Hindawi
Computational Intelligence and Neuroscience
Volume 2022, Article ID 4636931, 15 pages
https://doi.org/10.1155/2022/4636931

Research Article

Managing and Retrieving Bilingual Documents Using Artificial Intelligence-Based Ontological Framework

Abdulaziz Fahad Alothman and Abdul Rahaman Wahab Sait

Department of Documents and Archive, Center of Documents and Administrative Communication, King Faisal University, P.O. Box 400, Al Hofuf 31982, Al-Ahsa, Saudi Arabia

Correspondence should be addressed to Abdulaziz Fahad Alothman; afalothman@kfu.edu.sa

Received 25 May 2022; Revised 20 July 2022; Accepted 6 August 2022; Published 25 August 2022

Academic Editor: Elpida Keravnou

In recent times, artificial intelligence (AI) methods have been applied in document and content management to make decisions and improve the organization's functionalities. However, the lack of semantics and restricted metadata hinders the current document management technique from achieving a better outcome. E-Government activities demand a sophisticated approach to handle a large corpus of data and produce valuable insights. There is a lack of methods to manage and retrieve bilingual (Arabic and English) documents. Therefore, the study aims to develop an ontology-based AI framework for managing documents. A testbed is employed to simulate the existing and proposed framework for the performance evaluation. Initially, a data extraction methodology is utilized to extract Arabic and English content from 77 documents. Researchers developed a bilingual dictionary to teach the proposed information retrieval technique. A classifier based on the Naïve Bayes approach is designed to identify the documents' relations. Finally, a ranking approach based on link analysis is used for ranking the documents according to the users' queries. The benchmark evaluation metrics are applied to measure the performance of the proposed ontological framework. The findings suggest that the proposed framework offers supreme results and outperforms the existing framework.

1. Introduction

The recent development in the information retrieval (IR) techniques facilitates effective document management (DM) functionalities in organizations. The process of retrieving relevant information by passing a query in a search engine is called IR [1–5]. A query is a text in a natural language to extract a relevant document. For instance, a search engine can fetch approximately one million webpages for a user query. Organizations apply business intelligence (BI) tools to process a large amount of data and retrieve valuable information [6–11]. To compete effectively, organizations should analyze and leverage a wide range of data, information, and expertise in order to make effective decisions. Decision Support Systems (DSS) are interactive computer-based systems designed to assist decision-makers in identifying and solving problems, completing decision process tasks, and making decisions [12–19]. These systems are becoming increasingly popular among managers due to this trend. However, the shortcomings include unstructured data and complex queries reducing IR technologies' performance. In other words, users failed to retrieve relevant documents for their queries [20–25]. Moreover, the absence of bilingual (English and Arabic) IR systems causes difficulties for organizations in the Middle East countries.

On the one hand, there is an availability of a wide range of IR systems. On the other hand, there is a lack of domain-specific ontologies or IR systems to serve an organization [26–30]. In the Kingdom of Saudi Arabia (KSA), most organizations offer a sophisticated application for employees and stakeholders to share the information and valuable documents. The internal communication between employees generates a larger amount of documents [31–35]. Organizations demand an artificial intelligence (AI) based system to generate knowledge from the documents [36–41].

In the current environment, organizations store documents in Portable Document Format (PDF) form and their relevant metadata in a different storage location. The AI tools widely use the metadata for making decisions [42–47]. There are many techniques for retrieving a document using a query. Thus, organizations cannot access the document's content without its metadata [48–51]. The KSA's Vision 2030 motivates researchers to apply innovative techniques to the current functionalities of the organization. Therefore, developing an ontological framework for document management can support organizations in satisfying their stakeholders. In addition, the role of natural language processing (NLP) in the ontological framework enables individuals to interact with the system in their natural language [52–54].

The objectives of the study are:

(1) Build a data extraction model for extracting text from Arabic and English PDF documents.

(2) Construct a name entity-relationship (NER) classifier for classifying the documents.

(3) Implement a ranking approach to retrieve relevant documents for a user query.

The remaining part of the study is organized as follows: Section 2 reports the features of existing literature and research gaps. Section 3 outlines the research methodology and Section 4 discusses the study's findings. Finally, Section 5 concludes the study with its future direction.

2. Literature Review

DM is one of the critical processes in an organization. The communication between the users of the internal and the external units of an organization may generate a document [1–5]. Organizations follow the government and the international archival policies to store and manage their documents [6–9]. The existing studies show many techniques and frameworks for managing documents and IR [9–15].

Zaman et al. proposed an ontological framework for retrieving scientific sources [1]. They employed fuzzy rule base and word sense disambiguation for extracting information from multiple scientific documents. The experimental outcome suggests that the framework was less sensitive to the document file format modifications. However, there is limited information on the performance of the framework.

Yao et al. developed an AI-based ontological model for predicting the side effects of medicines [2]. The model had certain entities such as value and relationships. The value and relationship are used to indicate the drug and its side effects. The AI model's fuzzy and dynamically defined latent attributions can redefine vital records. The performance of the IR model is affected by the limitations, including the lack of negative data and the smaller dataset.

Crimp and Trotman proposed a linguistic model using Roget's and WordNet [3]. They employed an Attre search engine and evaluated the model using the mean average precision (MAP) metric. The outcome highlights the better performance of the linguistic model. However, the authors utilized a limited set of features from Roget's and WordNet.

Vocabulary mismatch is one of the limitations of the IR system. To overcome this limitation, query expansion (QE) techniques are developed. However, QE techniques are based on specialization and context relationships [4]. Raza et al. discussed that domain-specific ontologies are widely used in medicine, agriculture, and other scientific fields [4]. Multiple automated QE systems are proposed in IR [5]. Yunzhi et al. constructed an Arabic ontology based on the Protégé and SPARQL language to extract candidate expansion terms [6].

Domain-independent ontologies serve as a valuable resource for multiple domains. Aggarwal and Paul extracted expansion concepts from DBPedia and Wikipedia ontologies using semantic analysis [7]. However, the shortcomings include ambiguous terms and a lack of unique ontological properties causes more complexities. Zingla et al. and Omar et al. proposed hybrid models for extracting expansion concepts from DBpedia and Wikipedia [8, 9]. They employed Microblog and TREC 2011 datasets for evaluating their ontological performance.

The existing studies focus on the specific domains, and there are no studies on the DM and IR [10–15]. There is a lack of bilingual ontological framework for the organizations in the KSA. Most studies considered the NER classification of webpages as a primary objective rather than the ranking approach [16–21]. Particle Swarm Optimization (PSO) is used to enhance and train Hidden Markov Model (HMM) estimate approaches (PSO). PSO identifies the optimal response for a user query. For instance, the metadata of a document can be extracted using this approach [22–27]. A text extractor can be built using the AI technique for the automated extraction of key terms from a document [28–34]. An ontology-based dynamic information extraction framework identifies a wide range of document resources published in the scientific community and extracts the whole structural information [35–41]. The accuracy and scope of information extraction can be improved using an entity-relationship-based framework [42–47]. Few research works employed the term—frequency methodology for ranking the webpages [48–54]. Thus, there is a demand for a practical ontological framework for managing documents and retrieving information based on the user query. Furthermore, the recent ontological frameworks, including Gohar Zaman et al. (GOF) and Yuazhe Yao et al. (YOF), are employed to compare the performance of the proposed ontological framework (POF).

3. Research Methodology

In order to achieve the objective of the study, researchers construct a bilingual (Arabic and English) ontological framework for retrieving documents. Figure 1 presents the proposed research framework of the proposed study. It covers four phases including data extraction, NER classification, ranking technique, and performance evaluation.

The first phase outlines the data extraction process for extracting text from PDF documents. The NER

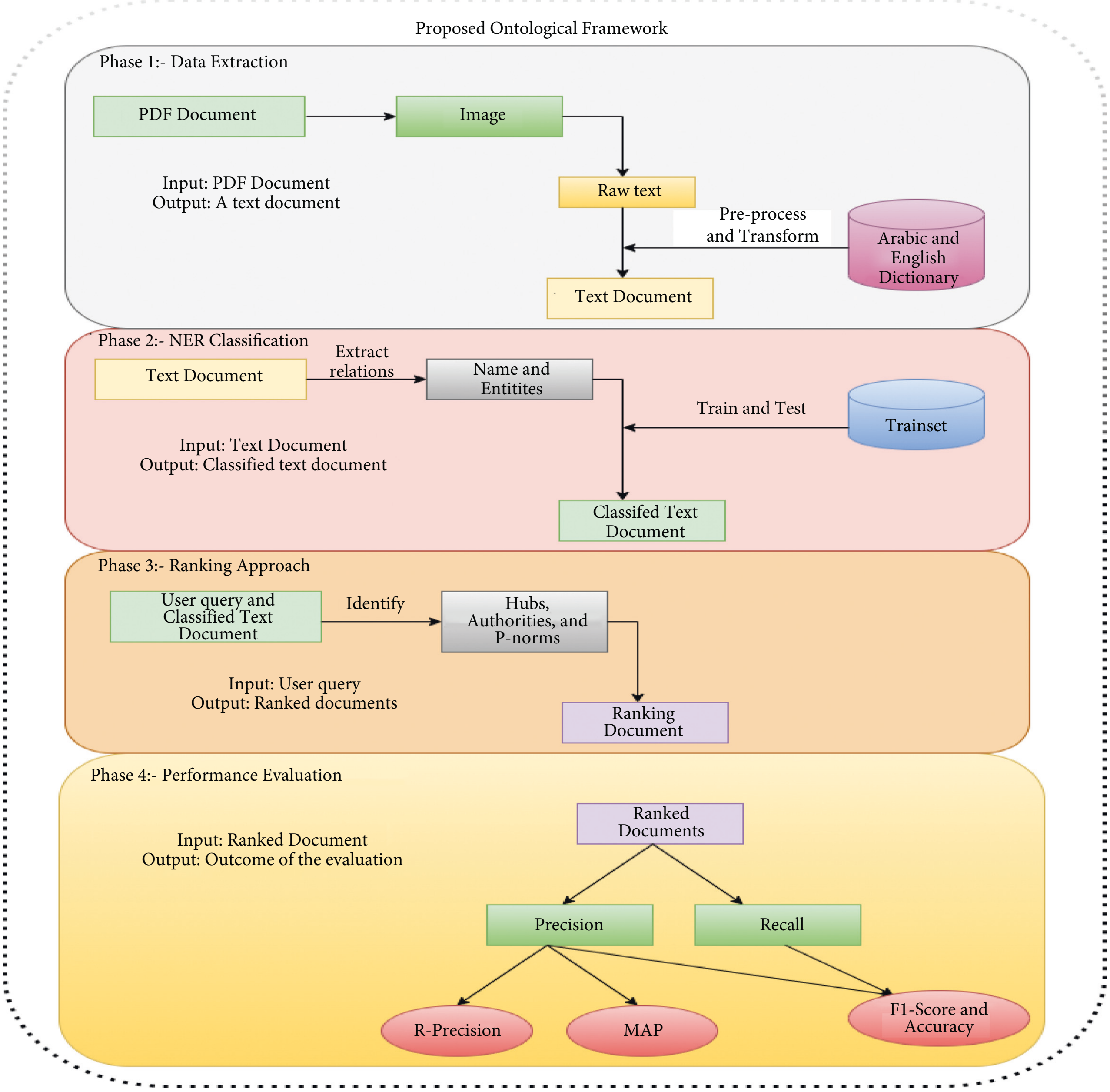


Figure 1: Proposed research framework.

classification using MNB is described in the second phase. The third phase highlights the ranking techniques to retrieve relevant documents. Lastly, the fourth phase evaluates the performance of the proposed ontological framework (POF).

3.1. Phase 1: Data Extraction.

This phase transforms the PDF document into a text document. It supports the retrieval process to extract relevant documents. During communication, employees or stakeholders widely use PDF documents for sharing information. It is difficult to search a PDF document using a user query. Therefore, A PDFtoWord is developed in order to automate the process of converting a PDF document to a Word document. However, a PDF document may contain handwritten content which cannot be converted into a Word document. In other words, converting handwritten text into standard text is challenging. Figure 2 shows the activities of phase 1. Initially, a document is converted to image format in order to extract the text. The extracted raw text is preprocessed and stored as a set of keywords and a word file. Phase 1 supports the proposed framework to search a document using a keyword. It overcomes the limitations of the searching document using metadata.

Thus, this study transforms the PDF document into an image, JPEG, or PNG format. The procedure of the data extraction process is as follows:

Step 1: Input a PDF document.

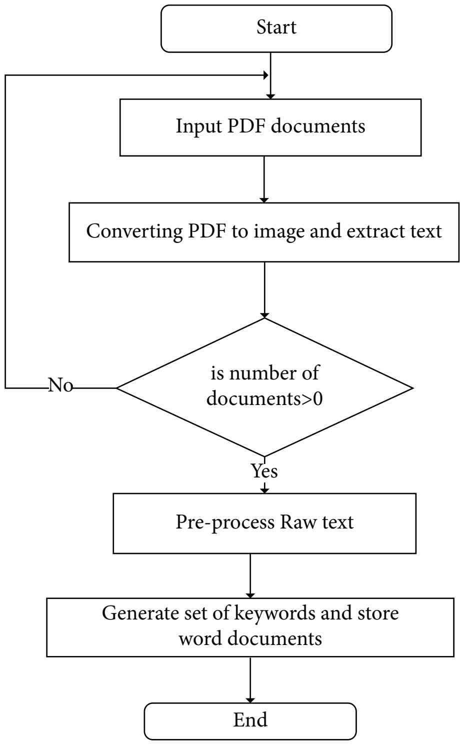

Start
Input PDF documents
Converting PDF to image and extract text
is number of documents>0
No
Yes
Pre-process Raw text
Generate set of keywords and store word documents
End

Figure 2: Text extraction process.

Step 2: Converting documents from a PDF form to JPEG or PNG format.

Let PD be the PDF document, ID be an image format of the PDF document. Doc_To_Img is a function for converting the documents from PDF to image structure and hres is the attribute to make the image with high resolution (1100 × 900 pixels at 600 pixels per inch). Equation (1) shows the expression of converting the PDF document into image format.

$$\mathrm{ID} = \mathrm{Doc\_To\_Img}(\mathrm{PD}) \cdot \mathrm{hres}(1100, 900, 600). \quad (1)$$

Step 3: Designing a text extractor.

A text extractor is designed using the AI-based Tessaract module that extracts the text from the image [55]. Nonetheless, the module is limited to the English Language. Thus, a dedicated Arabic dictionary is developed and integrated with the Tessaract module. Let Tessaract() be a function to extract text from an image, P_process be a preprocess function, RT is a raw text, and d be the document's content. Equations (2) and (3) outline the extraction and preprocessing of text.

$$\mathrm{RT} = \mathrm{Tessaract}(\mathrm{ID}), \quad (2)$$

$$D = P\_\mathrm{process}(\mathrm{RT}). \quad (3)$$

The P_process function employs an Arabic and English dictionary to ensure the RT is correct. During the text extraction, the extracted text may contain some errors. For instance, "name" may be misspelt as "mame." Thus, the dictionary corrects the erroneous content.

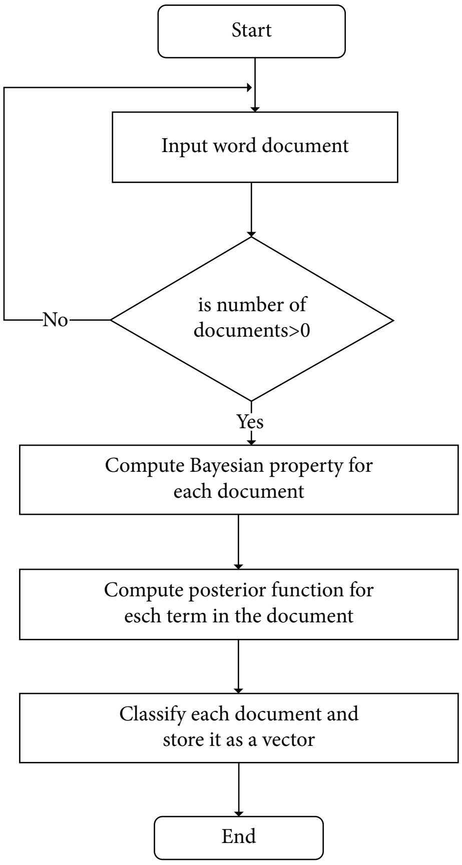

Start
Input word document
is number of documents>0
No
Yes
Compute Bayesian property for each document
Compute posterior function for esch term in the document
Classify each document and store it as a vector
End

Figure 3: Entity–relationship classification.

3.2. Phase 2: NER Classification. In this proposed study, the researchers employed the Multinomial Naïve Bayes (MNB) for classifying the documents [56]. Each document is a collection of words. A class or label consists of homogeneous documents. MNB algorithm is widely used in NLP applications. It classifies documents based on the statistical outcome of the content. Figure 3 outlines the processes of phase 2. The word document is processed using the Bayesian property. The posterior function is computed for each term in the document. Finally, each document is stored as a vector. The following section explains the computation of Bayesian property and posterior function in detail.

The classification assigns a text segment to a class using the probability of documents in the class of other documents. The process of grouping similar documents under a specific class is called labeling. Let S be the document to be classified. Each document in S is treated as a string related to one or multiple documents based on a class L. The classification of documents is based on a train set that contains the classified documents according to the document relationship in Figure 4. Figure 5 shows the classification of documents using the train set.

Let f be the vector in S, f_i be the feature in f representing the i^{th} term in L. The core of the MNB model is the evaluation of probability-based decision function. The Bayesian probability for the documents is expressed in equations (4) and (5). The likelihood of the i^{th} term f_i

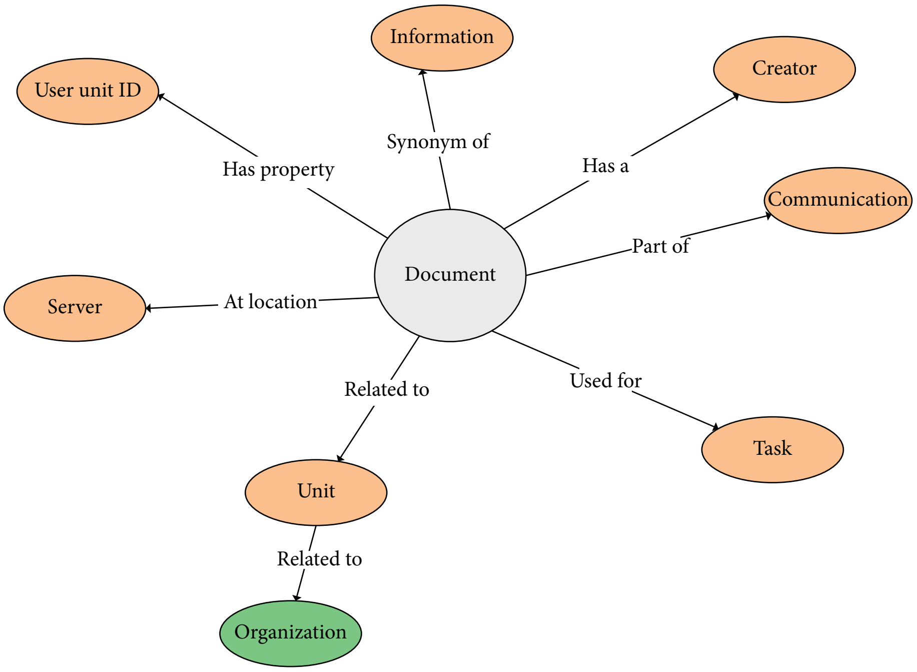


Figure 4: Document relationship.

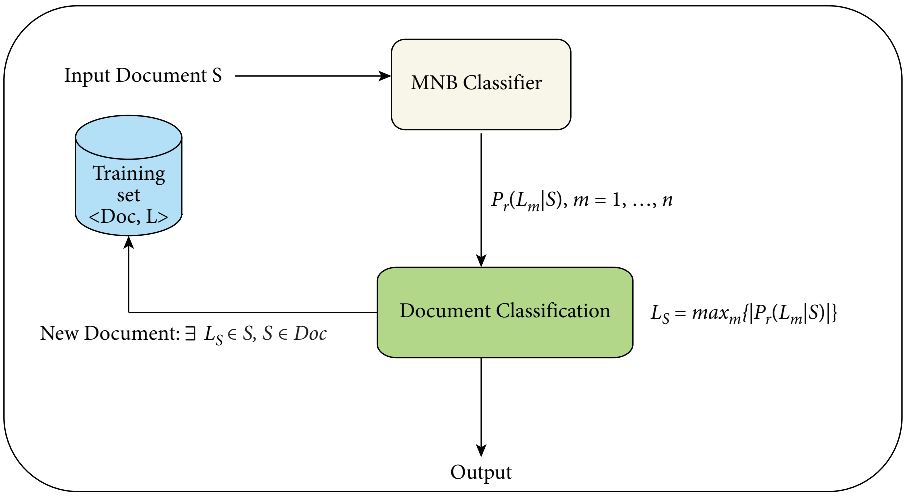

Input Document S
MNB Classifier
Training set <Doc, L>
$P_r(L_m|S)$, $m = 1, \ldots, n$
Document Classification
$L_S = max_m\{|P_r(L_m|S)|\}$
New Document: $\exists\ L_S \in S$, $S \in Doc$
Output

Figure 5: Document classification model [24].

belonging to the class L_m is shown in equation (6). Equation (7) outlines the MNB in the log space. The evaluation $\log(P)$ is expressed in (8).

$$P(L_m|f) = \frac{P(L_m)XP(fL_m|)}{P(f)}, \quad (4)$$

$$P_r(f|L_m) = \frac{\left(\sum_{i=1}^{n} f_i\right)!}{\left(\prod_{i=1}^{n} f_i\right)!} X \prod_{i=1}^{n} p_{m_i}^{f_i}, \quad (5)$$

$$P_r(f|L_m) = \prod_{i=1}^{n} P(f_i|L_m), \quad (6)$$

$$\log P_r(L_m|f) \propto \log P(L_m) + \sum_{i=1}^{n} f_i X \log P(f_i|L_m), \quad (7)$$

$$\log(P) = \begin{cases} \ln(P), & P < 1, \\ 1.0, & P \leq 1. \end{cases} \quad (8)$$

The following steps are followed for classifying the documents using the MNB classifier:

Step 1: Divide the documents (S) into a group of n-terms.

Step 2: Repeat the following process for each i^{th} term in S.

Step 2(a): Compute the Bayesian probability using equation (4).

Step 2(b): Evaluate the $P(Lm)$ function for each document i in L.

Step 2(c): Compute the posterior function by integrating the prior function to the sum of each term using equation.

$$P_r(L_m|f) = \log P(L_m) + \sum_{i=1}^{n} [f_i * \log P(f_i|L_m)]. \quad (9)$$

Step 3: Compute L_S of S using Eqn.

$$L_S = \underset{m \in (1 \ldots n)}{\text{argMax}} |P_r(L_m|f)|. \quad (10)$$

Step 4: Repeat Steps 1 to 3 with the train set.

Step 5: Classify the documents and store them as a vector.

3.3. Phase 3: Ranking Approach. In this phase, the researchers apply the ranking approach based on the study [19]. Figure 6 highlights the flow of processes in phase 3. Phase 3 initializes the vector and computes Hub and authorities similar to the HITS algorithm. However, a random walk feature is employed for updating Hub and authority weights.

The approach is the combination of PageRank [20], HITS [21], and SALSA [22] algorithms. It is a link-based ranking technique. Assume a_i be the authority weight, h_i be the hub weight. This ranking approach considers the document with higher a_i as better authorities and higher h_i as better hubs. Figures 7(a) and 7(b) show the authorities and Hub pointing with P. The weights of h_i and a_i are updated dynamically.

Documents are ranked according to the user query based on the weights of h_i and a_i. It works similar to HITS using bipartite graph (G) and seed set (R_f). In addition, the P-norms, a parameter, assign multiple normalized weights to each document link. A duplicative feature is employed to initiate Hub and authority, and vice-versa. The random walk feature of SALSA is used to identify the highly reachable node in G. Finally, normalization of the $\overrightarrow{A}$ generates the ranked documents. The following procedure is applied for the ranking documents:

Step 1: Input user query and initialize the N_h and N_a node and the parameter (P), P-norm value.

Step 2: Initialize $\overrightarrow{A} = 1$ $(\overrightarrow{A} \longrightarrow N_a)$

Step 3: For each element i in N_h

Step 3a: For each element I in the set of nodes pointed by i^{th} node

Compute Temp = Temp + $a_j P/|B(j)|$

Step 3b: Compute $h_j = \sqrt[p]{\text{Temp}}$

Step 4: For each element k in N_a

Step 4a: For each element l in $B(k)$

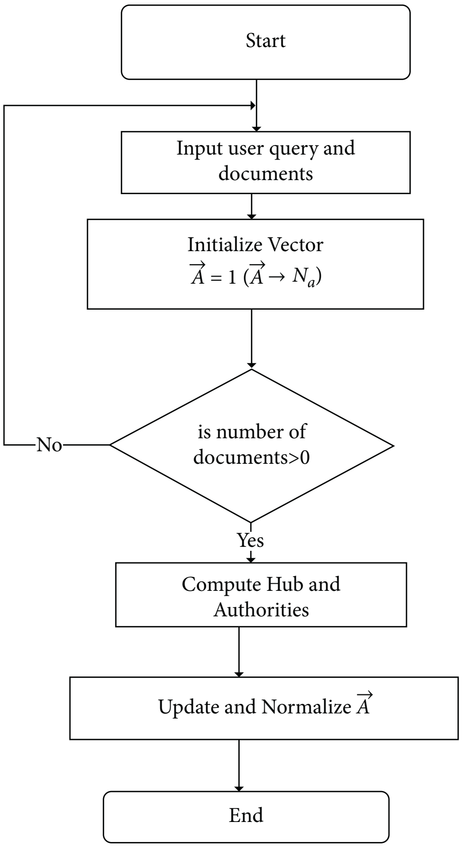

Start
Input user query and documents
Initialize Vector
$\overrightarrow{A} = 1$ $(\overrightarrow{A} \rightarrow N_a)$
is number of documents>0
No
Yes
Compute Hub and Authorities
Update and Normalize $\overrightarrow{A}$
End

Figure 6: Proposed ranking approach.

Compute Temp = Temp + $h_l P/|F(l)|$

Step 4b: Compute $a_k = \sqrt[p]{\text{Temp}}$

Step 5: Repeat Step 3 to 5 until weight converges

Step 6: Update $\overrightarrow{A}$ with authority weight

Step 7: Normalize $\overrightarrow{A}$, ranked documents.

3.4. Phase 4: Performance Evaluation. Phase 4 evaluates the ontological framework using the benchmark metrics. Precision, Recall, F1-measure, and Accuracy are the widely used metrics to measure the performance of IR systems. The following terms are applied in the evaluation metrics to ensure the effectiveness of the outcome generated by the frameworks.

True Positive (TP): The number of correctly predicted positive documents.

True Negative (TN): The number of correctly predicted negative documents.

False Positive (FP): The number of incorrectly predicted positive documents.

False Negative (FN): The number of incorrectly predicted negative documents.

Based on the above terms, the metrics are computed as follows:

Precision is a set of retrieved documents relevant to the user query.

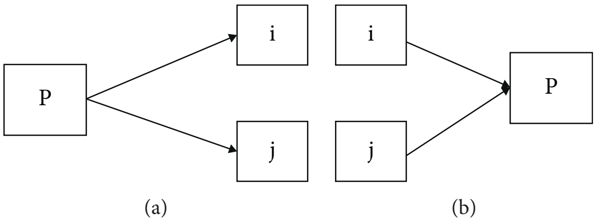


Figure 7: (a) Hub and (b) authority assignments.

$$\text{Precision} = \frac{\text{Number of relevant documents} \cap \text{Number of retrieved documents}}{\text{Number of retrieved documents}},$$
$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \tag{11}$$

It returns the number of documents divided by the number of retrieved documents. It can be computed for the topmost retrieved documents. For instance, Precision @10 indicates the top 10 retrieved documents.

The recall is a set of retrieved relevant documents. In other words, it is a number of documents divided by the number of relevant documents.

$$\text{Recall} = \frac{\text{Number of relevant documents} \cap \text{Number of retrieved documents}}{\text{Number of relevant documents}},$$
$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \tag{12}$$

$F1$–score is the harmonic mean of Precision and Recall.

$$F1 - \text{score} = 2 * \left(\frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}\right). \tag{13}$$

Accuracy is the number of retrieved documents for a user query.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}. \tag{14}$$

R–precision is used to ensure that the returned documents are relevant to a user query. It computes the recall value at R^{th} position.

Mean Average Precision (MAP) is the average precision for each user query.

MAP $= \sum_{q=1}^{n} \text{Average Precision}\,(q)/n$ where n is the number of queries (q).

4. Results and Discussion

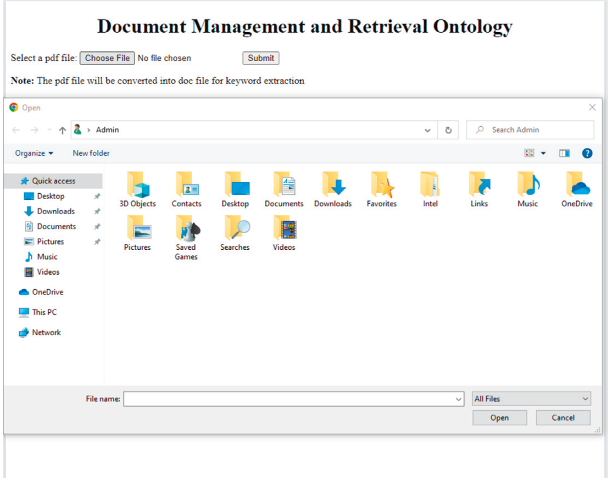
To evaluate the performance of the proposed ontological framework (POF), a testbed containing 77 documents in PDF form is developed. Python 3.9.12 in Windows 10 professional environment is utilized for implementing the frameworks. Initially, a text extractor is employed to extract the text from the PDF. Figure 8 illustrates the application interface for uploading the PDF file to convert it to a word file and extract key terms.

An Arabic dictionary is integrated with the text extractor to extract the Arabic content. MNB is used for building the ontology by classifying the documents with NER. Finally, the LBR method is applied for ranking the documents according to the user query. Table 1 outlines the Arabic and English queries for evaluating the framework's performance. It comprises the five frequently used queries by the organizations to retrieve the documents.

Figure 9 shows the list of documents for the term "salay issues." POF searches the documents and retrieves 27 documents based on the key terms. Using the hyperlink, the user can view the specific document.

Table 2 reports the findings of the performance evaluation of the POF. It outlines that the POF achieved compelling results. For instance, in Precision@77 for English Query 1, the POF offered Precision, Recall, $F1$-Score, and Accuracy of 97.3%, 97.1%, 97.2%, and 98.3. Similarly, in Precision@77 for Arabic Query, the POF presented Precision, Recall, $F1$-Score, and Accuracy of 97.7%, 98.4%, 98.05%, and 98.1%. It is evident from the outcome that the POF has produced a similar set of results for English and Arabic queries, respectively. The NER classification and link-based ranking approach have supported the POF in retrieving an optimal set of documents for user queries.

Figure 10 highlights the POF's overall performance (Precision @77) for the English and Arabic queries. The POF achieved an average $F1$-Score of 97% for five English and Arabic queries. It is noticed that the POF retrieved relevant



Document Management and Retrieval Ontology

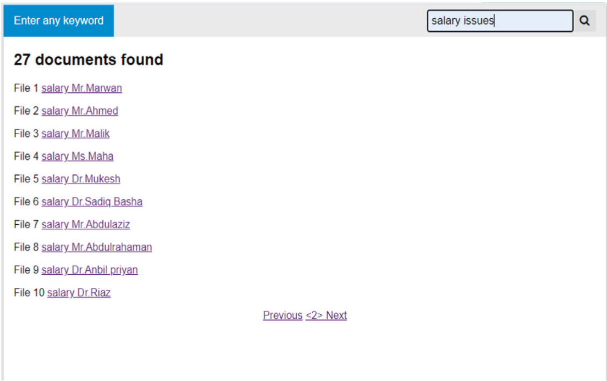
Select a pdf file: Choose File No file chosen Submit

Note: The pdf file will be converted into doc file for keyword extraction

Figure 8: Document conversion and extraction interface.

Table 1: User queries.

Queries	English	Arabic
1	What are the terms or words highly communicated by unit *A*?	ماهي الكلمة الأكثر استخداماً في الوحدة أ؟
2	What type of documents are accessed through the unit *B*?	ما نوع الوثائق التي يتم الوصول إليها من خلال الوحدة ب ؟
3	How many times unit *D* uses the term “center” in their communication?	كم مرة تستخدم الوحدة د مصطلح “مركز”في اتصالاتهم؟
4	What are the documents communicated by employee *A*?	ماهي الوثائق المرسلة من قبل الموظف أ؟
5	Who uses the word “delay” in the documents?	من يستخدم كلمة “تأخير في الوثائق”؟



Enter any keyword

salary issues

27 documents found

File 1 salary Mr.Marwan

File 2 salary Mr.Ahmed

File 3 salary Mr.Malik

File 4 salary Ms.Maha

File 5 salary Dr.Mukesh

File 6 salary Dr.Sadiq Basha

File 7 salary Mr.Abdulaziz

File 8 salary Mr.Abdulrahaman

File 9 salary Dr.Anbil priyan

File 10 salary Dr.Riaz

Previous <2> Next

Figure 9: Results window.

documents for Arabic queries. Thus, it can support Saudi organizations in extracting effective results for employees and stakeholders. Table 3 presents the findings of the comparative analysis of the ontological frameworks.

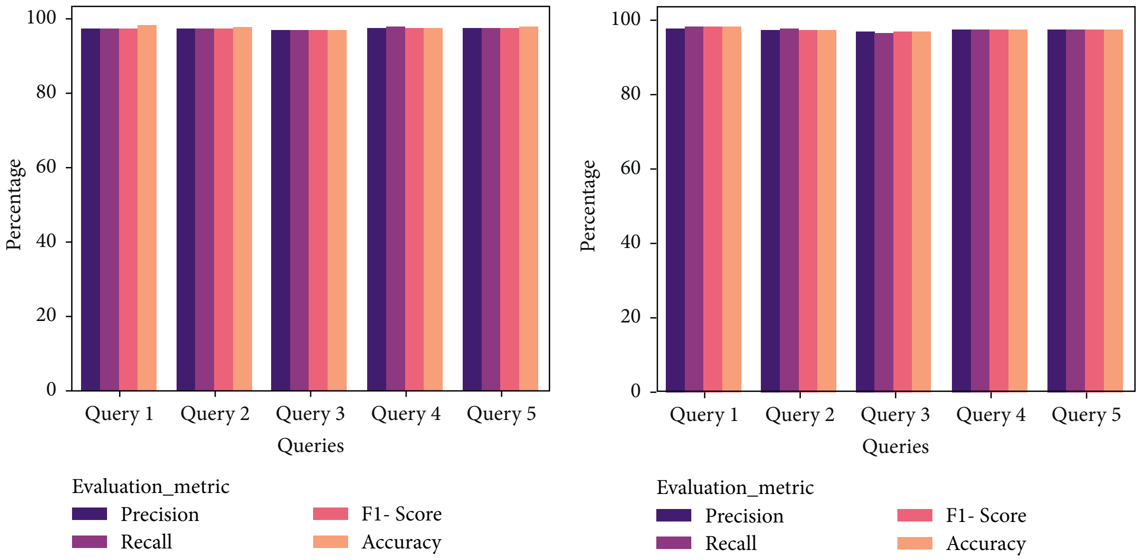
The frameworks have produced a better outcome for both English and Arabic queries, respectively. For context, the POF presented Precision, Recall, *F*1-Score, and Accuracy of 97.3%, 97.1%, 97.2%, and 98.3%, whereas the GOF and YOF have achieved Precision, Recall, *F*1-Score, and Accuracy of 97.1%, 96.4%, 96.75%, and 97.8% and 96.4%, 96.1%, 96.25%, and 96.4%. In addition, Figure 11 portrays the performance of the ontological framework for English queries, while Figure 12 presents the outcome for Arabic queries.

Figure 11 portrays the comparative analysis of the frameworks for the English queries. It represents that the POF has gained better Precision, Recall, *F*1-Score, and accuracy. Similarly, the GOF and YOF have accomplished higher Precision, Recall, *F*1-Score, and accuracy.

Likewise, Figure 12 presents the results for the Arabic queries. The frameworks have achieved a better result. However, the POF's overall performance is better than the existing frameworks. In addition to the benchmark metrics, Table 4 reveals the findings of *R*-Precision and MAP analysis. The POF outperforms both GOF and YOF, respectively. For instance, the value of *R*-Precision and MAP of the POF for Query 1 is 98.4% and 98.2, whereas GOF and YOF have offered R-Precision and MAP of 97.5% and 96.4% and 95.6%

Table 2: Performance analysis of the POF.

Queries	No of documents	English				Arabic			
		Precision	Recall	$F1$-score	Accuracy	Precision	Recall	$F1$-score	Accuracy
1	@10	98.2	97.8	98	98.2	98.1	97.6	97.85	98.3
	@30	97.4	97.6	97.5	97.6	97.5	97.4	97.45	97.6
	@50	97.5	98.3	97.9	98.1	97.6	97.7	97.65	97.9
	@77	97.3	97.1	97.2	98.3	97.7	98.4	98.05	98.1
2	@10	98.6	98.7	98.65	98.6	98.5	98.3	98.4	98.7
	@30	97.6	97.9	97.75	98.1	97.6	97.5	97.55	97.3
	@50	97.1	97.5	97.3	97.7	97.5	98.4	97.95	98.4
	@77	97.4	97.3	97.35	97.5	97.2	97.7	97.45	97.5
3	@10	98.2	98.4	98.3	98.6	98.4	98.6	98.5	98.7
	@30	97.9	97.2	97.55	97.9	97.5	97.6	97.55	97.1
	@50	97.4	97.6	97.5	97.5	97.3	97.4	97.35	97.3
	@77	96.7	96.8	96.75	96.7	96.8	96.5	96.65	96.8
4	@10	98.8	98.7	98.75	98.8	98.7	98.5	98.6	98.4
	@30	97.9	97.7	97.8	97.9	97.7	97.5	97.6	97.8
	@50	97.5	97.6	97.55	97.5	97.6	97.4	97.5	97.5
	@77	97.2	97.6	97.4	97.3	97.3	97.5	97.4	97.1
5	@10	98.6	98.8	98.7	98.6	98.5	98.6	98.55	98.7
	@30	97.7	97.5	97.6	97.7	97.6	97.3	97.45	97.5
	@50	97.1	97.3	97.2	97.6	97.3	97.6	97.45	97.7
	@77	97.3	97.4	97.35	97.5	97.2	97.4	97.3	97.1



(a)

English (b)

Figure 10: Performance analysis of POF (a) English (b) Arabic.

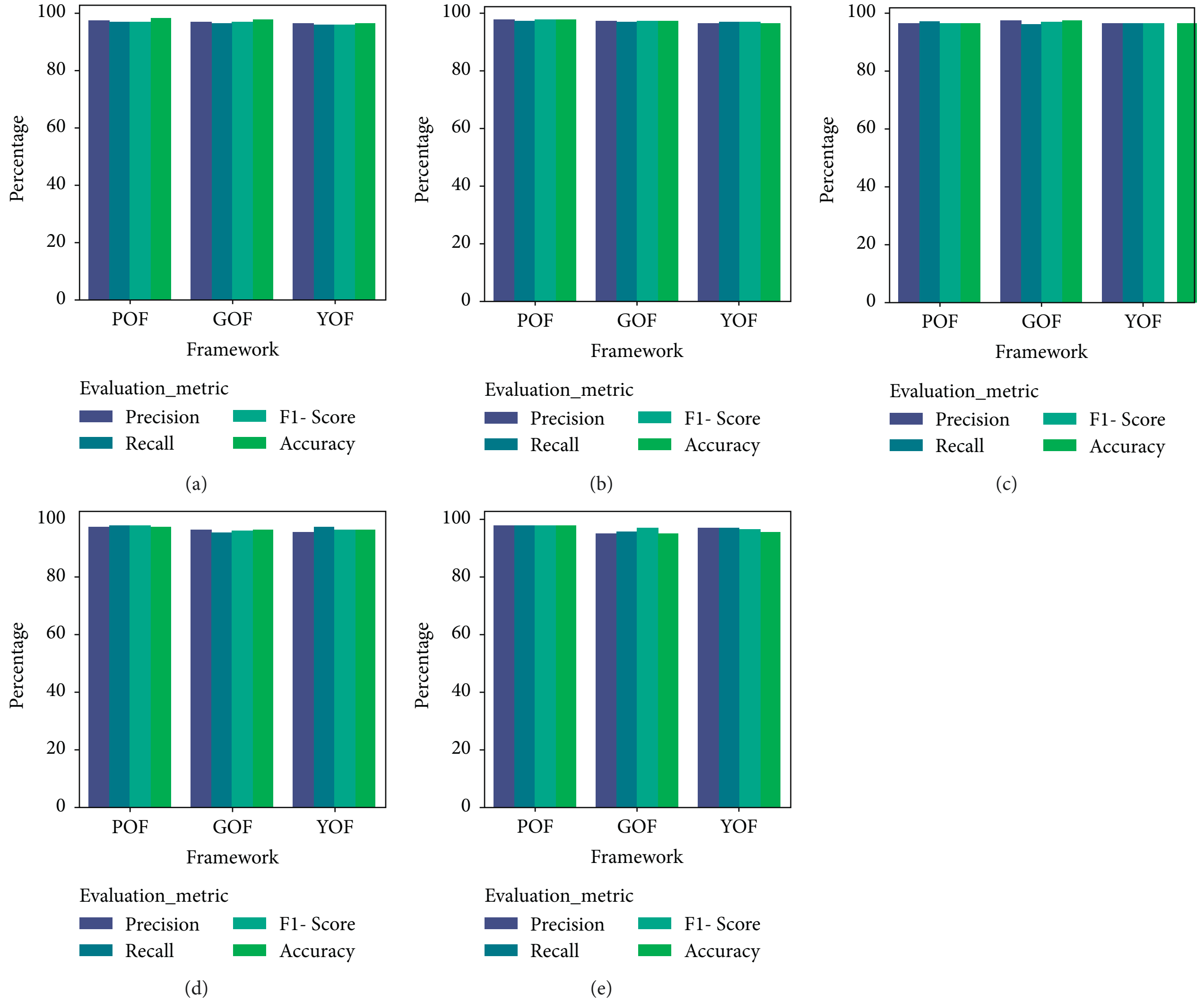
and 95.8%, respectively. The features of HITS and SALSA have favored the POF to retrieve a compelling set of documents compared to other frameworks.

Figure 13 shows that the POF offered a supreme outcome for the English and Arabic queries compared to the GOF (Figure 14) and the YOF (Figure 15). It reveals that the effectiveness of data extraction, NER classification, and ranking approach supported the proposed framework to produce better results.

POF achieves a better Precision, Recall, $F1$-score, and Accuracy for both Arabic and English languages, respectively. It can be applied in any kind of document management environment. However, GOF and YOF are the ontological frameworks for specific documents which cannot be applied for general applications. In addition, POF offers a ranking technique for searching a bilingual document rather than GOF and YOF. It is a link-based searching technique, whereas GOF and YOF rank the documents

Table 3: Comparative analysis of the frameworks.

Queries	Framework	English				Arabic			
		Precision	Recall	*F*1-score	Accuracy	Precision	Recall	*F*1-score	Accuracy
1	POF	97.3	97.1	97.2	98.3	97.7	98.4	98.05	98.1
	GOF	97.1	96.4	96.75	97.8	97.1	97.8	97.45	97.4
	YOF	96.4	96.1	96.25	96.4	96.7	96.8	96.75	97.5
2	POF	97.4	97.3	97.35	97.5	97.2	97.7	97.45	97.5
	GOF	97.2	96.8	97	97.1	97.5	96.7	97.1	98.1
	YOF	96.4	96.5	96.45	96.4	96.7	97.1	96.9	97.8
3	POF	96.7	96.8	96.75	96.7	96.8	96.5	96.65	96.8
	GOF	97.4	96.2	96.8	97.4	97.2	96.1	96.65	94.6
	YOF	96.4	96.5	96.45	96.4	96.1	95.7	95.9	95.3
4	POF	97.2	97.6	97.4	97.3	97.3	97.5	97.4	97.1
	GOF	96.2	95.3	95.75	96.4	94.7	97.1	95.88	96.4
	YOF	95.4	97.2	96.29	96.4	95.7	97.2	96.44	97.6
5	POF	97.3	97.4	97.2	97.5	97.2	97.4	98.05	97.1
	GOF	94.6	95.1	96.75	94.7	94.7	95.2	97.45	97.2
	YOF	96.5	96.5	96.25	95.2	95.1	95.3	96.75	96.8



(a)

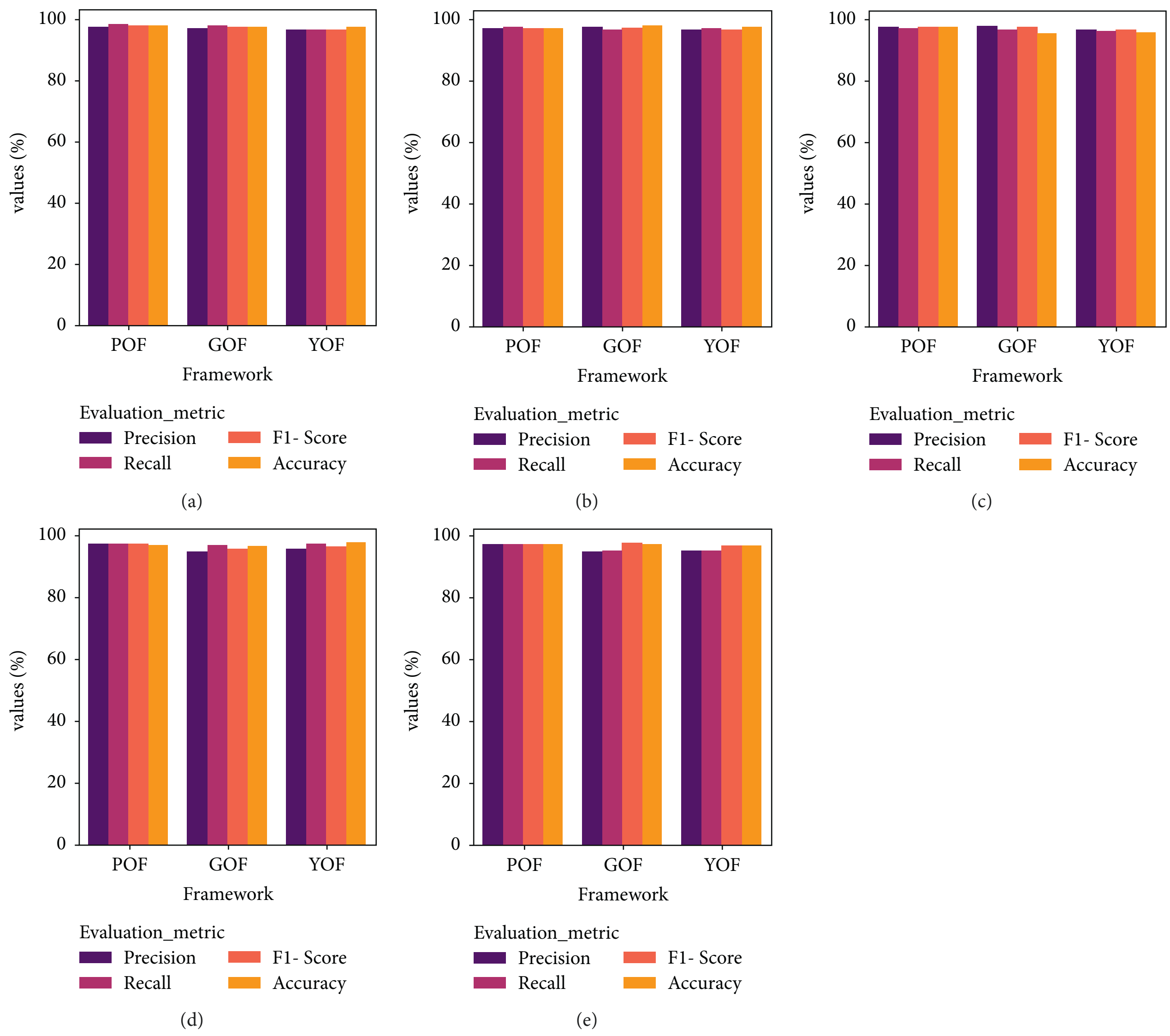
(b)

(c)

(d)

(e)

Figure 11: Comparative analysis of frameworks for English (a) query 1, (b) query 2, (c) query 3, (d) query 4, and (e) query 5.



(a)

(b)

(c)

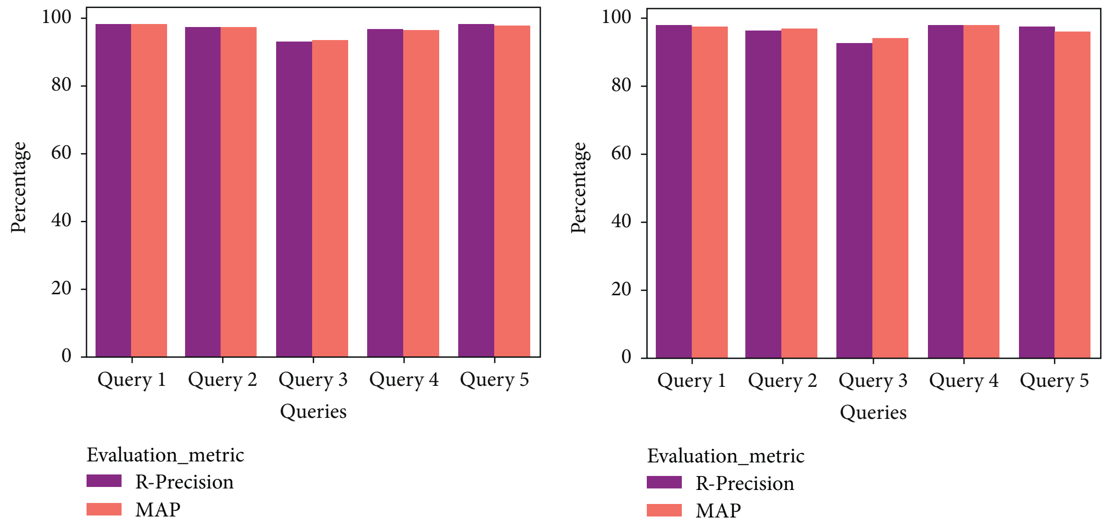
(d)

(e)

Figure 12: Comparative analysis of frameworks for Arabic (a) query 1, (b) query 2, (c) query 3, (d) query 4, and (e) query 5.

Table 4: Findings of *R*-precision and MAP analysis.

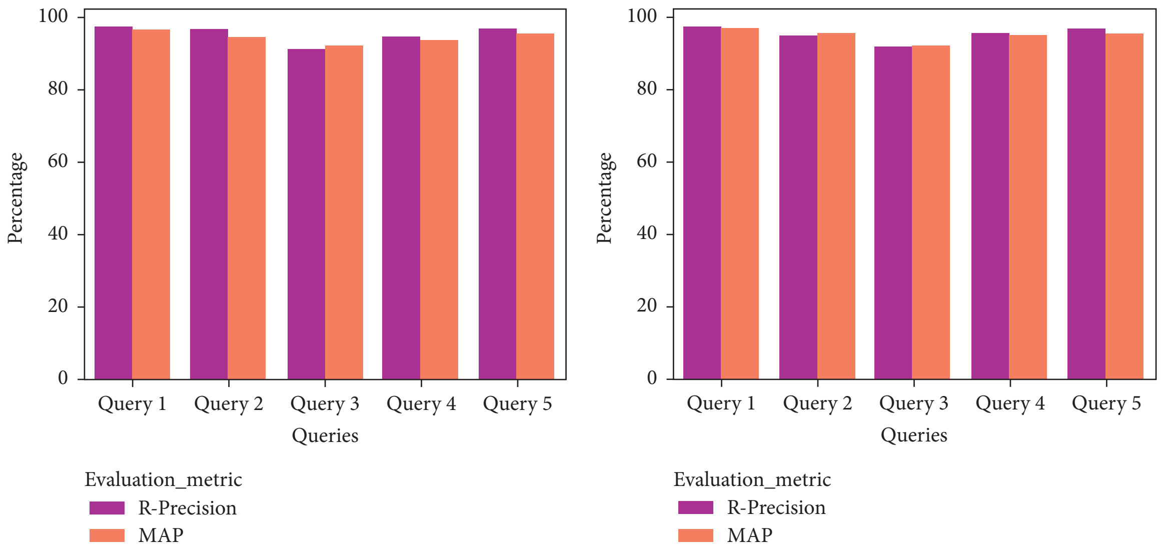
Queries	Framework	English		Arabic	
		R-precision	MAP	*R*-precision	MAP
1	POF	98.4	98.2	97.6	96.8
	GOF	97.5	96.4	97.2	96.4
	YOF	95.6	95.8	96.3	95.3
2	POF	97.5	97.2	95.9	96.3
	GOF	97.1	94.4	94.8	95.2
	YOF	96.3	95.1	96.4	95.3
3	POF	93.2	93.4	92.1	93.5
	GOF	91.2	92.4	91.5	91.7
	YOF	92.4	91.5	91.6	90.8
4	POF	96.8	96.2	97.3	97.6
	GOF	94.6	93.7	95.1	94.6
	YOF	94.3	93.8	94.6	92.5
5	POF	98.3	97.6	97.1	95.8
	GOF	96.7	95.6	96.4	95.1
	YOF	95.6	94.8	94.3	95.1



(a)

(b)

Figure 13: R-Precision and MAP analysis of POF (a) English and (b) Arabic.



(a)

(b)

Figure 14: R-Precision and MAP analysis of GOF (a) English and (b) Arabic.

according to the user query and term frequencies of the document. Thus, POF enables an effective searching environment for users compared to GOF and YOF.

4.1. Applications of the Proposed Framework.

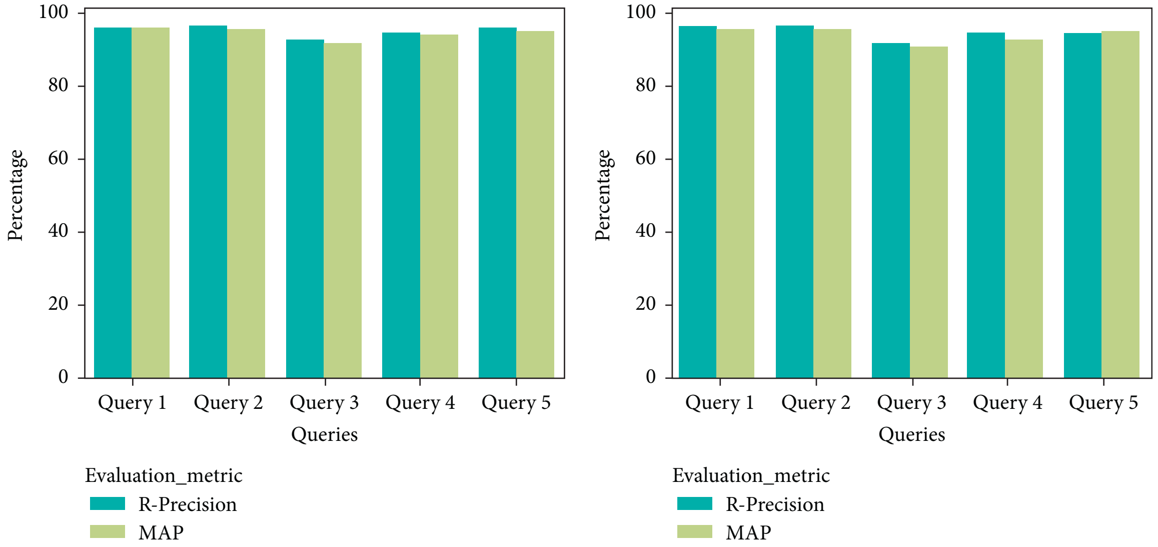
The proposed ontological framework can be applied in the real-time document management and retrieval environment. It enables an opportunity for the users to retrieve relevant documents based on the keywords. In addition, it offers the following applications for society.

Digital library: Using the proposed framework, a large corpus of documents can be developed to support the organization in facilitating a digital library for the employees to share information and manage their routine tasks.

Chatbot: The advent of AI techniques leads to the development of the question-answering system (Chatbot service) for the employees and stakeholders of an organization. The proposed framework can support the developers in training and test the Chatbot applications. The NB classifier offers the relation-based documents which the Chatbot system can use to provide relevant answers for the user queries.

Recommender system: Using phases 1 and 2, a recommender system can be developed for the employees to furnish useful data during document creation. The documents' data can be used as a keyword or metadata to search a document.

Furthermore, the bilingual feature of the proposed ontology supports Arabic and English-speaking users to share



(a)

(b)

Figure 15: R-Precision and MAP analysis of YOF (a) English and (b) Arabic.

information effectively. It assists the user in overcoming the communication barrier and completing their routine tasks without difficulties.

5. Conclusion

This study developed an ontological framework for managing Arabic and English documents in Saudi Arabian organizations. The proposed framework comprises three phases for converting the PDF documents into ordinary word documents with a set of unique terms; a Naïve Bayes-based entity-relationship document classifier and a ranking technique for arranging documents as per the user query. The conversion technique uses a modified text extractor for extracting Arabic and English terms from the images. Furthermore, the entity-relationship technique arranges the document as per the relationship among the terms of the documents. The ranking technique combines the features of the HITS and SALSA ranking algorithm to rank the documents at a faster rate. A set of 77 documents were utilized to compare the performance of the proposed frameworks with the recent techniques. The outcome reveals that the proposed ontological framework achieves adequate Precision, Recall, *F*1-score, and Accuracy for the bilingual documents using a user query. In addition, it offers an effective bilingual document management environment for employees and stakeholders of Saudi Arabian organizations. The proposed framework can be extended to other languages. Furthermore, the ranking technique can be improved using metadata with the newer deep learning techniques.

Data Availability

The data supporting the results can be available on request to the corresponding author.

Conflicts of Interest

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

Acknowledgments

This work was supported through the Annual Funding Track by the Deanship of Scientific Research, Vice Presidency for Graduate Studies and Scientific Research, King Faisal University, Saudi Arabia (Project no. AN000669).

Supplementary Materials

The file contains the code for implementing the ontology. After implementing the ontology, we have to train and test with the dataset. The necessary code is mentioned in the article and included as an attachment. (*Supplementary Materials*)

References

[1] G. Zaman, H. Mahdin, K. Hussain, J. Abawajy, J. Abawajy, and S. A. Mostafa, “An ontological framework for information extraction from diverse scientific sources,” *IEEE Access*, vol. 9, pp. 42111–42124, 2021.

[2] Y. Yao, Z. Wang, L. Li et al., “An ontology-based artificial intelligence model for medicine side-effect prediction: taking traditional Chinese medicine as an example,” *Computational and Mathematical Methods in Medicine*, vol. 2019, Article ID 8617503, 7 pages, 2019.

[3] R. Crimp and A. Trotman, “Automatic term reweighting for query expansion,” in *Proceedings of the 22nd Australasian Document Computing Symposium*, pp. 1–4, New York, NY, USA, December 2017.

[4] M. A. Raza, R. Mokhtar, N. Ahmad, M. Pasha, and U. Pasha, “A taxonomy and survey of semantic approaches for query expansion,” *IEEE Access*, vol. 7, pp. 17823–17833, 2019.

[5] S. Jain, K. R. Seeja, and R.. Jindal, "A fuzzy ontology framework in information retrieval using semantic query expansion," *International Journal of Information management Data Insights*, vol. 1, no. 1, Article ID 100009, 2021.

[6] C. Yunzhi, L. Huijuan, L. Shapiro, R. S. Travillian, and L. Lanjuan, "An approach to semantic query expansion system based on Hepatitis ontology," *Journal of Biological Research-Thessaloniki*, vol. 23, no. 1, pp. 11–22, 2016.

[7] N. Aggarwal and B. Paul, "Query expansion using Wikipedia and dbpedia," in *Proceedings of the CLEF (Online Working Notes/Labs/Workshop)*, Rome, Italy, September 2012.

[8] M. A. Zingla, C. Latiri, P. Mulhem, C. Berrut, Y. Slimani, and Y. Slimani, "Hybrid query expansion model for text and microblog information retrieval," *Information Retrieval Journal*, vol. 21, no. 4, pp. 337–367, 2018.

[9] O. El Midaoui, B. El Ghali, A. El Qadi, and M. D. Rahmani, "Geographical query reformulation using a geographical taxonomy and WordNet," *Procedia Computer Science*, vol. 127, pp. 489–498, 2018.

[10] N. S. Selvan, S. Vairavasundaram, and L. Ravi, "Fuzzy ontology-based personalized recommendation for internet of medical things with linked open data," *Journal of Intelligent and Fuzzy Systems*, vol. 36, no. 5, pp. 4065–4075, 2019.

[11] S. T. R. Rizvi, D. Mercier, S. Agne, S. Erkel, A. Dengel, and S. Ahmed, "Ontology-based information extraction from technical documents," in *Proceedings of the 10th International Conference on Agents and Artificial Intelligence*, no. 2, pp. 493–500, Madeira, Portugal, January 2018.

[12] M. Dragoni, S. Poria, and E. Cambria, "OntoSenticNet: a commonsense ontology for sentiment analysis," *IEEE Intelligent Systems*, vol. 33, no. 3, pp. 77–85, 2018.

[13] F. Wu, Y. Ji, and W. Shi, "Design of a computer-based legal information retrieval system," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 6942773, 10 pages, 2022.

[14] Y. Maatouk, *Building AIPedia ontology to evaluate research impact in artificial intelligence area*, p. 2, Academia Letters, San Francisco, California, 2021.

[15] Y.-H. Lee, P. J. H. Hu, W.-J. Tsao, and L. Li, "Use of a domain-specific ontology to support automated document categorization at the concept level: method development and evaluation," *Expert Systems with Applications*, vol. 174, Article ID 114681, 2021.

[16] P. Pham, P. Do, and C. D. Ta, "Automatic topic labelling for text document using ontology of graph-based concepts and dependency graph," *International Journal of Business Information Systems*, vol. 36, no. 2, pp. 221–253, 2021.

[17] V. Adithya and G. Deepak, "OntoReq: an ontology focused collective knowledge approach for requirement traceability modelling," in *Middle Eastern, North African Conference on Management & Information Systems*, A. European, Ed., Springer, Cham, pp. 358–370, 2021.

[18] E. Manziuk, I. Krak, O. Barmak, O. Mazurets, V. Kuznetsov, and O. Pylypiak, "Structural alignment method of conceptual categories of ontology and formalized domain," in *Proceedings of the CEUR Workshop Proceedings, International Workshop of IT-professionals on Artificial Intelligence (ProfIT AI 2021)*, vol. 3003, Kharkiv, Ukraine, November 2021.

[19] S. Goel, R. Kumar, M. Kumar, and V. Chopra, "An efficient page ranking approach based on vector norms using sNorm (p) algorithm," *Information Processing & Management*, vol. 56, no. 3, pp. 1053–1066, 2019.

[20] W. Xing and A. Ghorbani, "Weighted pagerank algorithm," in *Proceedings of the Second Annual Conference on Communication Networks and Services Research 2004*, pp. 305–314, IEEE, Fredericton, NB, Canada, May 2004.

[21] H. Deng, M. R. Lyu, and I. King, "A generalized co-hits algorithm and its application to bipartite graphs," in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Paris, France, January 2009.

[22] A. Farahat, T. LoFaro, J. C. Miller, G. Rae, and L. A. Ward, "Authority rankings from HITS, PageRank, and SALSA: existence, uniqueness, and effect of initialization," *SIAM Journal on Scientific Computing*, vol. 27, no. 4, pp. 1181–1201, 2006.

[23] J. F. Banu, P. Muneeshwari, K. Raja, S. Suresh, T. P. Latchoumi, and S. Deepan, "Ontology based image retrieval by utilizing model annotations and content," in *Proceedings of the 2022 12th International Conference on Cloud Computing, Data Science & Engineering (Confluence)*, pp. 300–305, IEEE, Noida, India, January 2022.

[24] Yu. Chen, "English translation template retrieval based on semantic distance ontology knowledge recognition algorithm," *Mathematical Problems in Engineering*, vol. 2022, Article ID 2306321, 11 pages, 2022.

[25] M. Hu, "Research on semantic information retrieval based on improved fish swarm algorithm," *Journal of Web Engineering*, vol. 21, no. 3, pp. 845–860, 2022.

[26] L. Yu, L. Hua, and J. Ding, "Research on the development support strategy of cultural enterprises based on fish swarm algorithm under the background of public health," *Journal of Environmental and Public Health*, vol. 2022, Article ID 6470147, 9 pages, 2022.

[27] Q. Ye, "Situational English language information intelligent retrieval algorithm based on wireless sensor network," *International Journal of Wireless Information Networks*, vol. 28, no. 3, pp. 287–296, 2021.

[28] T. Qiu, P. Xie, X. Xia, C. Zong, and X. Song, "Aggregated boolean query processing for document retrieval in edge computing," *Electronics*, vol. 11, no. 12, 1908.

[29] E. Novak, L. Bizjak, D. Mladenić, and M. Grobelnik, "Why is a document relevant? Understanding the relevance scores in cross-lingual document retrieval," *Knowledge-Based Systems*, vol. 244, Article ID 108545, 2022.

[30] U. D. Dixit, M. S. Shirdhonkar, and G. R. Sinha, "Automatic logo detection from document image using HOG features," *Multimedia Tools and Applications*, vol. 6, pp. 1–16, 2022.

[31] J. Mackenzie, M. Petri, and A. Moffat, "Efficient query processing techniques for next-page retrieval," *Information Retrieval Journal*, vol. 25, no. 1, pp. 27–43, 2022.

[32] M. Yuan, J. Zobel, and P. Lin, "Measurement of clustering effectiveness for document collections," *Information Retrieval Journal*, vol. 1, pp. 1–30, 2022.

[33] K. Alsubhi, A. Jamal, and A. Alhothali, "Deep learning-based approach for Arabic open domain question answering," *PeerJ Computer Science*, vol. 8, p. e952, 2022.

[34] A. Y. Muaad, H. J. Davanagere, D. Guru et al., "Arabic document classification: performance investigation of preprocessing and representation techniques," *Mathematical Problems in Engineering*, vol. 2022, Article ID 3720358, 16 pages, 2022.

[35] K. Keyvan and J. X. Huang, "How to approach ambiguous queries in conversational search? A survey of techniques, approaches, tools and challenges," *ACM Computing Surveys (CSUR)*, ACM, New York, NY, USA, 2022.

[36] N. Ul A. Ali, W. Iqbal, and H. Afzal, "Carving of the OOXML document from volatile memory using unsupervised learning

techniques," *Journal of Information Security and Applications*, vol. 65, Article ID 103096, 2022.

[37] T. Werner, "A review on instance ranking problems in statistical learning," *Machine Learning*, vol. 111, no. 2, pp. 415–463, 2022.

[38] B. Ofoghi, M. Mahdiloo, and J. Yearwood, "Data Envelopment Analysis of linguistic features and passage relevance for open-domain Question Answering," *Knowledge-Based Systems*, vol. 244, Article ID 108574, 2022.

[39] D. Yadav, N. Lalit, R. Kaushik et al., "Qualitative analysis of text summarization techniques and its applications in health domain," *Computational Intelligence and Neuroscience*, vol. 2022, Article ID 3411881, 14 pages, 2022.

[40] P. S. Sharma, D. Yadav, and R. N. Thakur, "Web page ranking using web mining techniques: a comprehensive survey," *Mobile Information Systems*, vol. 2022, Article ID 7519573, 19 pages, 2022.

[41] S. J. De Sousa, T. M. R. Dias, A. L. Pinto, and A. L. Pinto, "A strategy for identifying specialists in scientific data repositories," *Mobile Networks and Applications*, vol. 3, pp. 1–11, 2022.

[42] S. V. Thambi and P. C. ReghuRaj, "Graph based document model and its application in keyphrase extraction,"vol. 1, pp. 92–98, in *Proceedings of the 2022 IEEE International Conference on Signal Processing, Informatics, Communication and Energy Systems (SPICES)*, vol. 1, IEEE, Thiruvananthapuram, India, March 2022.

[43] G. McDonald, C. Macdonald, and I. Ounis, "Search results diversification for effective fair ranking in academic search," *Information Retrieval Journal*, vol. 25, no. 1, pp. 1–26, 2022.

[44] R. Srivastava, P. Singh, K. Rana, and V. Kumar, "A topic modeled unsupervised approach to single document extractive text summarization," *Knowledge-Based Systems*, vol. 246, Article ID 108636, 2022.

[45] A. S. Alqahtani, P. Saravanan, M. Maheswari, and S. Alshmrany, "An automatic query expansion based on hybrid CMO-COOT algorithm for optimized information retrieval," *The Journal of Supercomputing*, vol. 78, no. 6, pp. 8625–8643, 2022.

[46] A. Chugh, V. K. Sharma, S. Kumar et al., "Spider monkey crow optimization algorithm with deep learning for sentiment classification and information retrieval," *IEEE Access*, vol. 9, pp. 24249–24262, 2021.

[47] Y. Djenouri, A. Belhadi, D. Djenouri, and J. C.-W. Lin, "Cluster-based information retrieval using pattern mining," *Applied Intelligence*, vol. 51, no. 4, pp. 1888–1903, 2021.

[48] G. Amudha, "Dilated transaction access and retrieval: improving the information retrieval of blockchain-assimilated internet of things transactions," *Wireless Personal Communications*, vol. 2, pp. 1–21, 2021.

[49] H. Abdirad and P. Mathur, "Artificial intelligence for BIM content management and delivery: case study of association rule mining for construction detailing," *Advanced Engineering Informatics*, vol. 50, Article ID 101414, 2021.

[50] X. Wang, C. Macdonald, N. Tonellotto, and I. Ounis, "Pseudo-relevance feedback for multiple representation dense retrieval," in *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval*, pp. 297–306, New York, NY, USA, July 2021.

[51] K. Thirumoorthy and K. Muneeswaran, "An elitism based self-adaptive multi-population Poor and Rich optimization algorithm for grouping similar documents," *Journal of Ambient Intelligence and Humanized Computing*, vol. 13, no. 4, pp. 1925–1939, 2022.

[52] N. Darapaneni, G. Singh, A. Reddy Paduri et al., "Customer support Chatbot for electronic components," in *Proceedings of the 2022 Interdisciplinary Research in Technology and Management (IRTM)*, pp. 1–7, IEEE, Kolkata, India, 2022.

[53] A. Tandon, S. K. Guha, J. Rashid et al., "Graph based CNN algorithm to detect spammer activity over social media," *IETE Journal of Research*, vol. 2, pp. 1–11, 2022.

[54] A. M. Ikotun, M. S. Almutari, and A. E. Ezugwu, "K-Means-Based nature-inspired metaheuristic algorithms for automatic data clustering problems: recent advances and future directions," *Applied Sciences*, vol. 11, no. 23, Article ID 11246, 2021.

[55] Text extractor: https://github.com/tesseract-ocr/tesseract available online.

[56] Name Entity relationship classifier: https://gist.github.com/arthurratz/%207a63d3938d0%2059907352a85%20c791aa5290 available online.