Article

A Machine Learning Python-based Search Engine Optimization Audit Software

Konstantinos I. Roumeliotis [1] and Nikolaos D. Tselikas [1,*]

[1] Department of Informatics and Telecommunications; University of Peloponnese; k.roumeliotis@uop.gr, ntsel@uop.gr

* Correspondence: ntsel@uop.gr; Tel.: +30 2710372216

Abstract: In the present-day digital landscape, websites have increasingly relied on digital marketing practices, notably search engine optimization (SEO), as a vital component in promoting sustainable growth. The traffic a website receives directly determines its development and success. As such, website owners frequently engage the services of SEO experts to enhance their website's visibility and increase traffic. These specialists employ premium SEO audit tools that crawl the website's source code to identify structural changes necessary to comply with specific ranking criteria, commonly called SEO factors. Working collaboratively with developers, SEO specialists implement technical changes to the source code and await the results. The cost of purchasing premium SEO audit tools or hiring an SEO specialist typically ranges in the thousands of dollars per year. Against this backdrop, this research endeavors to provide an open-source Python-based Machine Learning SEO software tool to the general public, catering to the needs of both website owners and SEO specialists. The tool analyzes the top-ranking websites for a given search term, assessing their on-page and off-page SEO strategies, and provides recommendations to enhance a website's performance to surpass its competition. The tool yields remarkable results, boosting average daily organic traffic from 10 to 143 visitors.

Keywords: search engine optimization; seo techniques; python seo tool; machine learning seo

1. Introduction

In the current competitive environment, websites of all industries must implement SEO techniques and technologies due to the increasing internet use, changes in web searcher habits, heightened competition, and the necessity of obtaining traffic to a website through organic sources, rather than paid advertisements. With the constant updates to search engine algorithms, these factors serve as only a starting point for a website to maintain its visibility and relevance.

In just three decades, the Web has rapidly developed, starting from the initial document-based Web 1.0 that was created in 1990 to the more mobile and social Web 2.0 established in 1999 and advancing to the semantic Web 3.0 [1].

During earlier times when the number of websites was considerably lower, users possessed a heightened ability to recall and enter the precise web address of the desired site. In previous times, when the quantity of websites accessible was considerably lower, individuals demonstrated an enhanced ability to recollect and manually input the exact website address they desired. However, in the present day where the Web boasts nearly 2 billion active websites, users have adopted the practice of utilizing search engines to retrieve information by inputting relevant keyword phrases [2]. In essence, search engines function as bookmarks that direct users to specific pages based on their search queries.

In the realm of website traffic optimization, there exist various channels for websites to increase their visibility and attract visitors. These channels include Organic Search, Paid traffic, Social, Referral, Direct, and, as elucidated by the Google Analytics application [3,4]. Organic traffic pertains to individuals who employ search engines as a means to locate their desired information or content. When a user inputs a specific keyword into a search engine, a compilation of websites utilizing that particular keyword is presented, with the volume of traffic a website receives being significantly contingent upon its ranking in search engines and the resultant organic traffic [5]. Achieving a high ranking in organic results (SERPs) is often tricky because it cannot be paid for, and specific ranking factors need to be met to achieve high search rankings, as described in the guidelines published by search engine companies [4]. While these factors are known, search engines have not fully disclosed their impact on search rankings, as they do not publicize their ranking algorithms or the factors used in the ranking [6]. Nevertheless, many studies have investigated the dominant SEO techniques and their impact on organic traffic [7,1].

In the field of website traffic optimization, there has emerged a need to adhere to search engine guidelines in order to attain a higher position in search results. Prior studies have recognized several SEO techniques and their importance in obtaining improved search engine rankings. However, there is a lack of explicit recommendations regarding which techniques can be employed by a web admin and in what order to maximize their website's SEO outcomes.

Under these circumstances, the necessity to optimize websites following the search engine guidelines has arisen, with the ultimate aim of attaining a higher position in the search results. Prior researches have successfully identified the available SEO techniques and their significance in achieving better positions in search results. However, none of these publications explicitly recommend which SEO techniques website owners should utilize and in what sequence to optimize their SEO outcomes.

Commercial SEO audit tools have been developed to address this gap by scanning the source code of websites to identify implemented SEO techniques and detect any shortcomings. These tools are offered in freemium versions that enable users to monitor a single web page and encourage them to purchase the SEO tool to extend scanning to additional websites. As a result, website owners are required to pay monthly to identify their competitors' strengths and weaknesses, regardless of whether they are in the early stages or have an established online presence. While the cost may be manageable for profit-generating websites, it can be overwhelming for new websites or non-profit organizations.

The present study aims to develop an open-source Python-based SEO audit software tool that will be accessible to the general public without charge and perform functions comparable to those offered by commercial SEO audit tools at a cost. The overarching objective is to produce an open-source SEO tool that will provide users with recommendations on appropriate SEO techniques based on analyzing their competitors' websites to optimize their websites for SEO and achieve improved search rankings and traffic.

Chapter 2 briefly presents an overview of the on-page and off-page SEO techniques used by the tool and recommended by Google Search Central Documentation’s Webmaster guidelines [4]. Chapter 3 explains the process and the rational behind the creation of the Python-based SEO tool, which analyzes the website's source code and competitor websites. Free APIs are used to extend tool’s functionality and machine learning to predict off-page SEO techniques and critical metrics.

Chapter 4 encompasses the utilization of a SEO tool to examine the competitive landscape of an operational e-commerce website. The findings suggested the incorporation of additional SEO techniques into the website's source code to enhance its prominence within search engine rankings. The findings of the study were remarkably significant, as the website experienced a substantial average increase of 143 daily organic visitors following the implementation of the recommended enhancements derived from the SEO tool. The incremented organic traffic coming of the proposed SEO tool is noteworthy, since the corresponding previous average was merely ten daily organic visitors obtained from search engine results before the integration of our enhancements.

2. Search Engine Optimization Overview

When it comes to optimizing a website for search engines, there are two main categories of SEO techniques: the on-page SEO and the off-page SEO. On-page SEO involves making changes and additions to a website's source code in order to meet various ranking factors. Off-page SEO, on the other hand, encompasses actions taken on external websites to establish credibility for the target website [8].

2.1. On-page SEO techniques

As previously stated, optimizing a website for search engines involves implementing on-page SEO techniques, which entails modifying the website's source code to fulfill the requirements of various ranking factors. The next section will focus on 19 on-page SEO techniques and technologies recommended in Google Search Central Documentation's Webmaster guidelines [4].

2.1.1. Title Tag

According to the World Wide Web Consortium (W3C), the title tag is an essential component of a website as it comprises a mix of terms and phrases that accurately represent the web page's content [9]. For a website to be visible in search results, it is crucial to have a title tag that is brief yet informative [4]. This helps users to grasp the content quickly and choose the most relevant result.

According to Moz's testing and experience, an optimal length for the title tag that satisfies both web users and search engines ranges between 6 and 78 characters [10]. It has been observed that placing keywords closer to the start of the tag can have a more profound effect on search rankings [10]. However, creating misleading title tags through keyword stuffing techniques can lead to search engines substituting the title with a more relevant tag based on the content of the web page [4,11].

2.1.2. SEO-Friendly URL

A Uniform Resource Locator (URL), also referred to as a RESTful, search-friendly, or user-friendly URL, is a text that is easily understandable by humans and outlines the organization of files on a web server. The URL is composed of three fundamental components: the access protocol, the domain name, and the path [12].

The URL of a website plays a crucial role in determining its ranking on search engines and connecting it to relevant search queries [7,1]. A well-structured URL can provide search engines and visitors with a clear understanding of the content of a page even before it is visited [13]. In order for URLs to be optimized for search engines, they should be concise, simple, and easy to comprehend, making use of words, hyphens, and slashes [4,8]. Conversely, URLs that include symbols such as ampersands, numbers, words, and question marks are considered non-SEO-friendly [14]. Additionally, according to the Moz SEO learning center, URLs should not exceed 2083 characters in length to ensure proper display across all browsers and visibility in search results [15].

2.1.3. Alternative Tags and Image Optimization

The file size of integrated images is a pivotal element that impacts the loading time of a website. When the file size of an image surpasses 100 kb, it poses a formidable obstacle for users to retrieve the image, despite advancements in internet speeds [1]. In order to tackle this predicament, numerous image compression formats have been developed. Among these, Google's WebP stands out as a contemporary image format that provides a lossless compression of up to 26% compared to the original image [16].

In the realm of search engine optimization (SEO), alternative tags (alt tags) hold a significant role in enhancing website accessibility and visibility. While computer vision and machine learning algorithms may not always accurately recognize image content, alt tags provide a text description that can benefit visually impaired users and search engine bots alike [4,17,18]. Moreover, incorporating targeted keywords in alt tags when integrating images into content can boost the website's SEO ranking [8,18]. Alongside alt tags, image file names are also essential for SEO, and the optimization techniques used for creating SEO-friendly URL structures can be applied to develop SEO-optimized image file names.

2.1.4. Link HREF Alternative Title Tags

In the context of website optimization, links play a vital role in both user navigation and search engine indexing. To ensure optimal user experience, users tend to hover over links to preview the destination page. Similarly, search engines rely on links to understand the content of the linked pages [19]. In this regard, the title tag serves as a crucial element in facilitating both user and search engine comprehension of a website's structure, internal linking, and external linking.

2.1.5. Meta Tags

The meta description tag, situated after the title tag within the head container of a webpage [9], serves as a concise summary of the content found on the page. When a user clicks on a search result, they are presented with the page's title, URL, and meta description. Search engines also use the meta description to provide searchers with an understanding of what the page contains [6,8]. SEO specialists work to optimize meta descriptions to make them compelling and relevant to the website's content [19]. To improve SEO rankings, the meta description should contain between 51 and 350 characters and should include the target keyword [1,17]. Although having metadata can improve a webpage's ranking, search engines like Google may choose not to use the meta description as the search result description due to its potential for being misleading [3].

2.1.6. Heading Tags

The HTML language specifies six heading tags, H1 through H6, adhering to w3c web standards [20]. Google Central Blog recommends the use of heading tags by web administrators to designate important content [4]. Search engines assign greater significance to headings than to regular body text, rendering them an essential element of a webpage [9]. Moreover, headings can benefit screen readers that aid users with visual impairments [18]. For SEO purposes, the primary keyword should appear in both the H1 and H2 tags, and the length of headings should range between 10 and 13 words [13]. Should the title tag be deemed misleading, search engines frequently substitute the H1 tag and present it in the SERPs.

2.1.7. Minified Static Files

Minification is a process that involves the removal of superfluous data from JavaScript, CSS, and HTML files while preserving their functionality, including the elimination of comments, formatting, and the shortening of variables and function names [4]. It is strongly recommended that website files be kept lightweight, as well as images, to enhance page loading times and improve the user experience [1]. By minimizing files, website loading times and search engine optimization (SEO) results can be improved; research has shown that load times can decrease by up to 16% and file sizes can be reduced by up to 70% [21]. Furthermore, minification can improve website security by substituting meaningful variable and function names with shorter and more obscure ones, making the code more difficult to interpret and comprehend [22].

2.1.8. Sitemap and RSS Feed

A sitemap is a file that furnishes information about a website's pages and their interconnections [4]. This, in turn, aids search engines in comprehending the crucial pages on the site, their update frequency, and the availability of alternate language versions [19]. Sitemaps are especially crucial for expansive websites with copious pages, as they enable search engine crawlers to navigate the site proficiently [4,19]. Multiple formats are supported for sitemaps, including XML, RSS, and plain text, with XML being the most prevalent. It is advisable for web administrators to submit their sitemap to search engine submission tools to expedite the indexing process [5].

The utilization of an RSS feed is a crucial determinant in enhancing a website's search engine visibility [7,23]. The primary purpose of RSS feeds is to notify visitors about newly added content on a website, and they are also utilized by search engine bots to expedite the discovery of new content.

Unlike static XML sitemaps, the RSS feed offers a more dynamic portrayal of the website's internal linking structure.

2.1.9. Robots.txt

The robot exclusion protocol, of which the robots.txt file is a component, delineates a roster of website URLs that ought not to be accessible to search engine crawlers [4]. This encompasses URLs such as administrative panels on websites constructed with Content Management Systems, as well as other pages that crawlers should avoid accessing. By enlisting URLs in the robots.txt file, search engines are notified that these pages should not be accessed and should not be included in search results. Nevertheless, there may be instances in which certain search engine crawlers disregard the robots.txt protocol [24].

2.1.10. Responsive and Mobile-Friendly Design

Roughly 50% of individuals who access the internet employ mobile devices during the information-seeking process [24]. Among these users, around two-thirds ultimately make a purchase. This trend has resulted in the prevalence of smartphones outpacing that of personal computers in many countries [4]. It is crucial for websites to be optimized for mobile use in order to maintain an online presence in the face of this increasing user cohort [4]. Google has developed a mobile-friendly test tool to assess whether a website is compatible with mobile devices by scrutinizing its constituent elements. CSS frameworks such as Bootstrap, UIKIT, Materialize, Bulma, and others that exhibit responsive design offer various design templates that are compatible with all devices [14].

2.1.11. Webpage Speed and Loading Time

The speed with which a webpage loads plays a pivotal role in the ranking of search engines and is of utmost importance in obtaining superior outcomes in the SERPs [26]. The speed of a webpage is regarded as one of the fundamental components of a website by search engines. Several tools, including GTMetrix, Google Lighthouse, SiteAnalyzer, and Pingdom, allow website owners and users to perform speed assessments on websites [27].

Google has created a pair of tools, namely Lighthouse and PageSpeed Insights, to facilitate website administrators in the monitoring and improvement of website efficacy. These tools gather data from a website and generate a performance rating, as well as an approximation of potential savings. The crucial divergence between these tools rests in the fact that Lighthouse utilizes laboratory data to evaluate website performance on a singular device and a predetermined set of network scenarios, whereas PageSpeed Insights employs both laboratory and field data to evaluate website performance across a diverse range of devices and real-world circumstances.

After assessing the performance score generated by the Lighthouse and PageSpeed Insights tools, website administrators can implement recommended modifications to enhance website performance [28]. Similar techniques are also utilized by the Pingdom tool [29].

2.1.12. SSL certificates and HTTPS

The utilization of HTTPS (Hypertext Transfer Protocol Secure) as a communication protocol on the internet is aimed at ensuring the confidentiality and integrity of data transmission between users' computers and websites. It is expected that users have a secure and private online experience while accessing websites [4]. Search engines prioritize websites with SSL certificates in their SERPs (Search Engine Results Pages) for security reasons [27]. Additionally, Google Chrome Browser underscores the importance of SSL certificates by alerting users with warnings during website navigation, indicating the security status of the website [30].

2.1.13. Accelerated Mobile Pages (AMP)

The proliferation of mobile devices has led to a significant increase in mobile users, surpassing 7.7 billion by the close of 2017 [31]. To optimize the mobile browsing experience, Google introduced the Accelerated Mobile Project (AMP) [32]. AMP eligibility is determined by Google and the source code of qualifying pages is cached on Google's web servers [33]. Mobile users who click on an AMP search result are immediately served the cached copy from Google's server, eliminating network delays [4]. Consequently, mobile users can access desired information swiftly. Web pages that adhere to AMP criteria earn higher rankings in mobile SERPs [32]. Nevertheless, conforming to AMP standards is arduous [31,34], despite the efficacy of the AMP technology in enhancing both web page performance and search engine ratings.

2.1.14. Structured Data

The collaborative community of Schema.org is devoted to the creation, maintenance, and promotion of structured data schemas for use on the internet, including web pages and email messages, among other applications [35,36]. Structured data provides a standardized format for conveying information about a website and organizing its content. Schema.org vocabulary can be employed with various encodings, including Microdata, JSON-LD and RDFa [14]. By integrating schema markup into a website's HTML source code through semantic annotations, search engines can comprehend the meaning of content fragments and furnish users with enriched information in search results [37]. The use of markup formats is increasingly prevalent in the semantic web, largely due to the endorsement of major search engines [38]. Google incentivizes the utilization of structured data by presenting rich results or snippets in SERPs, including information about product pricing and availability [1].

2.1.15. Open Graph Protocol (OGP)

OGP, which stands for Open Graph Protocol, is a type of structured data that has been developed by Facebook. This protocol enables external content to be seamlessly integrated into the social media platform [38]. With the use of OGP, any web page can be transformed into a rich object within a social graph, thereby providing it with the same level of functionality as any other object found on Facebook [39]. Whenever a link to a website is shared on social media, bots on the platform will typically look for three important elements on the webpage: its title, the image, and a brief summary or description. By pre-marking these elements using OGP, web administrators can facilitate the display of desired results by social media bots. OGP, like other structured data formats, provides a more enriched experience for both search engines and social media users, which can positively impact their decision to visit the website [14].

2.2. Off-page SEO techniques

Off-page search engine optimization (SEO) techniques encompass ranking factors that are external to a web page's content and are susceptible to various external influences [2,27]. The strategy employed for off-page optimization involves creating backlinks on other credible websites, which can enhance the authority of the website's domain and pages [5]. The most critical factor in off-page SEO is the quantity and quality of backlinks to a website, which can be accomplished by generating valuable content that people are likely to link to [8]. Search engines consider links as votes, and web pages with numerous links are frequently positioned higher in search results [3]. A variety of off-page techniques, such as profile backlinks, guest posts, comment backlinks, and Q&A, can influence a website's authority and position in search results [8]. According to Google's founders in their 1998 publication, anchored links comprise the target keyword of the destination website, which can provide a more precise description of web pages than the pages themselves [40]. To rank higher in search results, the target keyword must be utilized in both on-page and off-page pages [3]. Although off-page SEO techniques can have a positive impact on website ranking, search engines warn website administrators that creating backlinks to manipulate search engine rankings could lead to the website being removed from search results [1].

SEO professionals utilize the domain authority (DA) metric to simulate a website's ranking based on its perceived significance. Developed by MOZ, this metric ranges between DA1 and DA100, with higher values indicating greater website importance [41]. Backlinks from high DA websites are more likely to result in preferential treatment by search engines in their rankings. It is important to note that search engines do not rely on the DA metric to determine web page placement in search results. Rather, the DA metric serves as a simulation tool utilized by SEO experts to model the algorithms utilized by search engines.

3. Materials and Methods

In this chapter, the strategies and methodologies employed in the development of the SEO Audit software are introduced. Furthermore, a comprehensive depiction is provided regarding the Python packages, external APIs, and machine-learning techniques utilized during the software development process. It is noteworthy to mention that the software developed for the purposes of this article is openly accessible as an open-source repository on GitHub [42].

3.1. SEO Tool Functionality and APIs

The current article presents an SEO tool that necessitates the provision of a live website's URL and a target keyword by the user, with the objective of attaining a first-page ranking for the specified keyword. To attain such a ranking, both on-page and off-page SEO techniques must be superior to or at least comparable with those employed by competitors. The SEO tool, therefore, adopts a methodology that entails the detection of competing websites for a specific keyword, followed by a comprehensive competitor analysis of their on-page SEO techniques. The tool then employs free APIs and machine learning to identify competitors' off-page techniques. Once these data are obtained, the SEO tool scans the user's website to identify both on-page and off-page SEO techniques following a similar methodology. Equipped with knowledge regarding the strategies employed by competing entities, the SEO tool proffers SEO recommendations for the website at hand. Implementation of these suggested solutions has the potential to facilitate a first-page ranking and increase the target website’s traffic.

Sections 3.1.1 to 3.1.3 present the methodology and rationale for implementing the SEO tool.

3.1.1. Retrieve Google Search Results (SERPs)

The intention was to utilize a software program for scraping Google's search results. However, while developing the corresponding method, it was discovered that multiple requests from the same IP were detectable by Google's search engine, leading to the software being blocked after a few requests. To mitigate this challenge, proxies were integrated into the SEO tool, allowing for the rotation of IP addresses at regular intervals. While the free-proxy package [43] was functional, it primarily utilized free public proxies often already blocked by Google. The purchase of premium proxies was considered but conflicted with the article's principles as an open-source solution.

Consequently, the ZenSerp API [44] was employed, which retrieves Google search results in JSON format through an API request. Users can access up to 50 API requests per month without a subscription by creating an account. Upon receiving a target keyword from the user, the SEO tool utilizes the ZenSerp API to make a request, resulting in the first-page search results being returned in a JSON-structured format. These results are then processed and stored in a Python dictionary.

3.1.2. SEO Techniques Methods and Python Packages

The code employs several libraries to enable its functionalities.

- The `requests` library is utilized to make API requests, while the `JSON` library is used to convert JSON data into a Python dictionary.
- The `urllib.parse` library is used to parse the URL in the `get_robots(URL)` function.

- The `re` library, which supports regular expressions, is also used in the code.
- The `csv` library is used to save the dictionary to a CSV file.
- The BeautifulSoup library is also employed for web scraping purposes, allowing the code to extract data from HTML files.
- Finally, the `Mozscape` library obtains a website's domain authority.

These libraries enable the code to perform various tasks, from making API requests and parsing URLs to scraping websites and retrieving data on domain authority.

The SEO audit software described in Section 3.1.1 comprehensively analyzes each website listed in the dictionary. To achieve this, the tool employs a web scraping technique that involves searching the website's source code for on-page SEO techniques. The tool utilizes 24 individual methods (a total of 9 classes) to facilitate this process, each designed to identify a specific on-page SEO technique within the code. A comprehensive list of these techniques and associated methods are provided in Appendix A, **Table 4**. The tool's ability to identify and analyze on-page SEO techniques plays a critical role in its effectiveness in proposing solutions for website optimization and ranking improvement.

3.1.3. External APIs

To enhance the functionality of the SEO tool, four APIs were integrated into its code to perform a thorough analysis of the targeted website. These APIs include the PageSpeed Insights API, Mobile-Friendly Test Tool API, Mozscape API, and the Google SERP API. By utilizing these APIs, the SEO tool can provide valuable insights into different aspects of the website's performance, including its mobile usability, page speed, search engine ranking, and domain authority. Integrating these APIs helps the SEO tool perform a deep analysis of the website and identify areas for improvement in terms of on-page and off-page SEO techniques. Additionally, using these APIs allows the SEO tool to provide specific recommendations on addressing any issues found during the analysis. Overall, integrating these APIs dramatically enhances the functionality and effectiveness of the SEO tool, enabling website owners to optimize their website's performance and improve its visibility on search engine result pages.

- Mobile-Friendly Test Tool API, developed by Google, is a web-based service verifying a URL for mobile-usability issues. Specifically, the API assesses the URL against responsive design techniques and identifies any problems that could impact users visiting the page on a mobile device. The assessment results are then presented to the user in the form of a list, allowing for targeted optimizations to improve the website's mobile-friendliness [25].
- PageSpeed Insights API, developed by Google, is a web-based tool designed to measure the performance of a given web page. The API provides users with a comprehensive analysis of the page's performance, including metrics related to page speed, accessibility, and SEO. The tool can identify potential performance issues and return suggestions on optimizing the page's performance. This allows website owners to make informed decisions regarding optimizations that can be made to improve user experience and overall page performance [45].
- Mozscape API, The Mozscape API, developed by MOZ, is a web-based service that provides accurate metrics related to a website's performance. Specifically, the API takes a website's URL as an input and returns a range of metrics, including Domain Authority. Domain Authority is a proprietary metric developed by MOZ that measures the strength of a website's overall link profile. The metric is calculated using a complex algorithm that considers various factors, such as the quality and quantity of inbound links. It objectively assesses a website's authority relative to its competitors. The Mozscape API is a valuable tool for website owners and SEO professionals seeking insights into their website's performance and improving their overall search engine rankings [41].
- Google SERP API, developed by ZenSerp API, is a web-based service that allows users to efficiently and accurately scrape search results from Google. The API is designed to provide users with a seamless experience, offering features such as rotating IP addresses to prevent

detection and blocking by Google and returning search results in a JSON-structured format. The tool is a valuable asset for SEO professionals and website owners seeking to gain insights into their website's performance and improve their search engine rankings. The Google SERP API developed by ZenSerp API is an efficient and reliable tool for web scraping, offering accurate search results and facilitating the process of data analysis and SEO optimization [44].

3.2. Machine Learning

Two additional factors that must be considered for ranking a website in searches, as mentioned in section 2.2, are the number of backlinks and linking domains. Backlinks refer to the number of links from third-party pages that point to the target page, while linking domains refer to the number of unique domain names that point to the target page.

In the third subsection of section 3.1, the Mozscape API was utilized to acquire the domain authority (DA) of both the user's webpage and its competitors. However, the DA is computed through intricate algorithms that consider the authority of each backlink directed toward the page. Gathering, collating, and storing links for each website incur significant costs for the providers, amounting to millions of dollars. Consequently, these companies offer limited information regarding the number of backlinks and linking domains only for individual searches rather than in an API format. While collecting this information through an API is feasible, it often involves a cost exceeding $3,000 per year and limitations on the Rows returned by the API.

Rather than subscribing to an external service, our approach involved developing two prediction models employing machine learning techniques. These models were designed to forecast the number of backlinks and linking domains based on the Domain Authority. To train these models, we manually collected data on the domain authority, the number of backlinks, and the number of linking domains for a sample of 150 live websites selected from the DMoz Open Directory Project (ODP). The data was compiled in a CSV file and utilized for model training. Subsequently, pre-trained files were generated from the trained models to enable the SEO tool to expedite predictions. Nevertheless, we must highlight that the accuracy of the model's predictions could be enhanced with a larger and more representative sample of websites.

3.2.1. Model Training

Appendix B **Figure 2** depicts the methodology employed for model training and the generation of pre-trained models employed by the SEO software. The implementation relies on the utilization of three distinct libraries.

1. `Pandas` is a widely-used Python library that offers an extensive array of functions for manipulating and analyzing data. Its functionalities include support for data frames and series, which enables structured data processing [46]. The software employs pandas to read the data from a CSV file, preprocess it, and create new data frames to store the independent and dependent variables.
2. `Scikit-learn` is a machine-learning Python library that offers a diverse set of tools for data analysis and modeling [47]. In this software, scikit-learn is employed to train and evaluate the Random Forest Regression models.
    a. `RandomForestRegressor` is a class implemented in the scikit-learn library that embodies the Random Forest Regression algorithm [47]. This class is utilized in the code to train Random Forest Regression models, which are utilized to make predictions of the dependent variables, namely Backlinks and LinkingDomains, based on the independent variable, DA.
    b. `train_test_split` is a function incorporated in the scikit-learn library, which is employed to partition the dataset into training and testing sets [47]. This function randomly splits the data into two separate subsets, where one is used for training the machine learning model, and at the same time, the other is utilized for testing its performance.

   c. The `mean_squared_error` function in scikit-learn is a mathematical function that calculates the mean squared error (MSE) between the actual and predicted values of the dependent variable. In the context of the presented software, this function is used to evaluate the performance of the trained Random Forest Regression models on the testing data. It measures the average squared difference between the actual and predicted values, where a lower MSE indicates a better fit of the model to the data [47].

3. `Joblib` is a Python library that provides tools for the efficient serialization and deserialization of Python objects [48]. In this software, it is used to store the trained models on disk and retrieve them later to make predictions on new data. By storing the models as files, the trained models can be shared and used in other applications without retraining. This also enables efficient storage and retrieval of models, which can be especially useful for larger models requiring significant computational resources.

During the model training phase, the first step involves loading a CSV file named 'Dataset.csv' into a pandas DataFrame object 'data'. Subsequently, the independent variable 'DA' and dependent variables 'Backlinks' and 'LinkingDomains' are extracted from 'data' and stored in separate pandas series labeled as 'X', 'y1', and 'y2', respectively.

Next, data is split into training and testing sets utilizing the `train_test_split` function of scikit-learn. Two Random Forest Regression models are then trained using the training data, one to predict 'Backlinks' (rf1) and the other to predict 'LinkingDomains' (rf2). Both models are trained using 100 trees and a random state of 42.

The trained models' performance is then assessed on the testing data, utilizing the mean squared error metric, which measures the average squared difference between the predicted and actual values of the dependent variable. The resulting mean squared error values are stored in 'mse1' and 'mse2' variables and printed to the console.

Finally, the pre-trained models are saved to disk using the `joblib.dump()` function. The function generates two separate files, named 'da-to-backlinks.joblib' and 'da-to-linking-domains.joblib', which will be utilized in production mode for making predictions.

The `ipynb` files for model training, as well as the CSV and `joblib` pre-trained files, are available on GitHub [42].

3.2.2. Predictions based on DA

In the Prediction Phase, a Python class called "Predict" is defined (Appendix B **Figure 3**), consisting of two methods that take a value of "DA" as input and use pre-trained machine learning models to make predictions. The pre-trained models are loaded from disk using the `joblib.load` function and stored in the variables rf1 and rf2. The `predict_backlinks` and `predict_linking_domains` methods utilize the rf1 and rf2 models, respectively, to predict the values for Backlinks and LinkingDomains for the new data point. The resulting predicted values for Backlinks and LinkingDomains are returned to the SEOTechniques file for further use.

4. Results

Chapter 2 provides a concise and informative analysis of the on-page and off-page search engine optimization (SEO) techniques implemented by the tool, officially sanctioned by the Webmaster guidelines of the Google Search Central Documentation [4]. Additionally, Chapter Three presents a comprehensive explanation regarding the development of the Python-based SEO tool, emphasizing the underlying processes. The tool's functionality encompasses an in-depth SEO analysis of the website's source code and competitor websites for a given keyword. The integration of freely available APIs was employed to expand its capabilities, while the incorporation of machine learning techniques facilitated the prediction of off-page SEO techniques and essential metrics.

In this chapter, the study undertook a comprehensive analysis of the competitive environment of an online e-commerce website using the SEO software. The principal aim was to propose additional SEO techniques to be integrated into the website's source code, thereby augmenting its visibility and positioning within search engine rankings for a given keyword.

The software initially requested the URL of the target website and the target keyword. Utilizing the ZenSerp API, it retrieved the organic search results from Google using the target keyword as the search term. For each of the eight competitive websites (**Table 1** C1 to C8), it identified the on-page SEO techniques and technologies they employed. Simultaneously, by making API requests to the Mobile-Friendly Test Tool, PageSpeed Insights, and Mozscape APIs, it extracted data related to the responsive SEO technique, web page speed, and Domain Authority for each website. Additionally, leveraging the pre-trained models and utilizing the Domain Authority data, predictions were made for each website's number of Backlinks and Linking Domains.

A similar procedure was employed for the URL of the provided e-commerce website (**Table 1** U). Subsequent to the successful execution of the script, three CSV files were generated, each corresponding to **Table 1**, **Table 2**, and **Table 3** for presentation and analysis.

Table 1 showcases the SEO techniques along the y-axis alongside the respective ranking positions of the competitive websites within the organic search results for the specified keyword. The x-axis encompasses the eight competitive websites (C1-C8) and the URL of the e-commerce website (U). A check symbol denotes the implementation of specific SEO techniques on each corresponding webpage.

Drawing insights from the data presented in **Table 1** and **Table 2**, the software proposes the adoption of SEO techniques commonly utilized by the majority of competitors but currently absent from the e-commerce website's implementation.

In contrast, **Table 3** provides recommendations for the implementation of previously unutilized techniques on the website. It offers specific code snippets that require correction and accompanying informative text explaining each pertinent SEO rule. Due to the comprehensive nature of the suggestions generated by the SEO tool for the website, a concise representation is chosen, wherein only one recommendation out of the twelve proposed by the SEO tool is presented.

Table 1. SEO Analysis for Competitors' and User's Websites (report-seo_competitors_table.csv)

SEO Techniques/ Page	C1	C2	C3	C4	C5	C6	C7	C8	U
position	1	2	3	4	5	6	7	8	0
amp									
images_alt	✓			✓	✓		✓		✓
links_title	✓						✓		
heading1									
heading2									
title									
meta_description									
opengraph			✓			✓		✓	
style	✓						✓		
sitemap									
rss									
script	✓						✓		
json_ld			✓	✓				✓	
inline_css	✓						✓		
microdata					✓				
rdfa									
robots		✓		✓	✓	✓		✓	
gzip		✓	✓			✓		✓	
web_ssl	✓		✓	✓	✓	✓	✓	✓	✓
seo_friendly_url		✓	✓	✓	✓	✓		✓	✓
speed	2.9	1.4	1.5	0.4	3.8	1.7	1.9	1.3	2.8
responsive	✓	✓	✓	✓	✓	✓	✓	✓	✓
da	10	10	20	16	19	5	18	23	4
backlinks	1191	1191	17324	2908	17324	6	3261	6054	4
linking_domains	84	84	211	119	211	4	214	183	3

[C1] pasxalineslampades.com, [C2] toolittle.gr, [C3] armoniecandles.com, [C4] tsago.gr, [C5] e-gerakis.gr, [C6] dekor.gr, [C7] nuovavita.gr, [C8] keri.gr, [U] messiniancandles.gr

Table 2. SEO recommendations based on Competition (report-suggestions_based_on_competition.csv)

Missing SEO Technique	Comments	SEO Rules
gzip	It is suggested to use the [gzip] SEO technique, which the majority of the competition have also applied.	- Google Developers - Optimize Encoding and Transfer Size of Text-Based Assets: https://developers.google.com/web/fundamentals/performance/optimizing-content-efficiency/optimize-encoding-and-transfer - Mozilla Developer Network - HTTP compression: https://developer.mozilla.org/en-US/docs/Web/HTTP/Compression - GTmetrix - What is Gzip Compression?: https://gtmetrix.com/enable-gzip-compression.html - BetterExplained - The Importance Of Gzipping Your Website: https://betterexplained.com/articles/how-to-optimize-your-site-with-gzip-compression/

Table 3. SEO recommendations based on SEO tool (report-seo_tool_suggestions.csv)

Missing SEO Technique	Comments	Data	SEO Rules
links_title	The links on the WebPage are missing title tags. It is recommended to add title tags to all links for better user experience and SEO optimization.	{'/service/xondriki-polisi': '', 'product/bazo-130-ml-451': '', 'product/bazaki-me-fello-524': '', 'product/bazaki-me-fello': '', 'product/bazo-130ml-396': '', 'product/lampada-gamou-f12cm-20-cm': '', 'product/pasxalini-lampada-strogguli-krakele-mob': '', 'product/pasxalini-lampada-strogguli-krakele-prasini': '', 'product/lampada-gamou-tetragoni-masif-12cm12cm-22cm': '', 'product/parafinelaio': '', 'product/pasxalini-lampada-plake-xusti-sapio-milo': '', 'product/ekklisiastiko-keri-n1': '', 'product/kormos-apo-keri-f9cm95cm-455': '', 'product/parafini': '', 'product/keri-citronella-n1': '', 'product/premium-futika-keria-me-kapaki': '', 'product/potiri-vintage': '', 'product/potiri-stroggulo': '', 'product/kormos-apo-keri-9cm-9-cm-15cm': '', 'product/bazaki-me-fello-323': '', 'product/antipagotiko-keri': '', 'product/pasxalini-lampada-strogguli-mob': '', 'product/pasxalini-lampada-plake-krakele-prasini': '', 'product/lampada-baptisis-roz': '', 'product/akatergasto-prosanamma': '', 'product/pasxalini-lampada-strogguli-xusti-galazia': '', 'product/premium-futika-keria-me-kapaki-418': '', 'product/pasxalini-lampada-plake-xusti-galazia': '', 'product/pasxalini-lampada-plake-galazia': '', 'product/lampada-baptisis-galazia': '', 'product/set-kormon-apo-keri-339': '', 'product/ekklisiastiko-keri-n1-500gr': '', 'product/potiri-funky-xalkino': '', 'product/potiri-koukounara': ''}	- Example: <a href="..." title="..."> - According to Google's Search Engine Optimization (SEO) Starter Guide, adding descriptive title tags to your links can help both users and search engines understand the content of your pages. Google also considers title tags as a ranking factor for search results. - The Web Content Accessibility Guidelines (WCAG) also recommend using descriptive text for links, including title attributes, to ensure accessibility for people with disabilities. - Moz, a leading SEO software provider, also recommends using descriptive title tags for links as a best practice for SEO optimization.

By adhering to the recommendations outlined by the SEO tool and incorporating a total of 17,324 backlinks originating from 211 distinct referring/linking domains, the e-commerce website achieved a notable placement on the first page of search results for the targeted keyword. Specifically, the website secured the fifth position in the search rankings.

Prior to the implementation of the SEO tool's recommendations on February 26, 2023, the specific e-commerce website under consideration only managed to attract a meager daily average of ten organic visitors. To visualize the website's traffic patterns, **Figure 1** presents the traffic data recorded by Google Analytics, spanning from February 26, 2023, to May 20, 2023. Remarkably, within a span of 85 days, subsequent to incorporating the suggested modifications proposed by the SEO tool, a substantial increase in traffic occurred, with an average of 143 additional visitors per day. Out of the total 6,270 visitors, the breakdown is as follows: 4,518 originated from organic search, 1,431 from direct traffic, 316 from paid traffic, and five from referral traffic.

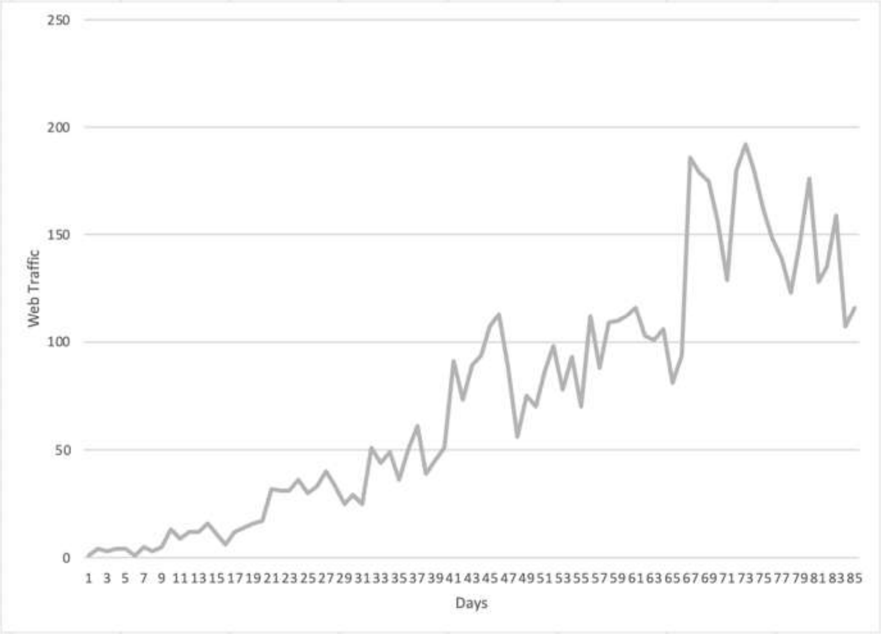


Figure 1. Google Analytics. Traffic trend from February 26, 2023, to May 20, 2023.

5. Conclusions

In conclusion, the development of an open-source Python-based SEO audit software has the potential to provide users and non-profit organizations with comparable functions to commercial SEO audit tools at no cost, thereby making it more accessible to a broader audience. By analyzing the source code of websites and their competitors and utilizing machine learning to predict off-page SEO techniques and critical metrics, the tool can provide recommendations for optimizing websites and achieving improved search rankings and traffic. The analysis of a live e-commerce website demonstrated the tool's effectiveness in significantly increasing daily organic visitors. The findings of this study highlight the importance and value of developing open-source tools for businesses and individuals seeking to improve their online presence and visibility.

Author Contributions: Conceptualization, K.I.R. and N.D.T.; methodology, K.I.R. and N.D.T..; software, K.I.R.; validation, K.I.R. and N.D.T.; formal analysis, K.I.R. and N.D.T..; investigation, K.I.R. and N.D.T..; resources, K.I.R.; data curation, K.I.R.; writing—original draft preparation, K.I.R. and N.D.T.; writing—review and editing, K.I.R. and, N.D.T.; visualization, K.I.R.; supervision, N.D.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Informed Consent Statement: Not applicable.

Data Availability Statement: Software source code and data supporting reported results can be found at https://github.com/rkonstadinos/python-based-seo-audit-tool (accessed on 12 May 2023).

Conflicts of Interest: The authors declare no conflict of interest.

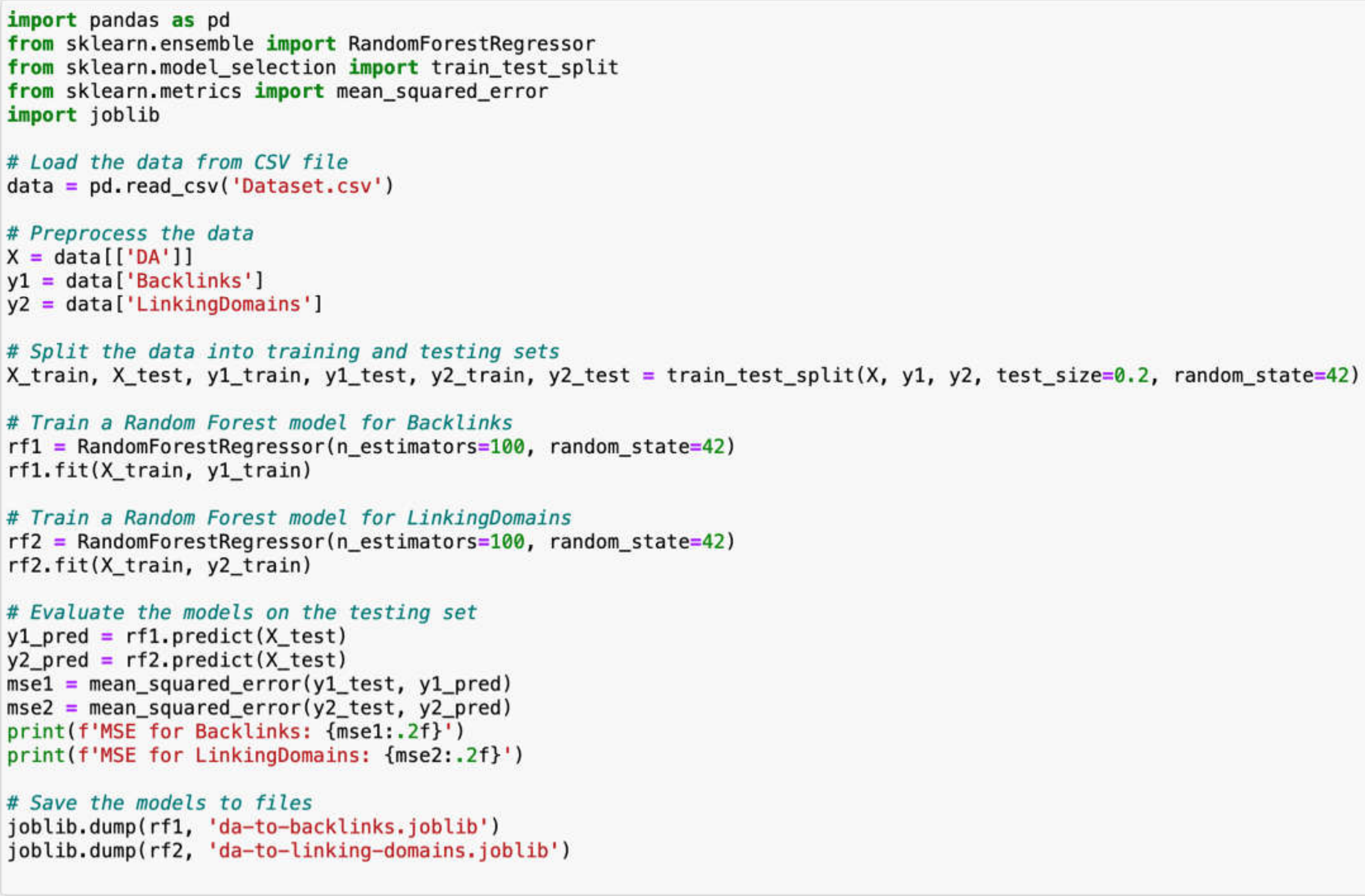
Appendix A

Table 4. Methods and their corresponding SEO Techniques

Method	SEO Technique	Description
`perform_seo_checks`	-	The method uses the `requests` method to get the website's source code and the `BeautifulSoup` library as `html.parser` to parse the code. Finally, it utilizes al the methods that detect SEO techniques by returning th results.
`get_organic_serps`	-	The method utilizes the ZenSerp application programming interface (API) to retrieve a comprehensive list of websites that are listed on the firs page of search results in response to a given keyword query.
`get_image_alt`	Image Alternative Attribute	The method is designed to identify and flag instances o missing alternative attributes for images from a provided list.
`get_links_title`	Link Title Attribute	The method involves the identification of missing title attributes within a list of links.
`get_h_text`	Heading 1 and 2 Tags	The method detects and counts the occurrences of H1 and H2 tags within the source code, followed by a search for the specified target keyword within these tags.
`get_title_text`	Title Tag	The method conducts a search for the presence of the title tag within the source code and verifies the existenc of the target keyword in conjunction with the tag.
`get_meta_description`	Meta Description	The method conducts a search for the presence of the meta description within the source code and verifies th existence of the target keyword in conjunction with the tag.
`get_meta_opengraph`	Opegraph	The method conducts a search for the presence of the opengraph tag within the source code and verifies the existence of the target keyword in conjunction with the tag.
`get_meta_responsive`	Responsive Tag	The method conducts a search for the presence of the viewport within the source code and verifies the existence of the target keyword in conjunction with the tag.
`get_style_list`	Minified CSS	The method identifies the stylesheets present in the source code and examines whether they have been minified.
`get_script_list`	Minified JS	The method identifies the scripts present in the source code and examines whether they have been minified.
`get_sitemap`	Sitemap	The method identifies the xml sitemap present in the source code.
`get_rss`	RSS	The method identifies the RSS feed present in the sourc code.
`get_json_ld`	JSON-LD structured data	The method identifies the JSON-LD present in the source code.
`get_item_type_flag`	Microdata structured data	The method identifies the Microdata present in the source code.
`get_rdfa_flag`	RDFA structured data	The method identifies RDFA present in the source code
`get_inline_css_flag`	In-line CSS	The method detects any in-line CSS code in source code

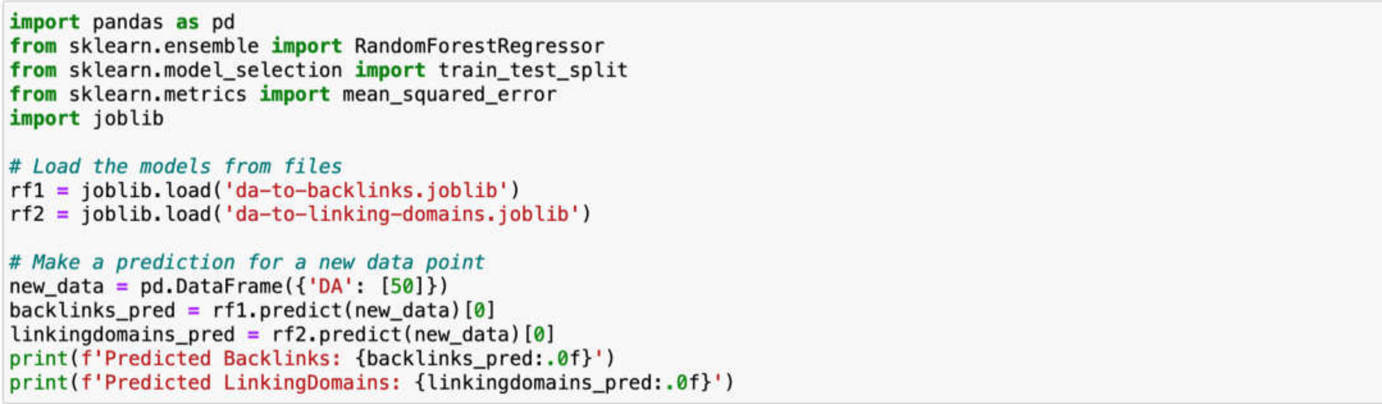
get_robots	Robots.txt	The method verifies the presence of a Robots.txt file in the root path of the website.
get_gzip	GZip	The method performs an evaluation to detect the presence of gzip Content-Encoding in response headers
get_web_ssl	SSL Certificates	The method performs an analysis to determine if the webpage has been provided with Secure Sockets Layer (SSL) certificates, which ensure that the connection between the user's browser and the website is encrypte and secure.
seo_friendly_url	SEO Friendly Url	The method examines whether the provided URL is optimized for search engine optimization (SEO), conforming to the best practices and guidelines.
get_speed	Loading Time	The method employs the Lighthouse API to measure th web page's loading time.
get_responsive_test	Responsive Design	The method employs the mobileFriendlyTest API to ascertain whether a given webpage is responsive, i.e., capable of rendering suitably on different devices and screen sizes.
get_da	Domain Authority	The method employs the MOZ API to obtain the Domain Authority (DA) of a website.

Appendix B



```python
import pandas as pd
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
import joblib

# Load the data from CSV file
data = pd.read_csv('Dataset.csv')

# Preprocess the data
X = data[['DA']]
y1 = data['Backlinks']
y2 = data['LinkingDomains']

# Split the data into training and testing sets
X_train, X_test, y1_train, y1_test, y2_train, y2_test = train_test_split(X, y1, y2, test_size=0.2, random_state=42)

# Train a Random Forest model for Backlinks
rf1 = RandomForestRegressor(n_estimators=100, random_state=42)
rf1.fit(X_train, y1_train)

# Train a Random Forest model for LinkingDomains
rf2 = RandomForestRegressor(n_estimators=100, random_state=42)
rf2.fit(X_train, y2_train)

# Evaluate the models on the testing set
y1_pred = rf1.predict(X_test)
y2_pred = rf2.predict(X_test)
mse1 = mean_squared_error(y1_test, y1_pred)
mse2 = mean_squared_error(y2_test, y2_pred)
print(f'MSE for Backlinks: {mse1:.2f}')
print(f'MSE for LinkingDomains: {mse2:.2f}')

# Save the models to files
joblib.dump(rf1, 'da-to-backlinks.joblib')
joblib.dump(rf2, 'da-to-linking-domains.joblib')
```

Figure 2. Train Model Phase: Domain Authority to Backlinks and Linking Domain



```
import pandas as pd
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import train_test_split
from sklearn.metrics import mean_squared_error
import joblib

# Load the models from files
rf1 = joblib.load('da-to-backlinks.joblib')
rf2 = joblib.load('da-to-linking-domains.joblib')

# Make a prediction for a new data point
new_data = pd.DataFrame({'DA': [50]})
backlinks_pred = rf1.predict(new_data)[0]
linkingdomains_pred = rf2.predict(new_data)[0]
print(f'Predicted Backlinks: {backlinks_pred:.0f}')
print(f'Predicted LinkingDomains: {linkingdomains_pred:.0f}')
```

Figure 3. Make Predictions Phase: Domain Authority to Backlinks and Linking Domain

References

1. Roumeliotis, K.I.; Tselikas, N.D. An effective SEO techniques and technologies guide-map. Journal of Web Engineering 2022, 21(5), 1603–1650.
2. Roumeliotis, K.I.; Tselikas, N.D.; Nasiopoulos, D.K. Airlines' Sustainability Study Based on Search Engine Optimization Techniques and Technologies. Sustainability 2022, 14, 11225. [CrossRef]
3. Matoševic, G.; Dobša, J.; Mladenic, D. Using Machine Learning for Web Page Classification in Search Engine Optimization. Future Internet 2021, 13, 9. [CrossRef]
4. Webmaster Guidelines, Google Search Central, Google Developers. Available online: https://developers.google.com/search/docs/advanced/guidelines/webmaster-guidelines (accessed on 12 May 2023).
5. Luh, C.-J.; Yang, S.-A.; Huang, T.-L.D. Estimating Google's search engine ranking function from a search engine optimization perspective. Online Inf. Rev. 2016, 40, 239–255. [CrossRef]
6. Iqbal, M.; Khalid, M. Search Engine Optimization (SEO): A Study of important key factors in achieving a better Search Engine Result Page (SERP) Position. Sukkur IBA Journal of Computing and Mathematical Sciences-SJCMS, 2022, 6, 1-15. [CrossRef]
7. Ziakis, C.; Vlachopoulou, M.; Kyrkoudis, T.; Karagkiozidou, M. Important Factors for Improving Google Search Rank. Future Internet 2019, 11, 32. [CrossRef]
8. Patil, V.M.; Patil, A.V. SEO: On-Page + Off-Page Analysis. In Proceedings of the International Conference on Information, Communication, Engineering and Technology (ICICET), Pune, India, 29–31 August 2018. [CrossRef]
9. Kumar, G.; Paul, R.K. Literature Review on On-Page & Off-Page SEO for Ranking Purpose. United Int. J. Res. Technol. (UIJRT) 2020, 1, 30–34. ISSN: 2582-6832
10. Fuxue, W.; Yi, Li; Yiwen, Z. An empirical study on the search engine optimization technique and its outcomes. In Proceedings of the 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC), Zhengzhou, China, 8–10 August 2011. [CrossRef]
11. (Meta) Title Tags + Title Length Checker [2021 SEO]–Moz. Available online: https://moz.com/learn/seo/title-tag (accessed on 12 May 2023).
12. Van, T.L.; Minh, D.P.; Le Dinh, T. Identification of paths and parameters in RESTful URLs for the detection of web Attacks. In Proceedings of the 4th NAFOSTED Conference on Information and Computer Science, Hanoi, Vietnam, 24–25 November 2017. [CrossRef]
13. Rovira, C.; Codina, L.; Lopezosa, C. Language Bias in the Google Scholar Ranking Algorithm. Future Internet 2021, 13, 31. [CrossRef]
14. Roumeliotis, K.I.; Tselikas, N.D. Search Engine Optimization Techniques: The Story of an Old-Fashioned Website. Business Intelligence and Modelling. IC-BIM 2019. Paris, France, 12–14 September 2019; Springer Book Series in Business and Eco-nomics. Springer: Cham, Switzerland, 2019. [CrossRef]
15. URL Structure [2021 SEO]–Moz SEO Learning Center. Available online: https://moz.com/learn/seo/url (accessed on 12 May 2023).
16. An Image Format for the Web∣WebP∣Google Developers. Available online: https://developers.google.com/speed/webp (ac-cessed on 5 May 2022).
17. Hui, Z.; Shigang, Q.; Jinhua, L.; Jianli, C. Study on Website Search Engine Optimization. In Proceedings of the International Conference on Computer Science and Service System, Nanjing, China, 11–13 August 2012. [CrossRef]
18. Roumeliotis, K.I.; Tselikas, N.D. Evaluating Progressive Web App Accessibility for People with Disabilities. Network 2022, 2, 350-369. [CrossRef]

19. Zhang, S.; Cabage, N. Does SEO Matter? Increasing Classroom Blog Visibility through Search Engine Optimization. In Proceedings of the 47th Hawaii International, Conference on System Sciences, Wailea, HI, USA, 07-10 January 2013. [CrossRef]
20. All Standards and Drafts-W3C. Available online: https://www.w3.org/TR/ (accessed on 12 May 2023).
21. Shroff, P.H.; Chaudhary, S.R. Critical rendering path optimizations to reduce the web page loading time. In Proceedings of the 2nd International Conference for Convergence in Technology (I2CT), Mumbai, India, 7–9 April 2017. [CrossRef]
22. Tran, H.; Tran, N.; Nguyen, S.; Nguyen, H.; Nguyen, T.N. Recovering Variable Names for Minified Code with Usage Con-texts. In Proceedings of the IEEE/ACM 41st International Conference on Software Engineering (ICSE), Montreal, QC, Canada, 25–31 May 2019. [CrossRef]
23. Ma, D. Offering RSS Feeds: Does It Help to Gain Competitive Advantage? In Proceedings of the 42nd Hawaii International Conference on System Sciences, Waikoloa, HI, USA, 5–8 January 2009. [CrossRef]
24. Gudivada, V.N.; Rao, D.; Paris, J. Understanding Search-Engine Optimization. Computer 2015, 48, 43–52. [CrossRef]
25. Mobile-Friendly Test Tool. Available online: https://search.google.com/test/mobile-friendly (accessed on 12 May 2023).
26. MdSaidul, H.; Abeer, A.; Angelika, M.; Prasad, P.W.C.; Amr, E. Comprehensive Search Engine Optimization Model for Commercial Websites: Surgeon's Website in Sydney. J. Softw. 2018, 13, 43–56. [CrossRef]
27. Kaur, S.; Kaur, K.; Kaur, P. An Empirical Performance Evaluation of Universities Website. International J. Comput. Appl. 2016, 146, 10–16. [CrossRef]
28. Google Lighthouse. Available online: https://developers.google.com/web/tools/lighthouse (accessed on 12 May 2023).
29. Pingdom Website Speed Test. Available online: https://tools.pingdom.com/ (accessed on 12 May 2023).
30. Google Chrome Help. Available online: https://support.google.com/chrome/answer/95617?hl=en (accessed on 12 May 2023).
31. Jun, B.; Bustamante, F.; Whang, S.; Bischof, Z. AMP up your Mobile Web Experience: Characterizing the Impact of Google's Accelerated Mobile Project. In Proceedings of the MobiCom '19: The 25th Annual International Conference on Mobile Com-puting and Networking, Los Cabos, Mexico, 21-25 October, 2019. [CrossRef]
32. Roumeliotis, K.I., Tselikas, N.D. Accelerated Mobile Pages: A Comparative Study. Business Intelligence and Modelling. IC-BIM 2019. Paris, France, 12–14 September 2019; Springer Book Series in Business and Economics. Springer: Cham, Switzerland, 2019. [CrossRef]
33. Start building websites with AMP. Available online: https://amp.dev/documentation/ (accessed on 12 May 2023).
34. A. Phokeer. On the potential of Google AMP to promote local content in developing regions, 11th International Conference on Communication Systems & Networks (COMSNETS), 2019, pp. 80-87. [CrossRef].
35. Welcome to Schema.org. Available online: https://schema.org/ (accessed on 12 May 2023).
36. Guha, R.; Brickley, D.; MacBeth, S. Schema.org: Evolution of Structured Data on the Web: Big data makes common schemas even more necessary. Queue 2015, 13. 10–37. [CrossRef]
37. Navarrete, R.; Lujan-Mora, S. Microdata with Schema vocabulary: Improvement search results visualization of open eductional resources. In Proceedings of the 13th Iberian Conference on Information Systems and Technologies (CISTI), Caceres, Spain, 13–16 June 2018. [CrossRef]
38. Navarrete, R.; Luján-Mora, S. Use of embedded markup for semantic annotations in e-government and e-education websites. In Proceedings of the Fourth International Conference on eDemocracy & eGovernment (ICEDEG), Quito, Ecuador, 19–21 April 2017. [CrossRef]
39. The Open Graph protocol. Available online: https://ogp.me/ (accessed on 12 May 2023).
40. Brin, S.; Page, L. Reprint of: The anatomy of a large-scale hypertextual web search engine. Computer Networks 2012, 56, 3825–3833 [CrossRef].
41. Mozscape API. Available online: https://moz.com/products/api (accessed on 12 May 2023).
42. SEO Audit Software: Available online: https://github.com/rkonstadinos/python-based-seo-audit-tool (accessed on 12 May 2023).
43. Free Proxy Python Package: Available online: https://pypi.org/project/free-proxy/
44. ZenSerp API: Available online: https://zenserp.com/ (accessed on 12 May 2023).
45. Pagespeedapi Runpagespeed. Available online: https://developers.google.com/speed/docs/insights/v4/reference/ pagespeedapi/runpagespeed (accessed on 12 May 2023).
46. McKinney, W. Data Structures for Statistical Computing in Python. Proc. Python in Science Conference 2020, 9, 56-61.
47. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Vanderplas, J. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research 2011, 12(Oct), 2825-2830.

doi:10.20944/preprints202306.0309.v1

48. Joblib Development Team. Joblib: Running Python Functions as Pipeline Jobs. 2019. Available online: https://joblib.readthedocs.io/en/latest/ (accessed on 12 May 2023).