Article

Semantic Features with Contextual Knowledge-Based Web Page Categorization Using the GloVe Model and Stacked BiLSTM

Amit Kumar Nandanwar * and Jaytrilok Choudhary

Computer Science & Engineering Department, Maulana Azad National Institute of Technology,
Bhopal 462003, Madhya Pradesh, India; jaytrilok@manit.ac.in

* Correspondence: amit.173112002@manit.ac.in; Tel.: +91-9827730050

Abstract: Internet technologies are emerging very fast nowadays, due to which web pages are generated exponentially. Web page categorization is required for searching and exploring relevant web pages based on users' queries and is a tedious task. The majority of web page categorization techniques ignore semantic features and the contextual knowledge of the web page. This paper proposes a web page categorization method that categorizes web pages based on semantic features and contextual knowledge. Initially, the GloVe model is applied to capture the semantic features of the web pages. Thereafter, a Stacked Bidirectional long short-term memory (BiLSTM) with symmetric structure is applied to extract the contextual and latent symmetry information from the semantic features for web page categorization. The performance of the proposed model has been evaluated on the publicly available WebKB dataset. The proposed model shows superiority over the existing state-of-the-art machine learning and deep learning methods.

Keywords: bidirectional long short-term memory; contextual features; GloVe; deep learning; semantic features; web page categorization

Citation: Nandanwar, A.K.; Choudhary, J. Semantic Features with Contextual Knowledge-Based Web Page Categorization Using the GloVe Model and Stacked BiLSTM. *Symmetry* **2021**, *13*, 1772. <https://doi.org/10.3390/sym13101772>

Academic Editors: José Carlos R. Alcántud and Lorentz Jäntschi

Received: 21 August 2021

Accepted: 18 September 2021

Published: 23 September 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Nowadays, information available on the World Wide Web (WWW) is growing exponentially, due to which finding user-relevant web pages has become challenging and tedious. The Search engine returns web pages as per the query submitted by the user, but these web pages may or may not be relevant to the user. For example, a search engine either returns too many results or misinterprets the user query due to linguistic ambiguity [1]. Typically, search engines collect the results of prior user searches to respond to future queries depending on the previous results with the highest hits [2]. Thus, there is a need for an efficient web page categorization method as per the user's requirements. Web page categorization is a supervised learning problem in which a set of labeled categories are assigned to web pages. Web page categorization plays a crucial role in improving the performance of search engines. Due to a lack of semantic and contextual information, many search engines rely solely on keywords and suffer from linguistic ambiguity. The main objective of the proposed work was to develop an effective technique for resolving linguistic ambiguity by considering the semantic and contextual knowledge of the web pages. Information retrieval, parenteral filtering, a recommendation system, and focused crawling are popular application areas of web page categorization [3,4].

Many researchers have concentrated on categorizing web pages based on textual information by counting the appearance of the terms in the textual corpus. This is called frequency-based web page categorization, and it examines the frequency with which text appears on the web page [5]. Content information such as 'title' 'description' and structure information such as 'keywords' and 'tags' may be available on web pages. In recent years, researchers have used structure-based feature vectors such as HTML tags to categorize web pages [6,7].

The term frequency and structure-based feature vector approaches are insufficient to discover the semantic meaning of textual information, and text sequence information remains unutilized. Researchers have used the Natural Language Processing (NLP) word embedding technique and the Long Short Term Memory (LSTM) layer in recent advances in deep learning [8]. The application domains of stated techniques include text classification, chatbots, sentiment analysis, object classification, and sequence problems [9–11]. The word embedding technique has been frequently utilized in NLP to produce feature vectors that are semantically correlated. To extract contextual features and latent symmetry information from sequential terms, the BiLSTM model has been utilized [10,12].

The principal approaches for the web page categorization discussed in the next section are based on the set of textual and structural features of the web pages. Previous methods lack the ability to utilize the semantic and contextual features of the web pages. Recent advancements in deep learning, specifically in NLP and LSTM, have motivated what is proposed in the present work. In this paper, a model is proposed for web page categorization that utilizes the GloVe word embedding model to capture the semantic features of terms and transform words into word vectors of web pages. In addition, the Stacked BiLSTM model has also been applied to conduct the feature extraction of sequential word vectors. Finally, the Softmax classifier has effectively categorized web pages based on rich semantic and contextual information. The main contributions of the proposed work are:

- The combination of GloVe and BiLSTM is used to categorize the web pages by resolving the linguistic ambiguity problem. Linguistic ambiguity is the ability to derive more than one meaning from a user query and is also referred to as lexical ambiguity in some articles [13].
- Previous research on the web page categorization mentioned in the next section uses term frequency of the text and structure of the web pages [5,6,14]. In comparison, the semantic and contextual features for efficient web page categorization are novel ideas suggested in this paper.

The rest of this paper is organized as follows. Section 2 presents the background and related works on web page classification and categorization. Section 3 describes the proposed model for the web page categorization. The experimental setup, dataset description, and hyperparameter settings are described in Section 4. Section 5 presents the experimental results and a performance comparison of the proposed model with other methods and existing methods. Finally, Section 6 concludes this work.

2. Background and Related Work

Earlier research approaches have addressed the problem of web page classification as per the user's preferences [15]. This is a simple document classification problem based on the textual contents and features of web pages. On the other hand, the classification of web pages is based on counting the frequency of text terms to form a term frequency feature vector. These feature vectors are applied to train the classifier to classify web pages. For example, Lipras et al. [16] classified the web pages of news articles into four categories using the frequency of unigram, bigram, trigrams, and four grams feature vector by random forest classifier.

Li et al. [5] proposed an entity similarity network based on Wikipedia to categorize web pages into ten classes using the term frequency feature vector. Feature vectors extracted from the title and main text of the web pages were utilized by the naive Bayesian classifier. The accuracy was improved by applying the probability distribution function (PDF) on all the terms in the feature vector; a Wikipedia entity word created these probability distribution functions.

Document frequency-based feature vectors have been used by past researchers to categorize web pages. Jinhua et al. [17] proposed a semi-supervised learning algorithm to categorize WebKB and 20 newsgroup web pages based on their textual content. In this study, a semi-supervised algorithm based on expectation maximization presented

the best results. The F1-score of this method was superior to the supervised machine learning algorithm.

Mulahuwaish et al. [18] recently proposed a method to classify web documents using the Support Vector Machine (SVM), Decision Tree (DT), and K-Nearest Neighbour (KNN). These approaches rely on document frequency-based features to improve classification efficiency.

The primary issue with the word frequency feature vector is that it may suffer from sparsity, which can be resolved using feature selection methods to categorize web pages. Tian et al. [19] proposed a method to categorize web pages into seven categories to resolve this issue. It leveraged information gain to reduce the dimensionality of the term frequency feature vector, which was then used to train an SVM.

To classify web pages of Yahoo sports news, Selamat et al. [20] utilized a principal component analysis (PCA) for a dimensionality reduction in the term frequency-inverse document frequency (TF-IDF) feature vector using a neural network. TF-IDF was a popular method for document classification and natural language processing, but the inclusion of PCA showed improved results.

Researchers have also used an unsupervised learning-based approach to reduce the dimension of feature vectors. Li et al. [21] proposed an approach that utilized the Autoencoder feature reduction method based on unsupervised learning to classify the emotions of social comments using the deep neural network. Wei et al. [22] improved the web page classification accuracy by utilizing the Latent Dirichlet Allocation (LDA) unsupervised learning model for feature reduction and classification using SVM.

In addition to textual information, a web page also contains structural information, i.e., labels and tags. In this situation, TF-IDF considers that the same term as different features appears in separate tags to improve web page classification accuracy. Ozal et al. [6] proposed a web page categorization technique based on a genetic algorithm using tagged terms as features. In this study, each web page was represented as a chromosome with a list of feature weights between 0 and 1. The genetic algorithm-based binary web page classification achieved a better classification accuracy than KNN and the naïve Bayesian classifier.

Lee et al. [7] proposed a binary classification of web pages using tagged terms as features and classified given data into either art or non-art, science or non-science, health or non-health, and computer or non-computer. This article used a meta-heuristic approach called simplified swarm optimization (SSO) to learn the best weight of every feature from the tagged term feature list. The SSO-based web page classifier achieved a better F1-score than a genetic algorithm, the Bayesian classifier, and the KNN classifier.

Vinod et al. [14] proposed an efficient method for web page categorization where feature vectors were based on feature extraction and a weight assignment process to each text term. These feature weights were assigned according to the domain-specific web page keyword list, and this extensive feature set was used by SVM to categorize web pages.

The web page ontology used for web page classification utilizes its semantic features. Recently, some research on web page categorization has focused on the semantic features of the textual content of web pages. Saleh et al. [23] proposed a web page classification technique called the classification of multilayer domain ontology (CMOD) based on a page analysis module, a page importance module, and a page classification module.

Wai et al. [24] developed a web page categorization technique by extracting the semantic features of webpages using ontology. Finally, the classification was performed by the enhanced version of C 4.5 and the Naïve Bays classifier.

Ensemble learning combines multiple machine models, each developed to solve a trivial problem to solve a large problem. Recently, Gupta et al. [25] proposed an ensemble learning-based method to web page classification. It utilized a pre-trained Bidirectional Encoder Representations from Transformers (BERT) model to create feature vectors and deep inception modeling with the residual connection.

These are state-of-the-art techniques in the field of web page classification. In summary, through a the study of existing works, a knowledge gap was identified related to the fact that web page categorization is based on a web page's textual and structural features.

In this paper, the novelty of the research is related to the utilization of the semantic and contextual features of web pages.

3. Proposed Methodology

The web page categorization may categorize web pages into a binary class or a multi-class. In the binary class, each web page $\{w_1, w_2, w_3, \dots, w_n\}$ is categorized into one of two categories $\{c_1, c_2\}$. However, in a multi-class classification, each web page $\{w_1, w_2, w_3, \dots, w_n\}$ is categorized into one of many categories $\{c_1, c_2, c_3, \dots, c_m\}$. The proposed web page categorization model categorizes web pages into different categories.

This section comprises a detailed description of the proposed model's overall methodology and a step-by-step procedure referred to as Algorithm 1.

Algorithm 1 Procedure of proposed work

1. The web pages from the WebKB dataset described in Section 4.2 were gathered
 2. Two samples of the WebKB dataset were taken to conduct experiments with the proposed model shown in Figure 1
 3. The first sample considered four categories, namely "Faculty", "Course", "Student" and "Project". Another sample had only three categories "Faculty", "Course" and, "Student"
 4. Data preprocessing, as explained in Section 3.1
 5. GloVe word embedding was used to make feature vectors for categorization as explained in Section 3.2, and was utilized to produce a semantically related feature vector which passes to the BiLSTM layer
 6. BiLSTM layer extract contextual features as explained in Section 3.3
 7. Contextual features were passed to the softmax classifier to categorize web pages into its category explained as output layer in Section 3.4
 8. Evaluation matrices precision, recall, accuracy, and the F1-score were calculated using the confusion matrix, and the ROC curve of the proposed model was calculated
 9. A comparative performance analysis of the proposed model with existing techniques was completed, as shown in Section 5.2
 10. For the generalization of the proposed model, experiments were performed on more a extensive dataset named as DMOZ in Section 5.3
 11. Evaluation matrices, the precision, recall, and accuracy, and the F1-score were also calculated with the DMOZ dataset using a confusion matrix
-

The proposed model is divided into four modules, as shown in Figure 1, beginning with the input of WebKB data for data preprocessing, followed by word embedding, the stacked BiLSTM module, and the output layer, as discussed in the following section. Section 4.2 provides a detailed description of the arrangements of the WebKB dataset used in this paper.

3.1. Data Preprocessing

The WebKB dataset is a collection of web pages that were used to perform the experiments for the proposed research. The web pages are in the HTML format, so the required data are extracted from the HTML files. First, in data preprocessing, the HTML body, paragraph, title, and tags are scanned to extract only text data. After that, punctuations and digits are removed from the extracted text data which is converted to lowercase. Finally, all the stop words are removed, and stemming is applied on preprocessed text data. For the tokenization, the preprocessed data are now passed to the word embedding phase.

Figure 1. Proposed model for web page categorization.

3.2. Word Embedding

Word embedding is the distributed representation of words which is called word vector representation [26]. The embedding layer extracts the semantic features from inputs of data preprocessing and passes them as input to the further layers of the Neural network learning model as the real-valued vectors. One hot encoding method and distributed word representation are widely used for word vector representations. The high dimensionality of word vector representation is caused by a one-hot encoding technique, which is removed using distributed word representation by mapping each word into a low dimensional dense vector [8]. In addition, distributed word embedding captures semantic information from words. So, words such as faculty and teacher are not considered to be two different features.

GloVe Word Embedding

GloVe word embedding technique [27] is used in the proposed model to extract semantic features from the text on web pages. The GloVe is an abbreviation of global vector, and GloVe embedding is an unsupervised learning algorithm for distributed word representation of the text extracted from the web pages [28]. The GloVe method is easier to train over the data due to its parallel implementation. It captures the semantic relationships of words in the vector space. A global co-occurrence matrix X was created using the words found in the Wikipedia dataset to train the GloVe word embedding model. In this paper, pre-trained word vector data named glove6b.zip with a vector with a size 100 embedding layer were used which were prepared on 400 k word vocabulary.

In the co-occurrence matrix, X : X_{ij} represents the number of context words i which appear with word j . The GloVe model minimizes the following objective function:

$$J = \text{Minimize} \sum_{i=1}^V \sum_{j=1}^V (W_i^t W_j + b_i + b_j - \log x_{ij}) \quad (1)$$

3.3. Stacked Bidirectional LSTM

The standard architecture of an LSTM cell to effectively overcome the vanishing gradient problem and convey useful information within the LSTM network is shown in Figure 2 [29]. Due to the vanishing gradient problem during model training, the traditional Recurrent Neural Network (RNN) may fail to capture long-term dependencies among feature vectors [30]. The LSTM cell consists of three gates, namely, the input gate (i_t), the forget gate (f_t), and the output gate (o_t). These gates control a memory cell activation vector.

Figure 2. LSTM Cell.

- **Forget gate:** This is used to determine the amount of information from a previous state c_{t-1} that should be retained or forgotten based on the present input x_t and hidden state h_{t-1} . The output of the forget gate lies between 0 and 1. The forget gate is expressed by Equation (2).

$$f_t = \text{Sigmoid} (w_{xf}x_t + w_{hf}h_{t-1} + b_f) \quad (2)$$

where b_f , w_{xf} , and w_{hf} are the bias vectors, the weight matrix is between x_t and f_t , and the weight matrix is between h_{t-1} and f_t , respectively.

- **Input gate:** This is used to determine the amount of the network's input x_t that must be retained to the current cell state c_t . The input gate is expressed by Equation (3).

$$i_t = \text{Sigmoid} (w_{xi}x_t + w_{hi}h_{t-1} + b_i) \quad (3)$$

where b_i , w_{xi} , and w_{hi} are bias vectors, the weight matrix is between x_t and i_t , and the weight matrix is between h_{t-1} and i_t , respectively.

- **Output gate:** This is used to determine the amount of information conveyed to the LSTM network from the cell state c_t through the current output value h_t . LSTM gates are fully connected networks whose input is a vector and whose output is a real number. The output gate is expressed by Equation (4).

$$o_t = \text{Sigmoid} (w_{xo}x_t + w_{ho}h_{t-1} + b_o) \quad (4)$$

where b_o , w_{xo} , and w_{ho} are the bias vectors, the weight matrix is between x_t and o_t , and the weight matrix is between h_{t-1} and o_t , respectively.

The final outputs of the LSTM cell are cell output state c_t and layer output h_t , which are expressed in Equations (5) and (6), respectively.

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c}_t \quad (5)$$

$$h_t = o_t \odot \tanh(c_t) \quad (6)$$

The intermediate cell input state is represented by $\tilde{c}_t$ which is expressed by Equation (7):

$$\tilde{c}_t = \tanh(w_{xc}x_t + w_{hc}h_{t-1} + b_c) \quad (7)$$

where b_f , w_{xc} , and w_{hc} are the bias vectors, the weight matrix is between x_t and $\tilde{c}_t$, and the weight matrix is between h_{t-1} and $\tilde{c}_t$, respectively.

In the recently published work by Peng et al. [31], LSTM was applied with fruit fly optimization for NN3 data for a time series analysis. On the other hand, LSTM was also used by Qin et al. [32] in a totally diverse domain to predict the life span of a gear in the mechanical transmission system. In this way, LSTM was identified as a robust and efficient concept useful to solve various research problems used in the proposed architecture.

LSTM cannot utilize the contextual information from the future tokens of web pages, so the bidirectional LSTM (BiLSTM) layer consists of two separate hidden LSTM layers in opposite directions to the same output [33,34]. With this structure, the output layer can utilize the long-term related information of web pages from previous and future text contexts.

A bidirectional LSTM (BiLSTM) comprises of two separate LSTMs that combine information from both directions of a phrase to obtain word annotations [35]. The information acquired as word annotations from the web pages is then integrated. The forward LSTM processes the input left to right and calculates the hidden state $\vec{h}_t$ based on x_t and $\vec{h}_{t-1}$, while the backward LSTM processes the input right to left and calculates the hidden state $\overleftarrow{h}_t$ based on x_t and $\overleftarrow{h}_{t-1}$. The forward and backward parameters in the BiLSTM network are independent of one another, although word embedding is shared. Finally, the hidden state of BiLSTM is defined by Equation (8), which concatenates the vectors of the forward and backward directions at time step t .

$$h_t = \begin{bmatrix} \vec{h}_t & \overleftarrow{h}_t \end{bmatrix} \quad (8)$$

The deep hierarchical model can further improve the performance of the web page categorization [36]. However, the stacking of BiLSTM is more efficient in the shallow learning model, so the proposed model defines a stacked BiLSTM network to utilize the local contextual and latent symmetry complex information of web pages. The output of the lower layer becomes the input of the upper layer in a Stacked BiLSTM network.

The summary in Sections 3.2 and 3.3 of the GloVe word embedding technique shows that it is helpful to identify relevant words from the text in the web pages by computing the probability ratio that those are semantically related. On the other hand, LSTM captures the long-term dependencies between token sequences, which is better for text classification. Due to these factors, we combined the GloVe Model and Stacked BiLSTM in our proposed model.

3.4. Output Layer

Finally, the output layer of the proposed model generates four output values for each category in order to predict the web page category with the highest output value. The SoftMax activation function is utilized in the output layer to ensure that the output value ranges from 0 to 1, which is then used as the predicted probability to predict the web page category. The predicted probability is computed using Equation (9).

$$y' = \arg \max p(c/x) \quad (9)$$

where c = actual class label of a web page and x = web page features.

The Adam optimizer adjusts the learning rate to attain the optimum hyperparameter values and categorical cross-entropy loss used to train the proposed model. The dropout layer is also utilized to avoid overfitting problems with a dropout value = 0.25. The cross-entropy loss increases as the predicated probability diverges from the actual category of web pages. The cross-entropy loss is calculated using Equation (10).

$$L(y, y') = - \sum_i^N y \log(y') \quad (10)$$

where, $L(y', y)$ is the categorical cross-entropy loss for the predicted results, y_i is the actual output for the i th web page (actual category), y'_i is the predicted output for the i th web page, and N is the total number of web pages.

4. Experiment Setup

This section discusses the datasets, performance metrics, and optimal hyperparameter values used during the implementation of the proposed model.

4.1. Dataset Description

The WebKB benchmark dataset [37] is used in the proposed model to categorize web pages. It is comprised of web pages from four distinct institutions: Cornell (867), Texas (827), Washington (1205), and Wisconsin (1263), as well as 4120 miscellaneous web pages from the CSE departments of other universities. The 8282 pages are manually classified into the following categories: student (1641) faculty (1124) staff (137), department (182) course (930), project (504) other (3764). The experiments of the proposed work focused on the following four category web pages: Course, Faculty, Project, and Student. Table 1 shows the number of Train and Test web pages of the WebKB dataset with the four categories considered in the experiments of the present research work.

Table 1. Train and Test web pages of the WebKB dataset.

Class Label	No. of Web Pages	Train Web Pages	Test Web Pages
Course	930	651	279
Faculty	1124	787	337
Project	504	353	151
Student	1641	1148	493

Another dataset available publicly is an extensive repository of the web pages named as the DMOZ dataset [38]. It consists of web pages of different domains, including Arts, Business, Computers, Games, Health, Home, News, Recreation, Reference, Science, Shopping, Society, and Sports used in the experiments of the present article. Table 2 shows the number of Train and Test web pages of the DMOZ dataset.

4.2. Performance Matrices

To measure the effectiveness of the proposed model, the well-known measures of precision, recall, F1-score, accuracy, and the confusion matrix that represent the performance of any classifier were evaluated [39].

Table 3 represents the confusion matrix for the binary classification. True positive (TP) represents the correctly identified web pages, false positive (FP) represents the incorrectly identified web pages, true negative (TN) represents the correctly rejected web pages, and false-negative (FN) represents the incorrectly rejected web pages.

Table 2. Train and Test web pages of DMOZ dataset.

Class Label	No. of Web Pages	Train Web Pages	Test Web Pages
Arts	193,914	135,740	58,174
Business	204,910	143,437	61,473
Computers	93,204	65,243	27,961
Games	36,454	25,518	10,936
Health	50,388	35,271	15,117
Home	22,241	15,569	6672
News	7488	5242	2246
Recreation	83,618	58,532	25,086
Reference	47,428	33,199	14,229
Science	96,766	67,736	29,030
Shopping	78,184	54,729	23,455
Society	195,880	137,116	58,764
Sports	85,376	59,763	25,613

Table 3. Confusion matrix for the binary classification.

	Predicted Positive	Predicted Negative
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

Precision denoted as (P) shows the correctness of the proposed model calculated using Equation (11),

$$\text{Precision (P)} = \frac{TP}{TP + FP} \quad (11)$$

Recall denoted as (R) shows the completeness of the proposed model calculated using the Equation (12),

$$\text{Recall (R)} = \frac{TP}{TP + FN} \quad (12)$$

Accuracy shows the exactness of the proposed model calculated using Equation (13),

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \quad (13)$$

F1-score is the harmonic mean of Precision and Recall calculated using Equation (14)

$$\text{F1 score} = \frac{2PR}{P + R} \quad (14)$$

4.3. Hyperparameter Setting

Many deep learning models provide an explicitly defined parameter to control memory, and the execution cost is called a hyperparameter. A deep learning model's success based on hyperparameter values must be set before applying a learning algorithm. The variable values are different for different tasks. Table 4 shows the proposed model's hyperparameter value.

Table 4. Values of the hyperparameters of the proposed model.

Hyperparameter	Value
Max Length	200
BiLSTM Nodes	64 (First Layer), 32 (Second Layer)
Embedding size	100
Loss	Categorical cross entropy
Optimizer	Adam
Batch size	128

5. Result Analysis

This section describes the performance of the proposed model and the performance comparison of the proposed model with machine learning classifiers, deep learning methods, methods published in recent articles, and the applicability with the DMOZ dataset.

5.1. Performance of the Proposed Model

The confusion matrix represents the TP, FP, TN, and FN values to calculate the evaluation matrices and to draw the Receiver Operating Curve (ROC) of the proposed model. Using these values, the Precision, Recall, and F1-scores were calculated for four categories of the WebKB dataset used in the experiments. Figure 3 illustrates the confusion matrix of the proposed model with four classes in which each column indicates the predicted category of the web page, while each row represents the actual category of the web page [40]. Thus, the sum of data values of cells in each row corresponds to the total number of data instances for that category. The experiments were also conducted by considering only three classes of the WebKB dataset for an apple-to-apple comparison with the existing work. Figure 4 illustrates the confusion matrix of the proposed model with three classes.

Actual	Course	Faculty	Project	Student
	262	9	5	3
	4	279	20	34
	5	36	97	13
Student	8	43	5	437
Predicted				

Figure 3. Confusion matrix of the proposed model with four classes.

Actual	Course	Faculty	Student
	252	19	8
	4	289	44
Student	10	40	443
Predicted			

Figure 4. Confusion matrix of the proposed model with three classes.

To better understand the significance of the values in the confusion matrix, the course category was considered, which produced fewer false-positive and false-negative values. This means that the proposed model correctly categorized 262 web pages in the course category. On the other hand, the project category generated more false-positive and false-negative values because they may have been associated with other categories.

Table 5 shows the evaluation matrices of the proposed model with four classes, which shows the Precision, Recall, and F1-scores for each category. The proposed model achieved an average F1-score of 83.03% and an average accuracy of 85.32%. In addition, Table 6 shows the evaluation matrices of the proposed model with three classes. In this case, the proposed model achieved an F1-score of 88.84% and an accuracy of 88.73%.

Table 5. Evaluation of the proposed model on the WebKB dataset with four classes.

Category	Precision	Recall	F1-Score	Average F1-Score (%)	Average Accuracy (%)
Course	0.9391	0.9391	0.9391	83.03%	85.32%
Student	0.8973	0.8864	0.8918		
Faculty	0.7602	0.8279	0.7926		
Project	0.7638	0.6424	0.6978		

Table 6. Evaluation of the proposed model on the WebKB dataset with three classes.

Category	Precision	Recall	F1-Score	Average F1-Score (%)	Average Accuracy (%)
Course	0.9474	0.9032	0.9248	88.84%	88.73%
Student	0.8949	0.8986	0.8968		
Faculty	0.8305	0.8576	0.8438		

The ROC was plotted between two parameters, i.e., the True Positive Rate (TPR) and the False Positive Rate (FPR). Thus, it shows the performance of the categorization model at all possible thresholds. For example, Figure 5 depicts the course category with the highest Area Under Curve (AUC) value of 0.99. At the same time, the AUC value for the faculty category was 0.95, which was the lowest. Similarly, Figure 6 depicts the ROC for three classes where the course category with the highest Area Under Curve (AUC) value of 0.99. At the same time, the AUC value for the faculty category was 0.94, which was the lowest.

Figure 5. ROC curve of the proposed model with four classes.

Figure 6. ROC curve of the proposed model with three classes.

5.2. Comparative Performance Analysis

This section is further divided into two subsections; the first subsection is dedicated to the experimental analysis which comprises the implementation details of the baseline models. Section 2 is an actual comparative analysis of the proposed model with the state-of-the-art methods from past research.

5.2.1. Experimental Analysis

In this section, to test the effectiveness of the proposed model, it is compared with the deep learning and machine learning models. For this purpose, the following models were implemented.

- **Naïve Bayesian (NB):** The Naïve Bayesian (NB) machine learning model is a simple multi-class classifier which is generally used in text classification [41,42]. It has a specific error rate because it determines the posteriority probability through prior knowledge and data. Nevertheless, NB produces accurate results in a short span of time with minimal training data. In this paper, the feature vectors formed by the TF-IDF approach and Multinomial Naïve Bayes (MNB) were applied with the value of parameter $\alpha = 1$ for web page categorization.
- **SVM:** The Support Vector Machine is a supervised machine learning model based on structural risk minimization and was introduced by Vapnik [11,42]. SVM creates a hyperplane that separates the data into two sets with the maximum margin. In this paper, TF-IDF was used to obtain the frequency of the words for utilizing the features of each web page. Web page categorization was performed using support vector classifier (SVC) with radial basis function (RBF) kernel and the value of parameter $C = 1$ [40].
- **LSTM:** The LSTM model captures the contextual information of the text of the web pages in a forward direction. Moreover, it utilizes the last hidden state to categorize the web pages [18,43]. The word embedding layer resolves the data sparsity problem in the BOW and TF-IDF machine learning approaches. In this paper, a deep learning model was also implemented, using a word embedding layer of 100 dimensions for feature extraction followed by an LSTM layer and finally a softmax layer for web page categorization.
- **BiLSTM:** As LSTM lacks a backward layer, it only gathers contextual information from a web page in one way. The BiLSTM model, on the other hand, collects the web page's contextual information simultaneously from left to right and right to left [6,23]. Another model was implemented in this paper using a word embedding

layer of 100 dimensions for feature extraction, followed by a BiLSTM layer and finally a softmax layer for web page categorization.

Various experiments have been conducted to evaluate the performance of different machine learning and deep learning models. Figure 7 shows the bar chart comparison on the basis of Precision, Recall, F1-scores, and the accuracy of four baseline models. In this bar chart, the machine learning approaches such as NB and SVM achieved F1-scores of 75.34% and 76.50%. Compared to this, the deep learning approaches such as LSTM and BiLSTM are suitable for a sequence processing task, achieving F1-scores of 80.02% and 80.50%, respectively, and showed a better performance. The proposed model outperformed the other models due to the semantic information generated by GloVe passed to stacked BiLSTM, and the stacking of BiLSTM extracted additional contextual features. This arrangement of the proposed model achieved a better performance in terms of precision (84.01%), recall (82.39%), F1-score (83.03%), and accuracy (85.32%).

Figure 7. Experimental performance analysis of the proposed model with four baseline models.

5.2.2. Comparative Performance Analysis with Existing Work

The comparative performance analysis of the proposed model with the state-of-the-art methods in terms of Precision, Recall, F1-score, and accuracy are shown in Tables 7 and 8.

Table 7. Comparative result analysis of the proposed model with existing methods considering four classes.

Author (s)	Method	Number of Classes	Precision	Recall	Accuracy	F1-Score
Gupta et al. [25]	BERT, Convolution Layer, Inception Layer	Course, Project, Student, and Faculty	0.3900	0.2200	0.7900	0.1300
Proposed model	GloVe, Stacked BiLSTM	Course, Project, Student, and Faculty	0.8401	0.8239	0.8532	0.8303

Table 8. Comparative result analysis of the proposed model with existing methods considering three classes.

Author (s)	Method	Number of Classes	Precision	Recall	F1-Score
Bhalla et.al. [14]	SVM	Course, Student, and Faculty	0.9233	0.8233	0.8633
Proposed model	GloVe, Stacked BiLSTM	Course, Student, and Faculty	0.8909	0.8865	0.8884

Gupta et al. [25] considered the same four categories that were implemented for the experiments of the proposed model. They reported an accuracy of 79.00% using the BERT pre-trained model with a deep residue inception layer module. The proposed model achieved an accuracy of 85.32%, which demonstrates a comparatively better performance, as shown in Table 7.

Bhalla et al. [14] proposed an automatic web page classification method in which a domain-specific feature extraction was followed by SVM classifier and achieved an F1-score of 86.33% with three classes of the WebKB dataset. The proposed model was also developed for the three-class categorization of web pages and showed a better performance with an F1-score of 88.84%, as shown in Table 8.

5.3. Applicability of the Proposed Model with the DMOZ Dataset

The proposed method was fine-tuned for the WebKB dataset, and it can also be easily fine-tuned on the other dataset of web pages for the categorization of the web pages. A few experiments were also performed with the DMOZ data set to prove the proposed model's generality and applicability. Figure 8 illustrates the confusion matrix of the proposed model with thirteen classes of the DMOZ dataset.

Actual	Arts	49450	1076	524	523	146	88	136	825	703	639	1022	2700	342
	Business	1026	51756	1553	88	460	132	118	661	377	1408	2549	1094	251
	Computers	608	1897	22282	490	72	69	29	243	283	968	290	659	71
	Games	578	101	362	8885	12	18	3	207	28	112	110	255	265
	Health	149	430	66	20	12225	76	15	161	373	447	103	929	123
	Home	183	284	87	27	171	4683	5	308	62	238	152	435	37
	News	158	137	25	9	5	6	1463	37	65	57	2	238	44
	Recreation	870	737	197	151	223	175	38	19190	245	596	516	1395	753
	Reference	643	536	396	31	338	35	110	309	8870	1182	72	1442	265
	Science	640	1876	807	118	432	169	53	522	1187	21160	170	1825	71
	Shopping	1160	2429	265	127	222	166	7	497	88	194	17693	393	214
	Society	2704	1097	505	209	987	260	201	1253	1412	1506	399	47869	362
	Sports	628	271	55	254	108	12	57	794	395	84	150	490	22315
		Predicted												
		Arts	Business	Computers	Games	Health	Home	News	Recreation	Reference	Science	Shopping	Society	Sports

Figure 8. Confusion Matrix of the proposed model with the DMOZ dataset.

The effectiveness of the proposed model evaluated in terms of Precision, Recall, and the F1-Score for each category of the DMOZ dataset is shown in Table 9. The accuracy of the proposed model with the DMOZ dataset was 80.23%, and the F1-score was 77.49%. These experiments prove that the proposed model performs well if applied with the real-world benchmarked datasets available to conduct future research.

Table 9. Evaluation of the proposed model with the DMOZ dataset.

Category	Precision	Recall	F1-Score	Average F1-Score (%)	Average Accuracy (%)
Arts	0.8410	0.8500	0.8455	77.49%	80.23%
Business	0.8264	0.8419	0.8341		
Computers	0.8215	0.7969	0.8090		
Games	0.8128	0.8125	0.8126		
Health	0.7938	0.8087	0.8012		
Home	0.7952	0.7019	0.7456		
News	0.6546	0.6514	0.6530		
Recreation	0.7674	0.7650	0.7662		
Reference	0.6296	0.6234	0.6265		
Science	0.7401	0.7289	0.7345		
Shopping	0.7617	0.7543	0.7580		
Society	0.8015	0.8146	0.8080		
Sports	0.8886	0.8712	0.8798		

5.4. Discussion

A summary of the experimental results conducted throughout the development of the present article is shown in Table 10. Detailed results are quoted and discussed in previous sections, but a summary is compiled here to demonstrate that the proposed method outperformed various scenarios. With the help of this table, one can easily observe that the proposed model showed a better F1-Score and accuracy than the machine learning and deep learning models. Still, the proposed model performed significantly well with a more extensive dataset such as DMOZ with higher categories.

Table 10. Summary of the experimental results.

Sr. No.	Method and Hyperparameter Settings	Dataset Arrangement	F1-Score (%)	Accuracy (%)
1	Naïve Bayesian	WebKB with Four Classes	75.34%	76.12%
2	SVM with RBF kernel, and C = 1	WebKB with Four Classes	76.50%	78.22%
3		WebKB with Four Classes	80.02%	81.56%
4	BiLSTM	WebKB with Four Classes	80.50%	82.78%
5	Proposed Model	WebKB with Four Classes	83.03%	85.32%
6	Proposed Model	WebKB with Three Classes	88.84%	89.09%
7	Proposed Model	DMOZ with Thirteen Classes	77.49%	80.23%

As far as applicability is concerned, the web page categorization model proposed here can be useful to improve the efficiency of search engines by providing keyword-based categorization of the web pages before mapping them with the user query.

6. Conclusions

This paper proposed and implemented a model for web page categorization that utilized the GloVe and Stacked BiLSTM. Feature extraction and classifier design are crucial processes to achieve this task, and many machine learning models have shown a better performance in this field. However, it is still challenging to understand the semantic features of words in web pages, and the categorization accuracy needs to be improved. In this paper, the GloVe model extracted the semantic features of words in web pages, followed by the Stacked BiLSTM model which extracted the contextual features of the web pages. Finally, the Softmax classifier was applied to the extracted features to categorize the web pages into pre-defined categories. The experimental results demonstrate that the proposed web page categorization model performed better than the machine learning classifiers, deep learning models, and existing methods. Specifically, implementing the proposed model considering three categories of the WebKB dataset gave an accuracy of

89.09%, the same dataset with four categories gave an accuracy of 85.32%, and the DMOZ dataset considering thirteen classes gave an accuracy of 80.23%.

As far as limitations are concerned, the proposed work based on a deep learning model required lots of hyperparameter tuning, and the computation time was also a bit longer. On the other hand, the proposed model gave good accuracy values and F-1 scores as compared to machine learning models such as SVM and NB. The proposed method can also be applicable with publicly available datasets, as DMOZ used in this article. The proposed model resolves the linguistic ambiguity, so it can also be applied to enhance the performance of the search engines. In the future, web pages categorized by utilizing the proposed model may be used further as a web corpus for domain-specific search engines, recommendation systems, and information retrieval to improve the quality of web searches. In addition, other word embedding models such as fast text, BERT, etc., may be utilized to enhance the accuracy of web page categorization.

Author Contributions: Conceptualization, A.K.N.; methodology, A.K.N.; software, A.K.N.; validation, A.K.N. and J.C.; formal analysis, A.K.N.; investigation, A.K.N.; writing—original draft preparation, A.K.N.; writing—review and editing, A.K.N. and J.C.; visualization, A.K.N.; supervision, J.C.; project administration, J.C. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data available in a publicly accessible repository that does not issue DOIs. This data can be found here: <http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/> (accessed on 16 August 2021).

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

References

- Chen, L.-C. A Study of Optimizing Search Engine Results Through User Interaction. *IEEE Access* **2020**, *8*, 79024–79045. [CrossRef]
- Li, C.; Liu, K. Smart Search Engine: A Design and Test of Intelligent Search of News with Classification. 2021. Available online: <http://urn.kb.se/resolve?urn=urn:nbn:se:du-37601> (accessed on 16 August 2021).
- Qi, X.; Davison, B.D. Web page classification. *ACM Comput. Surv.* **2009**, *41*, 1–31. [CrossRef]
- Hashemi, M. Web page classification: A survey of perspectives, gaps, and future directions. *Multimed. Tools Appl.* **2020**, *79*, 11921–11945. [CrossRef]
- Li, H.; Xu, Z.; Li, T.; Sun, G.; Raymond Choo, K.K. An optimized approach for massive web page classification using entity similarity based on semantic network. *Futur. Gener. Comput. Syst.* **2017**, *76*, 510–518. [CrossRef]
- Özel, S.A. A Web page classification system based on a genetic algorithm using tagged-terms as features. *Expert Syst. Appl.* **2011**, *38*, 3407–3415. [CrossRef]
- Lee, J.H.; Yeh, W.C.; Chuang, M.C. Web page classification based on a simplified swarm optimization. *Appl. Math. Comput.* **2015**, *270*, 13–24. [CrossRef]
- Xu, G.; Meng, Y.; Qiu, X.; Yu, Z.; Wu, X. Sentiment analysis of comment texts based on BiLSTM. *IEEE Access* **2019**, *7*, 51522–51532. [CrossRef]
- Wadawadagi, R.; Pagi, V. Sentiment analysis with deep neural networks: Comparative study and performance assessment. *Artif. Intell. Rev.* **2020**, *53*, 6155–6195. [CrossRef]
- Hameed, Z.; Garcia-Zapirain, B. Sentiment Classification Using a Single-Layered BiLSTM Model. *IEEE Access* **2020**, *8*, 73992–74001. [CrossRef]
- Vishwakarma, G.; Thakur, G.S. Hybrid system for MPAA ratings of movie clips using support vector machine. In *Advances in Intelligent Systems and Computing*; Springer: Berlin/Heidelberg, Germany, 2019; Volume 817, pp. 563–575.
- Chen, Q.; Xie, Q.; Yuan, Q.; Huang, H.; Li, Y. Research on a Real-Time Monitoring Method for the Wear State of a Tool Based on a Convolutional Bidirectional LSTM Model. *Symmetry* **2019**, *11*, 1233. [CrossRef]
- Omar, A.; Aldawsari, M. Lexical Ambiguity in Arabic Information Retrieval: The Case of Six Web-Based Search Engines. *Int. J. English Linguist.* **2020**, *10*, 219–228. [CrossRef]
- Bhalla, V.K.; Kumar, N. An efficient scheme for automatic web pages categorization using the support vector machine. *New Rev. Hypermedia Multimed.* **2016**, *22*, 223–242. [CrossRef]

15. Nandanwar, A.K.; Choudhary, J.; Singh, D.P. Web Search Personalization based on the Principle of the Ant Colony. *Procedia Comput. Sci.* **2021**, *189*, 100–107. [\[CrossRef\]](#)
16. Liparas, D.; HaCohen-Kerner, Y.; Moumtzidou, A.; Vrochidis, S.; Kompatsiaris, I. News articles classification using random forests and weighted multimodal features. *Lect. Notes Comput. Sci.* **2014**, *8849*, 63–75. [\[CrossRef\]](#)
17. JingHua, B.; Xiao Xian, Z.; ZhiXin, L.; XiaoPing, L. Mixture Models for Web Page Classification. *Phys. Procedia* **2012**, *25*, 499–505. [\[CrossRef\]](#)
18. Mulahuwaish, A.; Gyorick, K.; Ghafoor, K.Z.; Maghdid, H.S.; Rawat, D.B. Efficient classification model of web news documents using machine learning algorithms for accurate information. *Comput. Secur.* **2020**, *98*, 102006. [\[CrossRef\]](#)
19. Tian, L.; Zheng, D.; Zhu, C. Image classification based on the combination of text features and visual features. *Int. J. Intell. Syst.* **2013**, *28*, 242–256. [\[CrossRef\]](#)
20. Selamat, A. Web page feature selection and classification using neural networks. *Inf. Sci.* **2004**, *158*, 69–88. [\[CrossRef\]](#)
21. Li, X.; Rao, Y.; Xie, H.; Lau, R.Y.K.; Yin, J.; Wang, F.L. Bootstrapping Social Emotion Classification with Semantically Rich Hybrid Neural Networks. *IEEE Trans. Affect. Comput.* **2017**, *8*, 428–442. [\[CrossRef\]](#)
22. Wei, Y.; Wang, W.; Wang, B.; Yang, B.; Liu, Y. A Method for Topic Classification of Web Pages Using LDA-SVM Model. In *Lecture Notes in Electrical Engineering*; Springer: Berlin/Heidelberg, Germany, 2018; Volume 458, pp. 589–596. ISBN 9789811064449.
23. Saleh, A.I.; Al Rahmawy, M.F.; Abulwafa, A.E. A semantic based Web page classification strategy using multi-layered domain ontology. *World Wide Web* **2017**, *20*, 939–993. [\[CrossRef\]](#)
24. Wai, H.P.M.; Tar, P.P.; Thwe, P. Ontology Based Web Page Classification System by Using Enhanced C4.5 and Naïve Bayesian Classifiers. In Proceedings of the 2018 International Conference on Intelligent Informatics and Biomedical Sciences, ICIIBMS, Bangkok, Thailand, 21–24 October 2018; pp. 286–291.
25. Gupta, A.; Bhatia, R. Ensemble approach for web page classification. *Multimed. Tools Appl.* **2021**, 1–12.
26. Liu, G.; Guo, J. Bidirectional LSTM with attention mechanism and convolutional layer for text classification. *Neurocomputing* **2019**, *337*, 325–338. [\[CrossRef\]](#)
27. Pennington, J.; Socher, R.; Manning, C. Glove: Global Vectors for Word Representation. In Proceedings of the Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Stroudsburg, PA, USA, 25–29 October 2014; pp. 1532–1543.
28. Hanson, E.R. Musicassette Interchangeability: The Facts Behind the Facts. *AES J. Audio Eng. Soc.* **1971**, *19*, 417–425.
29. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [\[CrossRef\]](#)
30. Banerjee, I.; Ling, Y.; Chen, M.C.; Hasan, S.A.; Langlotz, C.P.; Moradzadeh, N.; Chapman, B.; Amrhein, T.; Mong, D.; Rubin, D.L.; et al. Comparative effectiveness of convolutional neural network (CNN) and recurrent neural network (RNN) architectures for radiology text report classification. *Artif. Intell. Med.* **2019**, *97*, 79–88. [\[CrossRef\]](#) [\[PubMed\]](#)
31. Peng, L.; Zhu, Q.; Lv, S.-X.; Wang, L. Effective long short-term memory with fruit fly optimization algorithm for time series forecasting. *Soft Comput.* **2020**, *24*, 15059–15079. [\[CrossRef\]](#)
32. Qin, Y.; Xiang, S.; Chai, Y.; Chen, H. Macroscopic–Microscopic Attention in LSTM Networks Based on Fusion Features for Gear Remaining Life Prediction. *IEEE Trans. Ind. Electron.* **2020**, *67*, 10865–10875. [\[CrossRef\]](#)
33. Graves, A.; Schmidhuber, J. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. In Proceedings of the Neural Networks, Pergamon, Turkey, 31 July–4 August 2005; Volume 18, pp. 602–610.
34. Graves, A.; Jaitly, N.; Mohamed, A. Hybrid speech recognition with Deep Bidirectional LSTM. In Proceedings of the 2013 IEEE Workshop on Automatic Speech Recognition and Understanding, Olomouc, Czech Republic, 8–12 December 2013; pp. 273–278.
35. Brahma, B.; Wadhvani, R. Solar irradiance forecasting based on deep learning methodologies and multi-site data. *Symmetry* **2020**, *12*, 1830. [\[CrossRef\]](#)
36. Wang, C.; Yang, H.; Meinel, C. Image Captioning with Deep Bidirectional LSTMs and Multi-Task Learning. *ACM Trans. Multimed. Comput. Commun. Appl.* **2018**, *14*, 1–20. [\[CrossRef\]](#)
37. McCallum the 4 Universities Data Set. Available online: <http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-20/www/data/> (accessed on 12 July 2021).
38. DMOZ—The Directory of the Web. Available online: <https://www.dmoz-odp.org/> (accessed on 16 August 2021).
39. Kaliyar, R.K.; Goswami, A.; Narang, P.; Sinha, S. FNDNet—A deep convolutional neural network for fake news detection. *Cogn. Syst. Res.* **2020**, *61*, 32–44. [\[CrossRef\]](#)
40. Vishwakarma, G.; Thakur, G.S. Comparative performance analysis of combined svm-pca for content-based video classification by utilizing inception V3. *Int. J. Emerg. Technol.* **2019**, *10*, 397–403.
41. El Hindi, K.M.; AlJulaidan, R.R.; AlSalman, H. Lazy fine-tuning algorithms for naïve Bayesian text classification. *Appl. Soft Comput. J.* **2020**, *96*, 106652. [\[CrossRef\]](#)
42. Khurana, A.; Verma, O.P. Novel approach with nature-inspired and ensemble techniques for optimal text classification. *Multimed. Tools Appl.* **2020**, *79*, 23821–23848. [\[CrossRef\]](#)
43. Du, J.; Vong, C.M.; Philip Chen, C.L. Novel Efficient RNN and LSTM-Like Architectures: Recurrent and Gated Broad Learning Systems and Their Applications for Text Classification. *IEEE Trans. Cybern.* **2021**, *51*, 1586–1597. [\[CrossRef\]](#)