Vol.64: e21210163, 2021
https://doi.org/10.1590/1678-4324-2021210163
ISSN 1678-4324 Online Edition

*Article - Engineering, Technology and Techniques*

# An Enhanced Focused Web Crawler for Biomedical Topics Using Attention Enhanced Siamese Long Short Term Memory Networks

**Joe Dhanith Pal Nesamony Rose Mary[1]***
https://orcid.org/0000-0002-9022-9145

**Surendiran Balasubramanian[1]**
http://orcid.org/0000-0001-5435-0880

**Raja Soosaimarian Peter Raj[3]**
http://orcid.org/0000-0002-7216-2207

[1]National Institute of Technology Puducherry, Karaikal, India; [2]School of Computer Science and Engineering, Vellore Institute of Technology, Vellore, Tamilnadu, India;

Editor-in-Chief: Alexandre Rasi Aoki
Associate Editor: Fabio Alessandro Guerra

Received: 2021.03.18; Accepted: 2021.04.13.

*Correspondence: joe.dhanith@gmail.com (J.D.P.N.R.M.).

**HIGHLIGHTS**

- This paper proposes a new focused crawler for biomedical topics.
- This paper proposes a novel Attention Enhanced Siamese Long Short Term Memory Networks.
- The proposed model is trained using ADAM optimizer with Batch Normalization.
- This paper produces an average harvest rate of 0.39.

**Abstract:** The Internet is chosen to be one among the primary source of biomedical information. To retrieve necessary biomedical information, the search engine needs an efficient, focused crawler mechanism. But the area of research concerned with the focused crawler for biomedical topics is notably scanty. However, the quantity, momentum, diversity, and quality of the available online biomedical information, challenges and calls for enhanced aid to crawl. This paper surmounts the challenges and proposes a new learning approach for focused web crawling adopting Attention Enhanced Siamese Long Short Term Memory (AE-SLSTM) Networks with peephole connections which predicts topical relevance of the web page. The proposed AE-SLSTM model accurately computes the semantic similarity between the topic and the web pages. The performance of the newly designed crawler is assessed using two well known metrics namely harvest rate $(h_{rate})$ and irrelevance ratio $(p_{rate})$. The presented crawler surpass the existing focused crawlers with an average $h_{rate}$ of 0.39 and an average $p_{rate}$ of 0.61 after crawling 5,000 web pages relating to biomedical topics. The results clearly depicts that the proposed methodology aids to download more relevant biomedical web pages related to the particular topic from the internet.

**Keywords:** focused crawler; attention mechanism; LSTM; peephole connection; Manhattan distance; Siamese network.

## INTRODUCTION

The expeditious growth of information initiates crucial computational inducement to the formatting, indexing, ranking, storing and extraction of data [1]. This enigma is predominant in many disciplines and has adopted diverse mechanisms dependent to their concerned anomaly. The web is a classical domain to work on required extracted feature although rapid surge of information floods the web. However, influence of the linked character of web data, keeps scientific and computational growths foiling. Web bots have been designed to download the web pages, extract the data and store it in a repository.

Ranking algorithms such as Google Page Rank [2] archive and arrange web pages according to their relevance score. This allows the search engines to supply catalog of items that can convincingly answer several user queries. This potentiality of the search engine provides an user friendly source of information from the web that supports answer user queries. The available web search technologies are inadequate, if the queries requested to be searched are complicated or loaded with diverse topics at the same time. Only a limited part of the web could actively participate to search web pages by modern search engines and omits the dynamic web pages, password protected web pages as well as the web pages that are connected through the scripts.

In the Computational Bioinformatics domain, the recent collection of genomics data is increasing gradually. The current potentiality of search engines in extracting relevant biomedical information is challenging due to various reasons. The first is the scattered, diverse and non-compatible nature of biomedical data with their diverse types stored in different formats. The important biomedical information is basically unstructured because they are naturally illustrated in free texts as provided in the discharge clinical reports. The electronic biomedical reports can improve the availability and distribution of biomedical data and information. The data available in the PDF format could not support extraction of information and querying problems. The second is the complex nature of biomedical queries which leads to poor retrieval of information.

The versatile structure of the clinical settings with their complexities pose forbidding technological and strategic challenges for an effective administration and utilization of biomedical data. Crawling, an active conceptualized method is propitious to resolve these intricacies. This paper proposes a new focused crawler, for the effective exploration of biomedical information present in the web. Focused crawler is a special purpose web crawler which could download topic relevant web pages. Currently research work is very limited related to the focused crawler which downloads relevant web pages of biomedical information.

This paper proposes a new learning-based crawler using Attention Enhanced-Siamese Long Short Term Memory (AE-SLSTM) Networks [3, 4] with peephole connections to resolve these issues. Here two LSTM [5] networks are implemented: one for the topic and the other for the web page contents. This work adopts pre-trained Global Vectors for word representation (GloVe) [6] embeddings to convert the input text sequences into embedding vectors. The peephole connection in the LSTM helps to learn the memory cells directly. The Manhattan metric used in this work computes the complex semantic similarity between the topic-web page pairs. The proposed focused crawler can efficiently format, extract, rank, index, and store the web pages related to biomedical topics.

This paper is organized as follows. Section 2 presents the Literature Survey, Section 3 establishes out the proposed methodology and Section 4 sets out the experimental design. Section 5 discusses the evaluation and analysis of performance, while Section 6 concludes the paper.

## Related Work

The topic specific relevant web pages are alone downloaded by the Focused crawler. Four types of available focused web crawlers are Vector Space Model (VSM), learning, semantic and ontology learning crawlers.

In the existing literature, Vector Space Model (VSM) [7] crawlers used cosine similarity which is weighted by Term Frequency-Inverse Document Frequency (TF-IDF) vectors to compute the relevance score of the web pages. Sekhar and coauthors[8] proposed a crawler based on the master-slave working principle. A graph-based ranking algorithm using TF-IDF was proposed to calculate the relatedness of the bioinformatics information enriched web pages. Srinivasan and coauthors [9] presented a crawler which used TF-IDF weighted average cosine similarity of the web page with respect to the topic to calculate the relatedness of the biomedical information source web pages. The target variables used in both the works were full-page text and anchor text. Average cosine similarity computation in the large web pages reveals to vast deviations in the result. This leads to the poor working of these crawlers.

Learning focused crawlers [10] overcomes the glitches by machine learning algorithms such as Naive Bayes (NB) [11, 12], Support Vector Machines (SVM) [13, 14], Decision Tree (DT) [15] and Artificial Neural Network (ANN) [16] to predict the topical relatedness of the web pages. These crawlers also employ the TF-IDF feature vectors to train the machine learning algorithms. Zowalla and coauthors[17] proposed a learning focused crawler for health information from the web. This work uses the SVM classifier to determine the relatedness of the web page. The TF-IDF-based features are extracted from the contents of the web page to train the SVM classifier. Approximately 87,562 web pages were collected from various medical sources used to train the SVM classifier. Abbasi and coauthors [18] proposed a two-tier graph propagation algorithm that leverages web graph data which has been used to filter out non-credible data from the web pages. Information gain Heuristic ranked n-gram-based features were used to train the SVM classifier. The trained SVM classifier predicts the topical and sentimental relatedness of the medical related web pages. Amalia and coauthors [19] used TF-IDF based features to train the NB classifier. The trained NB classifier was used to predict the topical relatedness of the health related web pages. Tang and coauthors [20] initiated a work which is a combination of quality and relevance scores. The quality was computed using the Relevance Feedback (RF) algorithm and the relevance score was computed using the Laplace corrected decision tree classifier. This hybrid approach helps to improve the crawling order of the medical related Uniform Resource Locators (URLs). Xu and coauthors [21] proposed a Rapid Automatic Keyword Extraction (RAKE) algorithm to identify a set of keywords on the web page. Keywords identified have been used as features for the training of the supervised learning model. An additive regression algorithm was used to calculate the relatedness score of the e-health related web pages using the extracted features. Yan and coauthors [22] presented a work which used both the VSM and the improved NB classifier to predict the topical relatedness of the medical related web pages.

The drawback of these learning crawlers is that if the size of the web pages increases or decreases, the feature space generated by the TF-IDF will also increase or decrease respectively. This inconsistent feature representation leads to poor results. Also, both the VSM crawlers and the learning crawlers consider only the lexical similarity and not the semantic similarity. This drawback leads to the low harvest rate of these crawlers.

Semantic focused crawlers were introduced to win over these obstacles and compute the semantic similarity between the topic and the target variables extracted from the web page. These semantic focused crawlers use domain-specific ontology to compute the topical relatedness of the web page. Semantic similarity algorithms such as Wu and Palmer [24], Resnik [26] and Li [29] have been used in the available literature to compute the semantic similarity between the topic and the target variables extracted from the web page. The major issue in this approach is that computing the semantic similarity between the topic word and the target variables on the web page is costlier and time-consuming in the dynamic internet.

Ontology learning-based approaches [30–33] have been introduced by combining both the ontology and the supervised learning methodologies to shrug off the snag. This type of crawler needs a huge dataset to train machine learning algorithms. Machine learning algorithms use domain-specific ontology to predict web page relevance. Zheng and coauthors [30] proposed a focused crawler for biomedical terms using Artificial Neural Network (ANN). This work uses Unified Medical Language System (UMLS) [34] ontology to compute the relevant concepts of the topic term. The term frequency of the relevant concepts extracted from the ontology in the web page is given as an input to the ANN to predict the relatedness of the web page. Learning similarity for each topic-web page pair using ontology increases both training time and crawling time. This approach improves the harvest rate but computationally costlier.

A scrutiny of the literature survey is listed as follows:

1. Research work is sparsely carried out in focused crawler for biomedical topics.
2. The existing focused crawlers struggled to handle the unstructured and complex biomedical queries.
3. The VSM and the learning crawlers considered only the lexical similarity and not semantic similarity which led to the low harvest rate.
4. Semantic and the ontology learning crawlers used domain specific ontology to compute the topical relevance of the web pages. These crawlers improved the harvest rate but is computationally costlier because of the static nature and the complex design of the ontology.
5. Design of ontology for the complex biomedical topics is also costlier.

*Contributions of this paper*

The beneficence of this paper are:

(1) This work proposed a new AE-SLSTM model with peephole connections to determine the topical relevance of the web page. The proposed model effectively computes the topical relevance of the web page.
(2) The attention mechanism introduced in the proposed model helps to maintain the fixed size vector representation of the topic and the web pages. This mechanism helps to handle the complex queries effectively.
(3) This work utilizes pre-trained GloVe embeddings to compute the input embedding vectors of the input text sequences.
(4) The Manhattan distance metric used in this work effectively computes the complex semantic similarity between the topic and the web pages. This improves the harvest rate of the proposed crawler.
(5) The proposed model is trained using the ADAM optimizer with Batch Normalization which handles the exploding gradient problem effectively.
(6) Five different focused crawlers namely the Breadth-First Search (BFS), VSM, Learning, Ontology learning (OL) and the proposed crawlers, are implemented and evaluated. Their performance is assessed using the harvest rate and the irrelevance ratio.

## Proposed Work

The proposed work consists of six agents namely, (i) crawl frontier, (ii) web page downloader, (iii) Validating Agent, (iv) web page repository, (v) parsing and extraction and (vi) Relevance Computation. Crawl frontier is a priority queue where the URLs are stored based on its priority. Seed URLs are initialized by the user in the crawl frontier which is the starting point of the crawling process. The seed URLs are then sent one by one to the web page downloader agent. This agent feeds the URLs to the validating agent for the validation of the downloadable web pages. If downloadable, the URL is returned to the web page downloader. The main goal of this component is to control the process of the web page downloader with the configuration policies such as selection policy, re-visit policy, politeness policy and the parallelization policy. The selection policy identifies the useful web pages for downloading. The re-visit policy checks for the dependency of the update on the web page. This policy checks whether the current content of the web page is similar to that of the previous visit or not. The politeness policy dodges the overloading of the web page. The parallelization policy handles the multithreading process. The web page downloader downloads the web page and stores in web page repository. The downloaded web pages are sent for parsing and then the extraction agent, where the HTML tags embedded in the web pages are removed and only the plain text is extracted. The parsing is done by using the lxml parser of BeautifulSoup python package. The extracted plain text is then sent to relevance computation module to check the relatedness of the web page with respect to the topic using AE-SLSTM algorithm. The specially designed mathematical framework is explained in Section 3.1. If the web page is predicted as relevant, all the URLs present in the web page is sent to the Crawl frontier for storage. These processes are repeated until a user defined depth has been reached.

### *Mathematical Model of the proposed Relevance Computation Agent*

### GloVe

GloVe [6] learns word representation owing to $(word, context)$ co-occurrence matrix, that combines the advantages of the local context window method and global matrix factorization. GloVe seeks to finely express each $wordw_i$ and each $contextw_j$ as $d$-dimensional vectors $\overrightarrow{w_i}$ and $\overrightarrow{w_j}$ by minimizing the cost function of the weighted least squares regression model as shown in Equation (1).

$$J = \sum_{i,j=1}^{V} f(X_{ij})\left({w_i}^T\widetilde{w_j} + b_i + b_j - \log X_{ij}\right)^2 \tag{1}$$

where $b_i$ and $b_j$ are $word$ and $context$specific biases respectively, the weighting function $f(X_{ij})$ can be set as follows in Equation (2):

$$f(x) = \begin{cases} \left(\frac{x}{x_{max}}\right)^{\alpha}, if x < x_{max} \\ 1, \ otherwise \end{cases} \tag{2}$$

So, when a pair of words that are extremely common is found (i.e. $X_{ij}$>$x_{max}$), the function cuts it off and returns 1. In the other case, the function returns a value in the range (0, 1) that is weighted by the value of $\alpha$, which is demonstrated to give the best performance outputs when $\alpha = \frac{3}{4}$.

Finally, by means of gradient descent and calculating the derivative of the cost function with respect to the important parameters ($w_i$, $w_j$, $b_i$, $b_j$), the values of the vectors get rectified through iterations until the cost function reaches a local minimum and the vectors reach a state of convergence. In this work, AdaGrad optimizer [35] is used to optimize the parameters.

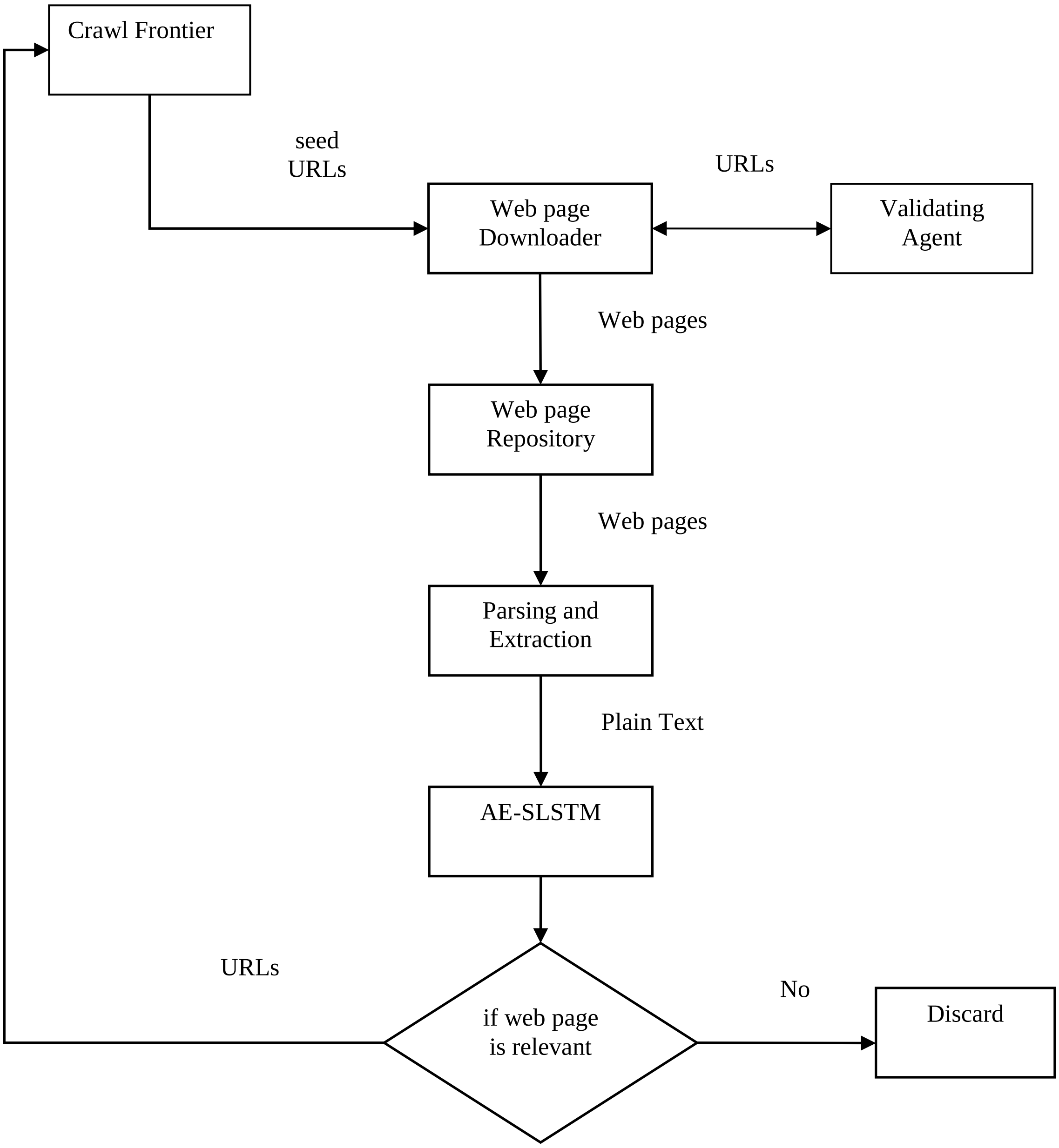


**Figure 1.** Proposed workflow architecture.

## Long Short Term Memory (LSTM) Network

The LSTM [5,36] has input gate, forget gate and output gate to solve the long term dependencies of basic RNN. The LSTM contains the peephole connections that are used to link the memory cells directly to the gates to learn the outputs in a precise time. Figure 2 shows the architecture of LSTM. At time step $t$, the formulas are given as follows from Equation (3) to Equation (7):

$$i_t = \sigma(w_{ix}x_t + w_{ih}h_{t-1} + p_i c_{t-1} + b_i) \tag{3}$$

$$f_t = \sigma\left(w_{fx}x_t + w_{fh}h_{t-1} + p_f c_{t-1} + b_f\right) \tag{4}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \varphi(w_{cx}x_t + w_{ch}h_{t-1} + b_c) \tag{5}$$

$$o_t = \sigma(w_{ox}x_t + w_{oh}h_{t-1} + p_o c_t + b_o) \tag{6}$$

$$h_t = o_t \odot \varphi(c_t) \tag{7}$$

Where $x_t$ and $h_{t-1}$ are the input and the recurrent vectors respectively, $i_t, o_t$ and $f_t$ are the input, output and the forget gate vectors respectively, $w_{ix}$ and $w_{ih}$ are the input and the recurrent weight matrices of the input gate respectively,$w_{ox}$ and $w_{oh}$ are the input and the recurrent weight matrices of the output gate respectively, $w_{fx}$ and $w_{fh}$ are the input and the recurrent weight matrices of the forget gate respectively, $b_i, b_o$ and $b_f$ are the bias vectors of the input, output and the forget gate respectively, $p_i, p_o$ and $p_f$ are the peep hole connection parameters of the input, output and the forget gates respectively, $\sigma$ and $\varphi$ are the sigmoid and the tangent hyperbolic functions respectively and $\odot$ represents the element wise multiplication of vectors.

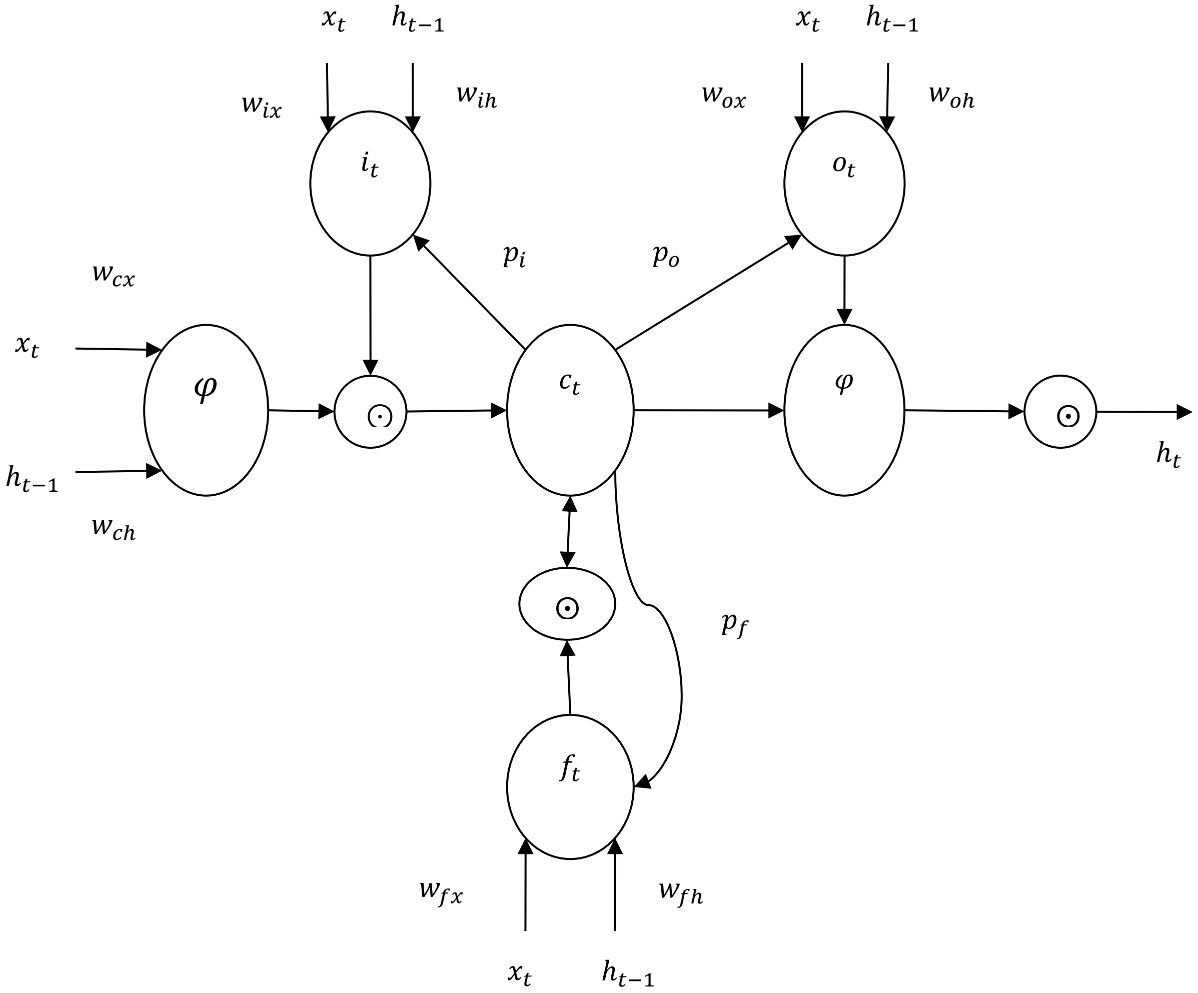
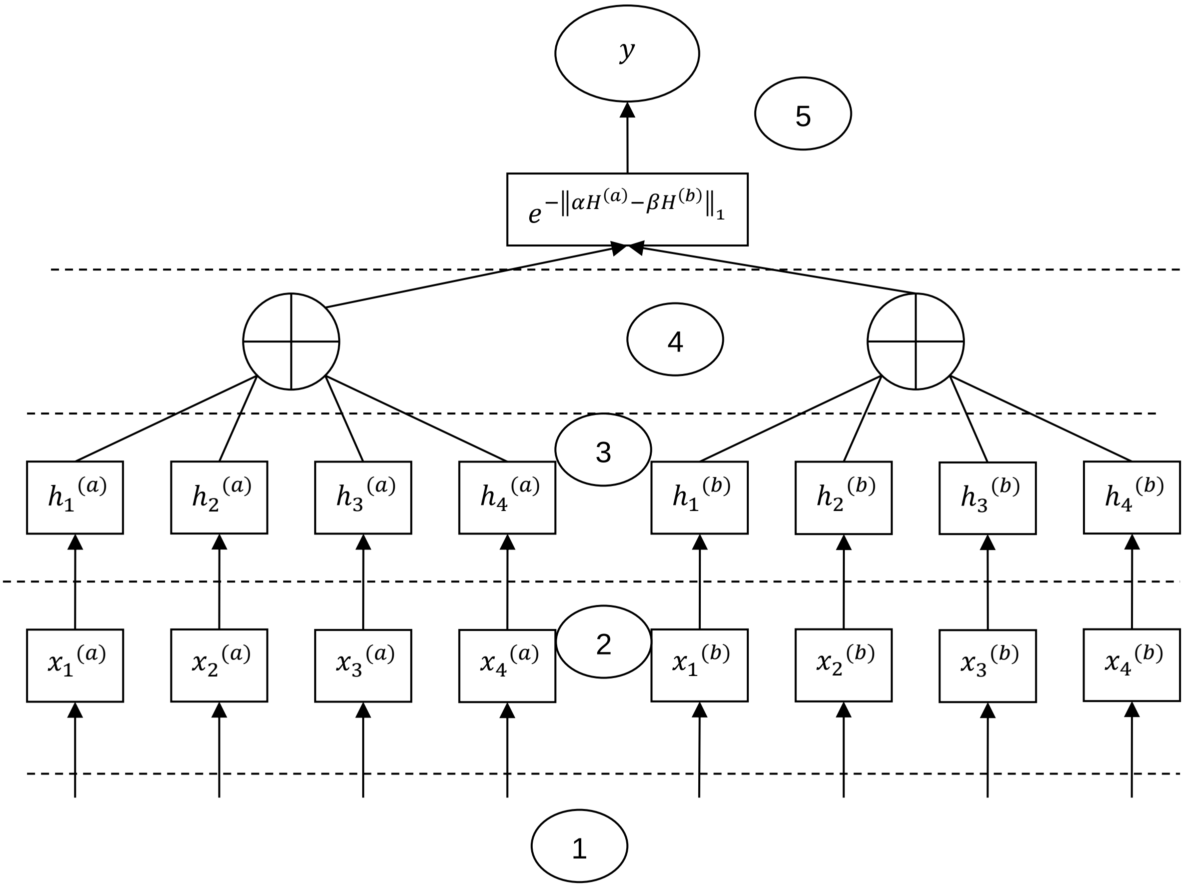


**Figure 2.** LSTM architecture with peephole connections.

Attention Enhanced Siamese Long Short Term Memory (AE-SLSTM) Networks

Figure 3 shows the proposed AE-SLSTM architecture. The proposed Attention Enhanced Siamese LSTM model contains two LSTM layers. The first LSTM layer is $LSTM_a$ which handles the given topic

$({x_1}^{(a)}, {x_2}^{(a)}, \dots, {x_{T_a}}^{(a)})$ and the second LSTM layer is $LSTM_b$ which handles the web page text $({x_1}^{(b)}, {x_2}^{(b)}, \dots, {x_{T_b}}^{(b)})$.



1. Input Layer, 2. Embedding Layer, 3. Hidden Layer, 4. Attention Layer, 5. Output Layer

**Figure 3.** Architecture of Siamese Enhanced Long Short Term Memory (AE-SLSTM) Networks.

The proposed AE-SLSTM model contains five layers: the first is the input layer where the topic is given as an input to $LSTM_a$ and the web page text is given as input to $LSTM_b$. The second is the embedding layer where the input topic and the web page text are represented in a low dimension vector using the GloVe word embedding model discussed in section 3.1. The third is the hidden layer where the high level features are learned. Here $H_a = [{h_1}^{(a)}, {h_2}^{(a)}, {h_3}^{(a)}, \dots, {h_{T_a}}^{(a)}]$ and $H_b = [{h_1}^{(b)}, {h_2}^{(b)}, {h_3}^{(b)}, \dots, {h_{T_b}}^{(b)}]$are the feature vectors of the topic and web page text where $T_a$ is length of the topic and $T_b$ is length of the web page text. $\alpha$is the weight of the topic $LSTM_a$ and $\beta$ is weight of the web page text $LSTM_b$. These hidden vectors are given as input to the fourth layer that is attention layer where it will produce weight vectors. Here the final vector representation of a topic $(r_a)$ and the web page text $(r_b)$ is computed as follows from Equation (8) to Equation (13):

$$M_a = \tanh(H_a) \quad (8)$$

$$\alpha = softmax(M_a) \quad (9)$$

$$r_a = H_a.\alpha \quad (10)$$

$$M_b = \tanh(H_b) \quad (11)$$

$$\beta = softmax(M_b) \quad (12)$$

$$r_b = H_b.\beta \quad (13)$$

Then the calculated $r_a$ and $r_b$vectors are sent to the output layer to predict the topical relevance. The semantic similarity between the topic vector and web page text vector is calculated by Manhattan distance metric. The Manhattan distance is formulated as follows in Equation (14):

$$g = e^{-\|r_a - r_b\|_1} \tag{14}$$

Where $g \in [1,5]$.

The AE-SLSTM model predicts the similarity between the topic and the web pages using $g$. This AE-SLSTM model is trained as a regression model using Back propagation through time (BPTT) under the Mean Squared Error loss function.

The optimization of the parameters in the proposed AE-SLSTM model is performed using the ADAM [37,38] gradient optimization algorithm with batch normalization to handle the exploding gradient problem effectively.

## Experimental Design and Analysis

The five focused crawlers implemented in this work are the BFS, VSM, learning, ontology learning, and the proposed crawlers. The crawler prototype was built with the Python3 [39], on the Spyder3.6 [40] platform. The 10 topics along with their seed URLs serve as inputs to all five focused web crawlers, as shown in Table 1. The performance of the five focused web crawlers are evaluated by the metrics, harvest rate and irrelevance ratio. The experimental results were analyzed and discussed in order to assess the efficiency of the five focused crawlers in section 5. Approximately 200,000 web pages have been manually collected from the internet for the ten topics listed in Table 1 to train machine learning algorithms. Approximately 20,000 web pages were collected for each topic, including 10,000 positive and negative samples. Out of 10000 positive samples, 5000 web pages were directly relevant and the other 5000 web pages were indirectly relevant. This division was done in order to check the crawler's capacity to download the indirectly relevant web pages. In this work, two seed URLs are used for each topic but depends on the depth of the crawling seed URLs can be increased or decreased.

**Table 1.**Topic and Initial URLs.

| S.No | Topic | Seed URL |
|---|---|---|
| 1 | Mitochondrion | https://www.britannica.com/science/mitochondrion<br>http://www.biology4kids.com/files/cell_mito.html |
| 2 | Glucose Transporter | https://www.frontiersin.org/articles/300363<br>https://www.ncbi.nlm.nih.gov/pmc/articles/PMC5425736/ |
| 3 | Amino Acid | https://www.britannica.com/science/amino − acid<br>https://pubmed.ncbi.nlm.nih.gov/8045288/ |
| 4 | Anemia | https://en.wikipedia.org/wiki/Anemia<br>https://www.medicalnewstoday.com/articles/158800 |
| 5 | Cholera | https://www.cdc.gov/typhus/epidemic/index.html<br>https://www.webmd.com/a − to − z − guides/what − is − typhus |
| 6 | Abarelix | https://www.rxlist.com/plenaxis − drug.htm<br>https://www.centerwatch.com/directories/1067 − fda − approved − drugs/listing/4014 − plenaxis − abarelix − for − injectable − suspension |
| 7 | Paraben | https://www.sigmaaldrich.com/catalog/substance/chlorobutanol18646600164511? lang = en&region = NL<br>http://www.athenstaedt.de/chlorobutanol − en.htm |
| 8 | Anesthesiology, also spelled anaesthesiology, medical specialty dealing withanesthesia and related matters, including resuscitation and pain. | https://pubs.asahq.org/anesthesiology<br>https://www.anesthesiologynews.com/ |
| 9 | Dermatology is a branch of medicine that deals with the skin | https://www.aad.org/<br>https://www.medicalnewstoday.com/articles/286743 |
| 10 | Ophthalmology is the study of medical conditions relating to the eye. | https://www.radboudumc.nl/en/research/departments/ophthalmology<br>https://www.medicalnewstoday.com/articles/326753 |

### *Performance Evaluation of AE-SLSTM*

The performance of the proposed AE-SLSTM algorithm with peephole connections is compared with the Siamese LSTM [4], Siamese bidirectional LSTM [41] and the Attentive Siamese LSTM without peephole connections [3] using the metrics Pearson's correlation co-efficient, Spearman's correlation co-efficient and the Mean Squared Error to prove that the proposed algorithm performed better than the existing algorithms. For training, the dataset discussed in Section 4 is used. ADAM optimizer with Batch Normalization is adopted to train all the learning algorithms. All the models for 12 epochs are executed successfully because maximum accuracy is achieved at the twelfth epoch.

Initially a pre-trained word embedding model is designed with around 100,000,000 words related to biomedical topics using GloVe. The GloVe model is trained using AdaGrad [35] optimizer algorithm. This pre-trained word embedding model is employed to convert the input topic and webpage text sequences into input embedding vectors for the $LSTM_a$ and $LSTM_b$ at the input layer for all the four models (Siamese LSTM, Siamese bidirectional LSTM, the Attentive Siamese LSTM and the proposed). These input embedding vectors are then passed through various layers of their corresponding architecture to produce the Manhattan distance score. This Manhattan distance score is then used to predict the semantic similarity between the text sequences.

The result comparison is shown in the Table 2. The Siamese LSTM and the Siamese Bidirectional LSTM model only considers the last hidden state vector of the topic and the web page text sequences in computing the Manhattan score revealing poor performance. The peephole connection in the LSTM helps to learn the memory cells directly. The proposed attention mechanism considers all the hidden state vectors to compute the Manhattan score which boosts up the performance. Hence the results manifested that the proposed method performed better than the existing methods in computing the semantic similarity between text sequences.

**Table 2.** Result Comparison of Siamese LSTM, Siamese Bidirectional LSTM, Attentive Siamese LSTM without peephole connections and the Attentive Siamese LSTM with peephole connections

| **Model** | **Pearson's correlation co-efficient** | **Spearman's correlation co-efficient** | **Mean Squared Error** |
|---|---|---|---|
| Siamese LSTM | 0.7923 | 0.7891 | 0.3998 |
| Siamese bidirectional LSTM | 0.7987 | 0.7931 | 0.3456 |
| Attentive Siamese LSTM without peephole connections | 0.8127 | 0.8014 | 0.3141 |
| Attentive Siamese LSTM with peephole connections (proposed) | 0.8149 | 0.8113 | 0.2934 |

*Performance Evaluation of crawling phase*

Performance Metrics

Harvest Rate

The harvest rate ($\mathrm{h_{rate}}$) is the ratio of the downloaded web pages which is relevant ($\mathrm{R_D}$) to the entirely crawled web pages ($\mathrm{R_T}$) and is calculated using Equation (15).

$$\mathrm{h_{rate}} = \frac{\mathrm{R_D}}{\mathrm{R_T}} \tag{15}$$

Irrelevance Ratio

The irrelevance ratio ($\mathrm{p_{rate}}$) is the ratio of the downloaded web pages which is irrelevant ($\mathrm{r_D}$) from the entirely crawled web pages ($\mathrm{r_T}$) to the total pages downloaded ($\mathrm{r_T}$) and is calculated using Equation (16).

$$\mathrm{p_{rate}} = \frac{\mathrm{r_D} - \mathrm{r_T}}{\mathrm{r_T}} \tag{16}$$

*Analysis of crawling phase*

The result analysis of the proposed work compared with the existing BFS, VSM, the learning, and the ontology learning crawlers is performed at this phase. The result analysis of the crawling phase for the existing crawlers is executed in three stages. The first is related to their first four topics shown in Table 1 where the seed URLs are directly relevant to the topic. The second is related to the topics (5-7) shown in Table 1 where the seed URLs are indirectly relevant to the topic. The third is related to the topics (8-10) shown in Table 1 where the topics are the sentences.

Initially, comparison is made by implementing BFS, VSM, the learning, ontology learning and the proposed crawler for the first four topics as shown in Table 1. BFS (42) crawler is a sequential crawler that works on the basis of the First-In First-Out (FIFO) principle. BFS crawler downloads all the downloadable URLs in the crawl frontier using the FIFO algorithm, without considering the relatedness of the web page due to the absence of the relevance computation module. Considering four (1-4) biomedical topics as shown in Table 1, after 5000 webpage downloads, the average $h_{rate}$ and the average $p_{rate}$ of the BFS crawler is 0.13 and 0.87 respectively.

VSM crawler which is a focused crawler implements the average cosine similarity metric to calculate the relatedness of the web page to the topic. These crawlers used TF-IDF weighted vectors, which calculates the similarity based on co-occurrence by assigning more weightage to infrequent words and low weightage to frequent words. The similarity score is found only when the topic term co-occurs in the target variables. This prevents the crawlers to download semantically relevant web pages on the particular topic. The VSM crawler produced an average $h_{rate}$ of 0.21 and an average $p_{rate}$ of 0.79 for first four biomedical topics as shown in Table 1 after 5000 webpage downloads.

The learning crawler used SVM classifier to predict the topical relatedness of the web page. Here the TF-IDF feature vectors are extracted from the web page and given as input to the SVM for prediction. The TF-IDF-based learning crawler faces two major issues (i) the TF-IDF crawler is based on co-occurrence, i.e. it calculates the relevance of the web page only if the topic term co-occurs in the target variables of the web page. (ii) As the number of words on the web page increases, the feature space generated by TF-IDF based crawlers also increases resulting in a lower performance of SVM algorithm. These drawbacks influenced the SVM along with the TF-IDF crawler to produce an average $h_{rate}$ of 0.27 and an average $p_{rate}$ of 0.73.

The ANN-based ontology learning crawler selects the synonyms of the topic term from the domain-specific ontology and then calculates the term frequency of the synonyms for the topic term. These term frequencies are then used as inputs to the ANN to predict the relatedness of the web page. The downside of this is that it considers only the synonyms of the web pages in order to calculate the relatedness of the web page and only uses full page text as target variables. This results in an average $h_{rate}$ of 0.34 and an average $p_{rate}$ of 0.66.

The proposed AE-SLSTM crawler uses pre-trained GloVe model to compute the input embedding vectors, which are then used to train the two LSTM layers considering one for the topic and the other for the web page. Their result is then sent to the output layer to compute the Manhattan distance. The web page is predicted as relevant if the Manhattan distance score is greater than or equal to the 3. The proposed crawler generates an average $h_{rate}$ of 0.39 and an average $p_{rate}$ of 0.61. The Manhattan distance metric computes the complex semantic relation between the topic and the web page that aids to improve the harvest rate. Figure4 shows the result comparison of BFS, VSM, learning, ontology learning and the proposed crawler with respect to the average $h_{rate}$ and Figure 5 shows the result comparison of BFS, VSM, learning, ontology learning and the proposed crawler with respect to the average $p_{rate}$ where the seed URLs are directly relevant to the topic.

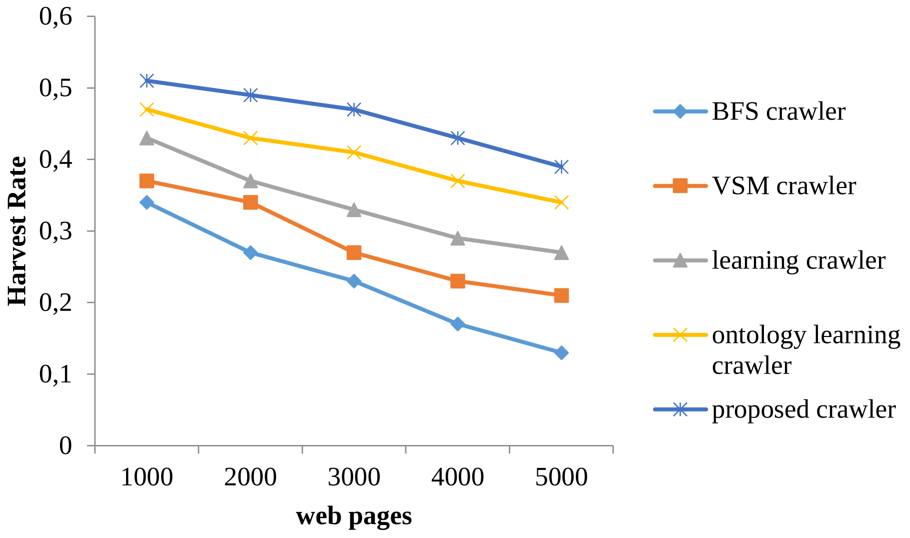


**Figure 4.** Average Harvest Rate of BFS, VSM, learning, ontology learning and the proposed crawler where the seed URLs are directly relevant.

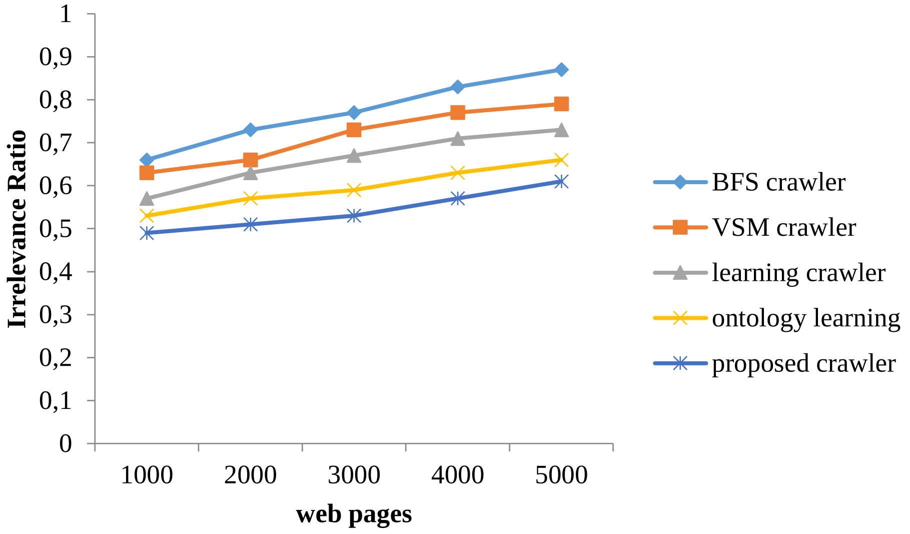


**Figure 5.** Average Irrelevance Ratio of BFS, VSM, learning, ontology learning and the proposed crawler where the seed URLs are directly relevant.

The second stage of the comparison was performed between the BFS, VSM, learning, ontology learning and the proposed crawler for the topics (5-7) as shown in Table 1. The second sub stage was conducted to prove that the proposed crawler performed better than the existing BFS, VSM, learning and the ontology learning crawler for the seed URLS indirectly relevant to the topic. After crawling 5000 web pages, the BFS, VSM, learning, ontology learning crawlers and the newly designed crawler produced an average $h_{rate}$ of 0.13, 0.16, 0.21, 0.26, 0.29 respectively and an average $p_{rate}$ of 0.87, 0.84, 0.79, 0.74, 0.71 respectively. The result proves that the proposed crawler downloaded more relevant web pages than the existing crawlers did, although the web pages are indirectly relevant. The specially designed Manhattan distance metric effectively computes the complex semantic relationship between the topic and the web pages. This metric helps to improve the harvest rate of the proposed crawler for the indirectly relevant web pages for the given topic. Figure6 shows the result comparison of BFS, VSM, learning, ontology learning and the proposed crawlers with respect to the average $h_{rate}$ and Figure7 shows the result comparison of BFS, VSM, learning, ontology learning and the proposed crawlers with respect to the average $p_{rate}$ where the seed URLs are indirectly relevant.

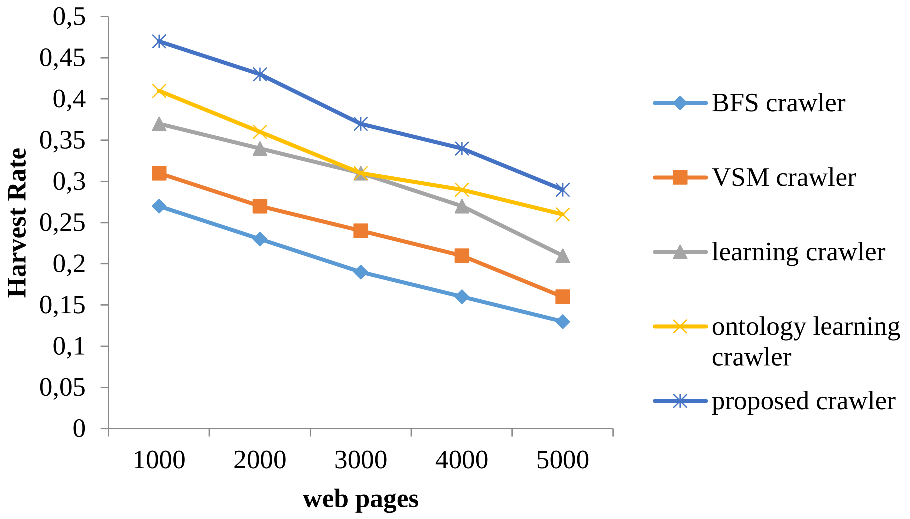


**Figure 6.** Average harvest rate of BFS, VSM, learning, ontology learning and the proposed crawler where seed URLs are indirectly relevant.

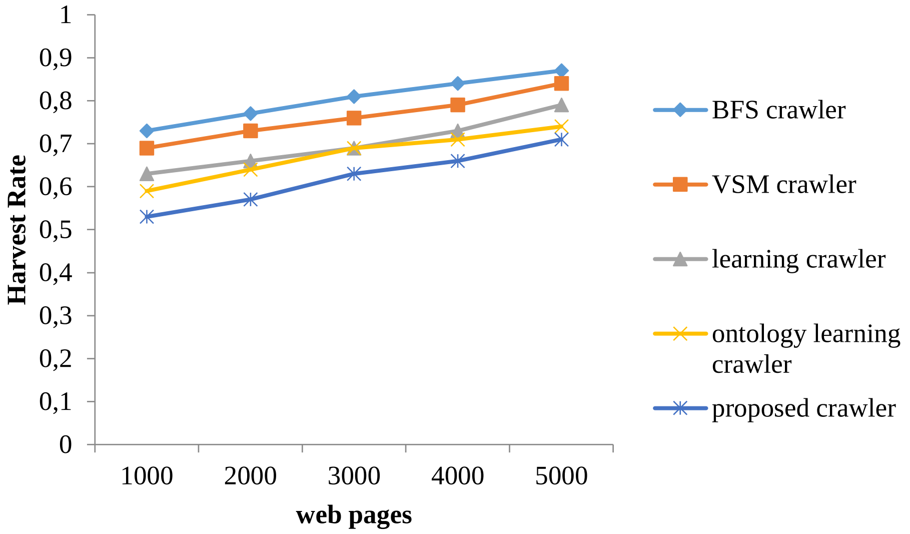


**Figure 7.** Average irrelevance ratio of BFS, VSM, learning, ontology learning and the proposed crawler where seed URLs are indirectly relevant.

The third stage of the comparison was performed between the BFS, VSM, learning, ontology learning and the proposed crawler for the topics (8-10) as shown in Table 1. The third sub stage was conducted to prove that the proposed crawler performed better than the existing BFS, VSM, learning and the ontology learning crawlers for the sentence based topics. After 5000 web page crawls, the BFS, VSM, learning, ontology learning and the newly designed crawler produced an average $h_{rate}$ of 0.16, 0.23, 0.27, 0.31, 0.34 respectively and an average $p_{rate}$ of 0.84, 0.77, 0.73, 0.69, 0.66 respectively. This certainly proves that the newly designed crawler computes the semantic similarity between two sentences potentially. The ability of the two LSTM layers in handling the sentences helps to compute the semantic similarity between the sentence pairs effectively. This helps to improve the harvest rate of the proposed crawler. Figure 8 shows the comparative result of BFS, VSM, learning, ontology learning and the proposed crawler with respect to the average $h_{rate}$ and Figure9 shows the comparative result of BFS, VSM, learning, ontology learning and the proposed crawler with respect to the average $p_{rate}$ for the sentence based topics.

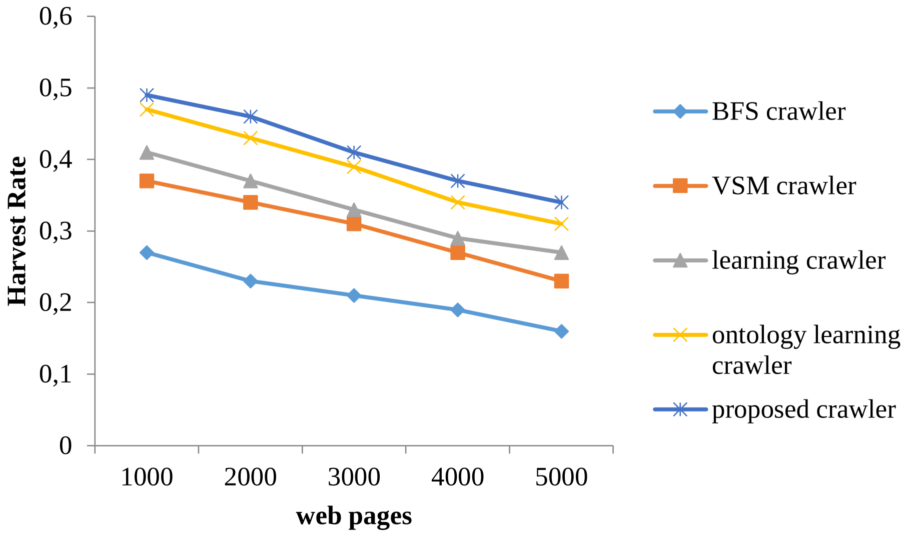


**Figure 8.** Average harvest rate of BFS, VSM, learning, ontology learning and the proposed crawler for the sentence based topics.

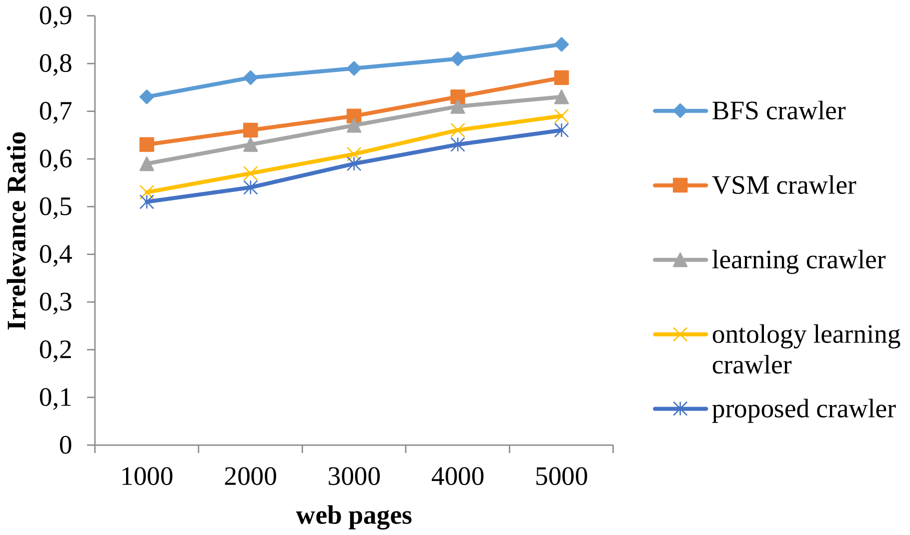


**Figure 9.** Average irrelevance ratio of BFS, VSM, learning, ontology learning and the proposed crawler for the sentence based topics.

## CONCLUSION

The scattered, diverse, non compatible and complex nature of biomedical data displays poor crawling of biomedical related web pages. In this paper, we proposed a novel, focused crawler for biomedical information implementing AE-SLSTM algorithm. The proposed work enhanced with attention mechanism, effectively handles the biomedical data and also improves the semantic similarity computation of the crawler. The proposed crawler is compared with various crawlers available in the literature. The proposed methodology produced an average harvest rate of 0.39 and an average irrelevance ratio of 0.61. The experimental results had proved that the proposed crawler outperformed the existing focused crawlers with respect to harvest rate and irrelevance ratio which outwardly enhanced the performance of the focused crawler for biomedical relevant web pages.

## REFERENCES

1. Bellazzi R, Masseroli M, Murphy S, Shabo A, Romano P. Clinical Bioinformatics: Challenges and opportunities. BMC Bioinformatics. 2012;13(SUPPL 1):1–8.
2. Brin S, Page L. The anatomy of a large-scale hypertextual Web search engine BT - Computer Networks and ISDN Systems. Comput Networks ISDN Syst [Internet]. 1998;30(1–7):107–17. Available from: http://dx.doi.org/10.1016/S0169-7552(98)00110-X%5Cnhttp://apps.webofknowledge.com/full_record.do?product=UA&search_mode=GeneralSearch&qid=6&SID=X1pOWPMuSmOv1SlwJ6f&page=1&doc=2
3. Bao W, Bao W, Du J, Yang Y, Zhao X. Attentive Siamese LSTM Network for Semantic Textual Similarity Measure. Proc 2018 Int Conf Asian Lang Process IALP 2018. 2019;312–7.
4. Mueller J, Thyagarajan A. Siamese recurrent architectures for learning sentence similarity. 30th AAAI Conf Artif Intell AAAI 2016. 2016;(2014):2786–92.
5. Sutskever I, Vinyals O, Le Q V. Sequence to sequence learning with neural networks. Adv Neural Inf Process Syst. 2014;4(January):3104–12.
6. Jeffrey Pennington, Richard Socher CDM. GloVe: Global Vectors for Word Representation Jeffrey. Proc 2014 Conf Empir Methods Nat Lang Process. 2017;1532–43.
7. Salton G, Wong A, Yang C. Information Retrieval and Language Processing: A Vector Space Model for Automatic Indexing. Commun ACM. 1975;18(11):613–20.
8. Mani Sekhar SR, Siddesh GM, Manvi SS, Srinivasa KG. Optimized focused Web Crawler with Natural Language Processing based relevance measure in bioinformatics web sources. Cybern Inf Technol. 2019;19(2):146–58.
9. Srinivasan P, Mitchell J, Bodenreider O, Pant G, Menczer F. Web Crawling Agents for Retrieving Biomedical Information. Proc Int Work Agents Bioinforma [Internet]. 2002;(January 2013). Available from: http://informatics.indiana.edu/fil/Papers/bixmas.pdf
10. Dhanith PRJ, Surendiran B, Raja SP. A Word Embedding Based Approach for Focused Web Crawling Using the Recurrent Neural Network. 2020;1–11.
11. Pawar N, Rajeswari K, Joshi A. Implementation of an Efficient web crawler to search medicinal plants and relevant diseases. Proc - 2nd Int Conf Comput Commun Control Autom ICCUBEA 2016. 2017;48:87–92.
12. Saleh AI, Abulwafa AE, Al Rahmawy MF. A web page distillation strategy for efficient focused crawling based on optimized Naïve bayes (ONB) classifier. Appl Soft Comput J [Internet]. 2017;53:181–204. Available from: http://dx.doi.org/10.1016/j.asoc.2016.12.028
13. Pant G, Srinivasan P. Link contexts in classifier-guided topical crawlers. IEEE Trans Knowl Data Eng. 2006;18(1):107–22.
14. Peng T, Liu L. Focused crawling enhanced by CBP-SLC. Knowledge-Based Syst [Internet]. 2013;51:15–26. Available from: http://dx.doi.org/10.1016/j.knosys.2013.06.008
15. Li J, Furuse K, Yamaguchi K. Focused crawling by exploiting anchor text using decision tree. Spec Interes tracks posters 14th Int Conf World Wide Web - WWW '05 [Internet]. 2005;1190. Available from: http://portal.acm.org/citation.cfm?doid=1062745.1062933
16. Menczer F, Menczer F, Pant G, Pant G, Srinivasan P, Srinivasan P. Topical Web Crawlers: Evaluating Adaptive Algorithms. ACM Trans Internet Technol. 2003;V(February):38.
17. Zowalla R, Wetter T, Math D, Pfeifer D. Crawling the German Health Web : Exploratory Study and Graph Analysis Corresponding Author : 2020;22:1–22.
18. Abbasi A, Fu T, Zeng D, Adjeroh D. Crawling credible online medical sentiments for social intelligence. Proc - Soc 2013. 2013;254–63.
19. Amalia A, Gunawan D, Najwan A, Meirina F. Focused crawler for the acquisition of health articles. Proc 2016 Int Conf Data Softw Eng ICoDSE 2016. 2017;
20. Tang TT, Hawking D, Craswell N, Griffiths K. Focused crawling for both topical relevance and qualify of medical information. Int Conf Inf Knowl Manag Proc. 2005;147–54.
21. Xu S, Yoon HJ, Tourassi G. A user-oriented web crawler for selectively acquiring online content in e-health research. Bioinformatics. 2014;30(1):104–14.
22. Yan H. Internet medicine information monitoring system based on focused crawler. 3rd Int Conf Inf Sci Interact Sci Chengdu. 2010;452–6.
23. Du Y, Liu W, Lv X, Peng G. An improved focused crawler based on Semantic Similarity Vector Space Model. Appl Soft Comput J [Internet]. 2015;36:392–407. Available from: http://dx.doi.org/10.1016/j.asoc.2015.07.026
24. Wu Z, Palmer M. Verbs semantics and lexical selection. 1994;133–8.
25. Dong H, Hussain FK. Self-adaptive semantic focused crawler for mining services information discovery. IEEE Trans Ind Informatics. 2014;10(2):1616–26.

26. Resnik P. Using Information Content to Evaluate Semantic Similarity in a Taxonomy. 1995;1. Available from: http://arxiv.org/abs/cmp-lg/9511007
27. Joe Dhanith PR, Surendiran B. An ontology learning based approach for focused web crawling using combined normalized pointwise mutual information and Resnik algorithm. Int J Comput Appl [Internet]. 2019;0(0):1–7. Available from: https://doi.org/1206212X.2019.1684023
28. Capuano A, Rinaldi AM, Russo C. An ontology-driven multimedia focused crawler based on linked open data and deep learning techniques. Multimed Tools Appl. 2019;
29. Li Y, Bandar ZA, McLean D. An approach for measuring semantic similarity between words using multiple information sources. IEEE Trans Knowl Data Eng. 2003;15(4):871–82.
30. Zheng HT, Kang BY, Kim HG. An ontology-based approach to learnable focused crawling. Inf Sci (Ny) [Internet]. 2008;178(23):4512–22. Available from: http://dx.doi.org/10.1016/j.ins.2008.07.030
31. Hussain HD and FK. SOF: a semi-supervised ontology-learning-based focused crawler. Concurr Comput Pract Exp. 2013;25(6):1755–70.
32. Chang S, Yang G, Jianmei Y, Bin L. An efficient adaptive focused crawler based on ontology learning. Proc - HIS 2005 Fifth Int Conf Hybrid Intell Syst. 2005;2005:73–8.
33. Hassan T, Cruz C, Bertaux A. Ontology-based approach for unsupervised and adaptive focused crawling. Proc Int Work Semant Big Data, SBD 2017 - conjunction with 2017 ACM SIGMOD/PODS Conf. 2017;1–6.
34. Bodenreider O. The Unified Medical Language System (UMLS): Integrating biomedical terminology [Internet]. Vol. 32, Nucleic Acids Research. 2004. Available from: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC308795/
35. Duchi J, Hazan E, Singer Y. Adaptive subgradient methods for online learning and stochastic optimization. COLT 2010 - 23rd Conf Learn Theory. 2010;257–69.
36. Palangi H, Deng L, Shen Y, Gao J, He X, Chen J, et al. Deep Sentence embedding using long short-term memory networks: Analysis and application to information retrieval. IEEE/ACM Trans Audio Speech Lang Process. 2016;24(4):694–707.
37. Kingma DP, Ba JL. Adam: A method for stochastic optimization. 3rd Int Conf Learn Represent ICLR 2015 - Conf Track Proc. 2015;1–15.
38. Ruder S. An overview of gradient descent optimization algorithms. 2016;1–14. Available from: http://arxiv.org/abs/1609.04747
39. Rossum G van. Python tutorial, Technical Report CS-R9526. Cent voor Wiskd en Inform (CWI), Amsterdam. 1995;
40. Spyder. Spyder Ide [Internet]. Spyder Project. 2018. Available from: https://www.spyder-ide.org/
41. Zhu Z, He Z, Tang Z, Wang B, Chen W. A semantic similarity computing model based on siamese network for duplicate questions identification. CEUR Workshop Proc. 2018;2242:44–51.
42. Najork M, Wiener JL. Breadth-first search crawling yields high-quality pages. Proc 10th Int Conf World Wide Web, WWW 2001. 2001;114–8.