АВТОМАТИЗАЦИЯ МОНИТОРИНГА ЦЕН С ИСПОЛЬЗОВАНИЕМ WEB-СКРАПИНГА И ТЕХНОЛОГИЙ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

Смирнова Марина Александровна[1], **Муталова Луиза Улугбековна**[2], **Васильева Гера Викторовна**[3], **Колебер Камила Леонидовна**[4]

[1,2,3,4]Карагандинский университет имени Е.А.Букетова, Караганда, Республика Казахстан
[1]E-mail: Smirnova_Marina@buketov.edu.kz
[2]E-mail: genkaire@gmail.com
[3]E-mail: yasslh@mail.ru
[4]E-mail: koleberkamila@gmail.com

В условиях стремительного роста электронной коммерции и увеличения конкуренции на рынке актуальной задачей становится оперативный мониторинг цен и ассортимента товаров. Компании, работающие в ритейле, стремятся отслеживать цены конкурентов, динамику спроса и наличие товаров в реальном времени. Одним из ключевых инструментов для этого является web-скрапинг - технология автоматического извлечения данных с веб-сайтов. Однако, простого сбора данных недостаточно: необходимо уметь интерпретировать, структурировать, фильтровать и анализировать большие массивы информации. На этом этапе критически важную роль начинают играть технологии искусственного интеллекта (ИИ). В данной статье рассматривается, как можно автоматизировать процесс мониторинга цен с использованием web-скрапинга и как ИИ усиливает возможности систем мониторинга цен, обеспечивая интеллектуальную обработку и интерпретацию данных, полученных через web-скрапинг.

Нами ставилась цель разработать программное средство для автоматизированного парсинга товаров с маркетплейсов и представления данных пользователю через удобный графический интерфейс. Для достижения этой цели решены следующие задачи: проведён обзор существующих методов и инструментов веб-скрапинга, выбран и обоснован технологический стек (язык Python, библиотеки Selenium, BeautifulSoup, pandas и др.); реализована программа, способная собирать данные о товарах (названия, цены, отзывы и пр.) с целевых сайтов с учётом возможных защит от бот-парсинга; разработан графический интерфейс пользователя (GUI), предоставляющий удобные средства для запуска парсинга, настройки параметров и просмотра результатов; выполнено тестирование и отладка приложения на реальных данных; сформулированы рекомендации по дальнейшему развитию проекта. Был создан инструмент, который может использоваться для мониторинга товарных позиций и цен конкурентов в режиме реального времени, что особенно важно для отделов маркетинга и аналитики в сфере электронной коммерции.

Возможности веб-скрапинга широко применяются в различных отраслях. В частности, для мониторинга цен и ассортимента: компании регулярно выгружают с конкурирующих сайтов сведения о товарах, ценах и отзывах, чтобы своевременно реагировать на изменения рынка. Другие распространённые случаи использования – агрегаторы контента (новостей, объявлений), системы лидогенерации (парсинг контактов, вакансий) и т.д. [1]. Таким образом, веб-скрапинг стал неотъемлемым инструментом анализа в бизнесе, позволяя собирать публично доступные данные автоматически.

Для решения поставленной задачи выбран язык Python, который предлагает богатый набор библиотек для веб-скрапинга. Существует два ключевых подхода: скрапинг статических страниц через HTTP-запросы и headless-скрапинг динамических страниц с эмуляцией браузера [1]. Первый подход опирается на такие инструменты, как библиотека Requests и парсер BeautifulSoup. Для современных динамических сайтов, активно использующих JavaScript для подгрузки данных, зачастую применяют второй подход – эмуляцию веб-браузера. В Python-developers de-facto стандартом стал инструментарий Selenium. С помощью Selenium можно автоматически запустить невидимый браузер (режим headless), загрузить страницу, дождаться выполнения скриптов и динамически сформированного контента, а затем получить итоговый DOM. По сути, Selenium имитирует действия реального пользователя: может прокручивать страницу, нажимать кнопки, заполнять формы и т.д. Благодаря этому, становятся доступны для парсинга те данные, которые появляются только после исполнения JS (например, списки товаров, подгружающиеся при скроллинге, или цены, обновляющиеся через API-запросы).

Данные, извлечённые с веб-страниц, обычно представляют собой набор структурированной информации (список товаров с атрибутами: название, цена, рейтинг, и т.п.). Для последующего анализа и импорта в другие системы их следует сохранить в удобном формате. В рамках проекта экспорт результатов парсинга реализован именно в CSV - по окончании сбора данных программа генерирует файл results.csv, где каждая строка содержит данные по одному товару (например, «Наименование»; «Цена»; «Рейтинг»; «Количество отзывов» и т.д.).

Помимо простого сохранения, часто требуется еще и предварительная обработка данных: очистка, приведение типов (например, цен к числовому формату без валютных символов), фильтрация дублей, агрегирование. Для этих целей в Python-деэкосистеме существует библиотека pandas. Графический пользовательский интерфейс (GUI) значительно повышает удобство работы с утилитой по сравнению с консольным запуском скриптов. GUI предоставляет визуальный и интуитивно понятный способ взаимодействия, тогда как командная строка требует знания специальных команд и синтаксис. Для разработки настольных приложений на Python имеется несколько популярных инструментов. Стандартным является библиотека Tkinter - официальная оболочка Tcl/Tk в составе Python.

Следуя принципам удобства и UX, при проектировании интерфейса особое внимание уделено простоте использования. Пользователю не должно требоваться выполнять много действий для запуска парсинга - достаточно минимального числа шагов. В нашем приложении предусмотрено всего одно окно, где сосредоточены основные элементы управления: поля ввода параметров и кнопка запуска. Интерфейс реализует понятную логику: пользователь вводит необходимую информацию (например, ссылку на категорию товаров или артикул) и нажимает кнопку «Начать парсинг», после чего может наблюдать ход выполнения. Все элементы UI оформлены последовательно и единообразно, в соответствии с принципом согласованности дизайна, что снижает когнитивную нагрузку на пользователя.

Таким образом, интерфейс приложения спроектирован так, чтобы даже не подготовленный технически пользователь мог интуитивно понять, как собрать данные: указать параметры и запустить процесс одной кнопкой. Такой подход повышает практическую ценность инструмента, делая его доступным для сотрудников отделов маркетинга и аналитики, не обладающих навыками программирования.

Многие крупные веб-сайты, особенно коммерческие, внедряют механизмы защиты от автоматизированного сбора данных. Это продиктовано как соображениями нагрузки на сервер, так и желанием защитить уникальный контент или предотвратить массовый парсинг конкурентов. Для обхода подобных защит разработчики скраперов применяют различные подходы. Простейшая мера - имитация работы реального браузера. В нашем проекте парсинг проводится в рамках разумного (десятки страниц за запуск) и носит сугубо аналитический характер, не нарушающий прав владельцев площадок.

Для верификации корректности работы инструмента были проведены тестовые запуски на разных категориях и сайтах. Разработанный инструмент демонстрирует эффективность веб-скрапинга для мониторинга цен в реальном времени. Он может найти применение для автоматизации конкурентной разведки на рынке e-commerce: компании смогут быстрее реагировать на ценовые изменения, а аналитики - собирать данные для исследований рынка. Инструмент масштабируем: добавив новые конфигурации, можно мониторить и другие сайты. Использование Selenium в сочетании с CustomTkinter показало, что даже настольное приложение с интерфейсом может быть создано на Python относительно быстро, интегрируя возможности браузерной автоматизации.

Данные, полученные через web-скрапинг, часто бывают неструктурированными, шумными и избыточными. Различия в форматах страниц, динамическая генерация контента (например, через JavaScript), ошибки в разметке, синонимы, дубли и различия в представлении характеристик товаров - всё это требует сложной обработки перед тем, как информация может быть использована для принятия решений. Здесь стандартные алгоритмы поиска и фильтрации оказываются неэффективными. В разработанном инструменте возможные улучшения автоматизации мониторинга цен. Могут на помощь прийти технологии искусственного интеллекта (ИИ), в частности машинное обучение (ML), обработка естественного языка (NLP), а также методы компьютерного зрения [2].

Одна из наиболее трудоёмких задач в мониторинге цен - это сопоставление товаров, представленных на разных торговых площадках. Один и тот же продукт может иметь различные названия, описания и изображения. Для автоматизации этого могут быть использованы алгоритмы машинного обучения:

- обучение с учителем (Supervised Learning) позволяет построить модели, которые распознают товарные позиции на основе обучающего набора;
- методы кластеризации (unsupervised learning) помогают группировать схожие товарные позиции без предварительной разметки. Это особенно полезно при мониторинге ассортимента, когда структура данных неизвестна заранее;
- метрики текстовой схожести (например, cosine similarity, Jaccard index) применяются к названиям и описаниям товаров, а также векторные представления слов (word embeddings) позволяют более точно выявлять семантическую близость.

Таким образом, ИИ помогает формировать «умные» сопоставления товаров, определять, является ли одна и та же позиция представленной у разных продавцов, и проводить корректный ценовой анализ.

Технологии обработки естественного языка позволяют анализировать текстовые описания товаров, извлекать ключевые характеристики (бренд, объём, цвет, комплектацию и т.д.), а также интерпретировать отзывы и пользовательские оценки. Это важно не только для идентификации товаров, но и для оценки их качественных характеристик:

- извлечение признаков: модели на основе трансформеров (например, BERT) исполь-

зуются для выделения сущностей и классификации текстов;
- анализ тональности (Sentiment Analysis): позволяет оценивать настроение отзывов и соотносить это с ценой и спросом;
- распознавание шаблонов: помогает адаптироваться к различным стилям описания товаров и изменяющимся форматам сайтов.

С помощью NLP можно будет автоматически выделять атрибуты товаров и корректно обрабатывать большие объёмы разнородного текстового контента, что раньше требовало участия человека.

Во многих случаях описание товара неполное или отсутствует вовсе, и единственным источником информации остаются изображения. В таких случаях применяются технологии компьютерного зрения:

- сверточные нейронные сети (CNN): позволят распознавать изображения и извлекать визуальные признаки товаров;
- семантическое сопоставление изображений: поможет определить, изображён ли один и тот же товар на разных сайтах;
- обнаружение объектов (Object Detection): может быть использован для анализа упаковки, брендов, маркировки и даже ценников на изображениях.

Эти методы существенно повышают точность сопоставления товаров и позволяют анализировать даже те предложения, где отсутствует структурированная текстовая информация.

Использование искусственного интеллекта в автоматизированных системах мониторинга цен радикально повышает их эффективность, адаптивность и интеллектуальные возможности. Машинное обучение, NLP и компьютерное зрение позволяют не просто собирать данные, а извлекать из них ценные бизнес-инсайты. Будущее таких систем заключается в их способности к самообучению, интеграции с ERP и CRM-системами, а также в использовании генеративных моделей для формирования аналитических отчётов. Применение ИИ уже сегодня позволяет компаниям принимать более обоснованные и своевременные решения, укрепляя свои позиции в конкурентной среде.

Список литературы

[1] Turlych. Парсинг Web-сайтов: взгляд изнутри // Хабр [Электронный ресурс] - URL: https://habr.com/ru/articles/803869/

[2] Stanford University - CS224N: Natural Language Processing with Deep Learning [Электронный ресурс]. URL: https://web.stanford.edu/class/cs224n/

ЖАСАНДЫ ИНТЕЛЛЕКТ ЖӘНЕ МАТЕМАТИКА: ҚАЗІРГІ ЗАМАНҒЫ ОҚУ ҮДЕРІСІНДЕГІ ИНТЕГРАЦИЯ

Тайболдина Қаламқас Радылхановна[1], **Оспанова Динара Манаповна**[2], **Асқар Шырайлым Ұланқызы**[3]

[1,2,3]SHAKARIM UNIVERSITY, Семей қаласы, Қазақстан
[1]E-mail: k.taiboldina@shakarim.kz
[2]E-mail: d.ospanova@shakarim.kz