IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 25 April 2024, accepted 21 June 2024, date of publication 28 June 2024, date of current version 9 July 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3420734

RESEARCH ARTICLE

Automatic Regular Expression Generation for Extracting Relevant Image Data From Web Pages Using Genetic Algorithms

CANAN ASLANYÜREK[1] **AND TARIK YERLIKAYA**[2]
[1]Kırklareli University, 39100 Kırklareli, Türkiye
[2]Trakya University, 22030 Edirne, Türkiye
Corresponding author: Canan Aslanyürek (c.aslanyurek@klu.edu.tr)

ABSTRACT In this study, a method that automatically generates regular expressions using genetic algorithms is designed to extract relevant images on web pages. Data extraction, which is usually done with web scrapers, can also be done with regular expressions. The complexity of regular expressions and the fact that they require expert knowledge make their writing difficult. With this study, a regular expression is automatically created to obtain relevant images of news content on websites. With the principle of genetic algorithms, the survival of the good and the elimination of the bad, a regular expression that can reach the most relevant image is produced. Thus, instead of a time-consuming and error-prone method such as creating the appropriate pattern for each site with web scraper tools, automatic regular expression generation using genetic algorithm methods can be used as a better method. A data set containing text-based related and irrelevant images from 200 websites collected from 58 countries was used in the study. There are 22,682 relevant images among 635,015 image data in the dataset. With the method developed using the genetic algorithm, the rate of accessing the relevant images by regular expressions produced by only looking at the relevant image data is approximately 98.49%.

INDEX TERMS Genetic algorithms, automatic regular expressions, web data extraction, image data extraction.

I. INTRODUCTION

Nowadays, as technology develops, the amount of data shared in the virtual environment increases. Creating meaningful information using this data is very important. The process of obtaining this data in the web environment is called web extraction. Web data obtained through web data extraction can be used in different areas such as sentiment analysis [1], image search, data mining, fake news detection [2], [3]. Researchers have mostly worked to extract text-based data such as titles [4], main content [5], [6], comments [7], tables [8], and layouts [9] on web pages. When the literature is examined, it is seen that web image data scraping is not a subject that has been studied much. However, web image data is important data that informs users about the content of a web page. Web scraping tools are generally used to extract web data. Text data extraction is mostly carried out using web scraping tools. Moreover, for this inference process, the expert has to examine all the pages of the website from which she will make the inference and make an appropriate design (pattern) according to the coding made. Performing these procedures can be time consuming and challenging. If a suitable pattern is not created, a situation such as obtaining unwanted data may occur. For this, the expert may have to create new patterns. Another method that can be used when extracting web data is regular expressions. Regular expressions are a method used to access texts in a specific desired format. However, since writing regular expressions manually is complicated and difficult, it requires serious expert knowledge. To minimize these problems, it may be important to use approaches that provide automatic solutions that can be integrated into manual web scraping tools. In this

The associate editor coordinating the review of this manuscript and approving it for publication was Hai Dong.

study, a method that tries to automatically generate regular expressions using genetic algorithms is proposed to scrape images related to page content on web pages. In the proposed method, features such as HTML elements that make up web pages are important. These are properties such as text, height and width of html image elements. In our approach, text data with the mentioned features is used. In the study, a data set containing relevant and irrelevant images of different pages of different websites collected from various countries was used. Each image data in the data set is coded with 30 features. The aim of our work is to produce an automatic regular expression that allows accessing the relevant image for each site. Thus, the relevant data was accessed in a shorter time than other methods and the accuracy rate was increased. An attempt was made to reach the best solution using genetic algorithms. Genetic algorithms produce an expression that can reach the highest number of relevant image data in the possible solution set.

When we look at the previous studies in the literature, they focused on extracting web image data using text data, unlike the web data extraction technique using character-based genetic algorithms. It has been observed that the method uses HTML elements to automatically scrape data faster and easier than manual web scraper programs. Unlike other presented approaches, automatic regular expressions were created for each site in the data set using textual data without the need to download images. Thus, automatic creation of regular expressions that are difficult to write and require specialized knowledge provides both convenience and time. In addition, a small mistake that may occur in the writing of regular expressions with a complex structure can cause a mismatch. In our study, the possibility of this mismatch was significantly reduced by automatically creating regular expressions and using genetic algorithms in creating this expression.

The other sections of the paper are organized as follows: The second section provides a detailed overview of web scraping, regular expression usage, and the literature on genetic algorithms. The third section defines the problem of extracting relevant images from the web and briefly introduces genetic algorithms. The fourth section elaborates step-by-step on the genetic algorithm-based approach utilized in this study. Following that, the fifth section presents the dataset used, performance metrics, and experimental results. Finally, the last section is dedicated to the conclusion and future works.

II. RELATED STUDIES

When the studies on web data extraction are examined, there are various methods that enable manual or automatic extraction. A study examining content classification and template cleaning in web data extraction was conducted by San and Aye [10]. In their study, it was suggested that the template cleaning process would remove only relevant content from web pages. In a study conducted using machine learning methods to classify the content and noise in web pages, it was carried out in two different environments where it was not known whether a controlled and processed web page containing structured content documents had structured content or not [11].

One of the methods used in web data extraction is regular expressions. Because regular expressions are complex and require specialized knowledge, researchers have tried to design tools that automate it for web data extraction. The topic of learning a regular expression using correct matches within a list has been well researched [12], [13]. Creating regular expression patterns is one of the most well-known techniques for manually extracting data from web pages [14]. However, the preparation of these patterns is very error-prone [15]. In the study conducted by Li et al. [16], they aimed to reduce the effort of manual data scraping by using regular expressions. They proposed a transformation-based algorithm called ReLIE and compared their method with other algorithms. Another study investigating automatic regular expression generation was conducted using Grammatical Evolution. The work is based on a subset of POSIX regular expression rules. Tests have shown that the Grammatical Evolution approach is promising in automatic regular expression creation [17]. Another study that addressed the problem of extracting a regular expression from a given string of text that is very similar to the regular expression that a human expert would write to describe the language was done by Prasse et al. The aim of the study is to automate an e-mail service that uses regular expressions to describe and filter e-mail spam campaigns for those who undertake this task [18]. In a thesis based on automatic regular expression extraction, an automatic programming tool Vnet (Visual Network Data Extraction and Programming Tool) was introduced to help biologists retrieve specific web pages on the Internet and extract useful information from these pages [19]. In one study proposed an approach that can automatically extract regular expressions based on a set of positive examples [20]. In this study, which was carried out to reduce the cost and time in web data extraction using the DOM tree, an intuitive approach called REGEXN was introduced that can automatically create these patterns through CSS selectors [21]. In a study in which a web information retrieval, matching and structure extraction model was created based on a search engine, a regular expression of basic tags was designed by obtaining the standard mathematical expression of URLs by analyzing the URL addresses of the search results and the DOM tree structure of the web pages [22]. In a study that introduced a clustering-based approach to learn regular expressions on large alphabets in the context of noisy and structureless texts, a method was designed to automatically generate regular expressions that can effectively capture patterns in text data using clustering methods [23]. In studies using genetic algorithms in the automatic creation of regular expressions in web data extraction processes, better results

have been obtained due to the survival of the good ones and the elimination of the bad ones. In the study conducted by Bartoli and his colleagues in 2012, they investigated the automatic generation of regular expressions from examples with genetic programming [24]. In another study, experiments were conducted on extracting data such as phone numbers, product names, and dates with regular expressions produced using genetic algorithms [25]. In a later study by Bartoli and his team, a system was designed to automatically generate regular expressions for text extraction tasks. Genetic Programming approaches have been shown to perform successfully [26], [27]. In another study by Bartoli et al., an automatic entity extractor was designed in the form of a regular expression from the desired inference examples from an unstructured text with a genetic algorithm [28]. In the study conducted by Coddy - Kenny et al. in 2017, it was aimed to improve the performance in regular expressions by using a genetic algorithm. A benchmark suite of candidate regular expressions is proposed for improvement [29]. In a study by Barrero et al., an updated state assessment of the web data extraction problem was made and an evolutionary computation approach based on genetic algorithms and regular expressions was introduced [30]. In a study introducing an extension of the tool named Searchy, which is an agent-based system specialized in data extraction and integration, it has been demonstrated that Genetic Programming performs well in generating regular expressions to address challenges such as the difficulty of evaluating or coding certain parts of evolved regular expressions, and it is considered a promising approach to overcome issues related to linearity in expression construction [31].

The literature review indicates a general lack of extensive research on extracting image data from web pages. In this context, when studies on extracting web image data are examined, a research effort utilizing machine methods has been observed. In this study, relevant image data from web pages is automatically extracted using a generated regular expression, demonstrating a faster extraction of data compared to web scraping programs [32]. In another study conducted for automatically extracting data from shopping web pages using classification methods based on the features of relevant and irrelevant image data, different methods were compared. Within the study, it was observed that faster data extraction could be achieved by leveraging the advantages of both manual extraction processes and automated approaches. When examining the prediction accuracy rate, it has been concluded that the f-score is 0.980, indicating a high level of accuracy in the predictions [33]. In another study conducted by Uzun et al. [34], they successfully increased the f-score value in the automatic extraction of web image data using machine learning methods by employing the AdaBoost technique. Support Vector Machines (SVM) method was employed in the study, and certain parameters were defined to enhance the performance of the SVM method. In the conducted research, a f-score of 0.439 was achieved [35].

The mentioned methods were generally implemented using supervised approaches. Among studies that utilized unsupervised approaches, one can refer to the work conducted by Helfman and Holland. In this study, heuristic measures within the web page were used, and each image was assigned a score [36]. In another study, as an extension of this research, it was recommended that the image size should be more than 120,000 pixels. However, in a scoring system implemented in this way, it has been observed to be useful for selecting a single image from a group of images within a web page [37].

III. BACKGROUND

This section will commence with an overview of the pertinent location of relevant images within a webpage, along with providing background information on the addressed problem. Subsequently, the genetic algorithms to be employed in the solution will be introduced.

A. PROBLEM DEFINITION

The structure of each web page is composed of different patterns. Web pages consist of HTML tags and CSS elements that handle formatting. Images displayed on the web page are created with HTML tags. These images can be relevant or irrelevant. Relevant images are those that are related to the content of the page, while irrelevant images are pictures such as logos or advertisements that are not related to the content of the page. The aim of this study is to extract the relevant image data on web pages using an automatically generated regular expression through the use of genetic algorithms. In the extraction of these images, it is necessary to examine HTML contents such as image sizes and titles. An example of relevant and irrelevant images belonging to a web page is shown in Figure 1.

Figure 1 depicts an image related to a news content found on a news website. As seen, the image labeled as 1 in green is relevant and associated with the page content, while all the images labeled as 0 in red are irrelevant images independent of the page content.

These images are added to web pages using HTML tags. The styles of web pages created with HTML tags are determined by CSS elements. In our study, the dataset comprises the 'img' tag along with its two parent tags extracted from textual data, as illustrated in Figure 1. A method has been developed within the study to automatically extract pertinent images from web pages, which are composed of these elements, utilizing genetic algorithms to generate automatic regular expressions. The objective of our approach is to easily obtain these relevant images with the highest accuracy.

The goal is to achieve access to a regular expression generated by using genetic algorithms that enables the extraction of the highest percentage of relevant images. It has been observed that in previous studies on web data extraction, there has been a focus on text data rather than image data. n a similar study, it was observed that the processing time was prolonged because character-based data

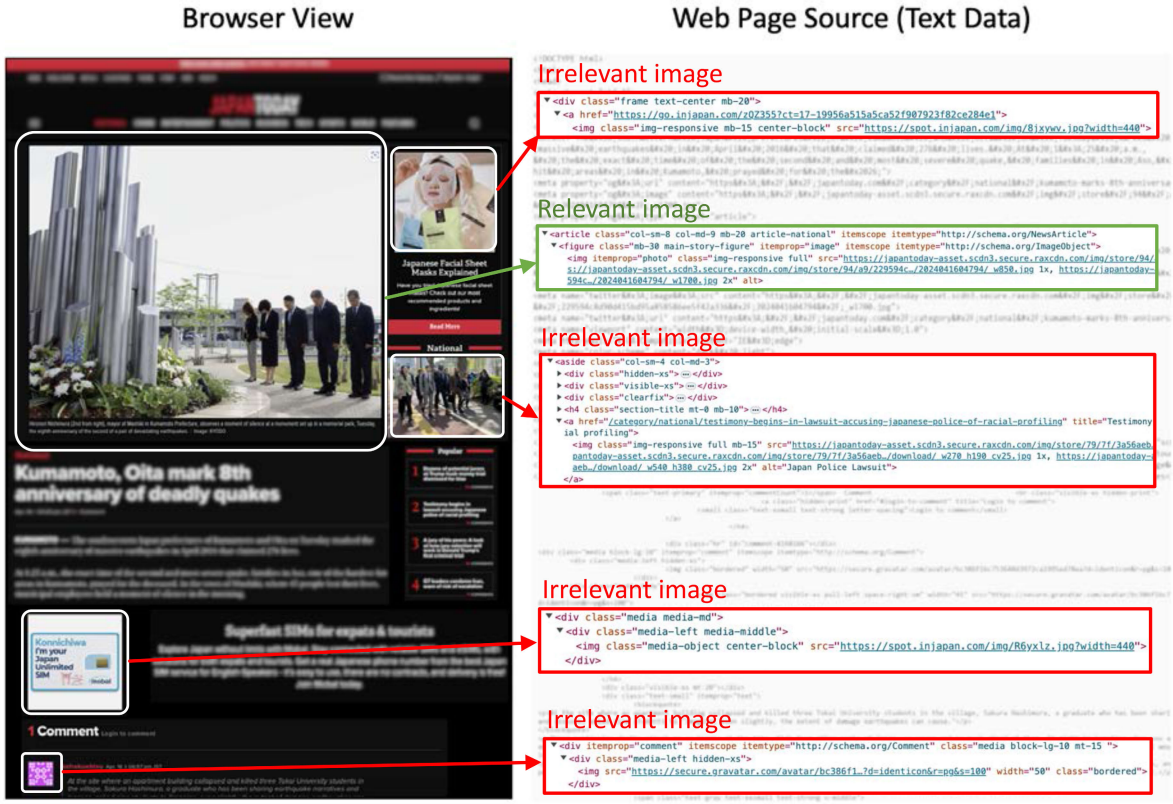

FIGURE 1. Relevant and irrelevant images along with their corresponding text sources on a web page.

extraction was performed [24]. In our study, automatic regular expression generation was performed on a dataset containing relevant and irrelevant images stored with HTML tags, on a word-by-word basis. This facilitates faster data extraction. Additionally, our study can be integrated into web scraping programs, potentially speeding up time-consuming data extraction processes performed by web scraping programs.

B. GENETIC ALGORITHMS

Within the domain of evolutionary algorithms [38], genetic algorithms [39] stand as a class of adaptive heuristic search techniques. Inspired by the tenets of natural selection and genetics, genetic algorithms offer an intelligent paradigm for navigating solution spaces. By capitalizing on historical information and incorporating randomized searches, genetic algorithms effectively steer the exploration process towards regions characterized by enhanced performance. Renowned for their capacity to generate high-caliber solutions across a spectrum of optimization and search problems, genetic algorithms have witnessed extensive application in various disciplines. This study leverages the potency of genetic algorithms to tackle the challenge of swiftly and precisely extracting images from web pages. Here, we propose the utilization of genetic algorithms for the generation of regular expressions, thereby enabling the rapid and accurate extraction of images embedded within web content. Through the integration of these methodologies, we enhance the augmentation of efficiency and accuracy within the realm of relevant image extraction.

IV. APPROACH

Our approach focuses on the automatic extraction of web image data using genetic algorithms based on text data. Unlike other methods that require expertise and skills, our approach involves generating automatic regular expressions with the knowledge of one or more relevant image data on web pages, allowing the extraction of that data. This process consists of two stages: the pre-processing stage and the genetic algorithm stage as shown in Figure 2.

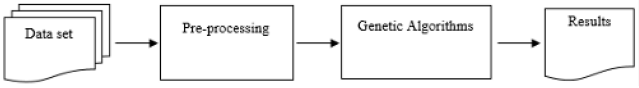

FIGURE 2. Steps of our approach.

A. PRE-PROCESSING STAGE

To perform data extraction from websites with automatic regular expressions, it is necessary to first decide which features from the dataset should be used. In the study, only 3 out of 30 features in the dataset have been utilized.

These are listed as feature-1 (theImg), feature-2 (Parent1) and feature-3 (Parent2). These features are used both individually and in combination to obtain word-based attributes. In the preprocessing stage of the study, the Bag of Words method and f-score, an attribute extraction method, were used to obtain features.

The f-score is a classification model evaluation metric used to measure the performance of results obtained from genetic algorithms [40]. It is defined as the harmonic mean of precision and recall values. Precision represents the ratio of correct positive predictions to the total positive predictions made by the model, while recall measures the proportion of true positives captured out of all actual positives. By providing a combined measure of both precision and recall values, the f-score evaluates whether the model demonstrates balanced performance. In the context of genetic algorithms, the f-score will be utilized to optimize the parameters of the model.

Initially, the Bag of Words was employed to decide which features from the dataset would be sufficient. The steps of the preprocessing stage are as follows:

Step 1: Extract features that can be used in the genetic algorithm using the features theImg, Parent1, and Parent2:

Words that can be used are extracted based on the applied feature extraction-selection methods. For example, ''imagen'', ''image_656_370'', ''display'' $\cdots$ etc. This stage will involve determining the best features from the words created using the features theImg, Parent1, and Parent2. Thus, the genes for individuals to be used in the genetic algorithm and the population size will be determined. To determine the best features, the Bag of Words feature extraction method has been used. In addition, features have been determined using the f-score, which is used to measure classification success. Since the f-score is calculated for each regular expression generated in the developed method (the reason for using f-score instead of accuracy is the imbalanced nature of the dataset), it is understood that it can also be used to determine features.

Step 2: Sub steps for determining features using the f-score:

1) Combine the features theImg, Parent1, and Parent2.
2) Remove punctuation marks in the resulting text (comma, period, exclamation mark, question mark, quotation marks, parentheses, etc.).
3) Split the text into words by spaces.
4) Add the obtained words to a list (allWords).
5) Remove words from the ''allWords'' list that can create regular expressions such as www, http, https, site name.
6) Create a list named ''words.''
7) Read the elements in the ''allWords'' list sequentially, and if the read word is not in the ''words'' list, add the word to the ''words'' list.
8) Calculate the relevance score (f-score) by searching for each word in the ''words'' list within the dataset to reach the relevant image.
9) Sort the ''words'' list in descending order based on the generated f-score.

B. GENETIC ALGORITHM STAGE

In the genetic algorithm part, first, the creation of the population is necessary. It should be determined how each individual will be encoded for this population consisting of individuals. When encoding individuals, a coding method should be selected that is suitable for the dataset. In the study, individuals were encoded using permutation encoding. The steps of the genetic algorithm are as follows:

Step 1: Create individuals based on the best f-scores of features according to the total number of individuals. When creating individuals, the positions of features are changed, or each feature is used once and encoded with the symbol ''.*?'' to have an equal number of genes.

Step 2: Determine the population size. In this stage, the population size also represents the number of individuals. For example, if the population size is set to 10, the initial population is determined as the best 10 words based on the method used. As the population-individual count increases, better regular expressions are ensured.

Step 3: Determine the length of an individual. It is decided how many genes each individual will consist of. For example, if the individual length is chosen as 5, an individual could be formed as ['.*?', 'imagen', '.*?', '.*?', '.*?'].

Here, each initially selected feature-word is distributed to form an individual, and the regular expression ''.*?'' is added until the individual length is completed. Thus, with genetic operators such as crossover, mutation, etc., more gene diversity is ensured. In the stage of creating individuals from the determined features, genes can be distributed randomly or sequentially starting from the first index of the list element.

- **Randomly Distributed Genes:** [['.*?', 'imagen', '.*?', '.*?', '.*?'], ['.*?', '.*?', 'noLazyImage', '.*?', '.*?'], ['.*?', '.*?', '.*?', '.*?', 'imagenes'], ['.*?', 'files', '.*?', '.*?', '.*?'], ['.*?', 'image_656_370', '.*?', '.*?', '.*?'], ['uploads', '.*?', '.*?', '.*?', '.*?'], ['style', '.*?', '.*?', '.*?', '.*?'], ['.*?', '.*?', 'display', '.*?', '.*?'], ['.*?', '.*?', '.*?', 'block', '.*?'], ['.*?', '.*?', '.*?', '.*?', 'La']]
- **Sequentially Distributed Genes:** [['imagen', '.*?', '.*?', '.*?', '.*?'], ['.*?', 'noLazyImage', '.*?', '.*?', '.*?'], ['.*?', '.*?', 'imagenes', '.*?', '.*?'], ['.*?', '.*?', '.*?', 'files', '.*?'], ['.*?', '.*?', '.*?', '.*?', 'image_656_370'], ['uploads', '.*?', '.*?', '.*?', '.*?'], ['.*?', 'style', '.*?', '.*?', '.*?'], ['.*?', '.*?', 'display', '.*?', '.*?'], ['.*?', '.*?', '.*?', 'block', '.*?'], ['.*?', '.*?', '.*?', '.*?', 'La']]

Step 4: Calculate the fitness value for each individual. This step is one of the most crucial in the genetic algorithm process. Once individuals are created with the determined features, the success of each individual in reaching the relevant image in the dataset is calculated using the f-score. In this stage, while calculating the f-score, a reduction in the

number of genes (removing unnecessary regular expressions) is applied. For example, let's consider an individual like ['imagen', '.*?', '.*?', '.*?', '.*?']. Since this individual consists of a single word, the regular expression is adjusted to be "imagen" and the f-score calculation is performed within the site, and the score is written next to it [(['.*?', 'imagen', '.*?', '.*?', '.*?'], 0,33739837398373984)]. If there are multiple genes containing more than one word within the individual, reduction is performed again. This time, after ensuring that the regular expression ".*?" comes between each word, the f-score calculation is done. For example, if the regular expression is ['imagen', 'image_656_370', '.*?', '.*?', '.*?'], it is adjusted to be "imagen.*? image_656_370 to ensure that ".*?" comes between each word.

- **Regular expression and scores for the first site (1st Generation):** [(['.*?', 'imagen', '.*?', '.*?', '.*?'], 0.33739837398373984), (['.*?', '.*?', 'noLazyImage', '.*?', '.*?'], 1.0), (['.*?', '.*?', '.*?', '.*?', 'imagenes'], 0.33739837398373984), (['.*?', 'files', '.*?', '.*?', '.*?'], 0.3394683026584867), (['.*?', 'image_656_370', '.*?', '.*?', '.*?'], 0.70940170940170), (['uploads', '.*?', '.*?', '.*?', '.*?'], 0.332648870636), (['style', '.*?', '.*?', '.*?', '.*?'], 0.3887587822014052), (['.*?', '.*?', 'display', '.*?', '.*?'], 0.72807017543859), (['.*?', '.*?', '.*?', 'block', '.*?'], 0.836363636363636), (['.*?', '.*?', '.*?', '.*?', 'La'], 0.5514950166112956)]

Step 5: Determine the number of generations to be created with the genetic algorithm. In this stage, the number of generations determines how many iterations-steps the program will run. It is determined considering the number of features-individuals used and the population size.

Step 6: Determine the selection method and its size. In this stage, the tournament selection method is applied to randomly determine the individual or individuals that will move on to the next generation. In the tournament selection method, individuals with higher f-scores will have a higher probability of being selected.

Step 7: Determine the crossover method and crossover probability. Apply crossover to the selected parents to produce two new individuals (children). A hybrid method, a mixture of single-point and ordered crossover methods, is used as the crossover method. Additionally, obtaining different results with different crossover probability values is explained in subsequent generations. In Figure 3, two examples of individuals used are provided. When performing the crossover operation on these individuals, a hybrid method is applied, which is a combination of both single-point and ordered crossover operations.

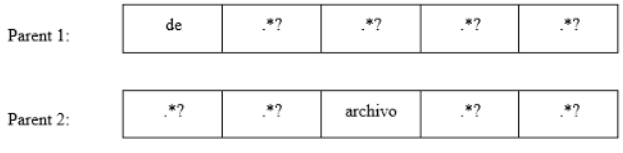

FIGURE 3. Two example individuals taken from the dataset.

Crossover method steps:

Step 7.1: In this sub step, a cutoff point is determined for both individuals by determining a random point. Figure 4 shows the cutoff point for two randomly selected individuals.

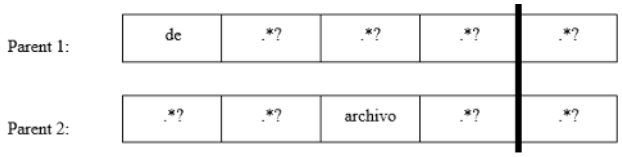

FIGURE 4. Determining the cutoff point.

Step 7.2: After the individuals are divided into two parts at the specified cutting point, segments are formed by leaving a part of each individual fixed. Figure 5 shows the segments formed after being divided into two parts at the cutting point.

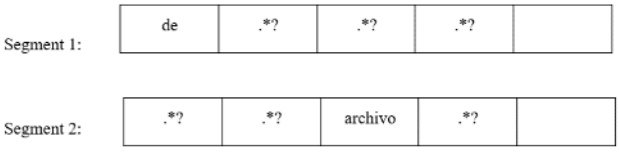

FIGURE 5. Segments formed after the cutoff point.

Step 7.3: The rank crossover method is applied by taking Individual 2 as a reference for Individual 1 and Individual 1 for Individual 2.

Step 7.4: According to the sorting crossover method, genes that are different from the genes in Individual 1 will be placed in Individual 2, respectively, instead of the genes that are not in Individual 1. The same process will be done for Individual 2. Figure 6 shows the children formed after the crossover method stages. Thus, as a result of the hybrid cross, two children with different genes are formed.

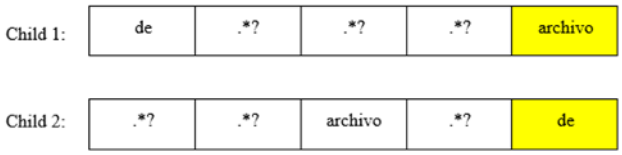

FIGURE 6. Formation of children.

Step 8: Determine mutation method and mutation probability. At this stage, genetic diversity is increased by swapping two randomly determined genes of a randomly selected individual, based on the mutation probability determined at this stage. Figure 7 shows the new individual (Individual 2) formed as a result of the mutual replacement of two genes of a selected individual (Individual 1).

Step 9: Determine elitism. Without applying genetic operators such as crossover or mutation to the next generation, the selected number of individuals or individuals with the best

TABLE 1. A simplified example of a web page illustrating relevant and irrelevant states for the 'img' tag and its two parent tags.

Image ID	Textual features	Textual content	main_image
1	theImg	<img class="img-responsive mb-15 center-block" src="img/BjKDpv.jpg?width=440">	0
	Parent1	<a href="com/v32WnZ">	
	Parent2		
2	theImg	<img itemprop="photo" class="img-responsive full" src="2024041604794_w850.jpg" alt="">	1
	Parent1	<figure class="mb-30 main-story-figure" itemprop="image" itemtype="http://schema.org/ImageObject">	
	Parent2	<article class="col-sm-8 col-md-9 mb-20 article-national" itemtype="http://schema.org/NewsArticle">	
3	theImg	<img class="img-responsive full mb-15" src="w270_h190_cv25.jpg" alt="Japan Police Lawsuit">	0
	Parent1	<a href="/category/national/testimony-begins-in-lawsuit" title="Testimony begins in lawsuit">	
	Parent2	<aside class="col-sm-4 col-md-3">	
4	theImg	<img class="media-object center-block" src="RvXlXv.jpg?width=440">	0
	Parent1		
	Parent2		
5	theImg	<img class="bordered" width="50" src="avatar/bc386f1bc753640d3972ca1995ed78ea?d=100">	0
	Parent1		
	Parent2		

FIGURE 7. Application of mutation.

f-score are passed to the next generation without changing their genes.

Step 10: Generate the best regular expression. If the best result (regular expression) is not produced (f score = 1.0) or 100 generations are not produced, continue with step 4.

V. EXPERIMENTS

Firstly, the experimental section will begin with a brief explanation of the dataset and performance metrics. Subsequently, f-score values and experimental results will be presented using sample websites.

A. DATASET

The data set used within the scope of the study consists of images obtained from 200 websites [34]. These websites encompass 58 different countries, including Albania, Australia, Austria, Azerbaijan, Bahrain, Bangladesh, Belarus, Bolivia, Bosnia and Herzegovina, Botswana, Brazil, Bulgaria, Cameroon, Canada, China, Cuba, Czechia, Egypt, Finland, France, Germany, Greece, Guatemala, India, Indonesia, Iran, Italy, Japan, Jordan, Kazakhstan, Kyrgyzstan, Laos, Latvia, Liberia, Macedonia, Madagascar, Malaysia, Mexico, Montenegro, Nepal, New Zealand, Pakistan, Philippines, Romania, Russia, Slovakia, Spain, Tanzania, Turkey, Uganda, Ukraine, USA, Uzbekistan, Venezuela, Vietnam, Zambia, and Zimbabwe. For each site in the dataset, 100 web pages have been downloaded. Thus, there are a total of 20,000 web pages in the dataset. The content of the dataset consists of HTML codes of images found on the web pages. A dataset containing a total of 635,015 images has been created. These images are encoded as relevant (1) and irrelevant (0). Out of these images, 22,682 are relevant image.

TABLE 2. Experiments with specific genetic parameters on the dataset.

Number of relevant images viewed	Feature extraction method	f-score	Number of sites where all relevant images were accessed
1	f-score	0.9581	136
	BoW(1,1)	**0.9671**	139
	BoW(1,2)	0.9647	**140**
2	f-score	0.9609	140
	BoW(1,1)	0.9644	137
	BoW(1,2)	**0.9666**	**141**
3	f-score	0.9615	135
	BoW(1,1)	**0.9638**	**139**
	BoW(1,2)	0.9632	138
5	f-score	0.9646	132
	BoW(1,1)	**0.9687**	**137**
	BoW(1,2)	0.9662	136
10	f-score	0.9683	139
	BoW(1,1)	**0.9714**	**142**
	BoW(1,2)	0.9726	139
20	f-score	0.9693	138
	BoW(1,1)	0.9704	**139**
	BoW(1,2)	**0.9708**	**139**

A website is composed of HTML attributes. For each image in the dataset used in the study, 30 attributes have been employed. Among these attributes, ''theImg'', ''Parent1'' and ''Parent2'' have been used in the study. For each image, the ''main_image'' attribute is the feature that determines whether the image is relevant or irrelevant. It is assigned 1 for relevant images and 0 for irrelevant images. Images related to the content of the page within a website are considered relevant, while images such as advertisements, logos, etc., found outside the page content are considered irrelevant. Table 1 shows examples for these four attributes.

B. PERFORMANCE MEASUREMENTS

The data set used in the study is an imbalanced data set. The number of irrelevant images in the data set is quite high. For this reason, accuracy is not used as a performance measure. The accuracy value can be misleading because it gives the ratio of predictions of both relevant and irrelevant images to

TABLE 3. Experiments with specific genetic parameters on several websites from the dataset.

Site Number	ID of Site Name	First regex	First f-score	PS	IL	NG	CP	MP	TS	E	Method	Feature	Final regex	Final f-score
3	1	slider.*?listing.*? status.*?standard	0.9595	300	10	100	0.5	0.1	2	2	f-score	first relevant	uploads.*?class.*? class	1
25	2	1280.*?span	0.9402	150	10	75	0.5	0.2	1	1	BoW (1,2)	top 5 related	low .*? 1280	1
36	3	prvky.*?fotogaleria	0.7359	120	3	75	0.2	0.1	2	2	BoW (1,1)	first relevant	galeria .*? ob .*? galeria	1
42	4	Content	0.9638	150	10	75	0.5	0.1	1	1	f-score	first relevant	id .*? images	1
69	5	main.*?part	0.9898	300	10	100	0.5	0.1	2	2	f-score	first relevant	main.*?cms	1
82	6	lazy.*?md.*?1600	0.9892	300	10	100	0.5	0.1	2	2	f-score	first relevant	lazy.*?images	1
87	7	article	0.9865	300	10	100	0.5	0.1	2	2	f-score	first relevant	article.*?rhd	1
97	8	_w1700	0.9826	150	10	75	0.5	0.1	1	1	f-score	top 20 related	_w1700 .*? item	1
110	9	100	0.9109	200	8	100	0.5	0.2	1	1	f-score	first relevant	news .*? col	1
123	10	margin	0.9677	150	10	75	0.5	0.1	1	1	f-score	first relevant	margin .*? 7	1
136	11	photo	0.4921	100	3	75	0.2	0.2	2	2	f-score	first relevant	262 .*? width .*? 262	1
140	12	620px	0.6667	300	10	100	0.5	0.1	2	2	f-score	first relevant	0px.*?img.*?img	1
142	13	AP	0.7208	300	10	100	0.5	0.1	2	2	f-score	top 3 related	https.*?https.*?https .*?data.*?data	1
158	14	660	0.9744	150	10	75	0.5	0.1	1	1	BoW (1,2)	top 5 related	660 .*? 43 relative	1
173	15	4.*?jpg.*?20	0.9897	300	10	100	0.5	0.1	2	2	f-score	first relevant	title.*?4	1
178	16	stories_images.*?100 .*?cover.*?story	0.6575	300	10	100	0.5	0.1	2	2	f-score	first relevant	resources.*? story.*?cover	1
197	17	yenisafak	0.6691	150	5	75	0.9	0.1	2	2	f-score	first relevant	yenisafak .*? 2020	1

ID: Site Names - 1: 24tanzania.com, 2: www.bild.de, 3: www.cas.sk, 4: cn.chinadaily.com.cn, 5: faktor.ba, 6: www.hbl.fi, 7: www.hurriyet.com.tr, 8: japantoday.com, 9: www.kurzemnieks.lv, 10: www.manobkantha.com.bd, 11: www.nikkan-gendai.com, 12: nra.lv, 13: www.nydailynews.com, 14: romanialibera.ro, 15: www.tehrantimes.com, 16: www.themalaysianinsight.com, 17: www.yenisafak.com

all predictions. For this reason, f-score is used as performance criterion value in the study.

C. EXPERIMENT RESULTS

In the study, attributes were created by combining theImg, Parent1 and Parent2 features in the data set. The genetic parameters used for the first regular expression created in Table 2 were determined as population size (PS) 50, individual length (IL) 5, number of generations (NG) 50, crossover probability (CP) 0.5, mutation probability (MP) 0.1, tournament size (TS) 1 and finally elitism (E) 1 for each test. Bow(1,1), BoW(1,2) and f-score were used as feature extraction methods. Although f-score is generally used as a performance evaluation criterion in the literature, in the study the objective function was created according to the f-score method and used as a feature extraction method. The f-score obtained as a result of the tests and the number of sites where all relevant images were accessed are shown in Table 2. According to these results, it has been observed that all relevant images of 142 sites can be accessed by looking at the first 10 relevant images. By looking at the first relevant image, it was observed that the maximum number of sites where all relevant images could be accessed was 140. This shows that 96% of all relevant images can be accessed by looking at the first relevant image data.

It has been observed that more relevant image data can be accessed with the final regular expressions created as a result of the different genetic parameters and methods in Table 3. As a result of the tests, it is shown in Table 3 that improvements were made to the regular expressions produced for 30 sites and new regular expressions were created that can access more relevant images. While previous tests created regular expressions that could access all relevant images of 142 sites at most, with the tests performed in this section, regular expressions that could access all relevant images of 17 more sites were created, and regular expressions that could access all relevant images of 159 of 200 sites in total were created. Thus, when all evaluations were made, regular expressions that could be accessed with 98.49% accuracy for 22,682 relevant images on 200 sites were created with the method using the developed genetic algorithm.

As a result of the studies, the pre-processing time was generally measured to be 8 seconds on average, and the genetic algorithm running time was measured to be 4 seconds on average for 200 sites. This period may vary depending on the amount of data in the sites and the number of images involved.

VI. CONCLUSION

In this study, a model that automatically generates regular expressions using genetic algorithms is presented to extract the relevant image data belonging to the actual page content shared on the web. The data set used in the study includes HTML codes of 635,015 image data coded as relevant and irrelevant. Each image data is encoded with 30 features. Relevant images in the data are coded as 1, and irrelevant images are coded as 0. A model was designed using theImg, Parent1, Parent2 features as shown Figure 1 in the data set. In the model designed in the study, the widely used Bag of Words method and the features extracted by using the f-score of the words obtained from the features were used to extract the features. f-score, accuracy, recall and precision methods are included to measure the success of test methods. However, since the data in the dataset is unbalanced, it has

been observed that performance measurement with f-score is more accurate.

In the experiments, only the relevant images were viewed. It has been observed that good results can be achieved by only looking at the relevant pictures in the tests performed by using the f-score and the attributes obtained in the form of both single words and two words with the Bag of Words method. In both methods, regular expressions have been found to be approximately 96% accurate by looking at just a few relevant images instead of seeing all the images. Experiments were also performed using different genetic parameters in the study. Looking at previous tests, it was seen that regular expressions were formed that could access all relevant images of a maximum of 142 sites. However, in these tests performed with different genetic parameters, it was observed that regular expressions were produced that could access all relevant images of 159 of 200 sites. Thus, thanks to the method developed using the genetic algorithm, regular expressions were created that could access 22,680 relevant image data from all 200 sites with 98.49% accuracy.

The processing time for the experiments taken using the method may vary for each site. This period may vary depending on the amount of data on the site. For example, while one site had 10,000 images, another site had 2,000 images. Generally speaking, it can be said that the processing time is shorter on sites where the amount of data is less. For the data set used in the study, it was observed that the general pre-processing time was approximately 8 seconds, while the genetic algorithm duration was approximately 4 seconds. Although this developed method is used only to access the relevant image data, it can be used for video, audio and text data in future studies. It can also be integrated into web scraping programs and used to perform web scraping operations easier and faster.

In future studies, efforts will focus on generating regular expressions for various textual data types such as titles, main content, and comments. Additionally, attention will be given to deep learning models in web content extraction and regular expression generation. Finally, experimental tests are planned for improvements to the approach to make it more efficient and faster.

REFERENCES

[1] Z. Zhao, H. Xue, Z. Xue, Z. Liu, J. Tian, M. C. H. Chua, and M. Liu, "An image-text consistency driven multimodal sentiment analysis approach for social media," *Inf. Process. Manage.*, vol. 56, no. 6, Nov. 2019, Art. no. 102097.

[2] D. K. Vishwakarma, D. Varshney, and A. Yadav, "Detection and veracity analysis of fake news via scrapping and authenticating the web search," *Cognit. Syst. Res.*, vol. 58, pp. 217–229, Dec. 2019.

[3] J. Xue, Y. Wang, Y. Tian, Y. Li, L. Shi, and L. Wei, "Detecting fake news by exploring the consistency of multimodal data," *Inf. Process. Manage.*, vol. 58, no. 5, Sep. 2021, Art. no. 102610.

[4] M. Mahrishi, S. Morwal, N. Dahiya, and H. Nankani, "A framework for index point detection using effective title extraction from video thumbnails," *Int. J. Syst. Assurance Eng. Manage.*, Jun. 2021.

[5] Q. Liu, M. Shao, L. Wu, G. Zhao, G. Fan, and J. Li, "Main content extraction from web pages based on node characteristics," *J. Comput. Sci. Eng.*, vol. 11, no. 2, pp. 39–48, Jun. 2017.

[6] E. Uzun, H. V. Agun, and T. Yerlikaya, "A hybrid approach for extracting informative content from web pages," *Inf. Process. Manage.*, vol. 49, no. 4, pp. 928–944, Jul. 2013.

[7] E. Uçar, E. Uzun, and P. Tufekci, "A novel algorithm for extracting the user reviews from web pages," *J. Inf. Sci.*, vol. 43, no. 5, pp. 696–712, Oct. 2017.

[8] W. Haider and Y. Yesilada, "Classification of layout vs. Relational tables on the web: Machine learning with rendered pages," *ACM Trans. Web*, vol. 17, no. 1, pp. 1–23, Feb. 2023.

[9] E. Uzun, E. Serdar Guner, Y. Kılıçaslan, T. Yerlikaya, and H. V. Agun, "An effective and efficient web content extractor for optimizing the crawling process," *Software: Pract. Exper.*, vol. 44, no. 10, pp. 1181–1199, Oct. 2014.

[10] P. E. San and N. Aye, "Main content extraction from dynamic web pages," Ph.D. dissertation, 2015.

[11] R. P. Velloso and C. F. Dorneles, "Web page structured content detection using supervised machine learning," in *Proc. 19th Int. Conf.*, 2019, pp. 1–14.

[12] F. Ciravegna, "Algorithms for learning regular expressions from positive data," in *Proc. 17th Int. Joint Conf. Artif. Intell.*, vol. 2, Aug. 2001, pp. 1–20.

[13] H. Fernau, "Detecting fake news by exploring the consistency of multimodal data," *Inf. Process. Manage.*, vol. 58, no. 5, Sep. 2009, Art. no. 102610.

[14] A. Sahuguet and F. Azavant, "ABuilding light-weight wrappers for legacy web data-sources using W4F," in *Proc. 25th Int. Conf. Very Large Data Bases*, Sep. 1999, pp. 1–19.

[15] J. E. Friedl and E. Oram, *Mastering Regular Expressions*. Sebastopol, CA, USA: O'Reilly Associates, 2002.

[16] Y. Li, R. Krishnamurthy, S. Raghavan, S. Vaithyanathan, and H. V. Jagadish, "Regular expression learning for information extraction," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2008, pp. 21–30.

[17] A. Cetinkaya, "Regular expression generation through grammatical evolution," in *Proc. 9th Annu. Conf. companion Genetic Evol. Comput.*, Jul. 2007, pp. 2643–2646.

[18] P. Prasse, C. Sawade, N. Landwehr, and T. Scheffer, "Learning to identify regular expressions that describe email campaigns," in *Proc. 29th Int. Conf. Int. Conf. Mach. Learn.*, 2012, pp. 1115–1122.

[19] Y. Lin, "Internet data extraction based on automatic regular expression inference," M.S. thesis, Iowa State Univ., Ames, IA, USA, 2007.

[20] F. Brauer, R. Rieger, A. Mocan, and W. M. Barczynski, "Enabling information extraction by inference of regular expressions from sample entities," in *Proc. 20th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2011.

[21] E. Uzun, "A regular expression generator based on CSS selectors for efficient extraction from HTML pages," *Turkish J. Electr. Eng. Comput. Sci.*, vol. 28, no. 6, pp. 3389–3401, 3389.

[22] J. Li, G. Jiang, A. Xu, and Y. Wang, "A regular expression generator based on CSS selectors for efficient extraction from HTML pages," *J. Softw.*, vol. 12, no. 3, pp. 180–188, 2017.

[23] R. Babbar and N. Singh, "Clustering based approach to learning regular expressions over large alphabet for noisy unstructured text," in *Proc. 4th Workshop Analytics Noisy Unstructured Text Data*, Oct. 2010.

[24] A. Bartoli, G. Davanzo, A. De Lorenzo, M. Mauri, E. Medvet, and E. Sorio, "Automatic generation of regular expressions from examples with genetic programming," in *Proc. 14th Annu. Conf. Companion Genetic Evol. Comput.*, Jul. 2012.

[25] K. Murthy, P. Deepak, and P. M. Deshpande, "Improving recall of regular expressions for information extraction," in *Proc. 13th Int. Conf. Web Inf. Syst. Eng.*, Nov. 2012, pp. 455–467.

[26] A. Bartoli, A. De Lorenzo, E. Medvet, and F. Tarlao, "Inference of regular expressions for text extraction from examples," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 5, pp. 1217–1230, May 2016.

[27] A. Bartoli, A. De Lorenzo, E. Medvet, and F. Tarlao, "Can a machine replace humans in building regular expressions? A case study," *IEEE Intell. Syst.*, vol. 31, no. 6, pp. 15–21, Nov. 2016.

[28] A. Bartoli, A. De Lorenzo, E. Medvet, and F. Tarlao, "Active learning of regular expressions for entity extraction," *IEEE Trans. Cybern.*, vol. 48, no. 3, pp. 1067–1080, Mar. 2018.

[29] B. Cody-Kenny, M. Fenton, A. Ronayne, E. Considine, T. McGuire, and M. O'Neill, "A search for improved performance in regular expressions," in *Proc. Genetic Evol. Comput. Conf.*, Jul. 2017.
[30] D. F. Barrero, D. Camacho, and M. D. R-Moreno, "Automatic web data extraction based on genetic algorithms and regular expressions," in *Data Mining Multi-agent Integration*. Cham, Switzerland: Springer, 2009.
[31] D. F. Barrero, M. D. R-Moreno, and D. Camacho, "Adapting searchy to extract data using evolved wrappers," *Expert Syst. Appl.*, vol. 39, no. 3, pp. 3061–3070, Feb. 2012.
[32] E. Uzun, "Scraping relevant images from web pages without download," *ACM Trans. Web*, vol. 18, no. 1, pp. 1–27, Feb. 2024.
[33] H. V. Agun and E. Uzun, "An efficient regular expression inference approach for relevant image extraction," *Appl. Soft Comput.*, vol. 135, Mar. 2023, Art. no. 110030.
[34] E. Uzun, E. Özhan, H. V. Agun, T. Yerlikaya, and H. N. Bulus, "Automatically discovering relevant images from web pages," *IEEE Access*, vol. 8, pp. 208910–208921, 2020.
[35] K. Vyas and F. Frasincar, "Determining the most representative image on a web page," *Inf. Sci.*, vol. 512, pp. 1234–1248, Feb. 2020.
[36] J. Helfman and J. Hollan, "Image representations for accessing and organizing web information," in *Proc. The International Society for Optical Engineering*. Bellingham, WA, USA: SPIE, 2000.
[37] A. Bhardwaj and V. Mangat, "An improvised algorithm for relevant content extraction from web pages," *J. Emerg. Technol. Web Intell.*, vol. 6, no. 2, May 2014.
[38] P. A. Vikhar, "Evolutionary algorithms: A critical review and its future prospects," in *Proc. Int. Conf. Global Trends Signal Process., Inf. Comput. Commun. (ICGTSPICC)*, Jalgaon, India, Dec. 2016, pp. 261–265.
[39] M. Melanie, *An Introduction to Genetic Algorithms*. Cambridge, MA, USA: MIT Press, 1996.
[40] L. Derczynski, "Complementarity, F-score, and NLP Evaluation," in *Proc. 10th Int. Conf. Lang. Resour. Eval.*, 2016, pp. 261–266.

CANAN ASLANYÜREK received the degree from the Department of Computer Engineering, Süleyman Demirel University, in 2014, and the master's degree in computer engineering from Trakya University, in 2018, where she is currently pursuing the Ph.D. degree. In her doctoral thesis, she is interested in genetic algorithms and regular expressions. She has been a Lecturer with Kırklareli University, since 2017. Her areas of interests include cryptography, machine learning, genetic algorithms, web data extraction, and regular expressions.

TARIK YERLIKAYA was born in Edirne, Türkiye, in 1977. He received the degree from the Electronics and Communications Engineering Department, Yıldız Technical University, Istanbul, Türkiye, in 1999, and the master's and Ph.D. degrees from the Computer Engineering Department, Edirne, in 2002 and 2007, respectively. In 2007, he completed the Ph.D. thesis in cryptography. In 1999, he started his academic career with the Computer Engineering Department, Trakya University. From 1999 to 2007, he was a Research Assistant with Trakya University, where he became an Assistant Professor (Dr.) with the Computer Engineering Department, in 2008. His research interests include cryptography, natural language processing, text classification, and information extraction.