)
            stack.pop()
            child_counters_stack.pop()
```

۲-٤. ذخیره سازی و تکمیل اطلاعات: اسکریپت بخش های ۳-۲ و ۲-۲، تنها زمانی اجرا می شوند که اسکریپت بخش ۱-۲ مقادیری خالی از محصولات صفحه را بازگرداند یا دچار خطا شود. پس از اجرایی کامل اسکریپت بخش های ۳-۲ و ۲-۲، جدولِ تگ‌های HTML دیجی کالا براساس لیست paths پر می شود و سپس جدولِ شمارش تگ‌ها جهت تشخیص انتخابگرهای محتمل بر اساس جدولِ تگ‌های HTML دیجی کالا تکمیل می گردد.

۳. اسکریپت های مربوط به یادگیری ماشینی و هوش مصنوعی برای انتخاب انتخابگرها: برای انتخاب انتخابگر یا همان xpath مناسب، ابتدا مقادیر مربوط به ستون مسیر اختصاصی از جدولِ تگ‌های HTML دیجی کالا با فیلتر تاریخ روز از پایگاه داده برداشت شده و مقادیر مربوط به ستون "استفاده شد" آن بر اساس داده های تاریخی همان جدول با کمک الگوریتم های سری زمانی پیشبینی می گردد و xpath هایی که مقدار "استفاده شد" در آنها بله باشد، جایگزین xpath موجود در اسکریپت سلنیوم می شوند و در نهایت کد سلنیوم مجدد اجرا می گردد و در صورتی که کد بدون خطا اجرا گردید، مقادیر "استفاده شد" در جدولِ تگ‌های HTML دیجی کالا اصلاح می گردد و در صورت بروز خطا تمام مقادیری که ستون "استفاده شد" آنها بله بود، حذف شده و مجدد الگوریتم های پیشبینی سری های زمانی فراخوانی می شوند. همچنین در صورتی که برخی المان ها مثلا فقط المان "درصد تخفیف" خطا داشت، xpath هایی که استفاده شده اند حذف می شوند و xpath هایی که خطا نداشتند، مقادیر "استفاده شد" در جدولِ تگ‌های HTML دیجی کالا آنها بله می شود و الگوریتم تا زمان یافتن xpath درست ادامه پیدا می کند. اگر برنامه نتواند با کمک سری های زمانی xpath درست را بیابد، آنگاه تگ های موجود در جدول تگ ها را پیمایش می‌کند. در ادامه جدول مربوط به الگوریتم های هوش مصنوعی استفاده شده و معیارهای سنجش آنها آورده شده است. شایان ذکر است که در شش ماه اول

از این پژوهش، اصلاح اسکریپت سلنیوم به صورت دستی و با نظارت مستقیم انجام گردید تا سطرهای این جدول برای آموزش مدل های هوش مصنوعی مناسب شود.

جدول ۲. الگوریتم های سری زمانی استفاده شده برای تحلیل و ارزیابی درستی پیشبینی xpath ها

الگوریتم ها	۰۶-۱۴۰۴ ۲۶	۰۵-۱۴۰۴ ۲۹	۰۴-۱۴۰۴ ۲۵	۰۳-۱۴۰۴ ۲۸
	R^2 (ضریب تعیین)			
LSTM	۱	۱	۰/۹۹۹	۱
ANN		۱	۰/۹۹۸	۰/۹۹۹
SVR	۰/۹۸۷	۰/۹۸۹	۰/۹۸۲	۰/۹۸۷
XGBoost	۰/۹٦٤	۰/۹۸۹	۰/۹۷۷	۰/۹۸۲

٤. بررسی آنچه انتخابگرها ارائه داده اند: در این مرحله، انتخابگرها تمام المان های اصلی صفحه که برنامه در تلاش برای جمع آوری آن بود را جمع آوری نموده اند، اما سوال اینجاست که از کجا انتخابگرها همان المان مورد نظر را باز می گردانند و به عنوان مثال مقدار یک فیلتر در صفحه یا یک مقدار کاملا غیر مرتبط را باز نمی گردانند؟. برای جلوگیری از چنین مشکلاتی چندین راهکار متفاوت در طول برنامه مورد استفاده قرار گرفته است. یکی اینکه تعداد xpath مشابه برای نام محصول، قیمت و تصویر باید حتما ۲۰ عدد باشد و برای درصد تخفیف و برچسب محصول کمتر از ۲۰ یا ۲۰ باشد. دوم نام محصول یک مقدار رشته ای است و طول آن نهایتا ۲۵۰ کاراکتر میباشد، درصد تخفیف مقداری عددی است که در کنار آن علامت % قرار دارد، تصویر محصول فایلی از نوع تصویر می باشد و برچسب محصول هم رشته ای است شامل مقادیر ثابت نظیر فروش شگفت انگیز، فروش ویژه و بلک فرایدی. با ترکیب این ۲ شرط در طول برنامه و در بخش های مختلف نظیر جداول و اسکریپت ها، کدهایی برای آنها نوشته شده است که درصورت وجود مشکل، برنامه از آن مطلع و انتخابگرها را اصلاح می نماید.

بیان یافته‌ها و جمع بندی

طبق بررسی های انجام شده، استفاده از روش پیشنهادی باعث ایجاد پایداری و مقاومت در کدهای سلنیوم برای برداشت اطلاعات از بستر وب گردید. طرح پیشنهادی از تاریخ ۵ مهر ۱۴۰۲ به صورت هفته ای یکبار شروع به جمع آوری تگ های موجود در ساختار html صفحه اول محصولات در دیجی کالا نمود و شش ماه اول هر هفته تقریبا ۲۰ هزار xpath استخراج شده از سایت دیجی کالا را به جدول تگ ها اضافه نمود و پس از آن هفته هایی که هیچ خطایی وجود نداشت تنها ۷ xpath اصلی به جدول افزوده شدند. درنتیجه در تاریخ ۱۴۰۴-۰۶-۲۶ که آخرین برداشت صورت گرفت، جدول تگ ها تقریبا دارای یک میلیون سطر داده بود. همچنین دقت اجرای الگوریتم های سری زمانی از طریق ضریب تعیین در ۴ تاریخ مختلف اندازه گیری شد که الگوریتم lstm در بسیاری از مواقع دقت ۱۰۰ درصدی داشته است. همچنین مدت زمان اجرای برنامه و وضعیت خطاهای آن درصورت استفاده و عدم استفاده از روش پیشنهادی در جدول ۳ آمده است که در بهترین حالت، یعنی زمانی که از اسکریپتی مشابه اسکریپت هفته قبل استفاده شد و خطا هم وجود نداشت، مدت زمان برداشت اطلاعات ۶۰ محصول از دیجی کالا ۱۸۰ ثانیه یا همان ۳ دقیقه طول کشیده است و در بدترین حالت که هم اسکریپت سلنیوم خطا داشته و هم الگوریتم هوش مصنوعی و برنامه مجبور به پیمایش کل جدول تگ ها شده، مدت زمان اجرای برنامه ۵۵۰ ثانیه یا همان ۹ دقیقه و میانگین مدت زمان اجرای کامل برنامه برای ۶۰ محصول در ۱۸ برداشت ۲۵۵/۲۲ ثانیه طول کشیده است که این مقدار به نسبت تکنیک های پردازش تصویر که ساعت ها اجرای آنها طول می کشد، بسیار بهتر است (جدول ۳). همچنین جدول ۴ گزارشی از تعداد اطلاعات برداشت شده به صورت خودکار از بستر وب و عدم اعمال تغییرات دستی در کد،

هنگام استفاده و عدم استفاده از روش پیشنهادی و میانگین تعداد اطلاعاتی که با موفقیت برداشت شده‌اند را ارائه می‌دهد. براساس جدول ۴، در صورت عدم استفاده از روش پیشنهادی، تقریبا اطلاعات ۲۶ محصول از ۶۰ محصول به صورت کامل برداشت شده است. به عبارت دیگر، درصورت عدم استفاده از روش پیشنهادی در هنگام برداشت اطلاعات به صورت کاملا خودکار، بیش از ۵۶ درصد از اطلاعات از بین می‌روند. درحالی که، هنگام استفاده از روش پیشنهادی ۱۰۰ درصد اطلاعات برداشت شده‌اند.

جدول ۳. مقایسه وجود خطا و عدم وجود خطا، قبل و بعد از اجرای روش پیشنهادی و مدت زمان اجرای کامل برنامه

ردیف	تاریخ	خطا در اسکریپت سلنیوم		مدت زمان اجرای کامل برنامه برای ۶۰ محصول (ثانیه)
		قبل از اجرای روش پیشنهادی	بعد از اجرای روش پیشنهادی	
۱	۱۴۰۴-۰۲-۳۱	خیر	خیر	۱۸۰
۲	۱۴۰۴-۰۳-۰۷	خیر	خیر	۱۸۰
۳	۱۴۰۴-۰۳-۱۴	بله	خیر	۲۸۰
۴	۱۴۰۴-۰۳-۲۱	بله	خیر	۳۱۰
۵	۱۴۰۴-۰۳-۲۸	بله	خیر	۲۱۰
۶	۱۴۰۴-۰۴-۰۴	بله	خیر	۲۱۰
۷	۱۴۰۴-۰۴-۱۱	خیر	خیر	۱۸۰
۸	۱۴۰۴-۰۴-۱۸	خیر	خیر	۱۸۰
۹	۱۴۰۴-۰۴-۲۵	بله	خیر	۳۴۴
۱۰	۱۴۰۴-۰۵-۰۱	خیر	خیر	۱۸۰
۱۱	۱۴۰۴-۰۵-۰۸	بله	خیر	۵۵۰
۱۲	۱۴۰۴-۰۵-۱۵	بله	خیر	۴۲۰
۱۳	۱۴۰۴-۰۵-۲۲	خیر	خیر	۱۸۰
۱۴	۱۴۰۴-۰۵-۲۹	بله	خیر	۲۰۰

۱۵	۱۴۰۴-۰۶-۰۵	خیر	خیر	۱۸۰
۱۶	۱۴۰۴-۰۶-۱۲	بله	خیر	۳۷۵
۱۷	۱۴۰۴-۰۶-۱۹	خیر	خیر	۱۸۰
۱۸	۱۴۰۴-۰۶-۲۶	بله	خیر	۲۵۵
میانگین مدت زمان اجرای کامل برنامه برای ۶۰ محصول در ۱۸ برداشت :				۲۵۵/۲۲

جدول ٤. تعداد اطلاعات برداشت شده به صورت خودکار از بستر وب و عدم اعمال تغییرات دستی در کد، هنگام استفاده و عدم استفاده از روش پیشنهادی

ردیف	تاریخ	عدم استفاده از روش پیشنهادی				استفاده از روش پیشنهادی			
		نام محصول	قیمت	تخفیف	نوع تخفیف	نام محصول	قیمت	تخفیف	نوع تخفیف
۱	۱۴۰۴-۰۲-۳۱	۶۰	۶۰	۱۷	۱۷	۶۰	۶۰	۱۷	۱۷
۲	۱۴۰۴-۰۳-۰۷	۶۰	۶۰	۲۲	۲۲	۶۰	۶۰	۲۲	۲۲
۳	۱۴۰۴-۰۳-۱۴	۱۱	۱۱	۵	۴	۶۰	۶۰	۲۹	۲۹
۴	۱۴۰۴-۰۳-۲۱	۲	۲	۰	۰	۶۰	۶۰	۱۲	۱۲
۵	۱۴۰۴-۰۳-۲۸	۴	۴	۲	۲	۶۰	۶۰	۱۴	۱۴
۶	۱۴۰۴-۰۴-۰۴	۱۲	۱۱	۴	۰	۶۰	۶۰	۳۷	۳۷
۷	۱۴۰۴-۰۴-۱۱	۶۰	۶۰	۳۱	۳۱	۶۰	۶۰	۳۱	۳۱
۸	۱۴۰۴-۰۴-۱۸	۶۰	۶۰	۲۱	۲۱	۶۰	۶۰	۲۱	۲۱
۹	۱۴۰۴-۰۴-۲۵	۰	۰	۰	۰	۶۰	۶۰	۲۵	۲۵
۱۰	۱۴۰۴-۰۵-۰۱	۶۰	۶۰	۳۳	۳۳	۶۰	۶۰	۳۳	۳۳
۱۱	۱۴۰۴-۰۵-۰۸	۰	۰	۰	۰	۶۰	۶۰	۱۹	۱۹
۱۲	۱۴۰۴-۰۵-۱۵	۵	۰	۰	۰	۶۰	۶۰	۱۳	۱۳
۱۳	۱۴۰۴-۰۵-۲۲	۸	۸	۴	۲	۶۰	۶۰	۲۷	۲۷
۱۴	۱۴۰۴-۰۵-۲۹	۶	۶	۵	۵	۶۰	۶۰	۱۵	۱۵
۱۵	۱۴۰۴-۰۶-۰۵	۶۰	۶۰	۴۰	۴۰	۶۰	۶۰	۴۰	۴۰
۱۶	۱۴۰۴-۰۶-۱۲	۰	۰	۰	۰	۶۰	۶۰	۳۸	۳۸

۱۷	۱۴۰۴-۰۶-۱۹	۶۰	۶۰	۲۷	۲۷	۶۰	۶۰	۲۷	۲۷
۱۸	۱۴۰۴-۰۶-۲۶	۱	۰	۰	۰	۶۰	۶۰	۲۱	۲۱
میانگین تعداد اطلاعات برداشت شده		۲۶/۱۱	۲۵/۷۸	۱۲/۵۶	۱۲/۱۷	۶۰	۶۰	۲۵/۳۳	۲۵/۳۳

مزایای این پژوهش و استفاده از روش پیشنهادی باعث می‌شود که نیاز به تغییر دستی انتخابگرها بعد از هر تغییر ظاهری سایت از بین برود، انتخابگرهای پایدار به صورت خودکار استخراج شوند، با حذف تلاش انسانی، سرعت و پایداری سیستم جمع‌آوری داده افزایش یابد، از داده‌های تاریخی برای تصمیم‌گیری خودکار در استخراج اطلاعات استفاده شود.

این مطالعه به‌منظور طراحی، پیاده‌سازی و ارزیابی یک سامانه مقاوم برای برداشت خودکار اطلاعات از وب انجام شده است. در گام نخست، ساختار کلی سیستم پیشنهادی طراحی شد که شامل دو بخش اصلی است:

۱) ماژول تولید انتخابگرهای پویا،

۲) لایه تصمیم‌گیری مبتنی بر هوش مصنوعی برای انتخاب بهینه انتخابگر در زمان اجرا.

در مرحله اول، مجموعه‌ای از داده‌های نمونه از ساختار صفحات وب جمع‌آوری شد تا تغییرات محتمل در چینش و ویژگی‌های موجو مورد تحلیل قرار گیرد. برای این منظور، صفحات هدف در بازه‌های زمانی مختلف ذخیره و مقایسه شدند تا الگوهای تغییر و رفتار عناصر قابل شناسایی باشند. خروجی این مرحله، مجموعه‌ای از ویژگی‌های قابل استناد برای طبقه‌بندی و انتخاب انتخابگرهای مناسب بود.

در مرحله دوم، الگوریتمی طراحی شد که برای هر عنصر هدف، چندین انتخابگر با اولویت‌های مختلف تولید می‌کند؛ این انتخابگرها در یک پایگاه داده ذخیره شده و برای تحلیل‌های بعدی در اختیار لایه تصمیم‌گیری قرار گرفتند.

در مرحله سوم، یک مدل یادگیری ماشین برای تصمیم‌گیری بین انتخابگرهای تولیدشده توسعه یافت. این مدل با استفاده از داده‌های ثبت‌شده از اجرای واقعی سیستم آموزش داده شد و توانست در صورت وقوع خطا، مناسب‌ترین انتخابگر جایگزین را بدون دخالت انسان انتخاب کند. در طراحی این مدل، معیارهایی همچون نرخ موفقیت انتخابگر، عمق جستجو، میزان وابستگی به ویژگی‌های تغییرپذیر و سرعت اجرا لحاظ شد.

در مرحله چهارم، سیستم پیشنهادی به‌صورت عملی بر روی وب‌سایت دیجی‌کالا پیاده‌سازی و به مدت دو سال، با تناوب هفتگی اجرا شد. در طول این مدت، داده‌های مربوط به تعداد شکست‌ها، نیاز به اصلاح دستی، زمان اجرای عملیات و میزان پایداری سیستم جمع‌آوری و ثبت گردید.

در مرحله نهایی، عملکرد روش پیشنهادی با روش استاندارد مبتنی بر انتخابگرهای ثابت مقایسه شد. معیارهای ارزیابی شامل نرخ شکست، نیاز به مداخله انسانی، هزینه نگه‌داری، پایداری در برابر تغییرات ساختار صفحه و زمان اجرای عملیات بود. نتایج نشان داد که روش پیشنهادی توانسته است تعداد شکست‌ها و نیاز به اصلاح دستی را به شکل معنی‌داری کاهش دهد و در محیطی پویا عملکرد قابل اتکاتری ارائه کند.

بحث و نتیجه‌گیری

هدف این پژوهش ارائه روشی بود که بدون نیاز به بازنویسی مداوم انتخابگرها پس از تغییر ساختار HTML سایت‌ها، بتوان فرآیند بازیابی و استخراج داده و در راستای آن بازیابی دانش را پایدار و خودکار نمود. در حالت معمول، تغییر در ساختار صفحه (مثلاً تغییر کلاس CSS

یا لایه‌بندی جدید) باعث شکست کامل اسکریپت استخراج داده می‌شود. رویکرد ارائه‌شده در این نوآوری، ۱. تحلیل کامل HTML صفحات هدف، ۲. استخراج همه تگ‌ها و مسیر کامل XPath با شماره‌گذاری ساختاری، ۳. استخراج الگوی تکرار و رفتار پایداری انتخابگرها و ۴. استفاده از الگوریتم تصمیم‌گیر هوش مصنوعی برای انتخاب انتخابگر مناسب توانست این مشکل را حل کند. از نتایج ثبت‌شده در جدول شمارش تگ ها مشخص شد که برخی انتخابگرها در طول زمان پایداری بیشتری دارند. این موضوع نشان داد که انتخابگرهای پایدار معمولاً وابسته به ساختار اصلی سایت و نه لایه بندی ظاهری هستند و می توانند مبنای تصمیم‌گیری الگوریتمی قرار گیرند.

نتایج حاصل از این پژوهش نشان می دهد که: انتخابگرهای پویا می‌توانند به صورت خودکار و بدون دخالت انسان انتخاب شوند، میزان شکست استخراج داده تا بیش از ۹۹٪ کاهش یافت، با استفاده از مدل‌های یادگیری، انتخابگرها بر اساس سری زمانی پایدارسازی شدند، همچنین سیستم طراحی شده قابلیت رشد دارد و با افزایش داده‌های تاریخی، دقت مدل هوش مصنوعی بهتر می‌شود. این روش می‌تواند به عنوان راهکاری عمومی برای وب‌اسکرپینگ مقاوم در برابر تغییرات سایت‌ها به‌کار رود. همچنین مدل داده‌محور استفاده‌شده، سیستم را به یک خزنده نیمه‌خودمختار تبدیل می‌کند که بدون بازنویسی کد قادر به ادامه فعالیت است.

در نتیجه با استفاده از روش پیشنهادی می‌توان داده های مورد نیاز را به صورت خودکار، در هر زمان و با دقت بالا، بازیابی نمود. در این روش، خزنده‌های وب به‌عنوان برنامه‌های خودکار، ضمن برطرف نمودن نیاز سازمان‌ها و پژوهشگران به بازیابی و تحلیل مستمر داده‌های موجود در بستر وب، می‌توانند به بازیابی دانش و اطلاعات موجود در این داده‌ها و ایجاد نظام‌های معنایی ارزشمند و واقعی و ساخت سامانه‌هایی نظیر بات‌های هوشمند بازیابی و ارائه دانش مانند چت جی پی تی و سیستم‌های مختلفی از جمله سیستم‌های تست

نرم‌افزاری تحت وب و گزارش خطا، تحلیل کسب‌وکار، پایش قیمت، رصد رفتار مشتریان، جمع‌آوری اطلاعات رقبا، پایش بازارهای مالی و بسیاری از خدمات مبتنی بر داده کمک شایانی کنند.

پیشنهادات آتی

با توجه به این که این برنامه برای یک وبسایت فروشگاهی نوشته شده، درنتیجه برای سایت های دارای اشیاء با ویژگی های متفاوت (به عنوان مثال در اینجا شی محصول و ویژگی های نام، قیمت، تصویر، تخفیف و برچسب در آن) مناسب می باشد. به عبارت دیگر، این برنامه برای سایت هایی نظیر دکتردکتر ، آمازون یا ترب که ویژگی های مشابهی دارند به خوبی کار می کند ولی برای سایت هایی مانند python.org یا برنامه های کاربردی تحت وب که ساختار متفاوتی دارند، بررسی نشد است. برای پژوهش های آتی می توان سایت هایی که ساختارهای متفاوتی دارند را براساس راهکار پیشنهادی این پژوهش مورد تحلیل و بررسی قرار داد.

منابع

تقی‌زاده کورایم, فرناز, کابارانزادقدیم, موسوی, & سیدعبداله‌امین. (۲۰۲۴). سازوکاری برای مدیریت زمان و افزایش دقت اطلاعات هنگام استفاده از کتابخانه سلنیوم. بازیابی دانش و نظام‌های معنایی, ۴۱(۱۱), ۲۰۴–۲۲۵. https://doi.org/10.22054/jks.2024.80235.1660 .

درون متن: (تقی زاده و همکاران ، ۲۰۲۴)

تقی زاده و همکاران (۲۰۲۴)

Coppola, R., Feldt, R., Nass, M., & Alégroth, E. (2025). Ranking approaches for similarity-based web element location. Journal of Systems and Software, 222, 112286.

Parenthetical citations: (Coppola et al., 2025)

Narrative citations: کوپولا و همکاران[1] (۲۰۲۵)

Degaki, R. H., Colonna, J. G., Lopez, Y., Carvalho, J. R., & Silva, E. (2022, November). Real Time Detection of Mobile Graphical User Interface Elements Using Convolutional Neural Networks. In Proceedings of the Brazilian Symposium on Multimedia and the Web (pp. 159-167).

1. Coppola et al.

Parenthetical citations: (Degaki et al., 2022)

Narrative citations: دگکی و همکاران[1] (۲۰۲۲)

Healenium. (2025). An open-source project developed by EPAM Systems. The project was initially released around 2019, and its latest update was on September 24, 2025. https://github.com/healenium.

Parenthetical citations: (Healenium, 2025)

Narrative citations: هیلنیوم[2] (۲۰۲۵)

Kirinuki, H., Tanno, H., & Natsukawa, K. (2019, February). COLOR: correct locator recommender for broken test scripts using various clues in web application. In 2019 IEEE 26th International Conference on Software Analysis, Evolution and Reengineering (SANER) (pp. 310-320). IEEE.

Parenthetical citations: (Kirinuki et al., 2019)

Narrative *citations*: کیرینوکی و همکاران[3] (۲۰۱۹)

Khaliq, Z., Khan, D. A., & Farooq, S. U. (2023). Using deep learning for selenium web UI functional tests: A case-study with e-commerce applications. Engineering Applications of Artificial Intelligence, 117, 105446.

Parenthetical citations: (Khaliq et al., 2023)

Narrative citations: خلیق و همکاران 4 (۲۰۲۳)

Kluge, A., & Stocco, A. (2025). Web Element Relocalization in Evolving Web Applications: A Comparative Analysis and Extension Study. arXiv preprint arXiv:2505.16424.

Parenthetical citations: (Kluge & Stocco, 2025)

Narrative citations: کلوگه و استوکو[5] (۲۰۲۵)

Nass, M., Alégroth, E., Feldt, R., Leotta, M., & Ricca, F. (2023). Similarity-based web element localization for robust test automation. ACM Transactions on Software Engineering and Methodology, 32(3), 1-30.

Parenthetical citations: (Nass et al., 2023)

Narrative *citations*: نس[6] و همکاران (۲۰۲۳)

Nass, M., Alégroth, E., & Feldt, R. (2024). Improving web element localization by using a large language model. Software Testing, Verification and Reliability, 34(7), e1893.

Parenthetical citations: (Nass et al., 2024)

---

1. Degaki et al.
2. Healenium
3. Kirinuki et al.
4. Khaliq et al.
5. Kluge & Stocco
6. Nass et al.

Narrative *citations*: نس[1] و همکاران (۲۰۲۴)

Saarathy, S. C. P., Bathrachalam, S., & Rajendran, B. K. (2024). Self-Healing Test Automation Framework using AI and ML. International Journal of Strategic Management, 3(3), 45-77.

Parenthetical citations: (Saarathy et al., 2024)

Narrative citations: ساراسی و همکاران [2] (۲۰۲۴)

Shayan Daneshvar, S., & Wang, S. (2024). GUI Element Detection Using SOTA YOLO Deep Learning Models. arXiv e-prints, arXiv-2408.

Parenthetical citations: (Daneshvar & Wang, 2024)

Narrative citations: دانشور و ونگ [3] (۲۰۲۴)

Taghizadeh Kourayem, Farnaz, Kabaranzad Ghadim, Mohammadreza, & Mousavi, Seyed Abdollah Amin. (2024). A mechanism to manage time and increase data accuracy when using the Selenium library. Knowledge Retrieval and Semantic Systems, 41(11), 204-225. https://doi.org/10.22054/jks.2024.80235.1660.

Parenthetical citations: (Taghizadeh et al., 2024).

Narrative citations: تقی زاده و همکاران (۲۰۲٤)

1. Nass et al.
2. Saarathy et al.
3. Daneshvar & Wang