Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 31 No. 5s (2024)

An Improved Information Retrieval System using Hybrid RNN LSTM for Multiple Search Engines

B. Sangamithra[1*], Dr. Asha K.H[2], Dr. M. Sunil Kumar[3]

[1]Research Scholar, Dept of Computer Science and Engineering, Don Bosco Institutue of Technology, Mysuru-Rode, Bangaluru-74

mithra197@gmail.com

[2]Associate Professor, Dept of Computer Science and Engineering, Don Bosco Institute of Technology, Mysuru-Rode, Bangaluru-74

asha.kh06@gmail.com

[3]Professor & Programme Head, Department of computer science and engineering, School of Computing, Mohan Babu University

(erstwhile Sree Vidyanikethan Engineering College), Tirupathi, AP, India

sunilmalchi1@gmail.com

Article History:

Received: *14-05-2024*

Revised: *22-06-2024*

Accepted: *04-07-2024*

Abstract

When searching for data on the internet, every user has their own personal context for doing so. The job of a search engine is to find the most relevant content from all the blogs on the internet based on the user's query. Information retrieval systems, both locally and globally, have been profoundly affected by the advent of the internet, and this includes the value of information left as comments on a page of SNS (Social Network Services). The concept of emotional and social connections is translated into a logical framework using the terms "node" and "link" to describe the structure of a social network. Efficient semantic models are more extensive training and evaluation materials, are needed to improve social network search capabilities. When compared with machine learning algorithms, the efficacy of traditional keyword-based search engines at understanding users' intentions is low. Recently, neural networks have gained popularity in the field of information retrieval because of their impressive vector representation learning capabilities. The usage of deep learning techniques for this purpose has recently been seen, and they have proven to be more effective than traditional machine learning techniques such artificial neural networks (ANNs). Improved results have been seen specifically using deep-learning techniques like long short-term memory (LSTM) & Recurrent Neural Network (RNN). In order to create a more personalized Information Retrieval(IR) system, this research suggests using a deep learning Hybrid RNN - LSTM model. Finally, the suggested method takes user comments into account and uses a hybrid RNN - LSTM to re-rank the data so that everyone is happy. Web search contest dataset is used for the implementation. Statistics like accuracy, precision, & recall are used to evaluate the data set on Bing and Duckduck go, two of the most prominent search engines. According to the findings, the proposed Hybrid method outperformed more traditional methods.

Keywords: Web personalization, RNN, LSTM, Information Retrieval, hybrid model.

1. Introduction:

The process of retrieving useful information from databases of records. Traditional information-retrieval approaches are unable to keep up with the growing volume of data and the desire for more accurate search results. Knowledge management's emphasis on information retrieval highlights the importance of ontology, a relatively new system for organizing knowledge. Rules for modeling text in recognition of patterns as well as additional domains constitute the basis of the current information retrieval models like the model of vector spaces (VSM) [1]. This abstract-looking text is broken down, filtered, and categorized by the VSM, and the statistics are carried on to the word frequency data. The computer processes the text in accordance with predetermined rules and generates statistics based on the text's word frequency data.

The probabilistic model [2] uses largely probabilistic operations so Baye’s rule to match details from the data, and all of the weight values for feature phrases are multi valued. To capture the user's interest, or their unique inquiry, the probabilistic model makes use of the index word. Unfortunately, there is not yet a standardized vocabulary set that includes both a semantic characteristic and a document label. The meaning of the document is missing from the traditional weighted method, making it unreliable for describing the material. Document meaning is expressed via domain ontology of the semantics relation [4] and weighted item frequency [3] based on the outcomes of semantic annotation.

Converting text into a vector space or probabilistic set is made easier by VSM and probability models. The number of times that a query word appears in the text of the article is described using the frequency property. The study simply calculates that the occurrence of a term is not sufficient for a summary, and this takes into account the segmentation details of the document. Meanwhile, here is no agreed-upon language for labeling documents or providing a collection of standardized semantic features. By incorporating ontology into the retrieval process, users' semantic information can be queried according to their specific retrieval requirements [5]. User information demand logic perspective is disjointed and wrong without the vocabulary set containing semantics description, which is necessary to convey the semantic of the user's requirement. The logical view doesn't show what the written content and the reader need, and the results of the search don't convince the user in this type of information retrieval model, even if we employ the right sort function R (R is the opposite of the split between points).

We construct a model using IR and a domain ontology repository to enhance the precision and speed of user retrieval. Combining an ontology-based retrieval system with. Improved recall and accuracy in document processing can be achieved through the use of a keyword-based IR system. Figure 1 shows an example of an effective, index-based, personalized information retrieval system.

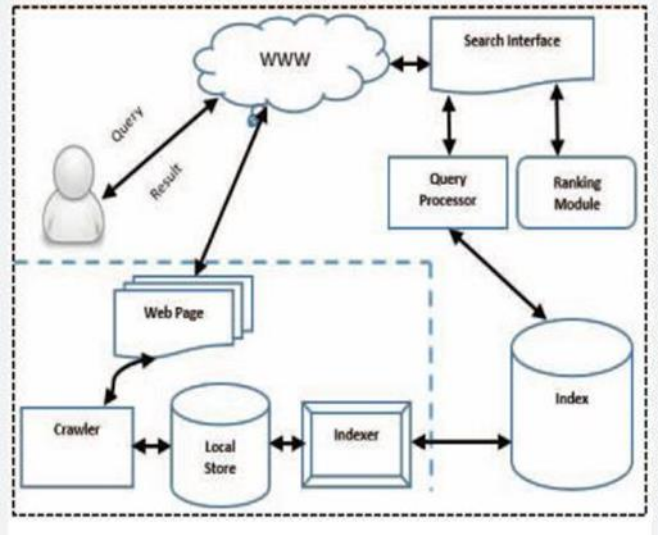

Figure 1.A Personalized Information Retrieval System [6]

Here is how the rest of the paper is laid out. In the second part of this paper, we review the recent research on Personalized Information Retrieval System systems and the algorithms which drive them. In Section 3, we covered the topics of RNN, LSTM, and the proposed hybrid model. The other search engines and their interpretation of the dataset are discussed in Section 4. Results and evaluation are covered in Section 5, and the process concludes up in Section 6.

2. Literature Survey:

Improvements in IR, or information retrieval, are numerous. IR has been used in commercial & intelligence contexts since at least 1960, long before the advent of the internet moved its focus to web searches and similar. The growth in computing power and storage space has led to a corresponding rise in the speed and precision of the same. As the area of IR has grown and evolved, so too have querying methods, which have quickly evolved from more laborious, library-based procedures to fully digital ones.

Brilliantly extending early IR technique[7], the references based search expansion was developed by Brshaw, Suheink, & Hammond at Northwestern University's Smart Information lab. Despite the fact that this approach can only be used with texts that are conceptually similar to one another, archives allow articles to be indexed based on how they are mentioned in other articles. To this day, Bradshaw's renowned and invaluable remark that "people rarely formulate queries of more than three words" stays true. They design the indexing system on the premise that users will only submit clear queries. Indexing documents was the primary method used to get relevant search results. Since the reference pair exactly matches the document that includes the necessary information, using references is a powerful method of indexing and ranking the documents.

Chau and Yeh [8] suggested a method that works with the heterogeneity document to address the limitations of reference-based query expansion by Bradshaw. The aim behind this was multilingual query extension, which would allow native Asians to perform the same search in their native language. To get around this, Chau and his subordinates provided a directory or structure of concepts that are translated into the target language, allowing the user to easily find the information

they need. After the user submits a query by selecting the concept of interest, the system displays the document that best fits the concept category.

Eze et al. [9] advocated for LIS students at the College of Nigeria, Nsukka to become familiar with and make use of Web 2.0 applications. The most popular Web 2.0 applications include wikis, Facebook, and YouTube. These developments are developed so that individuals can get to know one another, share their thoughts on life in general, and keep in touch with friends and relatives. The best ways to learn more about Web 2.0 are through personal experience and the guidance of trusted friends. UNN's LIS majors may be unfamiliar with RSS feeds, podcasts as well as and social bookmarking tools, but they're experts at using social networking sites, wikis, messaging apps, and blogs.

Challenges and Opportunities for a System for Collecting Blog Data is a novel method proposed by Hussain et al. This paper provides an innovative approach to collecting data from blogs. In this work, we shed light on the challenges involved in collecting data from blogs and demonstrate the importance of blog tracing by applying it to a real-world example. Experts and scholars can easily access the blog datasets, as the blog tracker tool indicates. This helped readers in their analysis of the structure and dynamics of the blog site, as well as its many noises and experiments. We implemented an automated crawling system to overcome the challenges exhibit by blog information and considerably boost the efficiency of the entire procedure based on data judgments.

Free-form web curation policies were portrayed by Kerne et al. [11] as creative methods of interacting with existing work. The students' commitment to the task was demonstrated by the addition of experiential, visual, and quantitative data on top of the unstructured web curation strategy, training, & structure. In this research, we formalize a framework for context-aware, hands-off online curation that takes into account both space and emphasis schemes. Customers can signify associations, determine while interpret by assembling self-made and prepared content components, and we built a freestyle online curation to facilitate this process. You can use these tactics: write, construct, gather, exhibit, sketch, and change viewpoint.

Librarians at East Azerbaijani Iranian university libraries were instructed by Pirshahid et al. [12] to determine how often Web 2.0 tools were used using an all-encompassing search. Librarians have extensive experience using Web 2.0 applications like weblogs & wikis. It's safe to assume that most people use Web 2.0 technologies for fun, curiosity, networking, teamwork, family and friend interaction, and trend monitoring. The librarians have been thinking about using an efficient Web 2.0 application to disseminate data about the library's holdings. Threatening the adoption of Web 2.0 by libraries include important incursions like a lack of training, a lack of access to fast web, and web separation.

Previous published findings for the combination NN-HMM model were surpassed by the CNN work of Abdel-Hamid et al. [13]. In comparison to continuous NNs with the same variety of hidden layers and weights, relative errors on the core TIMIT set of tests were reduced by more than 10% with the help of local filtering & max-pooling. Abdel-Hamid et al.'s study [14] also investigated time-frequency convolution simultaneously.

Researchers Sainath et al. [15] looked at how best to train CNNs to outperform DNNs in large-vocabulary continuous voice recognition. They tested the performance of DNNs and GMMs with CNNs' derived NN features on several LVCSR tasks. The experiment showed a 13-30% relative increase over GMMs and a 4-12% relative increase over DNNs on the 400-hour broadcast news task and the 300-hour switchboard task, respectively. An experimental study of CNN-based models based on sounds for low-resource languages shows that CNNs are superior to DBNs in terms of durability and enhanced generalization [16].

It was discovered that the work of Sak et al. [17] was the first to apply LSTM networks to a Google voice search with a wide vocabulary. They introduced a model architecture for LSTM RNN models that allows more efficient use of the model characteristics when building acoustic models for applications with an extensive vocabulary. They used various parameter and configuration numbers to train LSTM, RNN, & DNN models and then compared their results. Their investigation demonstrates that LSTM models converge rapidly and excel in moderately small-sized frameworks.

For robust voice detection in a noisy and reverberant settingIn a hybrid dynamic modeling framework, an LSTM RNN was proposed by Gulbar et al. [18]. The experiment employed data from the second CHiME independent speech and recognition challenge, namely the medium-vocabulary recognition tracks. The authors compared LSTM network-based phoneme prediction networks with LSTM network-based state prediction networks. The outcome demonstrated that LSTMs excel at state prediction as opposed to phoneme prediction via networks. To undertake large-scale modeling, recent advances in RNN-HMM hybrid systems have been made possible by the incorporation of deep unidirectional (DB) Distributed training methods, states in the LSTM input space, and LSTM-based models of pitch for CD phonetic elements [19].

LSTM RNN has been the subject of numerous research efforts. Almost all thrilling outcomes generated by RNNs has accomplished using LSTM. As a result, it is now the epicentre of IR-related deep learning [20]. To our knowledge, this is the very first occasion that an LSTM RNN is being used across several Search engines, but LSTMs have been widely used in IR jobs due to their strong learning ability [17], [22].

3. Methodology:

The RNN shall be presented first, followed by a discussion of the LSTM, in this section. Finally, we'll look into the Hybrid LSTM-RNN.

3.1 Recurrent Neural Network(RNN)

RNNs are an artificial neural network subset. The same three layers—input, hidden, and output—exists. Figure 2 depicts the general layout of an RNN model. Nodes in a conventional feed forward NN are not connected to one another if they are located on the same secret layer. The RNN, on the other hand, has connections between each node in a single hidden layer. RNNs are unique in that they can efficiently learn time series data by incorporating previously acquired information into the current covert layer's learning process. One node gt's mapping will be written as:

$$g_t = f\,(\mathbf{U}x_t + \mathbf{W}g_{t-1}) \quad (1)$$

where xt is the current input, gt is the current hidden state, and r is the network's memory unit; The nonlinear function is denoted by f (•), and the shared parameters W & U were used in each layer.

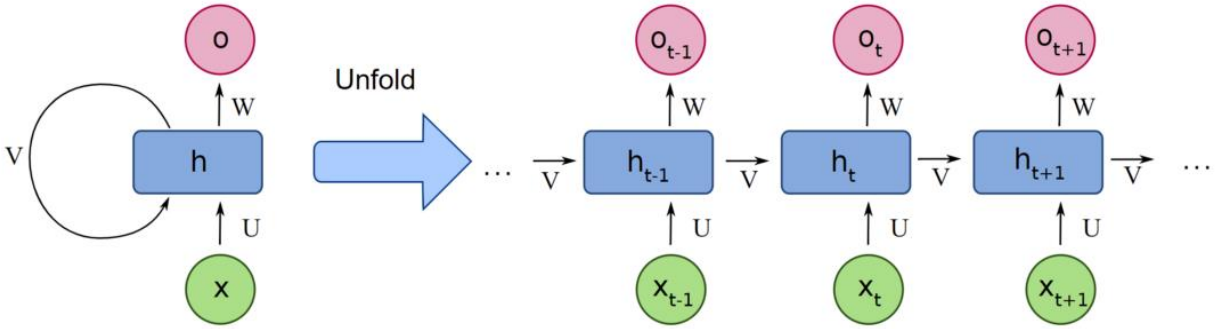

Figure2 .The structure of the typical RNN model

The RNN can be thought of as an ordered graph along a sequence, formed by the links between its nodes. Because of this, it is able to display dynamic temporal behaviour for a given time sequence. RNNs have the ability to remember previous inputs, which feed forward neural networks lack. In theory, an RNN can be used to make accurate predictions by analysing historical data. Unfortunately, the RNN's predicted results are not always satisfactory in real-world contexts due to its inability to effectively memorise the preceding information when the time gap between it and the present forecast position is large and the issue of vanishing gradients continues. Recently, a network called LSTM was proposed to improve the RNN's performance by addressing this drawbacks.

3.2 Long-Short-Term Memory Network

A long short-term memory network (LSTM) is a type of RNN that uses LSTM units [22, 23]. In Figure 3 we see the basic structure of the LSTM unit. This diagram depicts the standard components of an LSTM unit, which include a cell, input gate, output gate, and forget gate. The LSTM's "memory," or "cell," is what keeps track of data over long periods of time. For LSTM, the input is a vector and the output can be either 0 or 1. This is the "gate" of the network. No data is transmitted when the resultant value is 0. When the resultant value is 1, no data is blocked from being transmitted.

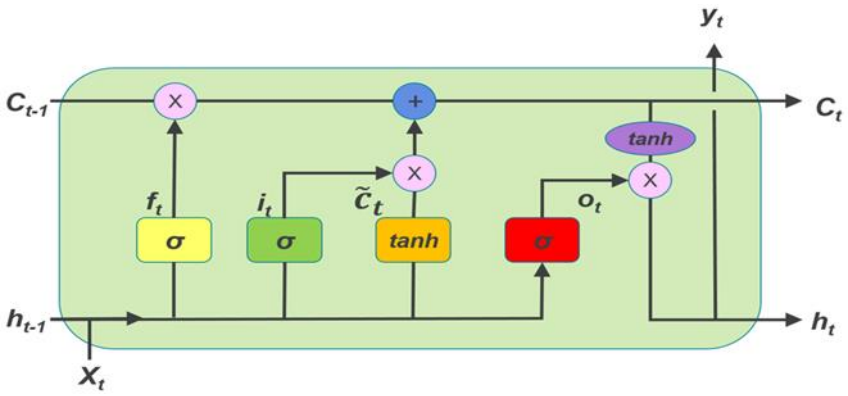

Figure 3. The structure of the LSTM unit

In order to overcome the gradient problem prevalent in very large networks, LSTM RNNs were developed. In addition, LSTM can gracefully deal with long-term distance dependencies inside a specified sequence. These features of LSTM justify its deployment in IR applications involving the sequential representation of sentences and textual materials. With LSTM, you can remenber only the steps that are actually useful. Various types of gates in the network make the call regarding what information (the current input and the concealed state) should be stored. A model of an LSTM network is constructed using Equation (6):

$$fg_t = (w_f * x_t + s_t * hd_{t-1} + b_f) \quad (2)$$

$$ip_t = (w_{ip} * x_t + s_{ip} * hd_{t-1} + b_{ip}) \quad (3)$$

$$M = \tanh(wm * xt + sm * hdt-1 + bm) \quad (4)$$

$$= ipt * \tilde{M}t + fgt * Mt-1 \quad (5)$$

$$op_t = (w_{op} * x_t + s_{op} * hd_{t-1} + b_{op}) \quad (6)$$

$$hd_t = opt * \tanh(Mt) \quad (7)$$

Where σ is the stimulation function, tanh is the elliptical tangent activation function, xt is the input data at the time instance, Mt1 is the previous memory, Hdt1 is the previous output, Mt is the current memory, Hdt is the current output, WF, WIp, WM, Wop SIp, SM, SOp, ST, are now weights, and BF, BIp, BM, BOp are bias. The current inputs and the prior cell output coordinate the various gate values. In order to choose what data to retain and what to discard, LSTM processes both historical and current information. By forgetting a subset of the current new input xt, the resulting memory cell is shown in Equation (2). Equation (3) shows that each LSTM cell produces some new memory. Memory & final storage of the gate can be calculated using Equations (4) and (5). The aggregated cell's result is a prediction for the entire sequence as a whole. Equation (7) uses the sigmoid activation function, which is the function of the final storage gate output to calculate the value of the hidden state.

3.3 Hybrid LSTM-RNN Architecture:

Previous research has shown that merging different neural network topologies can help remedy the drawbacks of each individual model. As a result, this paper investigates the effect of adding layers from different architectures to the preexisting neural network model. When compared with the recurrent neural networks (RNN) [25,26] (Fig. 3), the LSTM (Long Short Term Memory) layer[24] (Fig. 2) is better able to remember long-distance relations. Below is a diagram of our proposed LSTM-RNN.

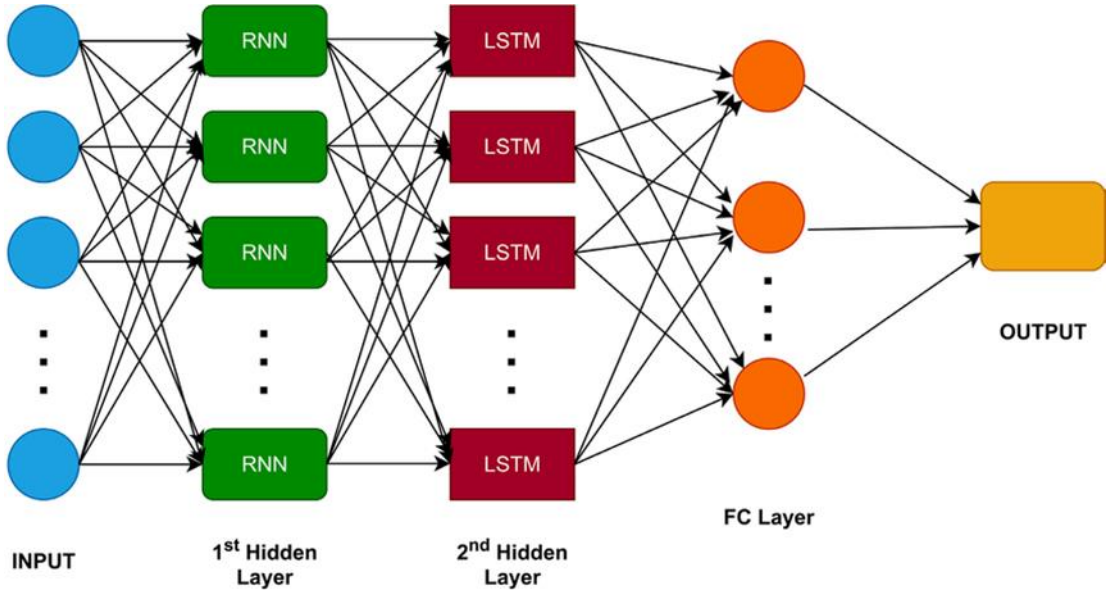

Figure 4.Proposed RNN-LSTM Architecture for IR System

Figure 4 depicts the general layout of the suggested hybrid model. In this diagram, we can see that once the data is entered, the LSTM model & the RNN model will both calculate their own outputs before being ensemble using the linear regression approach to get the final result of the hybrid model. The learning algorithms discussed above can be used to build both the LSTM & RNN models used in this hybrid model. Now all that's left to do in order to create this hybrid model is to settle on values for each parameter of the linear Regression, as shown in equations (1) through (7).

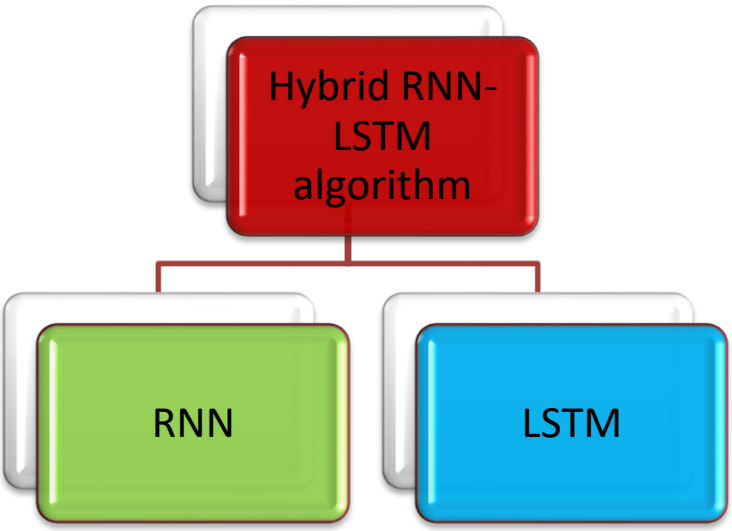

Figure 5.Hybrid RNN-LSTM Algorithm

3.4 Working of RNN-LSTM :

In IRS's Figure 4, a single word or phrase serves as the fundamental building block of your work. In addition, numerous algorithms use vector representations of text. Word representation is a popular application of our Hybrid method. Word similarities cannot be displayed using this method. However, due to the large dimensionality of the input documents, employing this approach in deep

learning results in issues such as dimensional disaster. It also misrepresents the depth of the links between the terms. Word embedding algorithms have superseded this problematic representation method. Word2vec, Glove, and Embedding layer are a few of the available word embedding algorithms. The word2vec embedding approach, a three-layer neural network structure, was utilised for representing the words in a decentralised fashion in this work. Word2vec primarily functions to transform text into vectors. Every phrase in the text is converted into a vector representation for processing. There are several different words that make up the vector representation. The word in the text is now represented by the notation Rd, where d is the dimension of the word embedding vector. Input text is symbolised by T, therefore the length of the text is used in Equation 8.

$$A = [x1, x2, \ldots xT] \in R^{T*d} \text{ ----------(8)}$$

This mixed-model neural network employs both RNN and LSTM. RNN refers to the network of neurons that operate together and communicate with one another. Here, the weights established connections between the neurons. This network architecture works well with time series problems and inputs of variable sizes. RNN's internal memory allows for two-way communication, which helps keep track of lengthy data sequences. In order to process sequences of varying lengths, RNN is frequently employed. A large amount of historical data obtained from lengthy text sequences is processed by RNN, as it is comparable to feed-forward NN. In addition to RNN, LSTM networks can also do Feature Extraction. Long text sequences also undergo semantic analysis, which fixes the vanishing gradient issue. With its three control gates plus memory cell, LSTM is able to efficiently store historical data in extended sequences. The previous information loss that occurred during training is avoided by using this structure. This concept of a network uses just three gates—the Input gate, the Forget gate, and the Output gate—to execute operations like reading, writing, and erasing data. The Input gate prevents any changes from taking effect. With the learnt weight in mind, the neuron's Forget gate inactivates a less-important neuron, and the resultant gate produces the output.

The recommended RNN-LSTM model uses a multi-layer LSTM to generate a vector as output, which is then fed into the RNN model. To further retrieve the attribute from the text input, the RNN model is constructed on top of the LSTM model. $S = [s1, s2,.. st]^T$ is the output form that results from the feature extraction performed by the multi-layer LSTM on the input sequence. Where T is the t-th word in the text sequence and st is an m-dimensional feature vector representing that word. In addition, the number of hidden layers in an LSTM model corresponds to the maximum vector distance. The length of a text sequence is equivalent to the number of steps "T" in an LSTM's growth process.

The RNN takes in data from a matrix in the form of $S \in Rm *T$. RNNs employ a convolution filter written as $F \in Ri*j$. Assume that "i" represents the number of words in the window and "j" represents the dimension of the word vector. Maximum features are retrieved using a number of filters, each with its own set of parameters. Now that the features have been retrieved, they are sent to a flatten layer, where max pooling plus activation functions are used to generate the highest possible probability.

The characteristics in this Deep Hybrid framework were extracted using a combination of a Recurrent Neural Network and a Long Short-Term Memory Network. Feature vectors were

generated using Feature Extraction. The feature extraction process does not require any human intervention. The Deep Neural Network extracts features effectively by using 2D convolution & max Pooling. The number of DNN layers utilized is a function of the complexity of the input. The filter price, kernel size, and activation function all play a role in the feature extraction process. The flattened layer is then used in the flattening process, and vectors of features are generated. The final classification will be carried out in the subsequent module using the standard machine learning classifier. This can therefore be implemented on major search engines such as Yahoo, Google, Bing, etc.

4. Experimental Analysis:

The kaggle standard dataset is used for model evaluation. This research makes use of a dataset (websearchchallenge.csv) collected from the website kaggle. The Kaggle page for the Yandex Personalised Web Search Challenge is https://www.kaggle.com/competitions/. Consolidating and evaluating the work done in commercial labs to personalise web search employing user-logged search behaviour context is made possible by the Personalised Web Search Challenge. Anonymized user identifiers, searches, query phrases, urls, url domains, and clicks are all included in the shared dataset provided by the Yandex search engine. There are a total of 21,073,569 distinct Queries in the dataset. Stop words, punctuation, and the like are cleaned out of the data before it's used. After the initial processing, the word encoding layer separates the data into test and train sets. The RNN-LSTM model is used to analyse train data. Only the machine learning test data should be used for actual testing. The document can be efficiently pre-processed to get features for feature extraction. In our suggested hybrid model, feature extraction is handled by a couple RNN-LSTM layers. Word2vec embedding technique is used to segment words from a text document. Each word in the text has been transformed into its own unique index. There's a cap of 100 characters, and a floor of 8. The model's input length has been set to 85 to improve model performance. Sequences shorter than 35 had leading zeros inserted, and those longer than 35 have been cut off. RNNs use feedback loops for serial processing and memory preservation; the parameters for the convolutional layer were 48 filters, a kernel size of 16, and the ReLu Activation function. The next layer, an LSTM with 256 neurons, receives the layer's output. The best features were picked out using the LSTM layer's probability value and loss function, then fed into the Machine Learning algorithm. AdaBoost & XGBoost algorithms are applied on bing and DuckDuckGo results to determine the efficacy of the suggested hybrid model.

5. Results and Evaluation:

Python simulations of our proposed RNN-LSTM algorithm have been run with various settings representative of industry practise and tailored to the datasets accessible via the two most popular search engines. The web searches challenge dataset was used to train our proposed RNN-LSTM algorithm, and its outcomes were analysed. We have used the web search challenge dataset to evaluate our suggested approach in Bing & Duck duck go search engines, along with other similar algorithms like LSTM, AdaBoost, and XGBoost.

The effectiveness of these Deep Learning algorithms is measured using a confusion matrix. Accuracy, precision, recall, and F1-score are just few of the statistical criteria used in evaluating

different Deep Learning Techniques, and these matrices of confusion have been put to good use in this process. It's laid out as a table with the actual numbers along the rows and the forecasts along the columns. Figure 6 depicts a simplified version of the confusion matrix. The outcomes of the Four techniques are provided in Table1 after thorough review of proposed and existing systems.

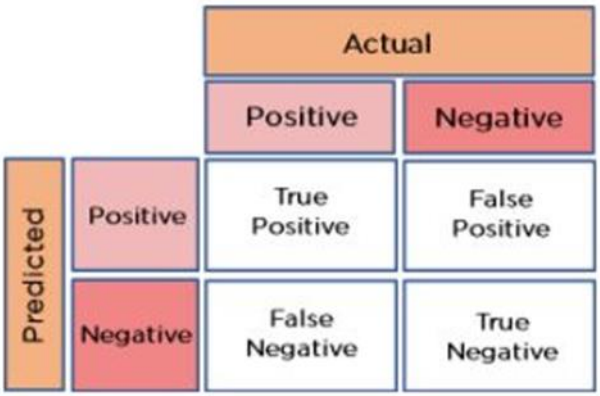

Figure 6. Confusion Matrix

The prediction accuracy is utilised as a metric to assess the efficiency of the proposed RNN-LSTM based hybrid model towards text classification. In text categorization, accuracy is an often measured parameter. The success rate of a text classifier is evaluated by the proportion of correctly classified texts. The proposed model is implemented in two different search engines and then tested for accuracy, precision, recall, and the F1-score. You can see the contrast in Table 1.

Table1.Experimental Results of Various models on Bing Search Engine

Name of the Model	Bing Search Engine			
	Accuracy	Precision	Recall	F1-Score
XG Boost	83.12	84.57	85.23	87.15
Ada Boost	86.18	86.98	87.48	88.54
LSTM	89.24	88.75	89.57	90.24
RNN-LSTM	94.58	93.98	95.48	96.15

Tables 1 and 2 show that when the suggested hybrid approach (RNN-LSTM) was pitted against the other DL methods, it fared better. Metrics including Accuracy, Precision, Recall, and F1 Score are used to assess the proposed hybrid model's performance alongside Ada Boost, LSTM, and XG Boost. Tables 1 and 2 and Figures 7 and 8 demonstrate the superior performance of our proposed method compared to the rest of the algorithms used by Bing Search Engine.

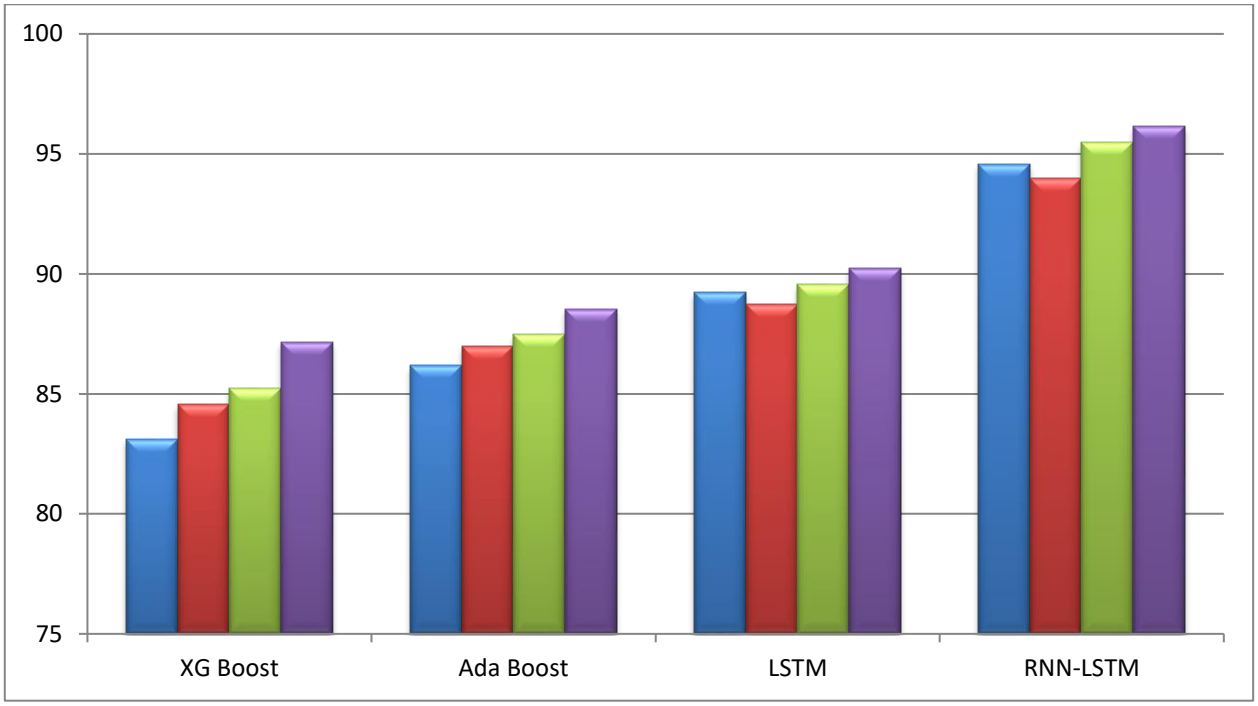

Figure 7.Reuslts of Various Algorithms in Bing Search Engine

Table2.Experimental Results of Various models on Duck Duck Go Search Engine

Name of the Model	Duck Duck Go Search Engine			
	Accuracy	Precision	Recall	F1-Score
XG Boost	85.26	86.89	87.28	88.32
Ada Boost	87.32	89.93	88.48	89.99
LSTM	90.58	91.89	90.57	91.54
RNN-LSTM	95.62	96.78	97.58	97.89

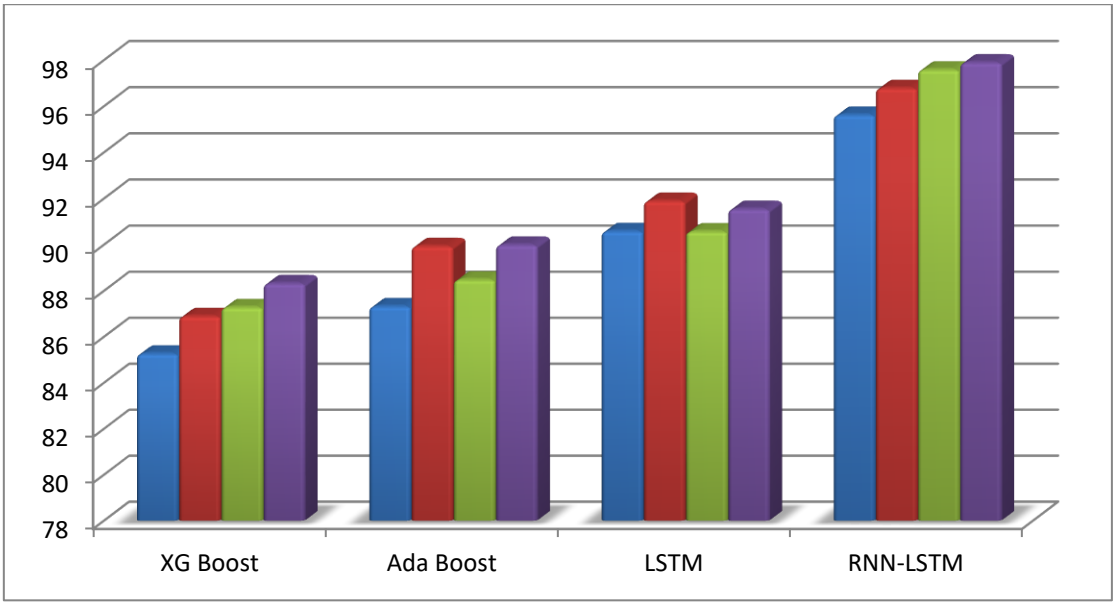

Figure 8.Reuslts of Various Algorithms in Duck Duck Go Search Engine

6. Conclusion and Future work

Right now, things are moving towards more personalised services. The old general retrieval method isn't able to meet the needs of retrieval in different places, for different reasons, and at different times. This paper suggested a mixed model based on RNN and LSTM with a Deep learning version for classifying text based on the questions given. The model takes advantage of LSTM and keeps past data in long text, which solves the Vector Representation problem. RNN, on the contrary hand, is used to pull out features from text. The model uses a type of Deep Learning called RNN to analyse the features that were extracted and make the end prediction. The model suggested in this work doesn't have the fully linked classification layers that are found in most Deep Learning models because it uses a different type of Deep Learning. Because of this, the suggested model works for less money and makes text classification more accurate. The results of the experiments show that the model is correct. The difficulty of text classification and extraction will rise as the length as well as complexity of questions change over time. An advanced mixed model can be made by combining the next level of models, like the Attention-based model, with a Deep Learning model. This makes it possible to accurately classify long text sequences. This work can also be expanded to include the other search engines as well.

References:

[1] M. Tang, Y. Bian, F. Tao, The research of document retrieval system based on the semantic vector space model. J Intelligence. 5(29), 167–177 (2020)

[2] C. Ma, W. Liang, M. Zheng, H. Sharif, A connectivity-aware approximation algorithm for relay node placement in wireless sensor networks[J]. IEEE Sensors J. 16(2), 515–528 (2022)

[3] Yang, X., Yang, D., Yuan, M. Scientific literature retrieval model based on weighted term frequency. Intelligent information hiding and multimedia signal processing (IIH-MSP), 2022 tenth international conference on. IEEE, 2014: 427–430

[4] M. Xu, Q. Yang, K.S. Kwak, Distributed topology control with lifetime extension based on non-cooperative game for wireless sensor networks[J]. IEEE Sensors J. 16(9), 3332–3342 (2021)

[5] Y. Yan, J. Du, P. Yuan, Ontology-based intelligent information retrieval system. J, Software 26(7), 1675–1687 (2022).

[6] Taksande, S.A., & Deoranka, A.V. (2021). Analytical Study on Personalized Information Retrieval System. Imperial journal of interdisciplinary research, 3.

[7] Bradshaw, Shannon, and Kristian Hammond. "Automatically indexing documents: content vs. reference." Proceedings of the 7th international conference on Intelligent user interfaces. 2022.

[8] Heenan, Charles H. "A Review of Academic Research on Information Retrieval." Unpublished manuscript, Engineering Informatics Group, Department of Civil and Environmental Engineering, Stanford University (2019).

[9] Eze EM (2020) Awareness and use of Web 2.0 tools by LIS Students at University of Nigeria, Nsukka, Enugu State, Nigeria, Library Philosophy and Practice

[10] Hussain MN, Obadimu A, Bandeli KK, Nooman M, Al-khateeb S, Agarwal N (2021) A framework for blog data collection: challenges and opportunities. In: IARIA

[11] Kerne A, NicLupfer, Linder R, Qu Y, Valdez A, Jain A, Keith K, Carrasco M, Vanegas J, Billingsley A (2017) Strategies of free- form web curation: processes of creative engagement with prior work. In: Proceedings of the 2022 ACM SIGCHI conference on creativity and cognition, ACM, pp 380–392

[12] Pirshahid SE, Naghshineh N, Fahimnia F (2022) Knowledge and use of web 2.0 by librarians in university libraries of East Azerbaijan, Iran. Electron Libr 34(6):1013–1030

[13] O. Abdel-Hamid, A.-R. Mohamed, H. Jiang, and G. Penn, ‘‘Applying convolutional neural networks concepts to hybrid NN-HMM model for speech recognition,’’ in Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), Mar. 2022, pp. 4277–4280.

[14] O. Abdel-Hamid, L. Deng, and D. Yu, ‘‘Exploring convolutional neural network structures and optimization techniques for speech recognition,’’ in Proc. Interspeech, vol. 11, 2023, pp. 5–73.
[15] T. N. Sainath, A.-R. Mohamed, B. Kingsbury, and B. Ramabhadran, ‘‘Deep convolutional neural networks for LVCSR,’’ in Proc. IEEE Int. Conf. Acoust., Speech Signal Process., May 2020, pp. 8614–8618.
[16] W. Chan and I. Lane, ‘‘Deep convolutional neural networks for acoustic modeling in low resource languages,’’ in Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), Apr. 2019, pp. 2056–2060.
[17] H. Sak, A. Senior, and F. Beaufays, ‘‘Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition,’’ 2020, arXiv:1402.1128.
[18] J. T. Geiger, Z. Zhang, F. Weninger, B. Schuller, and G. Rigoll, ‘‘Robust speech recognition using long short-term memory recurrent neural networks for hybrid acoustic modelling,’’ in Proc. 15th Annu. Conf. Int. Speech Commun. Assoc., 2019, pp. 1–5.
[19] H. Sak, A. Senior, K. Rao, and F. Beaufays, ‘‘Fast and accurate recurrent neural network acoustic models for speech recognition,’’ 2020, arXiv:1507.06947.
[20] Y. Yu, X. Si, C. Hu, and Z. Jianxun, ‘‘A review of recurrent neural networks: LSTM cells and network architectures,’’ Neural Comput., vol. 31, no. 7, pp. 1235–1270, Jul. 2019.
[21] A. Graves, A.-R. Mohamed, and G. Hinton, ‘‘Speech recognition with deep recurrent neural networks,’’ in Proc. IEEE Int. Conf. Acoust., Speech Signal Process., May 2018, pp. 6645–6649.
[22] Hochreiter, S.; Schmidhuber, J. Long short-term memory. Neural Comput. 1997, 9, 1735–1780.
[23] Manjunathswamy B. E., Sunil Kumar M., B. S. . (2024). Designing a Framework for Developing an Adaptive Information Retrieval System that Personalizes Information. International Journal of Intelligent Systems and Applications in Engineering, 12(21s), 1325–1333.
[24] Sangamithra, B., Swamy, B. M., & Kumar, M. S. (2023). Evaluating the effectiveness of RNN and its variants for personalized web search. Optical and Quantum Electronics, 55(13), 1202.
[25] Sangamithra, B., B. E. Manjunath Swamy, and M. Sunil Kumar. "Personalized Ranking Mechanism Using Yandex Dataset on Machine Learning Approaches." Proceedings of the International Conference on Cognitive and Intelligent Computing: ICCIC 2021, Volume 1. Singapore: Springer Nature Singapore, 2022.
[26] Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555