Intelligent Automation & Soft Computing
DOI:10.32604/iasc.2022.022209
Article

Tech Science Press

An Automated Word Embedding with Parameter Tuned Model for Web Crawling

S. Neelakandan[1,*], A. Arun[2], Raghu Ram Bhukya[3], Bhalchandra M. Hardas[4], T. Ch. Anil Kumar[5] and M. Ashok[6]

[1]Department of Information Technology, Jeppiaar Institute of Technology, Sriperumbudur, 631604, India
[2]Department of Computer Science and Engineering, SRM Institute of Science and Technology, Kattankulathur, 603203, India
[3]Department of Computer Science and Engineering, Kakatiya Institute of Technology & Science, Warangal, 506 015, India
[4]Department of Electronics Engineering, Shri Ramdeobaba College of Engineering and Management, Nagpur, 440013 India
[5]Department of Mechanical Engineering, Vignan's Foundation for Science Technology and Research, Guntur, 522213, India
[6]Department of Computer Science and Engineering, Rajalakshmi Institute of Technology, Chennai, 600 124, India
*Corresponding Author: S. Neelakandan. Email: snksnk17@gmail.com
Received: 31 July 2021; Accepted: 04 October 2021

Abstract: In recent years, web crawling has gained a significant attention due to the drastic advancements in the World Wide Web. Web Search Engines have the issue of retrieving massive quantity of web documents. One among the web crawlers is the focused crawler, that intends to selectively gather web pages from the Internet. But the efficiency of the focused crawling can easily be affected by the environment of web pages. In this view, this paper presents an Automated Word Embedding with Parameter Tuned Deep Learning (AWE-PTDL) model for focused web crawling. The proposed model involves different processes namely pre-processing, Incremental Skip-gram Model with Negative Sampling (ISGNS) based word embedding, bidirectional long short-term memory-based classification and bird swarm optimization based hyperparameter tuning. The SGNS training desires to go over the complete training data to pre-compute the noise distribution before performing Stochastic Gradient Descent (SGD) and the ISGNS technique is derived for the word embedding process. Besides, the cosine similarity is computed from the word embedding matrix to generate a feature vector which is fed as input into the Bidirectional Long Short-Term Memory (BiLSTM) for the prediction of website relevance. Finally, the Birds Swarm Optimization-Bidirectional Long Short-Term Memory (BSO-BiLSTM) based classification model is used to classify the webpages and the BSO algorithm is employed to determine the hyperparameters of the BiLSTM model so that the overall crawling performance can be considerably enhanced. For validating the enhanced outcome of the presented model, a comprehensive set of simulations are carried out and the results are examined in terms of different measures. The Automated Word Embedding with Parameter Tuned Deep Learning (AWE-PTDL) technique has attained a higher harvest rate of 85% when compared with the other techniques. The experimental results highlight the enhanced web crawling performance of the proposed model over the recent state of art web crawlers.

Keywords: Focused web crawling; deep learning; word embedding; parameter optimization; cosine similarity

1 Introduction

Due to the rapid development of network data, the Internet must turn into an effective database. To acquire the domain knowledge from huge data is a great challenge. However, the aim is to gather appropriate data from the Internet, i.e., crawling web pages [1]. Thus, to crawl web pages efficiently, scientists have presented Web Crawlers (WC). WC is a program that collects data from the Internet. It is segregated into general and special purpose WCs [2]. General purpose WCs retrieves huge number of web pages in each field from the Internet. To store and find the website, general-purpose WCs should contain immense hard-disk space and long running times. But the special-purpose WCs known as focused crawlers, produce better accuracy and recall by limiting themselves to a restricted region [3]. In contrast to general purpose WCs, focused crawlers apparently need a less run-time and hardware assets. Hence, focused crawler turns to be highly significant in collecting data from the website for resource limitations and are utilized in several applications like information extraction, search engines, text classification and digital libraries [4].

The practice of indexing information on web sites using a computer or automated script is referred to as crawling. A crawler, spider, spider bot, or similar automated software is referred to by various names, including web crawler, spider, and spider bot. The website's robot.txt file is downloaded by web crawlers as they start their crawling operation. Sitemaps for URLs that the search engine may crawl are included in the file. As soon as web crawlers begin to explore a webpage, they find new pages by following links. In order to explore these newly found URLs at a later time, these crawlers add URLs to the crawl queue. Web crawlers are now able to search and index any page that is related to others thanks to these new approaches.

From a huge asset on the web, almost all of them are not relevant to target domain. For that reason, Focused Web Crawlers (FWC) are highly preferable for retrieving websites. The FWC depends on methods like ML (classification) for identifying appropriate web pages, including local database [5]. These models are feature based, modelling an input region of interest for classifying appropriate web pages. When the websites are classified effectively, their URL is queued and extracted by the frontier model. In few FWC methods [6,7], the classification model depends on the document similarity measures for filtering related and non-related web pages. But such methods do not consider the expressiveness of web page contents, i.e., they do not explore their semantic contents or utilize the data in the filter procedure [8]. And it is the beginning for iteratively extracting the URLs. Specifically, FWC analysis the content of seed URLs for determining the significance of their content for a region of interest. This content analysis depends on the methods such as machine learning, query expansion, ontology-based approach [9,10]. Few methods need a primary dataset for creating a module (ML methods) or a group of keywords for producing certain domain queries (query extension).

The Semantic Web (SW) is considered as an expansion of current Web based on RDF for expressing data in a well-determined manner [11,12]. Fig. 1 demonstrates the functioning system of an FWC, in which the primary URL is fixed by the user for a provided topic. Depending upon the relevance score, precedence is stored and assigned in web page archives.

This paper presents an AWE-PTDL model for focused web crawling. The proposed model initially performs different stages of pre-processing to transform the raw data into compatible format. In addition, the proposed model involves an Incremental Skip-gram Model with Negative Sampling (ISGNS) based word embedding technique. At last, the Birds Swarm Optimization (BSO) with BiLSTM based classification model is used to classify the webpages. The BSO technique is used to fine tune the

parameters of the BiLSTM model and to accomplish maximum classification performance. A wide range of simulations takes place to highlight the better performance of the proposed AWE-PTDL model over the recent state of art techniques.

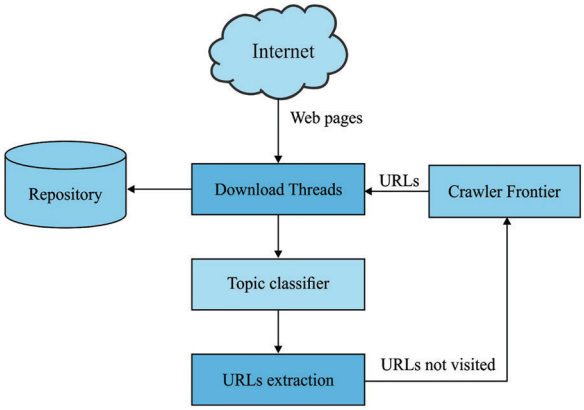

Figure 1: General flow of focused web crawler

2 Related Works

This section reviews the existing WC techniques available in the recent literature. Sekhar et al. [13] proposed a resolution to predict the page significance, depending upon the Natural Language Processing. The occurrence of the keyword on the top-rated sentence of the page defines the significance of the page within genomics sources. The presented resolution utilizes a Text Rank method for ranking the sentence and ensure the accurate classifications of Bioinformatics webpage. At last, the method is authenticated using a breadth first search. A novel approach which incorporates the Adagrad-optimized Skip Gram Negative Sampling (A-SGNS) based word embedding and Recurrent Neural Networks (RNN) [14].

Alexandrino et al. [15] addressed the problem of calculating and designing a WC for finding two OGC web service standards namely WMS and WFS. Commercial search engines like Bing and Google are certainly performing a beneficial task as general-purpose search engine. But few applications require domain specific search engines and WCs for finding comprehensive data on the Web. Thus, this study presents Spati Harvest, WC emphasis on finding WMSes and WFSes. Spati Harvest is an integration of the most developed methods established in the survey. The primary aim in [16] is to detect the bottlenecks in the traditional structure. It considers the research framework that shows an enhanced method for accelerating the crawling procedure. Research has been carried out on hybrid, Eclat, Declat, Apriori methods. The relative calculation of memory used reflects the minimum utilization by the hybrid framework. The major feature of this presented Technique is its capability of tunnelling via pages with a lower score.

Judith et al. [17] improved on the efficacy of focused crawling by suggesting a method based on RL approach. This method calculates the hyperlinks more effectively in the long run and selects the more promising links based on this calculation. To precisely model the crawling platform as a Markov decision procedure, they proposed novel representation of state-action that considers both link structure and content information. The size of the state-action space is decreased by the generalization procedure. According to this generalization, they utilize linear function approximation for updating value function. They examine the trade-off between asynchronous and synchronous approaches. Lu et al. [18] proposed a novel focused crawler. First, they construct webpage classifiers according to weighting method (ITFIDF), for gaining high appropriate web pages.

Sankaralingam et al. [19] proposed a query-based model in which a group of keys appropriate to the domain knowledge of the end user is utilized for shooting queries on searching interface. Hernandez et al. [20] presented a new SFWC based on schema for modelling the crawler's domain, hence decreasing the cost and complexity of creating a proper depiction while utilizing ontologies. Moreover, similarity of metrics depends on the integration of IDF measure, SD and the arithmetical mean presented for the SFWC. These measures filter webpage contents according to the target domain in the crawling process.

In Hosseini et al. [21], various approaches for crawler detection are examined. Log files of an instance of compromised websites are analyzed and optimal features for detecting crawlers have been extracted. After comparing and testing various ML methods like SVM, BN and DT, the optimal method is established with the most relevant features and its accuracy is calculated. Zhang et al. [22] proposed an advanced technique for efficient and feasible attainment of sewage outfall data by integrating remote sensing interpretation and WC techniques.

3 The Proposed Model

The initial phase is the topic pre-processing phase, where the topic is pre-processed with the help of Parts-of-Speech (POS) tagging, tokenization, synonym and stemming searches, senseless word filtering. The pre-processed topic is saved in storage. The next phase is the crawling phase, in which the webpages are download from the web using the allocated seed URL. When the downloading is completed, the webpages are transmitted to the term extraction phase. In the extraction phase, the webpages are analyzed to plaintext by eliminating HTML information tags. After parsing, the target parameters like anchor and webpage texts are extracted from the webpage. Later, in pre-processing phase, the extracted target parameters are pre-processed with the help of POS tagging, tokenization, senseless word stemming and filtering. The ISGNS-based word embedding matrix is created during the feature extraction step. The classification phase uses the cosine feature vector as an input, and the BSO-BiLSTM network categorises the webpage to determine its importance. Fig. 2 displays the flow chart of the presented work.

3.1 Design of ISGNS Based Word Embedding Technique

To provide the word sequence, w_1, w_2, …, w_n, the skip-gram process is employed to minimize the following function for word embedding.

$$\mathcal{L}_{\mathrm{SG}} = -\frac{1}{n}\sum_{i=1}^{n}\sum_{\substack{|j| \le c \\ j \neq 0}} \log p(w_{i+j}|w_i),$$

where w_i implies the target word, w_{i+j} represents the context word within window of size c and $p(w_{i+j}|w_i)$ signifies the probability that w_{i+j} performs with the neighbor w_i and is determined as follows,

$$p(w_{i+j}|w_i) = \frac{\exp(\mathrm{t}_{w_i} \cdot \mathrm{c}_{w_{i+j}})}{\sum_{w \in \mathcal{W}} \exp(\mathrm{t}_{w_i} \cdot \mathrm{c}_w)}, \tag{1}$$

where t_w and c_w are w's embedded that performs as target and context respectively. $\mathcal{W}$ implies the vocabulary set.

As it can be expensive for optimizing the above objective, the negative sampling speed up skip-gram training is used [23]. This estimates Eq. (1) utilizing sigmoid functions and k arbitrarily sampled words are named as negative samples. The resultant function is provided as follows.

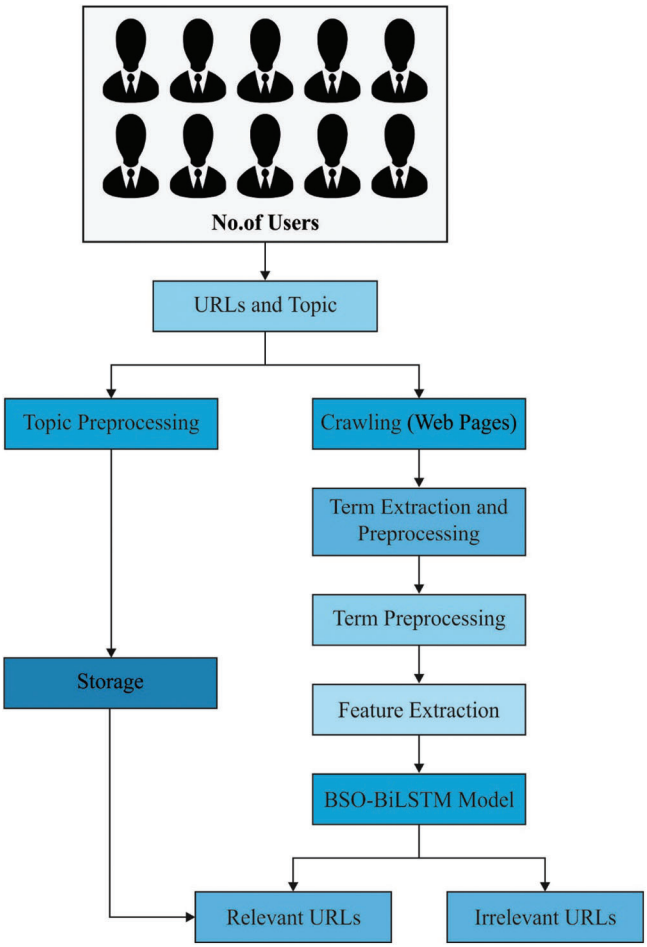

Figure 2: Block diagram of AWE-PTDL

$$\mathcal{L}_{\text{SGNS}} = -\frac{1}{n}\sum_{i=1}^{n}\sum_{\substack{|j| \le c \\ j \neq 0}} \psi^{+}_{w_i,w_{i+j}} + k\mathrm{E}_{v\sim q(v)}[\psi^{-}_{w_i,v}],$$

where $\psi^{+}_{w,v} = \log\sigma(\mathrm{t}_w \cdot \mathrm{c}_v)$, $\psi^{-}_{w_i,v} = \log\sigma(-\mathrm{t}_w \cdot \mathrm{c}_v)$ and $\sigma(x)$ implies the sigmoid functions. The negative instances v represents the smoothed unigram probability distribution mentioned as noise distribution $q(v) \propto f(v)^{\alpha}$, where $f(v)$ signifies the frequency of word v in the trained data and α refers the smoothing parameter $(0 < \alpha \le 1)$.

The objective is to optimize by SGD. Provided a target-context word pair (w_i and w_{i+j}) and k negative instances $(v_1, v_2, \ldots, v_k)$ drawn from noise distribution, the gradient of $-\psi^{+}_{w_i,w_{i+j}} - k\mathrm{E}_{v\sim q(v)}[\psi^{-}_{w_i,v}] \approx -\psi^{+}_{w_i,w_{i+j}} - \sum_{k'=1}^{k}\psi^{-}_{w_i,v_{k'}}$ is calculated. Then, the gradient descent is carried out for updating t_{w_i}, $\mathrm{c}_{w_{i+j}}$ and $\mathrm{c}_{v_1}, \ldots, \mathrm{c}_{v_k}$.

The training of SGNS requires the total trained information for pre-computing the noise distribution $q(v)$ before carrying out SGD. This makes it complex for performing incremental method upgrades if additional trained data is given.

Algorithm 1 proposes ISGNS that goes with trained data in single pass for updating word embedded incrementally. Different from the original SGNS, it does not pre-compute the noise distribution. Instead, it delivers the trained data word by word1 for incrementing upgrade of the word frequency distribution and the noise distribution by carrying out SGD. Henceforth, the original SGNS was mentioned as batch SGNS for emphasizing, where the noise distribution is calculated in batch fashion.

The rate of learning for SGD is adjusted by utilizing AdaGrad. The linear decay function is extremely utilized for trained batch SGNS and the adaptive techniques namely AdaGrad is further appropriate for incremental trained, as the number of training information is unknown in the development or is improved unboundedly.

It is straightforward in extending the ISGNS to mini-batch setting, by analyzing a subset of trained data (or mini-batch) instead of a single word at a time for updating the noise distribution and carry out SGD.

Algorithm 1: ISGNS

$f(w) \leftarrow 0$ for all $w \in \mathcal{W}$

for $i = 1, \ldots, n$ do

$$f(w_i) \leftarrow f(w_i) + 1$$

$q(w) \leftarrow \frac{f(w)^{\alpha}}{\Sigma_{w' \in \mathcal{W}} f(w')^{\alpha}}$ for all $w \in \mathcal{W}$

for $j = -c, \ldots, -1, 1, \ldots, c$ do

draw k negative instances from $q(w) : v_1, \ldots, v_k$

utilize SGD for updating t_{w_i}, $c_{w_{i+j}}$ and $c_{v1}, \ldots, c_{v_k}$

end for

end for

3.2 Cosine Similarity

From the proposed word embedded module, The cosine parity related the topic and the web page content is given in Eq. (2).

$$sim(t, d) = \frac{\vec{t}^{T} \cdot \vec{d}}{\| \vec{t} \| \cdot \| \vec{d} \|} \tag{2}$$

whereas $\vec{t}^{T}$ denotes the vector equivalent to the content of Topic, $\vec{d}$ represents the embedded vector equivalent to the webpage content.

3.3 Design of BSO-BiLSTM Based Classification Model

The features from the cosine similarity are passed into the BiLSTM model to classify the web pages. RNN is a special variety of normal ANN that demonstrates sequential information using recurrent connections [24]. Basically, it continues with the hidden state that is regarded as "memory" of preceding input. It is determined that all neurons signify an estimated function of every preceding data. An input unit $\{\ldots, x_{t-1}, x_t, x_{t+1}, \ldots\} = (x_1, x_2, x_3, \ldots, x_N)$ are associated with hidden unit $h_t = (h_1, h_2, \cdots, h_M)$ in the hidden layers, through associates determined as weight matrix W_{IH}. All hidden units are associated with the next one with recurrent associates provided as W_{HH}. All hidden units are expressed as follows.

$$h_t = f_H(o_t) \tag{3}$$

where,

$$o_t = W_{IH} + W_{HH} h_{t-1} + b_h \tag{4}$$

F_h implies the non-linear function namely tanh, ReLU or sigmoid and b_H represents the bias vectors. The hidden layer is also linked to the resultant layer of weight W_{HO}. Eventually, the output $y_t = (y_1, y_2, \ldots, y_P)$ is determined as follows.

$$y_t = f_O(W_{HO} h_t + b_o) \tag{5}$$

A similar approach as hidden layers, f_O represents the activation functions and b signifies the bias vectors. While this technique continues a memory of earlier states from vanishing gradient issue in long-term dependency. The different kind of RNN is named as LSTM established in 1997 to overcome this problem. The LSTM cell further follows a sophisticated approach which utilize "forget" gate for selecting what to forget. The state of LSTM memory unit accepts the following mathematical model.

$$i_t = \sigma(W_{xi} + W_{hi} h_{t-1} + \mathrm{W}_{ci} c_{t-1} + b_i)$$

$$f_t = \sigma(W_{xf} x_t + W_{hf} h_{t-1} + \mathrm{W}_{\mathrm{cf}} c_{t-1} + b_f)$$

$$c_t = f_t \otimes c_{t-1} + i_t \otimes tanh(W_{xc} x_t + W_{hc} h_{t-1} + b_c)$$

$$o_t = \Gamma(W_{xo} x_t + W_{ho} h_{t-1} + W_{co} c_t + b_o)$$

$$h_t = o_t \otimes tanh(c_t)$$

The subscripts related to all matrices signifies (for instance, W_{hf} implies the hidden forget weight matrix). Also, f, i, o and c is related to forget, input, output and cell gate vectors respectively. But the framework of LSTM network only carries passes on sequential data that eventually means that the data dependency is only uni-directionally modelled. Therefore, by integrating these two together, a Bidirectional LSTM (Bi-LSTM) is generated that is utilized for modelling dependency bi-directionally. Fig. 3 demonstrates the architecture of Bi-LSTM technique.

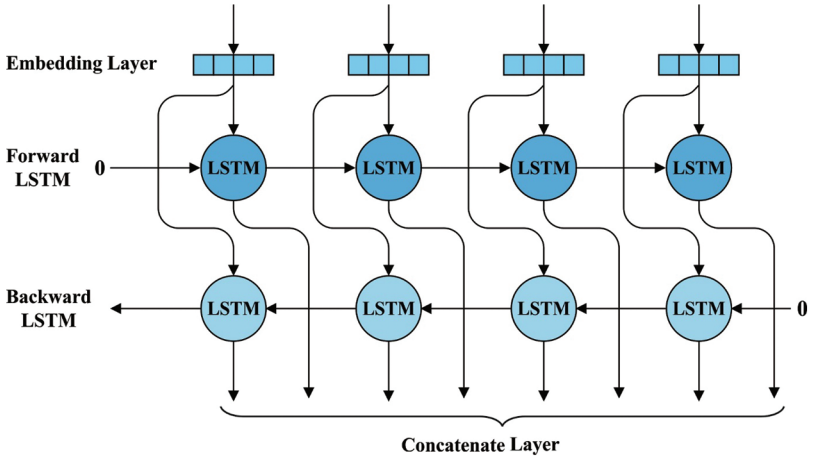

Figure 3: Structure of Bi-LSTM

For improving the outcomes of the BiLSTM model, the BSO algorithm is applied on it. The BSO is a new metaheuristic approach to solve optimization applications. It simulates the birds' flight, foraging and vigilance behaviour for solving the global optimization problem. In the procedure of foraging, every bird

search for food based on population and individual experiences. Such behaviour could be defined as below,

$$x_{i,j}^{t} = x_{i,j}^{t-1} + (p_{i,j} - x_{i,j}^{t-1}) * C * rand(0,\ 1) + (g_{best,j} - x_{i,j}^{t-1}) * S * rand(0,\ 1) \tag{6}$$

whereas $\chi_{i,j}^{t}$ represents the value of jth component of ith solution at tth generation, rand $(0,\ 1)$ represents a uniform distribution function, $p_{i,j}$ indicates the optimal prior location for jth component of ith bird and $g_{best,j}$ signifies the jth component of global best solution. C and S are the two positive numbers namely cognitive and social accelerated coefficient, respectively. During vigilance, every bird attempt to shift toward the center of swarm and will certainly compete with each other. The vigilance behaviour is given below,

$$x_{i,j}^{t} = x_{i,j}^{t-1} + (mean_j - x_{i,j}^{t-1}) * A1 * rand\ (0,\ 1) + (p_{i,j} - x_{i,j}^{t-1}) * A2 * rand\ (0,\ 1) \tag{7}$$

$$A1 = a1 * \ \exp\ \left(- \frac{pFit_i}{SumFit + \epsilon} * N \right) \tag{8}$$

$$A2 = a2 * \ \exp\ \left(\left(\frac{pFit_i - pFit_k}{|pFit_i - pFit_k| + \epsilon} \right) \frac{pFit_k * N}{SumFit + \epsilon} \right) \tag{9}$$

whereas $k(k \neq i)$ denotes a positive integer, i.e., arbitrarily selected from one to N. Then, $a1$ and $a2$ indicates two positive constants in $[0,\ 2]$, $pFit_i$ represents the ith bird's optimal fitness value and SumFit indicates the swarm's optimal fitness value. ε, utilized for avoiding zero-division error and it is the smallest constant in the computer. $Mean_j$ signifies the jth component of average location of the entire swarm [25,26].

Birds fly to different positions from time to time. While flying to other positions, birds might frequently shift between scrounging and producing. The birds with the maximum fitness value will be producers, whereas the ones with the minimum fitness value will be scroungers. The fighting behaviour of the scroungers and producers could be defined as follows,

$$x_{i,j}^{t} = x_{i,j}^{t-1} * (1 + randn\ (0,\ 1)) \tag{10}$$

$$x_{i,j}^{t} = x_{i,j}^{t-1} + FL * (x_{k,j}^{t-1} - x_{i,j}^{t-1}) * rand\ (0,\ 1) \tag{11}$$

whereas rand $n(O,\ 1)$ denotes a Gaussian distribution with mean 0 and standard deviation 1, $k \in [0, N]$, $k \neq i$. $FL \in (0, 2)$ represents the likelihood of the scroungers follow the producer for seeking food. Fig. 4 illustrates the flowchart of BSO technique.

Assume the individual variances, the Arithmetical Problems in Engineering FL value of every scrounger will arbitrarily choose from 0 to 2. The bird switches to fight at each FQ time step. Algorithm 2 defines the execution of BSO. In Algorithm 2, the variable N represents the number of populations, M signifies the highest number of iterations, FQ signifies the frequency of birds' fighting behaviour and P indicates the foraging likelihood for food.

Algorithm 2: Structure of the BSO

Parameter Initialization: N, M, FQ, P ;

Population Initialization of N birds and determine the N individual fitness value.

While ($t < M$)

If (% $FQ \neq 0$)

For $i = 1$ to N

If rand $(0,\ 1) < P$

(continued)

Algorithm 2: (continued)

Birds forage for food
Else
Birds keep vigilance by Eq. (7)
End if
End for
Else
Split the swarm into scroungers and producers.
For $i = 1$ to N
If (= =producer)
Birds fight by Eq. (10) //producer
Else
Birds fight by Eq. (11) //scrounger
End if
End for
End if
Calculate the fitness value of the novel solution;
If ($f_{new} < f_{01d}$)
$f_{01d} = f_{new}$;
Upgrade the global optimum solution;
$t++$
End While
Output the global optimum solution;

4 Performance Validation

This section performs the experimental validation of the proposed AWE-PTDL technique in various aspects. Tab. 1 provides a comprehensive classification result analysis of the AWE-PTDL technique in terms of different measures. Fig. 5 displays the precision analysis of the AWE-PTDL technique on classification of two classes. The figure depicts that the Artificial Neural Network-Tern Frequency-Inverse Document Frequency (ANN-TFIDF) technique has the least outcome with the minimal precision of 50% each under class 1 and class 0 respectively. In addition, both Support Vector Machine-Term Frequency-Inverse Document Frequency (SVM-TFIDF) and Naïve Bayse- Term Frequency- Inverse Document Frequency (NB-TFIDF) techniques have attained an identical performance with the precision of 50% under class 1% and 51% under class 0 respectively. The RNN-ASGNS technique has obtained moderately increased performance with the precision of 62% and 57% under class 1 and class 0 respectively. However, the proposed AWE-PTDL technique has attained the enhanced performance result with the maximum precision of 89% and 86% under class 1 and class 0 respectively.

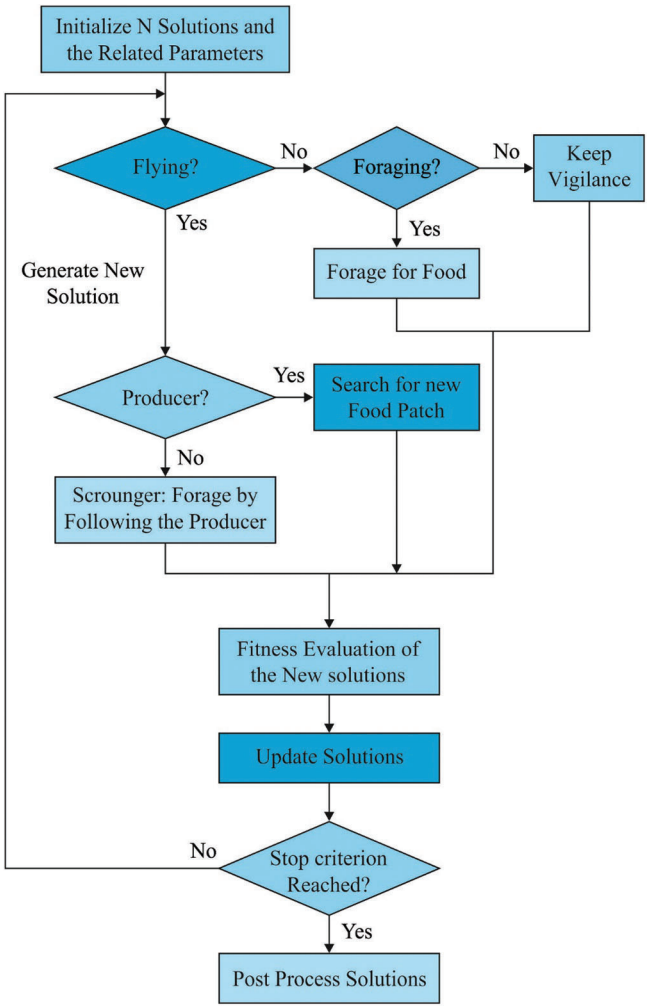

Figure 4: Flowchart of BSO algorithm

Table 1: Result analysis of existing with proposed AWE-PTDL model in terms of precision, recall and F1-score

Methods	Precision (%)		Recall (%)		F1-score (%)		Accuracy (%)
	Class-1	Class-0	Class-1	Class-0	Class 1	Class 0	
SVM-TFIDF	50.00	51.00	62.00	39.00	55.00	44.00	62.00
NB-TFIDF	50.00	51.00	63.00	39.00	55.00	44.00	62.00
ANN-TFIDF	50.00	50.00	42.00	58.00	45.00	53.00	70.00
RNN-ASGNS	62.00	57.00	45.00	73.00	52.00	64.00	81.00
AWE-PTDL	89.00	86.00	84.00	88.00	85.00	89.00	86.00

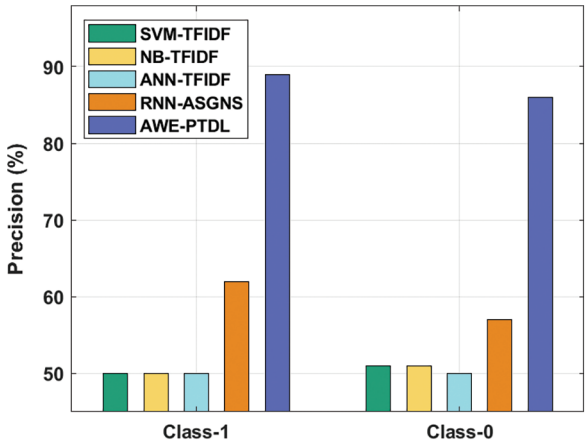

Figure 5: Precision analysis of AWE-PTDL model

Fig. 6 shows the recall analysis of the AWE-PTDL approach on classification of two classes. The figure shows that the ANN-TFIDF technique has the minimum recall of 42% and 58% under class 1 and class 0 respectively. Also, the SVM-TFIDF and NB-TFIDF algorithms have attained the recall of 62% and 63% under class 1 respectively while it is 39% under class 0 for both the techniques. Then, the RNN-ASGNS technique has achieved an improved performance with the recall of 45% and 73% under class 1 and class 0 respectively. Finally, the presented AWE-PTDL method has resulted in enhanced performance with the maximal recall of 84% and 88% under class 1 and class 0 respectively.

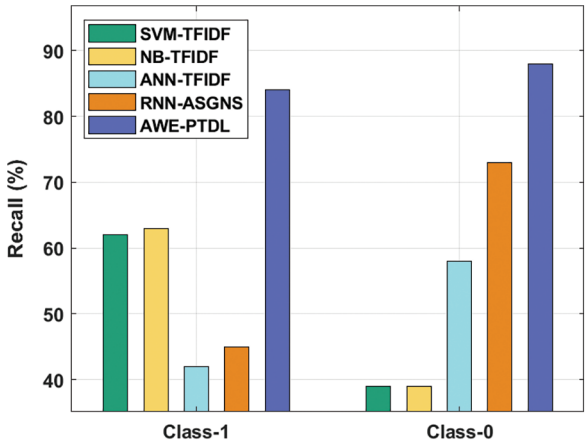

Figure 6: Recall analysis of AWE-PTDL model

Fig. 7 depicts the F1-score analysis of the AWE-PTDL method on classification of two classes. The figure demonstrates that the ANN-TFIDF algorithm has exhibited minimum outcome with the lower F1-score of 45% and 53% under classes 1 and 0 respectively. Furthermore, the SVM-TFIDF and NB-TFIDF methods have gained an identical performance with the F1-score of 55% under class 1% and 44% under class 0 respectively. Next, the RNN-ASGNS technique has an increased performance with the F1-score of 73% and 52% under class 1 and class 0 respectively. Eventually, the projected AWE-PTDL method has resulted in increased performance with superior F1-score of 85% and 89% under class 1 and class 0 respectively.

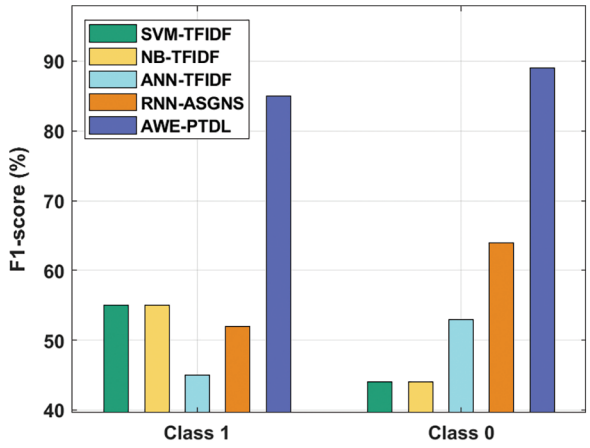

Figure 7: F1-score analysis of AWE-PTDL model

Accuracy analysis of the AWE-PTDL technique with existing techniques is shown in Fig. 8. The figure demonstrates that both the SVM-TFIDF and NB-TFIDF techniques have attained a lesser and identical accuracy of 62%. Then, the ANN-TFIDF technique has gained slightly improved outcome with an accuracy of 70%. Simultaneously, the RNN-ASGNS technique has accomplished reasonable outcome with an accuracy of 81%. However, the proposed AWE-PTDL technique has resulted with a maximum accuracy of 86%.

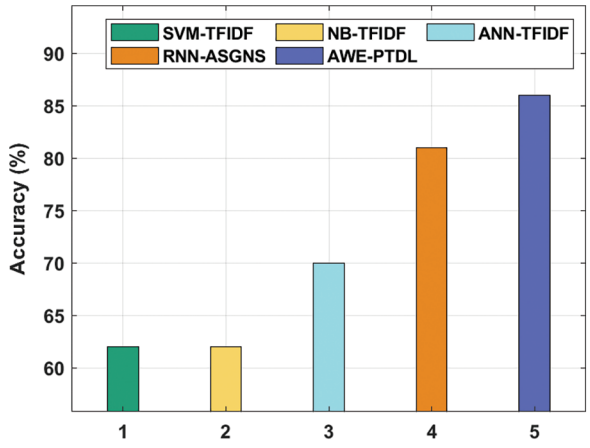

Figure 8: Accuracy analysis of AWE-PTDL model

A harvest rate analysis of the AWE-PTDL technique with existing techniques under varying number of web pages downloaded is shown in Tab. 2 and Fig. 9. From the attained results, it is evident that the AWE-PTDL technique has a better performance under all the distinct number of web pages downloaded. For instance, with 1000 web pages downloaded, the AWE-PTDL technique has attained a higher harvest rate of 85% whereas the RNN-ASGNS, ANN-TFIDF, SVM-TFIDF, NB-TFIDF, VS Model and Breadth First Search (BFS) Model has obtained a lower harvest rate of 79%, 63%, 61%, 41%, 48% and 26% respectively. Meanwhile, with 2000 web pages downloaded, the AWE-PTDL approach has obtained an improved harvest rate of 70% whereas the RNN-ASGNS, ANN-TFIDF, SVM-TFIDF, NB-TFIDF, VS Model and BFS Model has attained a lesser harvest rate of 64%, 56%, 54%, 39%, 39% and 18% respectively. Eventually, with 3000 web pages downloaded, the AWE-PTDL approach has achieved a

maximal harvest rate of 63% whereas the RNN-ASGNS, ANN-TFIDF, SVM-TFIDF, NB-TFIDF, VS Model and BFS Model has a lower harvest rate of 56%, 48%, 47%, 37%, 33% and 17% respectively. Simultaneously, with 4000 web pages downloaded, the AWE-PTDL method has obtained a higher harvest rate of 56% whereas the RNN-ASGNS, ANN-TFIDF, SVM-TFIDF, NB-TFIDF, VS Model and BFS Model has obtained a lower harvest rate of 47%, 39%, 37%, 33%, 26% and 15% respectively.

Table 2: Result analysis of existing with proposed AWE-PTDL model in terms of harvest rate (%)

Methods	No. of web pages downloaded				
	1000	2000	3000	4000	5000
AWE-PTDL	85.00	70.00	63.00	56.00	52.00
RNN-ASGNS	79.00	64.00	56.00	47.00	42.00
ANN-TFIDF	63.00	56.00	48.00	39.00	34.00
SVM-TFIDF	61.00	54.00	47.00	37.00	32.00
NB-TFIDF	41.00	39.00	37.00	33.00	29.00
VS Model	48.00	39.00	33.00	26.00	24.00
BFS Model	26.00	18.00	17.00	15.00	13.00

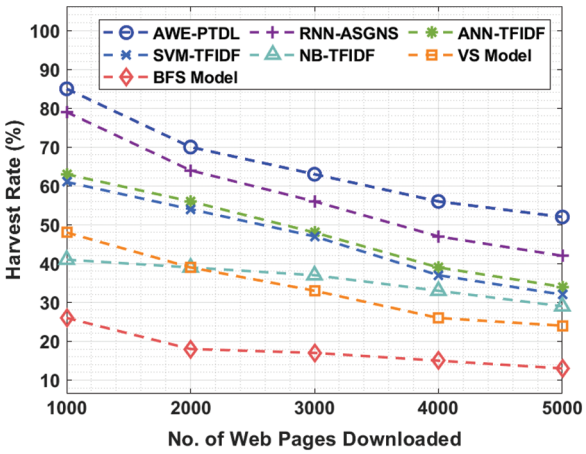

Figure 9: Result analysis of AWE-PTDL model in terms of harvest rate

At last, with 5000 web pages downloaded, the AWE-PTDL methodology has obtained a maximum harvest rate of 52% whereas the RNN-ASGNS, ANN-TFIDF, SVM-TFIDF, NB-TFIDF, VS Model and BFS Model have attained a minimum harvest rate of 42%, 34%, 32%, 29%, 24% and 13% respectively.

Finally, an irrelevance ratio analysis of the AWE-PTDL technique with other techniques is shown in Tab. 3 and Fig. 10. The results demonstrate that the AWE-PTDL technique shown better performance with the minimal irrelevance ratio. For instance, with 1000 web pages downloaded, the AWE-PTDL technique has resulted in the least irrelevance ratio of 15% whereas the RNN-ASGNS, ANN-TFIDF, SVM-TFIDF, NB-TFIDF, VS Model and BFS Model has an increased irrelevance ratio of 21%, 37%, 39%, 59%, 52% and 74% respectively. Concurrently, with 2000 web pages downloaded, the AWE-PTDL method has resulted with a minimum irrelevance ratio of 30% whereas the RNN-ASGNS, ANN-TFIDF,

SVM-TFIDF, NB-TFIDF, VS Model and BFS Model has an improved irrelevance ratio of 36%, 44%, 46%, 61%, 61% and 82% respectively. At the same time, with 3000 web pages downloaded, the AWE-PTDL method has resulted with an irrelevance ratio of 37% whereas the RNN-ASGNS, ANN-TFIDF, SVM-TFIDF, NB-TFIDF, VS Model and BFS Model has an irrelevance ratio of 44%, 52%, 53%, 63%, 67% and 83% respectively.

Table 3: Result analysis of existing with proposed AWE-PTDL model in terms of irrelevance ratio (%)

Methods	No. of web pages downloaded				
	1000	2000	3000	4000	5000
AWE-PTDL	15.00	30.00	37.00	44.00	48.00
RNN-ASGNS	21.00	36.00	44.00	53.00	58.00
ANN-TFIDF	37.00	44.00	52.00	61.00	66.00
SVM-TFIDF	39.00	46.00	53.00	63.00	68.00
NB-TFIDF	59.00	61.00	63.00	67.00	71.00
VS Model	52.00	61.00	67.00	74.00	76.00
BFS Model	74.00	82.00	83.00	85.00	87.00

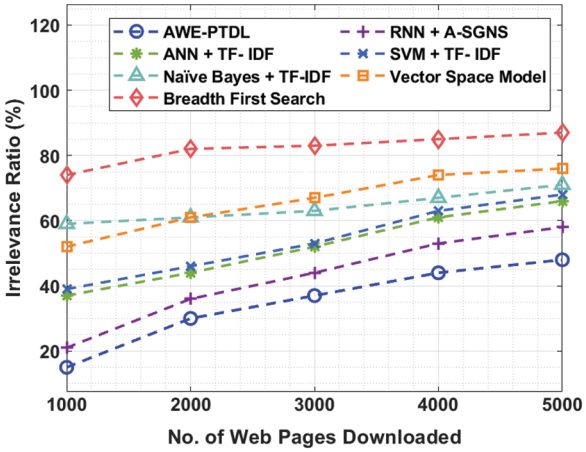

Figure 10: Comparative analysis of AWE-PTDL Model in terms of Irrelevance Ratio

Eventually, with 4000 web pages downloaded, the AWE-PTDL algorithm has resulted in the least irrelevance ratio of 44% whereas the RNN-ASGNS, ANN-TFIDF, SVM-TFIDF, NB-TFIDF, VS Model and BFS Model has obtained an irrelevance ratio of 53%, 61%, 63%, 67%, 74% and 85% respectively. At last, with 5000 web pages downloaded, the AWE-PTDL methodology has resulted with an irrelevance ratio of 48% whereas the RNN-ASGNS, ANN-TFIDF, SVM-TFIDF, NB-TFIDF, VS Model and BFS Model has attained a higher irrelevance ratio of 58%, 66%, 68%, 71%, 76% and 87% respectively. From the above-mentioned result analysis, it is apparent that the AWE-PTDL technique is an effective focused web crawler tool and can be employed in real time scenarios.

5 Conclusion

In this study, a new AWE-PTDL technique is developed to achieve effective outcome of the WC. This study analyzed the syntax and semantic similarities between the web page documents and topic. The AWE-PTDL technique initially determines the ISGNS base word embedding from document terms and cosine similarity of the topics are determined. The similarity of vectors is provided as input to the BSO-BiLSTM model to categorize the web pages based on the relevance. The employment of BSO technique to fine tune the parameters of the BiLSTM model also paves a way to accomplish maximal classification performance. The experimental outcome portrays that the AWE-PTDL technique has enhanced the performance of the focused crawler. The AWE-PTDL technique has attained a higher harvest rate of 85% whereas the RNN-ASGNS, ANN-TFIDF, SVM-TFIDF, NB-TFIDF, VS Model and BFS Model has obtained a lower harvest rate of 79%, 63%, 61%, 41%, 48% and 26% respectively. As a part of future extension, the focused crawler can be designed using hybrid CNN-LSTM model for eliminating the vanishing gradient problem. Moreover, the presented model can be extended to design focused WCs in real time application such as e-commerce and education.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] Y. Du, W. Liu, X. Lv and G. Peng, "An improved focused crawler based on semantic similarity vector space model," *Applied Soft Computing Journal*, vol. 36, no. 3, pp. 392–407, 2015.

[2] E. Madhan, S. Neelakandan and R. Annamalai, "A novel approach for vehicle type classification and speed prediction using deep learning," *Journal of Computational and Theoretical Nano Science*, vol. 17, no. 5, pp. 2237–2242, 2020.

[3] F. Ahmadi-Abkenari and A. Selamat, "An architecture for a focused trend parallel Web crawler with the application of clickstream analysis," *Information Sciences*, vol. 184, no. 1, pp. 266–281, 2012.

[4] P. V. Rajaraman and M. Prakash, "Deep reply-An automatic email reply system with unsupervised cloze translation and deep learning," *ICTACT Journal on Soft Computing*, vol. 10, no. 3, pp. 2090–2095, 2020.

[5] A. Vinothini and S. Baghavathi Priya, "Survey of machine learning methods for big data applications," in *2017 Int. Conf. on Computational Intelligence in Data Science (ICCIDS)*, Chennai, India, IEEE, pp. 1–5, 2017.

[6] S. Neelakandan and D. Paulraj, "An automated learning model of conventional neural network based sentiment analysis on twitter data," *Journal of Computational and Theoretical Nano Science*, vol. 17, no. 5, pp. 2230–2236, 2020.

[7] T. Salah and S. Tiun, "Focused crawling of online business Web pages using latent semantic indexing approach," *ARPN Journal of Engineering and Applied Science*, vol. 11, no. 10, pp. 9229–9234, 2016.

[8] R. Annamalai and J. Srikanth, "Accessing the data efficiently using prediction of dynamic data algorithm," *International Journal of Computer Applications*, vol. 116, no. 22, pp. 39–42, 2015.

[9] M. Kumar, R. K. Bhatia and D. Rattan, "A survey of web crawlers for information retrieval," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 7, no. 6, pp. 1218–1224, 2017.

[10] S. Neelakandan and S. Muthukumaran, "Transformation-based optimizations framework (tof) for workflows and its security issues in the cloud computing," *International Journal of Engineering and Computer Science*, vol. 4, no. 8, pp. 13746–13753, 2016.

[11] P. Bedi, A. Thukral, H. Banati, A. Behl and V. Mendiratta, "A multi-threaded semantic focused crawler," *Journal of Computer Science and Technology*, vol. 27, no. 4, pp. 1233–1242, 2012.

[12] A. Batzios, C. Dimou, A. L. Symeonidis and P. A. Mitkas, "Biocrawler: An intelligent crawler for the semantic web," *Expert System Application*, vol. 35, no. 3, pp. 524–530, 2008.

[13] S. M. Sekhar, G. M. Siddesh, S. S. Manvi and K. G. Srinivasa, "Optimized focused web crawler with natural language processing-based relevance measure in bioinformatics web sources," *Cybernetics and Information Technologies*, vol. 19, no. 2, pp. 146–158, 2019.

[14] P. R. Joe Dhanith, B. Surendiran and S. P. Raja, "A word embedding based approach for focused web crawling using the recurrent neural network," *International Journal of Interactive Multimedia & Artificial Intelligence*, vol. 6, no. 6, pp. 45–58, 2021.

[15] V. M. Alexandrino, G. Comarela, A. S. Silva and J. Lisboa-Filho, "A focused crawler for web feature service and web map service discovering," in *Int. Symp. on Web and Wireless Geographical Information Systems*, Springer, Cham, China, pp. 111–124, 2020.

[16] P. Lambhate, A. Hambarde, M. Emmanuel and S. Hambarde, "Hybrid algorithm on semantic web crawler for search engine to improve memory space and time," in *2021 6th Int. Conf. for Convergence in Technology (I2CT)*, Mumbai, India, IEEE, pp. 1–6, 2021.

[17] A. M. Judith and S. B. Priya. "Multiset task related component analysis for ssvep frequency recognition in bci," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 5, pp. 5117–5126, 2021.

[18] H. Lu, D. Zhan, L. Zhou and D. He, "An improved focused crawler: Using web page classification and link priority evaluation," *Mathematical Problems in Engineering*, vol. 2016, no. 6406901, pp. 1–10, 2016.

[19] B. P. Sankaralingam and R. Thangavel, "An optimal scheduling algorithm for real time applications in grid system," *International Journal of Computer Science Issues*, vol. 10, no. 1, pp. 145–154, 2013.

[20] J. Hernandez, H. M. Marin-Castro and M. Morales-Sandoval, "A semantic focused web crawler based on a knowledge representation schema," *Applied Sciences*, vol. 10, no. 11, pp. 3837, 2020.

[21] N. Hosseini, F. Fakhar, B. Kiani and S. Eslami, "Enhancing the security of patients' portals and websites by detecting malicious web crawlers using machine learning techniques," *International Journal of Medical Informatics*, vol. 132, no. 10, pp. 103976, 2019.

[22] J. Zhang, T. Zou and Y. Lai, "Novel method for industrial sewage outfall detection: Water pollution monitoring based on web crawler and remote sensing interpretation techniques," *Journal of Cleaner Production*, vol. 312, no. 4, pp. 127640, 2021.

[23] V. Sindhu and S. Nivedha, "An empirical science research on bioinformatics in machine learning," *Journal of Mechanics of Continua and Mathematical Sciences*, vol. 10, no. 7, pp. 86–94, 2020.

[24] A. Pogiatzis and G. Samakovitis, "Using BiLSTM networks for context-aware deep sensitivity labelling on conversational data," *Applied Sciences*, vol. 10, no. 24, pp. 8924, 2020.

[25] M. Lin, Y. Zhong, J. Lin and X. Lin, "Discrete bird swarm algorithm based on information entropy matrix for traveling salesman problem," *Mathematical Problems in Engineering*, vol. 2018, no. 9461861, pp. 1–15, 2018.

[26] T. Ravichandran, "An efficient resource selection and binding model for job scheduling in grid," *European Journal of Scientific Research*, vol. 81, no. 4, pp. 450–458, 2012.