ITC 3/52
Information Technology and Control
Vol. 52 / No. 3 / 2023
pp. 731-743
DOI 10.5755/j01.itc.52.3.31945

Crop Information Retrieval Framework Based on LDW-Ontology and SNM-BERT Techniques

Received 2022/07/26 | Accepted after revision 2022/09/15

HOW TO CITE: Ezhilarasi, K., Hussain, D. M., Sowmiya, M., Krishnamoorthy, N. (2023). Crop Information Retrieval Framework Based on LDW-Ontology and SNM-BERT Techniques. *Information Technology and Control*, 52(3), 731-743. https://doi.org/10.5755/j01.itc.52.3.31945

Crop Information Retrieval Framework Based on LDW-Ontology and SNM-BERT Techniques

K. Ezhilarasi
Dr. Dharambal polytechnic college for women, Chennai, 600113, India

D. Mansoor Hussain
School of Computer Science and Engineering (SCOPE), VIT University, Chennai,600127, India

M. Sowmiya
Sri Eshwar College of Engineering, Coimbatore, India

N. Krishnamoorthy
School of Information and Technology, Vellore Institute of Technology , Vellore, 632014, India

Corresponding author: kezhilarasi22@outlook.com

Currently, on the Internet, the information about agriculture is augmenting extremely; thus, searching for precise, relevant data of various details is highly complicated. To deal with particular difficulties like lower relevancy rate, false detection of retrieval resources, poor similarity rate, unstructured data format, multivariate data, irrelevant spelling, and higher computation time, an intelligent Information Retrieval (IR) system is required. An IR Framework centered on Levenshtein Distance Weight-centric Ontology (LDW-Ontology) and Sutskever Nesterov Momentum-centred Bidirectional Encoder Representation from Transformer (SNM-BERT) methodologies is presented here to overcome the complications as mentioned earlier. Firstly, the data is pre-processed, transmuting the unstructured data into a structured format, thus mitigating the error probabilities. Then, the LDW-Crop Ontology construction is done regarding the structured data. In the methodology presented, significance, frequency, and the suggestion of word in mind are considered to build Crop ontology. In the MongoDB database, the data being constructed are amassed. Then, by utilizing SNM-BERT, the data is trained for IR regarding clustered input produced by Inter Quartile Pruning Range-centred Hierarchical Divisive Clustering (IQPR-HDC) model. The LDW is computed for the provided user query; subsequently, the similarity evaluation outcomes are obtained from the database. The experiential evaluation displays that when analogized with the prevailing methodologies, a better accuracy of 94 % for simple queries and 92% for complex queries is achieved. Along with retrieval rate with lower computation time is achieved by the proposed methodology.

KEYWORDS: Information Retrieval, Crop Ontology, MongoDB, Query, Natural Language Processing.

1. Introduction

The data requested by the user on the Internet is processed using an information retrieval system. In recent years, on the web, a large quantity of information has been stocked in electronic format, and due to this, there is a huge demand for IR systems. Regarding the user query, IR focuses on discovering appropriate documents from huge document repositories [17]. By implementing the semantic-centered meaning of the words in the context as a substitute for simple keyword-centered matching, semantic IR focuses on expanding the classical retrieval models [5]. In recovering related documents from a huge repository of commercial, agricultural, medical, and educational, along with other documents, semantic information has its application [16, 23].

The agricultural experience motivates this research habits, related information, values concept, expert knowledge, and Agricultural Knowledge (AK), which is not only presented in the file but also utilized in agricultural production together with research in day-to-day life, practices, procedures, work, case studies, and norms is termed as AK [21, 6]; in addition, it is action-oriented, focused on dynamic organization, user needs, along with driving programmers, which helps people in engaging directly in agricultural production and economical operations. On the Internet, a massive quantity of data is available, which is increasing exponentially [8, 15]. In IR techniques, analogous technological developments have not matched this unrestricted information expansion. An online search does not return related outcomes at all times for enormous causes [19]. Initially, the keywords users submit can be relevant to multiple topics; thus, the search outcomes are not focused on the subject of interest [1]. Next, the question given by the user can be too short of capturing properly. Till the user sees the outcome, the user is not sure about what the person is searching for; even if the person knows, the user does not know how to create suitable questions [25].

The many questions conveyed to the IR system are ambiguous and imprecise [18]. The retrieval job is conducted by utilizing question representation along with document representation match scores [20]. The real text of the document is not used by the retrieval methodology; alternatively, the documents are embodied by a catalog of indexes/keywords [10]. This provides the records in a sensible order. Most IR techniques deploy standard keyword-matching algorithms to retrieve the related documents [2, 9, 27]. Alternatively, only some IR methodologies generate more related and actual outcomes. Conversely, when the ontology is merged with web languages, it is efficient to symbolize information effectively in a structured and semi-structured manner [12]. The web search methodology for information representation not only helps in the symbolization of information with hierarchy but also in the preservation of inheritance in the hierarchy [4, 13]. These sorts of methodologies will recover data not only with single-level legacy but also with multilevel inheritance. Thus, semantic IR centered on ontology is more efficient for semantic evaluation and retrieval of related information.

The main motivation of this research is to present IR in a semantic way so that relevant data can be quickly fetched with high accuracy. But, there remain specific issues like low relevancy rate, unstructured data format, poor similarity rate, irrelevant spelling, multivariate data, high computation time, etc., which make the IR rate imprecise for agriculture. The system had evolved a crop IR based on LDW-Ontology and SNM-BERT methodologies to deal with those problems.

The balance of the paper is arranged as below: the review of current IR in the literature is given in Section 2; the proposed techniques are illustrated in Section 3; a comparative evaluation with other related methods to validate the proposed methodology is explained in Section 4; the conclusion along with the directions for further study is given in se Section 5 concludes the paper with future orders.

2. Review of Literature

Jain et al. [22] developed an ontology centered on domain-specific knowledge. Pre-defined domain ontology, together with a global ontology, was deployed. A fuzzy ontology was made with the concept Net. The most semantically similar terms for a question were established by the evolved fuzzy ontology. A fuzzy membership function was built for the semantic links in the Global ontology Concept Net. Precision was significant to the web series engine. Every indicator was enhanced by about 10% regarding the framework. On

different search engines, precision ranges from 0.75-0.81 before query expansion, while accuracy ranges from 0.85-0.89 after query expansion. After the query expansion, the number of documents recovered was nearly enhanced by 1/1000, but it was impossible to handle multivariate data.

Boukhari et al. [11] utilized two methodologies to determine the link betwixt text and a specific concept they are (a) Vector Space Model (VSM) and (b) Description Logic (DL). VSM formed a partial match on documents and keywords from external assets. DL transmitted information in a suitable manner for enhanced competition. The contribution alleviated the constraint of the accurate match. It was utilized to index papers that used Medical Subject Headings (MeSH) thesaurus services with a close match. The trials were carried out on enormous corpora, which produced enhanced outcomes (+ 25% advancement in average accurateness analogized to earlier methodologies). Nevertheless, the similarity rate was low.

Mahalakshmi et al. [14] produced DL techniques for text along with images individually. Firstly, based on Convolution Neural Networks (CNN), A VGGNet-19 technique was used as a feature extractor; in addition, for picture retrieval, Euclidian distance-based similarity measurement was utilized. Concurrently, the Bidirectional-Long Short-Term Memory (BiLSTM) technique was utilized to recover textual materials. Every word was evaluated by the BiLSTM technique in a phrase consecutively, recovered the details, and incorporated them in the semantic vector. The constructed retrieval algorithms were analyzed on text and graphics for standard and specialized areas (agriculture) with Yahoo, Google, and Corel10K datasets. Compared to other techniques, this approach attained accuracy, recall, and F-score of 93 %, 85 %, and 90 %, respectively. Nevertheless, the relevance rate was low.

Chiranjeevi et al. [3] developed a text IR scheme based on the Recurrent Convolution Neural Network (RCNN), which effectively retrieved text documents and information for the user query. For pre-processing, retrieval with Term Frequency (TF)-Inverse Document Frequency (TF-IDF), and an RCNN classifier to extract contextual information, Tokenization, and stemming were utilized. An actual-time sophisticated search scheme was built on a massive gathering of MAHE University datasets. The produced RCNN-centred text document IR technique performed better regarding accuracy and recall, together with F-measure. A high-quality, high-performance text document retrieval search scheme was established. However, the technique needed to be more capable of handling unstructured data formats.

Qiu et al. [7] presented a fuzzy IR methodology that merged deep learning and fuzzy set theory techniques to capture the associations between words and query language. This technique was used to place the related characteristics of terms and get word embedding by large-scale data and the continuous-bag-of-words model. To develop retrieval effectiveness, it evaluated the relativity of words through word embedding with the feature of symmetry. According to the experimental data, the recall, accuracy, and harmonic average of two ratios of the devised technique surpassed those of the standard methodologies. Nevertheless, a significant amount of processing time is required to attain data.

Viji C. et al. [24] describes efficient Fuzzy based k-Nearest Neighbor Technique for Web Services Classification. It developed a farmer-centric crop ontology and IQPR (Inter Quartile Pruning Range) based Hierarchical divisive clustering technique to improve the precision and recall in agricultural information retrieval. According to the experimental data, it produced a better recall and accuracy ratio. The space complexity can be improved by implementing the Merkle tree, and precision can still be enhanced using deep learning algorithms.

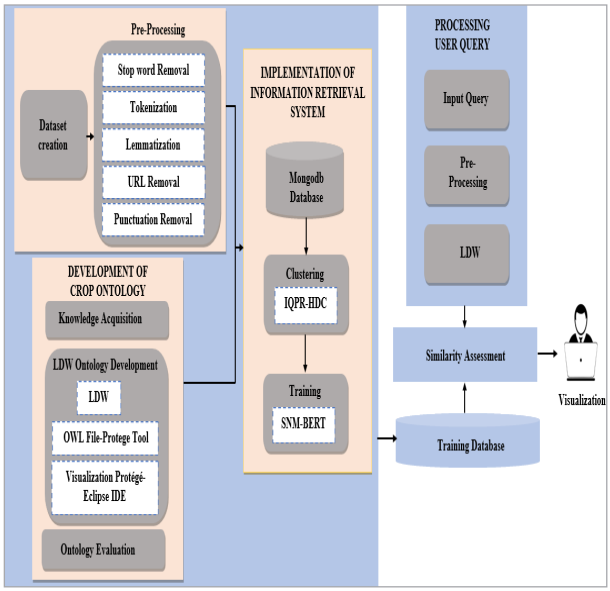
The literature work discussed in the article has many Pros. However, all techniques do not equalize their results. Some methods produce only accuracy and some techniques only concentrate on precision and recall. Moreover, a few methodologies improve the time of Information retrieval. This research focuses on improving the result in all metrics accuracy as well as time.

3. Methodology

Information technology has explored the increase of text document data in numerous businesses; thus, the structural arrangement of massive data is highly complicated. The retrieval of crop data is a complex task owing to factors like unstructured data format, lower relevance, wrong spelling, poor similarity rate, longer computing time, multivariate data, et cetera. Crops IR

centered on LDW-Ontology and SNM-BERT methodologies, exhibited in Figure 1, are presented here to overcome the aforementioned problems.

Figure 1
Proposed Crop Information retrieve framework



3.1. Pre-processing

The input data are structured in the pre-processing phase to enhance the quality of the text, potentially impacting the crop data evaluation.

a Stop word removal

In any language, stop words are the commonly utilized words. "The" is the most frequently seen stop word. "a", "and", "but", "how", "or", "what", et cetera are certain other stop words that are frequently noticed in the database. The words are prevented from being indexed with the aid of these stop words. Information pertinent to web services is not possessed by stop words in the data. Thus, the stop words must be taken away. To avoid these stop words, the proposed methodology is developed.

b Tokenization

Firstly, by utilizing tokenization, the text is divided into a structured format. Here, the text is partitioned into smaller units termed tokens. The tokens can be words, sub-words, or characters. The text can be understood effortlessly with the aid of this process.

$$\xi_1 = \xi_i^{tok}[D_i]. \quad (1)$$

c Lemmatization

Here, to identify the dictionary form of the word and mitigate the sparsity, the inflection is removed by eliminating the unnecessary characters (usually suffixes or prefixes) like ic/ical, less, ly, etc.

$$\xi_2 = \xi_i^{lem}[D_i]. \quad (2)$$

d URL removal

URL is a text which provides a reference to a location. To evaluate crop data, no extra data is provided by it.

$$\xi_3 = \xi_i^{url}[D_i]. \quad (3)$$

e Punctuation removal

The unsupportive parts of the data are eliminated by removing punctuations like apostrophes, commas, question marks, quotes, et cetera.

$$\xi_4 = \xi_i^{Pun}[D_i]. \quad (4)$$

Therefore, healthier text data is offered by the pre-processing in which the highest priority is provided to the words, which aids in the crop data evaluation

3.2. Crop Ontology

In crop ontology, the complex data are structured into simplified data; thus, supporting the data to be amassed and retrieved effortlessly. However, poor IR outcomes are produced regarding relevance while retrieving data centered on ontology. To ameliorate the relevance efficacy, the LDW-Ontology methodology is presented here, thus, conquering the aforementioned problem. To calculate the vital data of a document by considering the word's frequency along with significance, TF-IDF is employed here. However, the prevailing TF-IDF does not assess the spell-checking of the word or auto-suggestion. Thus, the working mechanisms of LD are amalgamated to resolve the issue mentioned above. With this mechanism, the exact document is retrieved by the user by retaining the relevancy even if the spellings are not correct. At first, to specify the term importance, the weighting factor ω_{ij} is gauged for every single term χ_j in document D_i. At last, every single document D_i is signified by a vector of weighted word stems.

$$\begin{cases} \omega_{ij} > 0 & if\ item\ j\ occurs\ in\ document\ i \\ \omega_{ij} = 0 & otherwise \end{cases}. \quad (5)$$

Terms, which are highly significant for content display, are differentiated with the aid of the term-weighting process. Numerous theories have been provided; in addition, the weight of a term in a document vector may be computed in a range of methodologies like TF-IDF. This word weighting means that in particular papers, the supportive terms will exist often; however, they exist rarely somewhere else. The TF factor and the inverse document frequency are the '2' components included in the term weight; these elements must be distinguished. The number of times a term exists in a document is mentioned as TF (*TF*).

$$TF_{ij} = \frac{F_{ij}}{L_i}, \quad (6)$$

where, the frequency of termin the document i is specified as F_{ij}, and the total number of keywords in the document i is signified as L_i. To differentiate one document from others, several weighting approaches are employed. This factor is termed IDF (*IDF*).

$$IDF_i = \log\left(\frac{N}{N_j}\right) + 1 \quad N_j > 0\ , \quad (7)$$

where the number of documents is defined as the number of documents in which term exists is proffered as N_j.

Regarding the LD, the resemblance amongst '2' words are calculated as,

$$\Gamma^{lev}_{w_i,q_i}(s,t) = \begin{cases} \max(s,t) & if\ \min(s,t) == 0 \\ \min\begin{cases} \Gamma^{lev}_{w_i,q_i}(s-1,t)+1 \\ \Gamma^{lev}_{w_i,q_i}(s,t-1)+1 \\ \Gamma^{lev}_{w_i,q_i}(s-1,t-1)+1_{(w_s \neq q_t)} \end{cases} & otherwise \end{cases} \quad (8)$$

Lastly, to create a composite weight for every single term in every single document, the conceptions of TF, IDF, along with LD are amalgamated.

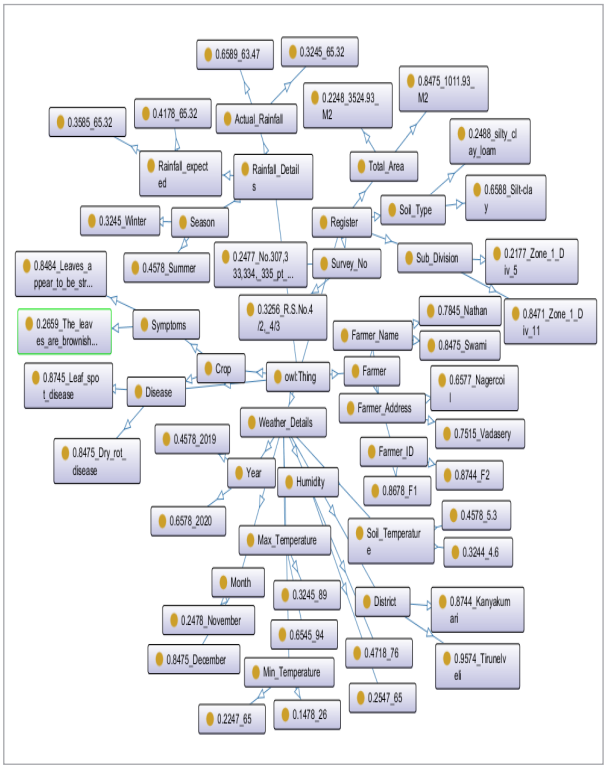
$$LDW = TF \times IDF + \Gamma^{lev}_{w_i,q_i}(s,t). \quad (9)$$

To build an ontology, the higher significant words are extracted along with they are organized in a data frame.

$$\Re = [w]_{s \times t}. \quad (10)$$

Classes, attributes, relations, instances, and axioms are the '5' significant components utilizing which the knowledge of data along with the cloud server is formulated by ontology construction. A family of knowledge representation languages for authoring ontology is mentioned as Web Ontology Language (OWL). Ontology is proffered as the specialization of conceptualization. Providing knowledge on certain domains, which are understandable by computers along with developers, is the intention of ontology. As exhibited in Figure 2, regarding entropy values, the OWL file is generated by utilizing the protégé tool.

Figure 2
LDW-Ontology Construction



For aiding ontology meant for the Semantic Web, is done using protégé OWL, which is an open-source tool. It is a plug-in expansion to the Protégé ontology development platform. Here, the users are permitted to alter ontology in the OWL and use a description logic classifier to retain the reliability of their ontol-

ogy. Following are the steps involved in the OWL's development using the protégé tool.

Step 1: The ontology's domain, along with scope, is estimated. After that, significant terms of the data frame are enumerated by reusing the prevailing ontology.

Step 2: Secondly, classes together with the hierarchy of the classes are proffered, for instance, the collection of individuals and an object in the world regarding the LDW.

Step 3: The facets of the slots are proffered; finally, the instances are created.

3.2.1. Visualization

Subsequent to the creation of the OWL file, by utilizing the protégé tool inbuilt with Eclipse Integrated Development Environment (IDE), it was visualized for further processing. Eclipse, which is an IDE, is utilized in computer programming. To customize the environment, a base workspace along with an extensible plug-in system is encompassed in this. Eclipse, which is written in Java, is primarily utilized to generate Java applications. However, plug-ins are also utilized to create applications in other programming languages like ABAP, Ada, C#, C, C++, FORTRAN, et cetera. Indented list, node-ling along with the tree, Zoom able, and focus context are the '4' methodologies involved in the visualization.

– **The indented list**

Here, the ontology's taxonomy is illustrated after the file system explore-tree view. These methodologies are spontaneous along with effortless to espouse. Via the intended list paradigm with sub-classes, it is demonstrated by " is-a " inheritance relationships; then, the corresponding superclasses are indented to the right.

– **Node link and tree**

It is a model often utilized for ontology visualization. A set of interlinked nodes denotes ontology. A better summary of the hierarchy, along with connections, is provided by this model. However, when utilized to visualize over hundred nodes, it may create clustered displays.

– **Zoom able**

The nodes in the lower levels of the hierarchy nested within their parents, along with smaller sizes, are presented by this methodology.

– **Focus + context**

The node in the center, together with the nodes linked around it to attain lesser space is represented by this model. Here, the nodes (classes) resist one another; conversely, the edges (links) attract them; therefore, semantically identical nodes are located close to one another. Thus, the OWL files are visualized by employing the above methodologies.

3.3. MongoDB Database

One of the document-oriented NoSQL databases is termed MongoDB. In the database, the ontology structured data is amassed. MongoDB permits users to enhance as the requirements alter since it possesses a flexible document data model. The data is amassed in collections created via individual document, which is nested in complex hierarchies; however, it is still simple to query along with the index. MongoDB has no pre-defined schema, unlike other relational databases. MongoDB, which offers an affluent document-oriented structure, is the fastest-emerging database.

3.4. Information Retrieval

In this, the data pertinent to crops or agriculture are retrieved from the database data.

3.4.1. Clustering

In accordance with the content, the data is grouped into different classes by performing clustering. To handle uncertain data along with boosting the IR rate, the grouping of data is performed. Nevertheless, to handle the multivariate data, a higher computation time is required by the prevailing clustering methodologies; also, whilst validating the larger data deeply, a higher error rate is obtained. The IQPR-HDC model is developed here to overcome the aforementioned complications. Partitioning a set of objects into consistent groups is the intention of this methodology.

Firstly, the data is regarded as a cluster $\left(\Re_1^C\right)$; then, concerning the similarity like $\left(r_1^C\right), \left(r_2^C\right), \left(r_3^C\right)$, and $\left(r_4^C\right)$, the cluster is split. Let $\left(\Re_1^C\right)$ is considered to be split randomly into '2' clusters $(\left(\Re_i\right), \left(\Re_j\right))$ by assigning $\left(K=2\right)$.

Let $\Re_i = \Re_1^C$ and $\Re_j = \beta$. Herein, the whole data points are specified as $\Re_1^C$ and the empty set is signified as β .Meanwhile, by utilizing Euclidean distance $(ED_{i,j})$, for every single data $(\beta_{i,j})$, the distance matrix is produced within the data points.

$$ED_{i,j} = \sum_{i,j=0}^{n} \left(\Re_i - \Re_j\right)^2 \tag{11}$$

$$ED_{I,J}\left(\Re_1^C\right) = \begin{bmatrix} \Re_{1\times1} & \Re_{1\times2} & \cdots & \Re_{1\times n} \\ \Re_{2\times1} & \Re_{2\times2} & \cdots & \Re_{2\times n} \\ \vdots & \vdots & \cdots & \vdots \\ \Re_{n\times1} & \Re_{n\times2} & \cdots & \Re_{n\times n} \end{bmatrix} \tag{12}$$

For every single $\beta_{i,j}^{+} \in \beta_i$, regarding the unweighted pair group mean average, the corresponding data are clustered. The average distance of the data points within the cluster $_i$ is calculated for the first iteration as,

$$D_1\left(\Re_i\right) = average\{ED_{i,j}\left(\Re_{n\times m}\right)\Re_{n\times m} \in \Re_i + \mathrm{I}\left(\Re_{n\times m}\right)\}, \tag{13}$$

where the Inter Quartile range (IQR), which retains the range of the data by the irrelevant data or outliers is specified as $\mathrm{I}\left(\Re_{n\times m}\right)$.The quartile score is obtained by dividing the absolute value of the individual feature value ($\Re_i$) minus the median value ($\widetilde{\Re}$) by its IQR (*IQR*). It is expressed as,

$$\Theta_{out} = \frac{\left|\Re_i - \widetilde{\Re}\right|}{I_U - I_L}, \tag{14}$$

where the upper and lower quartiles are signified as I_U and I_L. It is computed as,

$$IQR = Q_3 - Q_1 \tag{15}$$

$$I_U = Q_3 + 1.5(IQR) \tag{16}$$

$$I_L = Q_1 - 1.5(IQR), \tag{17}$$

where the IQR is specified as *IQR*, the third quartile range of the second half is signified and the second quartile range of the first half is proffered as Q_1. The median value, which is calculated by partitioning the data into '2' halves is represented as the second and first half; in this, the first half is lesser than the median value whereas the second half is higher than the median value; subsequently, the median value is estimated for the corresponding half.

The region with the largest distance is formed into the cluster $\left(\Re_j\right)$. Meanwhile, the alteration in the cluster is notated as,

$$\Re_j = \Theta\left\{\Re_n^{\Phi}\right\}. \tag{18}$$

Afterward, the following formula is utilized to compute the average distance for the subsequent sections.

$$D(\beta_{i,j}^{+}) = average\{ED_{i,j}\left(\Re_{n\times m}\right)\Re_{n\times m} \in \Re_i + \mathrm{I}\left(\Re_{n\times m}\right)\} - average\{ED_{i,j}\left(\Re_{n\times m}\right)\Re_{n\times m} \in \Re_j + \mathrm{I}\left(\Re_{n\times m}\right)\} \tag{19}$$

Until attaining the negative distance value, the iteration is repeated.After stopping the iteration, by calculating the diameter of the clusters and $\left(\delta_{dia}\left(\Re_j\right)\right)$, the splitting of the clusters takes place.

$$\delta_{dia}\left(\Re_i\right) = \max\{ED_{i,j}\left(\Re_i\right)\} \tag{20}$$

$$\delta_{dia}\left(\Re_j\right) = \max\{ED_{i,j}\left(\Re_j\right)\}. \tag{21}$$

Next, a standardized cost pruning, which eliminated certain cluster parts like base roots, and branches, is utilized here to prevent the overfitting of the clustering; thus, promoting the healthy growth of the hierarchical clustering. It is executed as,

$$\aleph_{path} = \delta_{dia}\left(\Re_i, \Re_j\right).\forall_{ccp}\left(\delta_{train\_test}\left|\Re\right|\right)$$
$$\psi_{ccp\_alphas} = \left[weak\Re_I^C\right]_v, \tag{22}$$

where the cluster's cost complexity measure is notated as $\forall_{ccp}$, the number of weak cluster nodes is symbolized as ψ_{ccp_alphas}.

At last, the process is repeated until obtaining an individual cluster, that is to say, $\left(r_1^C\right)$, $\left(r_2^C\right)$, $\left(r_3^C\right)$, and $\left(r_4^C\right)$.

3.4.2. Training

To retrieve the data, the clustered data are trained under the SNM-BERT, which maintains the balance betwixt bias and variance.A better IR rate with minimized error loss can be achieved by balancing the bias and variance. A Transformer, which is an attention mechanism, is utilized in this model. It recognizes the contextual relations amongst words or sub-words in a text.Let r_i^C be the clustered input text sequence;

similarly, the corresponding vectoring model can be expressed as $f(r) = \mathrm{T}$. After that, for all possible labels in the pre-defined category set $\mathrm{T} = \{\mathrm{T}_1 \ldots\ldots \mathrm{T}_c\}$, the conditional probability distributions are established.

Following are the '3' primary parts utilized for the construction of BERT.

1 Input Layer

Here, a tokenized sequence input text, which contains word, is pondered. $r_k^i = \left[r_1^1, r_2^2, r_3^2 \ldots\ldots r_n^n\right]$ symbolizes the i^{th} word in the sequence.A simple text sequence or two text sequences in one token sequence (that is to say, [Question, Answer]) is explicitly represented by an input sequence. In this, the first token, which includes the special classification embedding, is [CLS]; to separate segments, another special token [SEP] is utilized, which may also specify the end of the sequence. Subsequently, to mitigate the vocabulary size, regarding Word Piece embeddings with a 30,000 token vocabulary, the tokens are segmented. For instance, the word "helping" is partitioned into "help" and "ing".After that, to transmute the one-hot vector for "help" $\Re^H$, an embedding matrix $(\mathrm{N} \times \Theta)$ is utilized. Lastly, to obtain the final input representations, position embedding is performed.

2 Bert Encoder

Bert encoder contains merely 12 Transformer blocks along with 12 self-attention heads; thus, it does not permit over 512 tokens. A hidden state vector or a time-step sequence of hidden state vectors is obtained as the output of the BERT encoder [27]. In this methodology, the special [CLS]'s final remote state vector $H[CLS] \in \Re^H$ is the aggregate representation of the sequence. Here, the dimension with a default value of 768 is notated as *h*.

By employing Scaling Transformation (ST), the tokens are trained under dense layers in the BERT Encoder. The prevailing linear transformation, which creates warping effect frequency, does not retain non-linear relationships, thus, degrading the BERT's performance. The ST utilized her to overcome the aforementioned problems. At first, by deploying the activation function, the text sequence is activated. It is signified as,

$$\Pi_i^{fc} = act_{relu}[\mathrm{Z}], \tag{23}$$

where

$$\mathrm{Z} = \sum_{i=1}^{n} w_i r_i + B_i \tag{24}$$

$$act_{relu} = \max(0, \mathrm{Z}), \tag{25}$$

where the fully connected layer for i^{th} layers is notated as $\prod_i^{fc}$, the rectified linear unit activation function is symbolized as act_{relu}, for every single word, the weight and bias are specified as w_i and b_i. Next, to the dense layer, the ST is performed as,

$$\forall_i^{fc} = \Psi_{ST}[\mathrm{Z}], \tag{26}$$

where the ST function is signified as Ψ_{ST}. It is formulated as,

$$\begin{bmatrix} \Gamma_i' \\ \mathrm{Y}_j' \end{bmatrix} = \begin{bmatrix} \Psi_i & 0 \\ 0 & \Psi_j \end{bmatrix} \begin{bmatrix} \Gamma_i \\ \mathrm{Y}_j \end{bmatrix} \tag{27}$$

3 Output Layer

A simple softmax classifier is presented on the BERT encoder's top to compute the conditional probability distributions over pre-defined categorical labels along with forming a vector representation of the text sequence. Let the set of all trainable parameters for FTS-BERT be θ; in the output layer, the vector $H[CLS]$ is transmuted into the conditional probability distributions $\mathrm{Y}_j \mid H_{[CLS]}, \theta$ over all categorical labels $\mathrm{T} = \{\mathrm{T}_1 \ldots\ldots \mathrm{T}_c\}$ as,

$$\begin{aligned} \mathrm{P}(\mathrm{T}_j \mid H_{[CLS]}, \theta) &= Soft\max(H_{[CLS]}, \gamma^T) \\ &= \frac{\exp(\mathrm{P}(\mathrm{T}_j \mid H_{[CLS]}, \theta))}{\sum_{j=1}^{c} \exp(\mathrm{P}(\mathrm{T}_j \mid H_{[CLS]}, \theta))}, \end{aligned} \tag{28}$$

where the trainable task-specific parameter matrix is exhibited as $\gamma^T \in \Re^{c \times h}$ and the number of labels is represented as *c*.

Let the true label of the input sequence *c* be *t*, the predicted outcome will be the label with the highest $\mathrm{Y}_\Gamma = \arg\max(\mathrm{P}(\mathrm{Y}_j \mid H_{[CLS]}, \theta))$ value, and regarding the canonical cross-entropy function, the standard calculation loss $\aleph(\mathrm{T}, \theta)$ is computed as,

$$\aleph(\mathrm{T}, \theta) = \begin{cases} -\wp \ln \mathrm{P}(\mathrm{T}_\Gamma) - (1 - \Gamma) \\ \ln(1 - \mathrm{P}(\mathrm{T}_\Gamma)) \quad ifc = 2 \\ -\ln(\mathrm{P}(\mathrm{T}_\Gamma)) \quad ifc > 2 \end{cases} \tag{29}$$

The number of every single training batch is signified by the parameter batch size. The regularization strat-

egy dropout is espoused along with the value always maintained at 0.1 to prevent overfitting. To optimize the error function, rather than the Adam optimizer, the SNM is utilized by the BERT. By achieving an effectual rate of weight decay per step by curtailing the original loss function, the SNM surpasses the Adam optimizer. During a smaller Learning Rate (LR), the prevailing optimizer starts to converge; however, the training / test accuracy is reduced when the LR is high. Thus, by utilizing the momentum and velocity updation, the SNM optimizer surpasses the prevailing difficulties.

$$\begin{aligned} w_{t+1} &= w_t + \vartheta_t \mu_t - \eta_t \nabla f\left(w_t + \vartheta_t \mu_t\right) \\ \mu_{t+1} &= \vartheta_t \mu_t - \eta_t \nabla f\left(w_t + \vartheta_t \mu_t\right) \end{aligned}, \tag{30}$$

where the updated weights and current weights are illustrated as w_{t+1} and w_t, the momentum and velocity update are depicted as ϑ_t and μ_t, and the learning rate is notated as η_t.

The weights are updated regarding the SNM optimizer; then, the error minimization occurs rapidly with accurate data training. Figure 3 illustrates the proposed SNM-BERT's pseudo-code.

Figure 3

Pseudo de for proposed SNM-BERT Information retrieval

Input: cluster crop data $\left(r_1^C\right)\left(r_2^C\right),\left(r_3^C\right)$, and $\left(r_4^C\right)$
Output: information retrieval

Begin
Initialize input text sequence r_i^C respective vectoring model $f(r)=\mathrm{T}$
Define conditional probability for each text sequence
$\mathrm{T}=\{\mathrm{T}_1,.....\mathrm{T}_\sigma\}$
Compute input layer
For k *in* T
Perform tokenization $r_k^i=\left[r_1^1,r_2^2,r_3^2,.....r_n^n\right]$
Perform segmenting of tokens
Perform position embedding of segmented tokens
End for
Compute BERT encoder
For i *in* k
Evaluate dense layer
$\Pi_i^{fc}=act_{relu}[\mathrm{Z}]$
Perform scaling transformation using,
$\forall_i^{fc}=\Psi_{ST}[\mathrm{Z}]$
End for
Compute Output layer
For j *in* i
Compute output using,
$\mathrm{P}\left(\mathrm{T}_j \mid H_{[CLS]},\theta\right)=Soft\max\left(H_{[CLS]},\gamma^T\right)$
$=\frac{\exp\left(\mathrm{P}\left(\mathrm{T}_j \mid H_{[CLS]},\theta\right)\right)}{\sum_{j=1}^{\sigma}\exp\left(\mathrm{P}\left(\mathrm{T}_j \mid H_{[CLS]},\theta\right)\right)}$
End for
Compute error loss
$w_{t+1}=w_t+\vartheta_t\mu_t-\eta_t\nabla f\left(w_t+\vartheta_t\mu_t\right)$
$\mu_{t+1}=\vartheta_t\mu_t-\eta_t\nabla f\left(w_t+\vartheta_t\mu_t\right)$
Return the retrieved data
End begin

3.5. Processing of User Query

By utilizing the SNM-BERT, the user query processing is performed following the IR system's execution. In this, the user's query by a semantic web search engine is inputted. The testing process of this model is identical to the training procedure. The user input query is pre-processed; in addition, established the LDW for the input query. Subsequently, the higher prioritized word is connected with the trained database; then, by employing the SNM-BERT, the outcome is retrieved.

4. Results and Discussion

This part illustrates the performance of the crop IR system for the set of documents database with various formats of documents along with the evaluation of the proposed work. Regarding publically available datasets, the proposed IR is executed on the working platform of JAVA.

4.1. Performance Analysis

Regarding precision, recall, f-score, accuracy, returned vs. effective information, retrieved results, and query retrieval time, the proposed SNM-BERT is examined; in addition, the achieved outcomes are analogized with the prevailing techniques like Long Short Term Memory (LSTM), Bidirectional Long Short Term Memory (BILSTM), BERT, Attention Network (AN). The appraisal of the proposed methodology with the prevailing approaches is given below.

The results of the prevailing techniques together with the proposed approach are illustrated in Table 1. On simple and complex queries, the examination of metrics like accuracy, precision, F -score, and recall is implemented; in addition, these metrics aid in evaluating the training together with the testing capability of the proposed SNM-BERT against various queries to recover the crop data. Sustaining high values of the metrics symbolizes an enhanced technique to handle IR. Thus, the higher accuracy attained by the proposed approach for simple queries is 94.56% and for complex queries is 92.65%, while the accuracy obtained by the prevailing techniques for simple and complex queries ranges betwixt 71.24%-77.89% and 68.78%-76.89%, respectively, are low. Regarding precision, recall, and F-score, the value of the proposed technique

Table 1
Evaluation of proposed SMN-BERT based on various metrics

Performance Metrics (%)	Simple Queries				
	LSTM	BIL-STM	BERT	AN	Proposed SNM-BERT
Precision	53.64	61.98	69.54	70.12	97.89
Recall	62.12	65.14	72.45	73.45	86.45
F-score	65.89	66.98	74.56	75.89	87.89
Accuracy	71.24	74.52	76.54	77.89	94.56
Complex Queries					
Precision	51.24	60.88	67.89	68.98	95.64
Recall	60.19	62.15	70.14	71.45	85.78
F-score	62.45	63.45	71.45	74.89	85.99
Accuracy	68.78	71.45	72.45	76.89	92.65

for simple along with complex queries ranges betwixt 86.45%-97.89% and 85.78% to 95.64%, but the prevailing techniques attain a lower value for simple along with complex queries, which ranges betwixt 53.64%-75.89% and 51.24%-74.89% respectively. Hence, for uncomplicated and intricate questions, the proposed SNM-BERT gives enhanced performance.

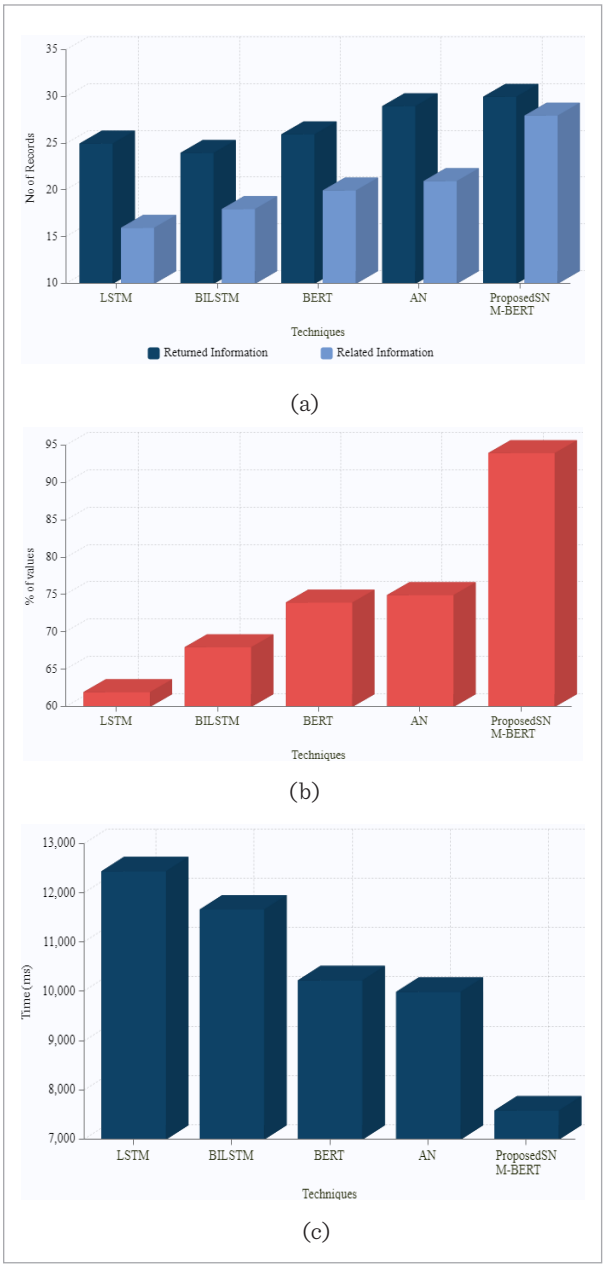
4.1.1. Performance Analysis for Simple Queries

In Figure 4, based on returned Vs effective information retrieved results, and query retrieval duration, the performance of the proposed SNM-BERT along with the existing methods is illustrated.

Regarding returned Vs effective information, the proposed SNM-Bert for simple query is pictorially shown in Figure 4(a). The proposed approach returns 30 records of which 28 are dynamic, which is related information, while the prevailing LSTM, BILSTM, BERT, and AN return average records of 26 of which 18 records are dynamic, which changes with a broad margin along with causes mismatch of data.

The recovery percentage of the proposed approach with the prevailing methodologies is illustrated in Figure 4(b). The scheme retrieves 94% of knowledge on the crop, while the information retrieved by the prevailing methodologies is LSTM (62%), BILSTM (68%), BERT (74%), and AN (75%) by using SNM-BERT. Thus, compared to the proposed techniques, the prevailing approaches obtain a low recovery rate on crop information.

Figure 4
Graphical demonstration of proposed SNM-BERT for simple queries based on (a) returned Vs effective information (b) retrieved outcomes (c) query retrieval duration



The amount of time taken by the methodologies to recover data is given in Figure 4(c). For query recovery, the SNM-BERT takes 7589ms, but the recovery rate of existing techniques is 12450ms (LSTM), 11670ms (BILSTM), 10235ms (BERT), and 9998ms (AN),

which is high. Thus, compared to prevailing techniques, the proposed SNM-BERT takes less duration to retrieve the query.

4.1.2. Performance Analysis for Complex Queries

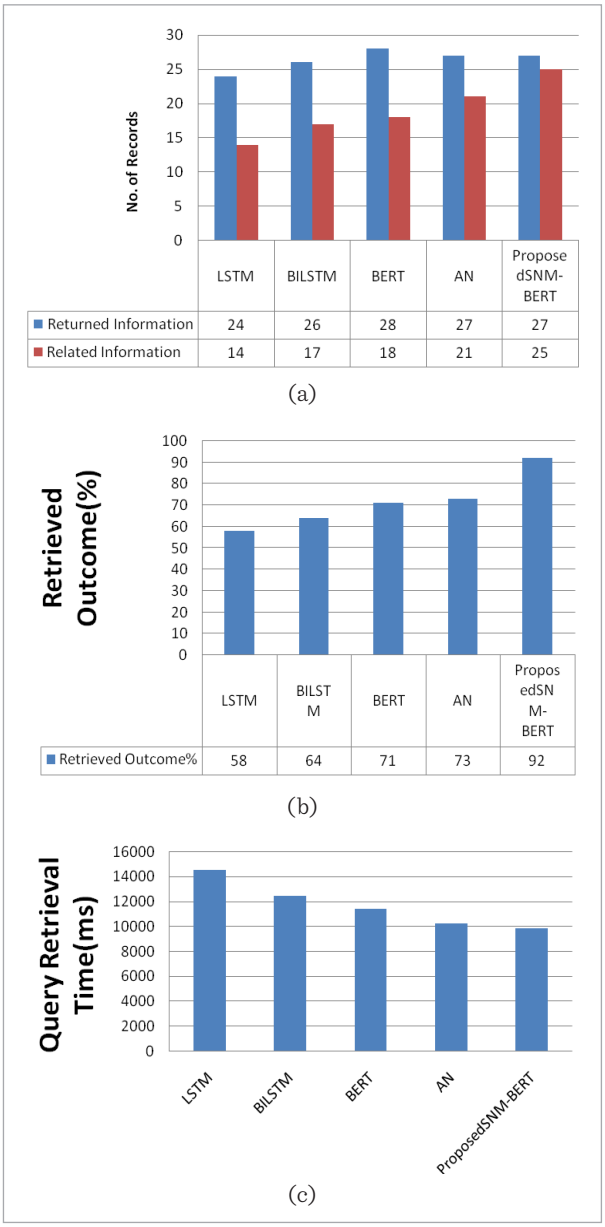
In Figure 5, the conclusion of the proposed SNM-BERT along with existing LSTM, BILSTM, BERT, and AN for intricate queries regarding returned Vs effective information, retrieved outcomes, and query retrieval duration is illustrated.

The proposed SNM-Bert for intricate queries on the basis of returned Vs effective information is shown in Figure 5(a). The proposed approach returns 27 records of which 25 are dynamic that is correlated information, while the prevailing techniques like LSTM, BILSTM, BERT, and AN returns an average of 26 records of which 17 are dynamic that varies widely and ends up in data mismatch.

The recovery percentages of the proposed together with prevailing methodologies for complex questions are analogized in Figure 5(b). By utilizing SNM-BERT, 92 % of crop information is retrieved by the scheme, but the existing techniques retrieve LSTM of 58%, BILSTM of 64%, BERT of 71%, and AN of 74%, which is low when analogized with the proposed methodologies.

Figure 5(c) depicts the time requisite by the approaches to achieve complex data. For complicated query recovery, the time used by the SNM-BERT is 9874ms, while 14560ms, 12458ms, 11478ms, and 10254ms are utilized by the current LSTM, BILSTM, BERT, and AN methodologies, which is comparatively high. Thus, compared to the prevailing techniques the proposed SNM-BERT performance is better.

Figure 5
Graphical demonstration of proposed SNM-BERT for complex queries based on (a) returned Vs effective information (b) retrieved outcomes (c) query retrieval duration



5. Conclusion

The procedure of searching and achieving specific information regarding the requisite from a pool of accessible resources is referred to as IR. Query IR aids users in meeting their requirements in agriculture applications. Nevertheless, owing to poor relevance rates, excessive data mismatches, and other factors, attaining information remains hard. A new framework centered on LDW-Ontology along with SNM-BERT methods is evolved, in order to resolve the limitations of the prevailing techniques and strengthen the query IR. To sustain a higher relevancy rate, the proposed approach collects deep information on the crop data. The Ontology construction has been done regarding frequency, importance, and the suggestion of words and then saved in a database. On the basis of clustered input, the saved data is used for training

the SNM-BERT. For simple and complex queries, the proposed approaches attain the accuracy of 94.56 % and 92.65 %, likewise, the SNM-BERT recovers 94 % and 92 % of the information. When recovering the data, SNM-BERT takes 7589ms and 9874ms for simple along with complex queries respectively. Thus, the proposed SNM-BERT surpasses the prevailing algorithms for basic and intricate queries.

References

1. Agnieszka, K. Towards Knowledge Handling in Ontology-Based Information Extraction Systems. Procedia Computer Science, 2018, 126, 2208-2218. https://doi.org/10.1016/j.procs.2018.07.228
2. Amarnath, P., Partha, P., Ranjita, D. LSTM Neural Network Based Math Information Retrieval. Proceedings of Second International Conference on Advanced Computational and Communication Paradigms, Gangtok, India, 2019.
3. Chiranjeevi, H.S., Manjula, K. Advanced Text Documents Information Retrieval System for Search Services. Cogent Engineering, 2020, 7(1), 1-16. https://doi.org/10.1080/23311916.2020.1856467
4. Damasevicius, R. Automatic Generation of Concept Taxonomies from WEB Search Data using Support Vector Machine. Proceedings of the 5th International Conference on Web Information Systems and Technologies, 2009, 673-680.
5. Debolina, M., Chandan, M., Soumya, P., Panda, J., Mohanty, A., Amit Mangaraj, A. Fuzzy-Cluster-Based Semantic Information Retrieval System. Proceedings of the fourth International Conference on Computing Methodologies and Communication, Erode, India, 2020.
6. Deepali, V., Shweta, T. A Text Preprocessing Approach for Efficacious Information Retrieval. Springer, Singapore, 2019.
7. Dong, Q., Haihuan, J., Shuqiao, C. Fuzzy Information Retrieval Based on the Continuous Bag-of-Words Model. Symmetry, 2020, 12(2), 1-11. https://doi.org/10.3390/sym12020225
8. Eko, W., Fenrianto, M., Jundi, H., Reko, S., Okky R., Sulaeman, R. Information Retrieval System for Searching JSON Files with Vector Space Model Method. International Conference of Artificial Intelligence and Information Technology (ICAIIT), Yogyakarta, Indonesia, 2019.
9. Ezhilarasi, K., Maria Kalavathy, G. Enhanced Neuro-Fuzzy-Based Crop Ontology for Effective Information Retrieval. Computer Systems Science and Engineering, 2022, 41(2), 569-582. https://doi.org/10.32604/csse.2022.020280
10. Jiafeng, G., Yixing, F., Liang, P., Yang Qingyao, A., Hamed, Z., Chen, W., Bruce, C., Xueqi, C. A Deep Look into Neural Ranking Models for Information Retrieval. Information Processing and Management, 2019.
11. Kabil, B., Mohamed, N. DL VSM based Document Indexing Approach for Information Retrieval, Journal of Ambient Intelligence and Humanized Computing, 2020, 11(1), 1-12.
12. Komala, A., Padmanabha, R. Information Retrieval Models in Neural Networks Framework: A Survey. Materials Today Proceedings, 2021.
13. Lakshmana Kumar, R., Kannammal, N., Krishnamoorthy, S., Kadry, S., Nam, Y. Semantics-Based Clustering through Cover-Means with Ontovsm for Information Retrieval. Information Technology and Control, 2020, 49(3), 370-380. https://doi.org/10.5755/j01.itc.49.3.25988
14. Mahalakshmi, P., Sabiyath Fatima, N. Ensembling of Text and Images using Deep Convolutional Neural Networks for Intelligent Information Retrieval. Wireless Personal Communications, 2021. DOI: 10.1007/s11277-021-08211-x. https://doi.org/10.1007/s11277-021-08211-x
15. Maryam, H., Hamid, R. Effective Retrieval of Related Documents based on Spelling Correction to Improve Information Retrieval System. Proceedings of 3rd Conference on Swarm Intelligence and Evolutionary Computation (CSIEC), Bam, Iran, 2018.
16. Mohammad, A., Ullah, S., Akhter, H. Ontology-based Information Retrieval System for University Methods and Reasoning. Springer, Singapore, 2019.
17. Naw, T., Wai Khin, N. Query Classification Based Information Retrieval System. International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS), Bangkok, Thailand, 2018. https://doi.org/10.1109/ICIIBMS.2018.8549988
18. Remi, S., Varghese, S. C. Domain Ontology has Driven Fuzzy Semantic Information Retrieval. Procedia Computer Science, 2015, 46, 676-681. https://doi.org/10.1016/j.procs.2015.02.122
19. Ridwan, A., Kambau, Z., Arifin, H. Concept-based Multimedia Information Retrieval System using Ontology Search in Cultural Heritage. Proceedings of Second In-

ternational Conference on Informatics and Computing (ICIC) Jayapura, 2017.

20. Ritika, B., Sonal, C. Design and Development of a Semantic Web-Based System for Computer Science Domain-Specific Information Retrieval. Perspectives in Science, 2016, 8, 330-333. https://doi.org/10.1016/j.pisc.2016.04.067
21. Saravana Kumar, C. S., Santhosh R. Effective Information Retrieval and Feature Minimization Technique for Semantic Web Data. Computers and Electrical Engineering, 2020, 81, 1-14. https://doi.org/10.1016/j.compeleceng.2019.106518
22. Shivani, J., Seeja, K. R., Rajni, J. A Fuzzy Ontology Framework in Information Retrieval using Semantic Query Expansion. International Journal of Information Management Data Insights, 2021, 1(1), 1-15. https://doi.org/10.1016/j.jjimei.2021.100009
23. Suma, V. A Novel Information Retrieval System for the Distributed Cloud Using a Hybrid Deep Fuzzy Hashing Algorithm. Journal of Information Technology and Digital World, 2020, 2(3), 151-160. https://doi.org/10.36548/jitdw.2020.3.003
24. Viji, C., Beschi Raja, J., Parthasarathi, P., Ponmagal, R. S. Efficient Fuzzy Based K-Nearest Neighbor Technique for Web Services Classification. Microprocessors and Microsystems, 2020, 76(103097), 1274-1278. https://doi.org/10.1016/j.micpro.2020.103097
25. Youcef, D., Asma, B., Fournier-Viger, J., Lin, C.-W. Fast and effective Cluster-based Information Retrieval Using Frequent Closed Itemsets. Information Sciences, 2018, 453, 154-167. https://doi.org/10.1016/j.ins.2018.04.008
26. Zahra, A., Saeedeh, M. Text-Based Question Answering from Information Retrieval and Deep Neural Network Perspectives: A Survey. Wiley Interdisciplinary Reviews Data Mining and Knowledge Discovery, 2021, 11(6), 1-40. https://doi.org/10.1002/widm.1412
27. Zongda, W., Shigen, S., Xinze, L., Xinning, S., Enhong, C. A Dummy-Based User Privacy Protection Approach for Text Information Retrieval. Knowledge-Based Systems, 2020, 195(4), 1-14. https://doi.org/10.1016/j.knosys.2020.105679.