Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.18576/amis/180514

Enhanced Cultural Algorithm for Information Retrieval System

*Doaa N. Mhawi[1], Haider W. Oleiwi[2], and Ammar Aldallal[3,]**

[1] Technical Institute for Management, Middle Technical University, Baghdad, Iraq
[2] Department of Electronic and Electrical Engineering, Brunel University London, UK
[3] Department of Telecommunication Engineering, Ahlia University, Bahrain

Received: 6 Mar. 2024, Revised: 8 Apr. 2024, Accepted: 29 May 2024.
Published online: 1 Sep. 2024.

Abstract: The number of web pages is rapidly increasing worldwide, making the information retrieval system (IRS) a crucial technology for search engines. Despite the enormous efforts made by researchers to improve the performance of IRSs, internet users still need to work on two main influential issues: imprecise content retrieved in response to their queries and accordingly resulting in irrelevant document retrieval. Thus, a high storage space was occupied while non-reasonable retrieval time was realized. In response to those challenges, this paper proposed a modified culture algorithm (MCA) allocated as an indexing method. The proposed system aimed to establish a way for indexing to retrieve specific documents rapidly, retrieve the relevant document for user queries/satisfaction, and reduce storage space. It applied the benchmark WebKb dataset with 8282 web pages that were semi-structured documents. According to the experimental findings, recall and precision metrics reached 99% for 40 test queries, whereas the storage size for document indexing occupied only 18 Megabytes. The relevant retrieved document was kept in a look-up table for quick access. Comparatively, the proposed system outperformed the state-of-the-art.

Keywords: Culture algorithm; evolutionary algorithm; information retrieval; indexing method; web pages.

1. Introduction

The exponential increase in data in recent years has posed new challenges for academia and businesses in developing innovative methods for extracting essential information in a shorter time. The concept of information retrieval is a daily experience for most people who use various IRSs as end users. The primary step in IR is to search through extensive data collection to find relevant results that meet user needs and provide information in response to a user query. The repository often contains various data types, including structured, unstructured, semi-structured, and heterogeneous data. Various strategies are used to extract crucial data from the repository. In the digital world, the vast information content in a global repository of knowledge and culture has facilitated the direct sharing of ideas and information at a high rate. Therefore, it is essential to extract data from this world as documents. The transformed documents are helpful for information retrieval by sharing information with all users. It is an automated process that generates a list of documents with relevance-ranked results in response to a query [1,2]. Using textual datasets to expedite the completion of essential activities has become a common practice among individuals and companies. Most datasets contain a large number of documents from various sources, such as scientific papers/articles, news items, electronic libraries, books, text messages, emails, and web pages. Therefore, finding information from databases that best satisfies the user is a significant challenge [3, 4, 5]. Effective data update and query activities require well-organized data. Therefore, indexing strategies are being used in addition to several indexing systems researching the targeted content [6, 7, 8, 9]. The performance and security of the datasets must be enhanced using indexing techniques. The information retrieval system (IRS) faces many challenges, such as the large indexing size, the inability to aggregate search output, and potentially significant security issues.

The web user still runs into two major issues with IRSs when retrieving documents concerning document retrieval relevancy to user queries. The first issue is multiple highly ranked retrieved documents irrelevant to the user query. Identifying documents relevant to the user's needs is one of the major difficulties in information retrieval. Technically, it is a ranking problem that requires a solution according to the relative documents'/user query's relevance after information retrieval [10, 11, 12]. The second issue represents the retrieval failure for many relevant documents in the dataset [13, 14, 15, 16, 17].

According to the challenges above, this paper proposes an indexing technique by modifying the evolutionary algorithm (i.e., culture algorithm (CA)) to retrieve relevant documents for the user query with high recall and precision. Furthermore, the proposed system is intended to reduce storage space and minimize retrieval time (complexity time).

The rest of this paper is structured as follows: section 2 includes the theoretical background of IRS, CA and an overview of the dataset. Section 3 explains summaries of similar attempts and some evolutionary algorithms that

[*] Corresponding author e-mail: aaldallal@ahlia.eu.bh

used the same dataset. Section 4 details explanations of methodology, while section 5 illustrates experimental results and discussions. Section 6 addresses a brief list of limitations for recommendation. Ultimately, section 7 indicates a conclusion and future work.

2. Theoretical Background

This section explains the main concepts used in this paper: the idea of an IRS and the CA concept.

2.1. The Concept of IRS

IRSs are designed to search and retrieve information or documents that the user community needs. They should provide the correct information to the user. Consequently, the IRS's goal is to gather and arrange information on any subject area to make it available to users as soon as possible, whenever they need it. IRS reveals the location and existence of the document in which the information would have been found and does not return it. An ideal IRS does not include irrelevant documents and only contains documents relevant to the user's query [18, 19, 20, 21, 22]. As shown in Figure 1, the IRS's general structure [23] consists of multiple stages. It starts by indexing the preprocessed documents. In the next stage, a query is received at the retrieval stage. In this stage, the IRS applies a specific algorithm to fetch the related documents. Next, these fetched documents are ranked based on the degree of relevance to the user query.

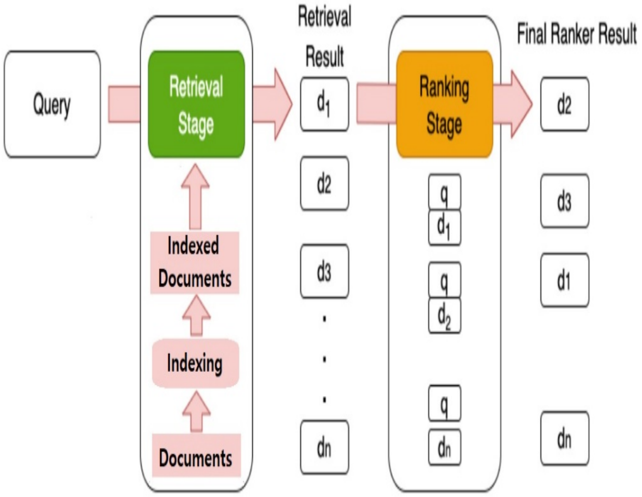


Fig. 1: Main steps and operations of the IRS

On the one hand, the retrieval stage fetches documents pertinent to the query. On the other hand, the ranking stage is responsible for re-ranking the documents according to a score of relevance. The retrieval objective is to locate pertinent documents in the collection that match the user's query. To do this, a variety of techniques including the Vector Space Model [23], Boolean Model [25, 26], Probabilistic Retrieval [26, 27, 28, 29, 30, 31], Latent Semantic Indexing [33], Language-based [34], and Genetic Algorithms (GAs) [35, 36, 37] are employed.

The primary goal of the second stage is to change the order in which the documents were initially obtained depending on the relevance score. The ranking process models are quite different from those employed for the retrieval stage because ranking focuses on increasing the efficacy of the findings rather than their efficiency.

Without the user's direct engagement, indexing occurs in the background and tackles the document's representation. The document is entirely or partially stored as part of the indexing process. A query is a typical term used to describe the user's information request. A query is initially composed of keywords, then the documents that match those keywords are looked up. The matching procedure involves further matching the query representation with the document representation kept in the index file.

2.2. Cultural Algorithm

The implicit processes used by conventional evolutionary computation techniques make it difficult to store, represent, and transmit the knowledge of one generation to the next. An evolutionary system with dual inheritance called the CA [38, 39] can offer explicit procedures for knowledge acquisition, storage, and refinement. A CA is an evolutionary system that relies on knowledge. It simulates two layers of evolution: the belief space level and the population space level. Evolutionary Programming, GAs, and Genetic Programming are evolutionary population models that can be used in the population space. With a population space, CA also has a belief space where the knowledge for solving problems can be gathered, debated, and improved. The belief space may describe the problem-solving knowledge using any symbolic form.

Within societies, cultures change along with time; however, it is considered the standard to interpret and document the individuals' behaviors of a particular society. Culture algorithms (CAs) were proposed for modeling cultural component development as they pick up knowledge and learn new things. CAs are an extension of GAs, whereas belief space acts as a conduit of knowledge for each generation under evolution. To this end, CAs are utilized to self-adapt evolutionary systems for various applications. CA was developed for evolution computations; it was used to re-engineer the commercial rule-based expert systems and the knowledge-discovery systems using decision trees and a top-to-down approach to reduce network complexity and improve performance [40, 41, 42].

2.3. Overview of the WebKb Dataset

The WebKb dataset consists of WWW (Worldwide Webpages) collected precisely from several universities with computer science departments (e.g., Texas, Washington, Cornell, Mis., and Wisconsin). It includes 8282 HTML-programmed semi-structured documents, as shown in Table 1. Each of its directories has five classes, as shown in Table 2.

Table 1: Names of directories in the WebKb dataset.

Name of each Directory	Number of Documents
Students	930
Faculty	181
Courses	3763
Projects	1124
Others	137
Staffs	504
Departments	1641
Total: 8282	

Table 2: University names with the number of documents.

University-name	Documents-number
Cornell:	867
Texas:	4120
Mis.:	1205
Washington:	1263
Wisconsin:	827

3. Related Works

This section elaborates on related work bi-directionally. It discusses different indexing methods to improve IRS performance and focuses on evolutionary algorithms to enhance retrieval outcomes.

Researchers have developed several techniques for IRSs, including document indexing and query expansion, aiming at achieving fast retrieval with high recall and precision. Gu et al [43] proposed a semantic indexing framework based on the multi-probe attention neural network, in which a KNN-derived MeSH masking mechanism was applied to generate a handful of candidate MeSH terms for each input article. This technique was used for the COVID-19 semantic indexing problem and proved effective.

Another IRS that considered hyphenated word fragments in untranscribed text images was proposed by [44]. They developed a probabilistic model that merged probabilistic indexing with optical prediction of hyphenated word fragments. The model further estimated a probabilistic model using machine learning (ML). Additionally, it offered the benefit of enabling appropriate trade-offs between storage utilization and information retrieval efficiency through the strategic selection of a probability threshold. This work has not stated explicitly how ML affects the improvement of obtained results.

Gil-Leiva et al [45] assessed the performance of three automatic indexing systems, namely SISA (Automatic Indexing System), KEA (Keyphrase Extraction Algorithm), and MAUI (Multi-Purpose Automatic Topic Indexing), in comparison to human indexing. SISA utilized an algorithm that relies on rules regarding the placement of terms within the document's various structural components. On the other hand, KEA and MAUI employed ML and statistical features of terms in their algorithms. Results showed that SISA is three times better than the other two models using F-measure, indicating that it was more similar to the human indexing.

Rats and Pede [46] proposed an ML model designed to automate the indexing and routing of incoming documents within an enterprise by classifying them into topics using a binary classification-based model. The architecture involved training classification bots for each major topic, which were then executed on incoming documents to predict their topics based on the training. Various ML techniques were examined, including supervised learning, text embedding methods (Bag of Words), and advanced embedding techniques, e.g., BERT and GPT. The results showed increased efficiency in automating document handling and minimized errors associated with manual processing.

The paper by Briciu et al [47] presented a comprehensive examination of ML and deep learning approaches for sentiment analysis across multiple levels (document, sentence, and aspect) using Romanian language reviews. It investigated the performance of five ML algorithms, which were logistic regression (LR), decision trees (DT), k-nearest neighbors (kNN), support vector machines (SVM), and naïve Bayes (NB). It introduced a balanced dataset from twelve product categories, demonstrating significant enhancements in sentiment analysis techniques. This work was based on latent semantic indexing and achieved an F1-score of 77% using the Romanian language.

Wang et al [48] introduced KenMeSH, an end-to-end model for MeSH indexing of biomedical articles, addressing the challenge of assigning multiple labels from an extensive and hierarchically organized collection of MeSH terms. The proposed model enhanced indexing by utilizing Graph Convolutional Neural Networks to integrate document features with MeSH label hierarchy and journal correlation features. The model achieved high performance across several measures.

Liu et al [49] presented an ontology-based method for categorizing clinical studies by conditions using structured vocabularies. This method leveraged the Observational Medical Outcomes Partnership Common Data Model and SNOMED CT for indexing and categorization, aiming to improve the accuracy and coverage of study categorization compared to traditional methods. The methodology achieved a high categorization accuracy (95.7%), demonstrating its effectiveness over the conventional MeSH-based method.

Kumar and Sharma [50] proposed an optimized query expansion strategy for improving semantic information retrieval using a novel hybrid algorithm combining spatial bound whale optimization and binary moth flame optimization. This approach addressed the insufficiencies of traditional keyword-based IRSs by focusing on semantic enrichment through query expansion. The approach utilized advanced algorithms to refine indexing processes, enabling more accurate and efficient retrieval of information based on semantic relevance rather than mere keyword matching. Their approach achieved a maximum F-score of 95.65%.

An algorithm was suggested by [51] for retrieving relevant information by incorporating user preference profiles and query expansion. That approach aimed to provide personalized search results. The effectiveness of the information retrieval process was evaluated using precision, recall, and mean average precision metrics. The maximum precision at ten documents (P@10) achieved using that technique was 86%. However, no accuracy was provided for the retrieved documents.

Xu et al [52] found that increasing the number of components of the IRS decreases its efficiency for fusion-based systems in terms of search performance. They proposed a clustering-based approach that used the Chameleon clustering algorithm. It was applied to divide candidates into clusters. Then, Sequential Forward Selection for fusion was used to select a representative for each cluster. That approach was examined using the TREC document set for medical information retrieval, and its efficiency was proved in terms of mean average precision and recall-level precision. Applying the TREC 2018 dataset, that work was evaluated using several metrics, i.e., Mean Average Precision over a group of queries, Recall-level Precision, P@10 level, and Mean Reciprocal Rank. The best P@10 was 71.66%, MRP was 89.14%, and PR was 43.76%.

Evolutionary algorithms have made significant progress by developing strategies to solve the challenges of delivering more precise, relevant documents to a user's query. In terms of retrieval outcomes, they outperformed the conventional IRS. Researchers used a variety of strategies to enhance retrieval outcomes and storage space. Nevertheless, current methods leave room for improvement, and a gap remains in the field. The following paragraph outlines the most related studies showing the current limitations of IRSs and the gap to be filled by this research.

Karim et al [6] used a method for indexing the dataset to reduce the memory size, achieving a reduction of up to 48 MB, but with an accuracy that did not exceed 90%. This work required a high processing time, and the two general issues of the IRS still needed to be solved. Karim et al [7] employed a new method for document indexing and preprocessing using a genetic algorithm (GA) as a search algorithm to decrease the storage size further. The memory size reached 40 MB, and the experimental results reached 93% for recall and 98% for precision. Although storage reduction was achieved by this work, the long time it took to retrieve the document and the specific documents not being retrieved for the user were the main limitations.

The modified GA with a method for indexing web page documents was presented by [8]. The proposed technique provided a storage reduction of up to 19.9 MB with an accuracy of 98%. The modified GA required a high complexity time, and some related documents were not retrieved by the user. Doaa et al [26] developed means of retrieval by using a hybrid of CA with GA as a searching algorithm and a document indexing method for documents' storage and arrangement. The experimental results showed that the researchers used a 75-population size with 15 iterations, a precision of 100%, and a recall of 98%. This work faced issues with some retrieved documents with high ranks not being a precise response to user queries.

The researchers in [53] used Sparse fitness with 100 population sizes and 10 clusters with interactive GAs based on a support vector machine. Experimental results reached 94% for recall, whereas the execution time was many hours. This work encountered the classifier's constraints, using the initial user’s selected examples. This method decreased while using a 250-population size and required a high execution time. Different methods of paired comparison [54] (PC- interactive GA) were used with 7000 generations, and different values of fitness function (FF) were applied. The main problem affecting this method's retrieval results was allowing the user to contrast two individuals and choose the best.

Interactive GAs based on Fuzzy logic were proposed in [55]. A surrogate model was built to evaluate individual fitness to enhance IEC or change user evaluation using ranking distances of either 10 or 20/FF values. It eliminated users' burdens using a fuzzy number for fitness allocation. Liaw et al [56] modified evolutionary operators with an adaptive learning evaluation to assess beauty in the evolutionary art system. The population size was 100, whereas the mutation rate was 0.5. Some researchers used an evolutionary computational approach that employed culture as a drive to store accessible, relevant information for the population over multiple generations [57]. The culture was considered an evolving data source influencing the patterns of behavior practiced by various populations. Hence, those methods required large memory sizes and long execution times. Real-world problems were the idea behind using semantic networks to represent the multiple relationships of those problems [30].

The main challenges of IR are ongoing. Therefore, current methods still need improvement in terms of the relevant document, required storage, and processing complexity. Based on the authors’ knowledge, no work has investigated the feasibility of applying the CA in information retrieval to tackle the abovementioned limitations.

4. Information Retrieval System Using MCA

Figure 2 demonstrates the general structure of the proposed system. It consists of four stages; the first stage is a preprocessing and document indexing method. It is carried out in a series of stages. The first stage is used to clean and prepare data in the best way (i.e., tokenization, stemming, stop word removal, and weighting each word according to tag weight). The second stage represents the queries of all users. The third stage is the evolutionary algorithm, and the final stage retrieves the related document and evaluation measurements (i.e., accuracy, recall, precision, and complexity time). All these stages are explained in detail in the following subsections.

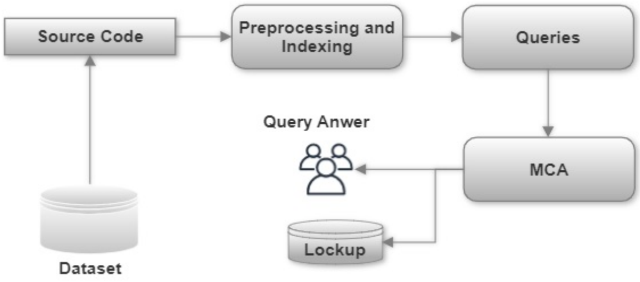


Fig. 2: A general framework of the proposed system

4.1. Stage 1: Preprocessing and Indexing Document Method

This stage consists of many steps to preprocess the dataset (i.e., WebKb) and index the document sufficiently using a novel method to reduce storage space. Algorithm 1 demonstrates the steps of this stage.

Algorithm 1 Preprocessing and Indexing Document

```
Input: WebKb dataset, Stemming List, Stop-Word list.
Output: The database contains a word table connected to the page's table.
Begin:
Initialization: Stemming-List= [ Return each word to the original].
        Stop-Word list= [a, an, the, and, it, for, or, but, in, my, your, etc.].
        Weighting according to the tags (i.e., Body=1, Headers, sub-headers=5, Title=6, Italic, bold=3, and anchor=4).
        Pages-table, and words-table.
Loop:
   For i=1 to 8282 do // all webpages in the Dataset (WebKb).
     Open all the webpage (document (i) as Text to treat with each word inside it.
       For j=1 to i do //for each webpage (documents (i))
          Tokenization process to get (Wj) // split webpage (documents to words).
          Return each Wj to the original by Checking them to the Stemming List.
            If the Wj is the Stop-Word, then Remove Wj
            Else
            If the Wj in the Body, then Wj=1
            Else
            If the Wj in the Title, then Wj=6
            Else
            If the Wj in the Headers or sub-Header, then Wj=5
            Else
            If Wj in the Anchor, then Wj=4
            Else
            If the Wj is Bold or Italic, then Wj=3
            End if
         End For
     Compute: (Total-weight-tag = Weight (Wi) tag + ∑Weight (Wi) tag).
            count Word(i) for each webpage(i).
          return Page-name, ID, and Pages list.
End for
End loop
Page table= Id, Page-name, words(j)count, and total weight tags)
Word table=Pages list.
End
```

As stated in algorithm 1, the process begins to apply the tokenization and prepare lists of the stemming. It is considered a way to return words to their origin, for example, mapping a comment group to one stem even if it is not a valid word linguistically. The stem represents the word's origin to which inflection (changes/derivatives), affixes (e.g., -s, -ed, -ing, -ize, -re, mis)), and stop word are added to reduce the text of each document (webpage) by excluding unnecessary data that adds processing time as well as additional storage space. Text Operations: (Tokenization-Stop word removal- and stemming).

Example

D1: Concepts of information technology.

D2: Data mining techniques.

D3: Web and Search engines.

D4: Information retrieval on the web.

D1: Concept inform technology.

D2: Data mining technique.

D3: Web search engine.

D4: Inform retrieve web.

Furthermore, it assigns a specific weight to each word according to the position in the text to split and distinguish the essential words from the text to facilitate the retrieval process. It computes the total weight tags of each webpage(i) and counts word(i) for each webpage(i), page name, ID, and page list. After these processes, two tables are created: a pages-table and a word table. Therefore, this proposed method reduces storage space compared to traditional and other related methods.

The main reason for using these weighted numbers in Table 3 is to reduce search space and retrieve all related documents in the dataset. Hence, the title receives the highest weight because it contains the most essential words in the user queries.

Table 3: Weighted tags according to the position in the document.

Names of each Tag	**Weight number**	**Impressions tags on the web page**	**General Description**
Body:	1	Appears once (no repetition)	It contains plain text (less important tag).
Title:	6		It contains terms near the user query (important tag).
Bold (B) & Italic (I):	3	n times appear	It provides document and scores explanations.
Anchor:	4		The tag includes a word that points to another word/link.
Header (i.e., h1, h2, h3):	5		Words found in these tags provide information about the structure.

4.2. Stage 2: Queries of All Users

It has many steps, starting with taking the queries as a string and then passing through a tokenization process (to get an array of words) for each word. This brings the specific page list with other details from the words/pages tables. Algorithm 2 demonstrates this stage.

```
Algorithm 2 Queries of all users
Input: a string of user query, (words, and pages) tables from algorithm 1.
Output: 1-D array of word [], ID_list array [].
Begin
  Initialization process:
  1-D array of word [], ID-list= [ ], Input user-query as a string, X [word] = [].
  LOOPING:
    Repeat from 0 to User Query Length.
      Splitting the S into words (Wi) through the tokenization process then storing in a 1-D array of the word [i].
      The 1-D array of the word [i]=array of the X [Wi].
      Repeat according to words that are stored in X [Wi] array
        1-D array [word, and pages-list] = obtain from words table.
        Aggregation ID-list [pages-list] for all user queries.
      End Repeat
    End Repeat
  End Loop
End
```

In algorithm 2, the queries of all users are converted, after being entered as a string, by the tokenization process to a group of words saved in a 1-D array. It obtains a word/page list from the words table and then stores them in the 1-D array. Each word from the pages-list is combined with the other pages-list words from the targeted query to create an array named ID-list, which contains the information for database words. To increase the probability of the population, the initial population is generated from the ID list (i.e., includes a repeated ID).

4.3. Stage 3: The Modified Culture Algorithm (MCA)

CA is considered a branch of evolutionary algorithms used in many fields. It was modified to be adaptable to IRS; the main steps for MCA are population space, belief space, and knowledge protocol.

For chromosome generation from the initial population, it takes the ID list (i.e., three main integer numbers representing the dataset directory) from algorithm 2 as an initial generation. This list includes the repeated or duplicated pages' lists, fitness function, and parent selection operators. Algorithm 3 demonstrates the main steps and processes of MCA.

```
Algorithm 3 Modified Culture Algorithm (MCA)
Input: ID-List [] from algorithm 2, stopping criteria.
Output: High fitness Chromosome (most relevant documents to the user queries).
Begin
Initialization:
population (Stochastic chromosomes (R1, R2) depending on ID-List [] from algorithm 2);
Belief-Network (0);
Channel communication (0);
evaluate population (0);
Parent selection;
Modified Fitness Function FF;
```

```
Number of Iteration Itr. =10;
Lock-up-table= [];
LOOPING:
    For i=0 to Itr. Do
        1. Population (i)= ID-list [i]. // create two chromosomes.
        2. Fitness Function FF (i)//the proposed FF as the following equations:
$\sum_{Chromosome=0} 10 - 1\sum_{gene=0}^{sharedpage-1} [(SharedPageTogether(gene) + 0.1| ID - array(gene) + (PID(gene) * 0.2) + \propto (gene)]$//$\propto$= FF(page)– index (page)
        3. Parent selection (i) using Tournament selection strategy // to select the fittest chromosome from the current population.
        4. Communicate Population (0), Belief network (Itr.);
        5. Adjust Belief network (Itr.);
        6. Communicate Belief network (Itr.), Population (Itr.);
        7. Modulate FF (Belief network (Itr.), Population (Itr.);
        8. Increase Itr. = Itr.+1.
        9. Select Population (Itr.) from Itr. - 1;
        10. Evolve Population (Itr.);
        11. Evaluate Population (Itr.);
        12. Stopping criteria: reporting the best solution when reaching the best result.
    End for
END LOOP
Storing the results in Lock-up-table;
End
```

In algorithm 3, the adaptation algorithm is used as a searching algorithm; it consists of three main stages (population, belief network, and communication channel (i.e., knowledge protocol).

The initial population takes the output of stage 2 (ID-list) from algorithm 2. The suggested system's optimal outcome was obtained with a population size of 80 after ten iterations. The ID of this list consists of three integer numbers: university number, directory number, and page code. Hence, the shape of the chromosomes is as follows:

Chromosome 1: 0-4-45, 0-1-362, 0-3-53, 0-2-18, 0-1-383, 0-1-399, 0-4-45, 0-2-18

Chromosome 2: 0-1-362, 0-3-53, 0-2-18, 0-1-383, 0-1-399, 0-4-45, 0-1-383, 0-3-53

For these chromosomes, compute a modified Fitness Function (FF). In this state, a new function is proposed to be suitable for solving this problem in algorithm 3.

This section explains the way the proposed function works with an example. Supposing the two chromosomes of the initial population consist of eight shared pages and only four genes of both chromosomes contain the words of the user queries, and the FF values are 1100 as follows:

Chromosome 1: 0-4-45, 0-1-362, 0-3-53, 0-2-18, 0-1-383, 0-1-399, 0-4-45, 0-2-18 1100

Chromosome 2: 0-1-362, 0-3-53, 0-2-18, 0-1-383, 0-1-399, 0-4-45, 0-1-383, 0-3-53 1100

Now check if that query is found in the pages brought together; if verified, it could be raised for every appearance of this query with 0.1 due to the Shared Page Together parameter representing many query's words' shared pages together. To determine which page contains all the user query's words in one place, search across all the pages generated at the time. According to the experiment, each time a group of query phrases appears together, the page fitness value can increase by 0.1. Thus, the value of FF for the first chromosome appeared five times and became as follows:

Chromosome 1: 0-4-45, 0-1-362, 0-3-53, 0-2-18, 0-1-383, 0-1-399, 0-4-45, 0-2-18 1100.5

And for the second chromosome appears three times and becomes:

Chromosome 2: 0-1-362, 0-3-53, 0-2-18, 0-1-383, 0-1-399, 0-4-45, 0-1-383, 0-3-53 1100.3

Furthermore, the Weight of Tags (WOT) is checked (i.e., it gives the weight according to the tag contained).

The weight begins from 1 to 6. For example, suppose the search query is GA. This word query appears in the tag's title, and its weight equals 6; the query now has a value of 6. Additionally, when a new page is added to the body, the weighting is increased to one because the body already has one.

Suppose chromosome 1 has the query; the GA is 1100, where the page's title includes the query and accordingly increases its value by 6:
FF = 1100+6= 1106
Assume the same query appears on another page in its body; FF is:
FF =1100+1= 1101
WOT prioritizes the query that appears in the title over the query that appears in the body. PID, which stands for page-id, multiplies for each page, comprising all queries coupled with a value of 0.2 chosen by the experiment to ensure this page's fitness remains high.

In algorithm 3, the straightforward value ensures that all pages pertinent to the user's query will still be present at the initial position of each chromosome.

After computing FF for each chromosome, choosing the best parent selection is necessary. The tournament is the technique used to select parents (select the fittest individual from the current population (i.e., 75). This strategy improves CA performance increasingly. Then, communicate between the current population after the parent selection, with a belief network that contains the previous population, to check the best after and before selection. Compute these steps for every iteration and modify after each result if needed, after FF checking and calculating for the population and belief network. Finally, the iteration is incremented, and the system will select the best population from iteration -1. The relevant results are recalled to the user queries and stored in a look-up table.

Look-up tables play a crucial role in IRSs by offering several benefits to efficient and effective retrieval processes. Here are some of the key benefits of using look-up tables in an IRS (i.e., Fast Data Retrieval, Constant Time Retrieval, Memory Efficiency, Reduced Computation, Consistency and Accuracy, Support for Complex Transformations, Adaptation to Hardware Constraints, Caching and Performance, Simplified Code, and Offline Preprocessing). It is important to note that while look-up tables offer many benefits, their design and implementation should be carefully considered. Choosing appropriate data structures, ensuring data integrity, and managing table updates are among the factors that need to be addressed to maximize the advantages of using look-up tables in an IRS.

4.4. Stage 4: Results and Evaluation Measurement

The indexing document method and modified culture algorithm (IDM-MCA) are evaluated using four main measures: recall, precision, storage space, and complexity time. Sklearn, Kears, and Tensor-flow libraries in Python are used to implement the proposed system on a laptop with the following specifications: Windows 11 as an Operating System with CPU Core i7 and 64-system bit.

Since a one-word query is meaningless, 40 queries with various word counts, ranging from 2 to 4, have been explicitly constructed to evaluate the suggested method with a predetermined number of linked documents. The number of queries is used to assess the precision of the proposed approach to find all documents connected to the queries. Two main phases are used: the first phase (intersection of web pages with the typed query words) locates/references every document that contains every user query's word. Phase 2 (selection of just shared pages) filters all pertinent documents to select only those that contain the search terms.

The suggested system's final phase involves using the following query to return relevant articles to a user query with a higher fitness value in the most recent iteration (i.e., iteration No. 10):

Query: GENETIC ALGORITHM STEPS

This query consists of four documents with shared pages; however, only one is relevant and includes all the query words.

5. Experimental Results and Discussions

The proposed IDM-MCA system is tested in two parts: the first tests the efficiency of memory space and the speed of retrieving relevant documents using IDM, whereas the second tests the efficiency of the modified CA.

5.1. Memory Efficiency

The suggested technique adds the necessary information and indexes all meaningful words; the data is stored in two tables. Table 1 stores words/page lists. Page lists contain ID lists with three entity codes (i.e., directory, university, and page). IDs decrease search time and retrieve relevant documents quickly. Memory space was remarkably reduced compared to conventional and other related methods, as shown in Figure 3. The conventional method needs more space of 2-byte (2×8 bit) for a document. For the dataset of 67,672 words, the required space is 896,186,560 MB, whereas the proposed technique needs a space of only 18 MB for data storage.

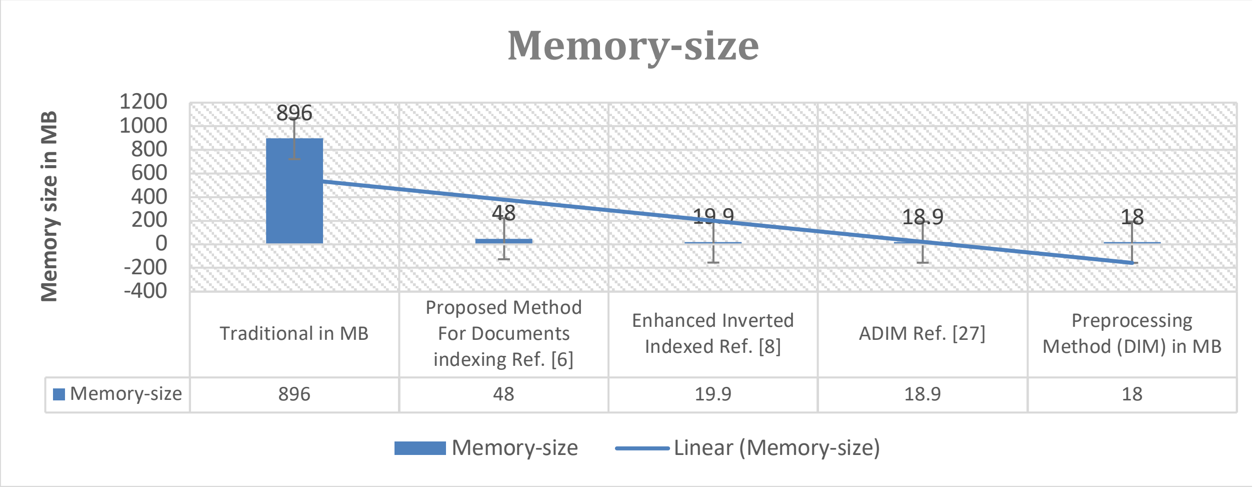


Fig. 3: Indexing document (Memory-Size)

5.2 Tested User Queries

The impact of the proposed system is assessed according to the main measurements (precision/recall). The Precision-Recall curve is a graphical representation that helps us understand the performance of the system in terms of retrieving relevant documents from a more extensive collection. The components and what they represent are as follows:

Precision: Precision measures how many retrieved documents are relevant to the query. Mathematically, it is the ratio of the number of true positive (relevant) documents retrieved to the total number of documents retrieved (both relevant and irrelevant). A high precision indicates that the system retrieves relevant primary documents and only brings in a few irrelevant ones.

Precision = TP / (TP + FP)

TP: True Positives (Relevant documents retrieved)

FP: False Positives (Irrelevant documents retrieved but mistakenly labeled as relevant)

Recall: Recall, also known as sensitivity or actual positive rate, measures how many relevant documents the system retrieved. It is the ratio of the number of actual positive documents retrieved to the total number of relevant documents in the collection. High recall means the system is good at retrieving a high proportion of relevant documents, even if it also retrieves some irrelevant ones.

Recall = TP / (TP + FN)

FN: False Negatives (Relevant documents not retrieved)

Therefore, Figure 4 demonstrates the measurement graph curve, and Figure 5 demonstrates the accuracy of different population sizes. The 40 various query lengths are shown in Table 4.

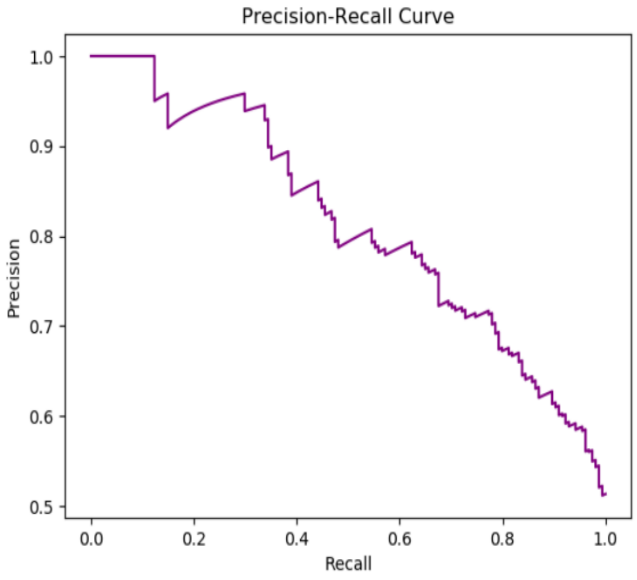


Fig. 4: Precision and recall curve

In Figure 4, The Precision-Recall curve is created by plotting different precision and recall values at various decision thresholds set by the IRS. Here's what the process looks like:

- The system retrieves a set of documents for a given query.
- The retrieved documents are ranked based on their relevance score or other criterion.

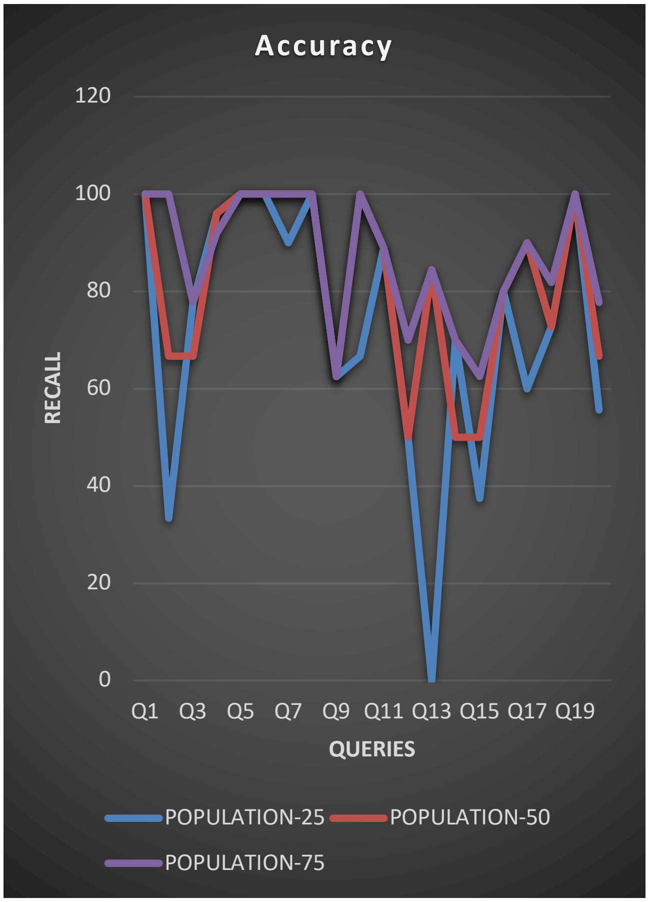


Fig. 5: Accuracy of population size (25, 50, and 75).

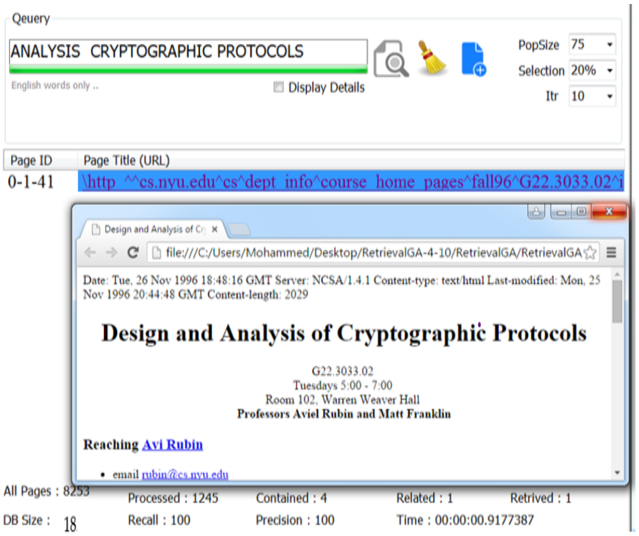
In Figure 5, the recall of population size 75 is better than that of the other two populations (25 and 50).

Different points on the Precision-Recall curve are plotted by changing the threshold for considering a document as relevant. The resulting curve typically starts near 1.0 precision and 0.0 recall (if all retrieved documents are relevant) and moves towards 0.5 precision and 1.0 recall (if all relevant documents are retrieved). The curve shows the trade-off between precision and recall as the system retrieves more documents. A system that balances high precision and high recall will have a curve that hugs the upper right corner of the graph. The Precision-Recall curve is a visual tool that helps evaluate an IRS's performance by illustrating how well it balances the trade-off between precision (relevance of retrieved documents) and recall (completeness of retrieval) for different decision thresholds.

Table 4: Results of executing 40 user queries.

User Query	No. of related documents.	Length of user Query	Semantic Results	Recall	Precision
CRYPTO SYSTEM DEPARTMENT	5	3	2 Of 2	100%	100 %
EFFICIENT SYSTEM	3	2	3 of 3	100%	100 %
INFORMATION GENETIC ALGORITHM	2	3	9 of 9	100%	100 %
INTRODUCTORY COMPUTER PROGRAMMING	2	3	25 of 25	100%	100 %
OPERATING SYSTEM	2	2	2 of 2	100%	100 %
PROGRAMMING LANGUAGE	3	2	1 of 1	100%	100 %
VIRTUAL MEMORY	9	2	10 of 10	100%	100 %
LOGIC DESIGN	25	2	1 of 1	100%	100 %
SUBJECT ENGINEERING TECHNIQUES	2	3	8 of 8	100 %	100 %
CRYPTOGRAPHIC ANALYSIS PROTOCOLS	1	3	3 of 3	100%	100%
PROGRAMMING LANGUAGES	10	2	9 of 9	100%	100%
IMAGE PROCESSING PROGRAM	1	4	9 of 10	90%	98%
DESIGN STYLE	8	2	11 of 13	90%	97%
NATURAL VLSI RESEARCH GROUP	3	3	10 of 10	100%	100%
OPERATING SYSTEM	10	2	8 of 8	100%	100%
NATURAL LANGUAGE	10	2	5 of 5	100%	100%
TEDER ROMER	13	2	9 of 10	90%	98%
MIKE DAHLING	10	2	10 of 11	90%	96%
DAVID ZUCKERMAN	8	2	12 of 12	100%	100%
WEB OPERATING SYSTEMS	5	3	9 of 9	100%	100%
SOFTWARE ENGINEERING PROJECT	10	3	6 of 7	90%	98%
NATURAL LANGUAGE	11	2	30 of 31	90%	99%
SOFTWARE SPECIFICATION	12	2	3 of 3	100 %	100%
COMPUTER COMMUNICATION NETWORKS	9	3	45 of 36	90%	100%
DANIEL WELD	7	2	17 of 18	90%	100%
CRAIG CHAMBERS	31	2	8 of 9	90%	100%
PROGRAMMING SOLUTIONS	3	2	3 of 3	100%	98%
CARL EBELING	36	2	8 of 8	100%	100%
STEVE HANKS	18	2	14 of 14	100%	100%
STEVEN TANIMOTO	9	2	4 of 5	90%	100%
PAUL YOUNG	3	2	14 of 15	90%	100%
QUALITY SYSTEMS	8	3	15 of 15	100%	100%
DISCRETE EXECUTION	14	2	24 of 25	96%	100%
SUFFICIENT PROGRAM	5	4	28 of 30	92%	100%
PROGRESS PROGRAMMING LANGUAGE	15	3	18 of 19	90%	100%
OLVI MANGASARIAN	15	2	6 of 7	89%	99%
MIRON LIVNY	25	2	5 of 5	100%	100%
SCIENTIFIC COMPUTATION	30	2	3 of 3	100%	100%
PROGRAMMING LANGUAGES COMPILERS	19	3	2 of 2	100%	100%
ADVANCED DIGITAL DESIGN	7	3	2 of 2	100%	100%
Average				**99 %**	**99%**

Table 4 demonstrates high efficiency in retrieving relevant documents. It shows a 99% recall value and precision with a response time of 0.8965 ms. Furthermore, Figure 6 illustrates the proposed system's execution.



Qeuery

ANALYSIS CRYPTOGRAPHIC PROTOCOLS

English words only ..

□ Display Details

PopSize 75

Selection 20%

Itr 10

Page ID	Page Title (URL)
0-1-41	\http ^^cs.nyu.edu^cs^dept info^course home pages^fall96^G22.3033.02^i

Design and Analysis of Cr

file:///C:/Users/Mohammed/Desktop/RetrievalGA-4-10/RetrievalGA/RetrievalGA

Date: Tue, 26 Nov 1996 18:48:16 GMT Server: NCSA/1.4.1 Content-type: text/html Last-modified: Mon, 25 Nov 1996 20:44:48 GMT Content-length: 2029

Design and Analysis of Cryptographic Protocols

G22.3033.02
Tuesdays 5:00 - 7:00
Room 102, Warren Weaver Hall
Professors Aviel Rubin and Matt Franklin

Reaching Avi Rubin

- email rubin@cs.nyu.edu

All Pages : 8253 Processed : 1245 Contained : 4 Related : 1 Retrived : 1

DB Size : 18 Recall : 100 Precision : 100 Time : 00:00:00.9177387

Fig. 6: Execution example of the proposed system.

5.3 Comparison with Similar Studies

Table 5 compares the traditional method with other related studies. Regarding recall, it is noted that the proposed system achieved 99% with enhancement between 1 and 19% over other systems considered in this study. Traditional indexing achieved a precision of 78%, 21% lower than the proposed system. The other systems considered achieved very close but lower precision results than the proposed system except [27], which achieved 100%. The proposed system elapsed 0.8965 ms to retrieve the required documents. This response time is remarkably fast compared to the traditional method or the one proposed by [8]. The last measure to be considered is the storage size. The proposed system uses 2% of the storage size utilized by the traditional method. Compared with the advanced IRS developed by [27], the proposed system reduced the required storage size by 4% and by 9.5% when compared with [8]. Generally speaking, [27] was slightly better than the proposed system in terms of precision and response time but somewhat worse than the proposed systems in terms of recall and storage size. In contrast, other systems were lower in the four measures than the proposed system.

Table 5: Comparison results with other related studies.

Method Name/ Ref.	**Preprocessing method**	**Recall**	**Precision**	**Response time/Execution time**	**Storage size in MB**
Traditional	N/A	80%	78%	Measures in hours.	Exceed 896
[7]	Enhanced inverted index	90%	97%	Measures in minutes.	48
[8]	Modified Genetic Algorithm for IRS	93%	99%		19.9
[27]	An efficient IRS using evolutionary algorithms	98%	100%	0.467478 ms	18.8
Proposed	Novel DIM-MCA	99%	99%	0.8965 ms	18

In Table 5, the proposed system had the best accuracy (precision and recall), minimum storage space (which belongs to the lookup table), and the best parameters with modified functions in the algorithm.

6. Limitations for Future work

In addition to addressing significant issues in IRSs in this work, mainly focusing on improving the relevance and retrievability of documents for users, areas still require further investigation/improvement. Potential limitations and recommendations for future work might include:

a) System scalability: To evaluate the system's scalability, testing with larger and more updated datasets is recommended. Hence, larger datasets can probably add further challenges, e.g., complexity and memory issues.

b) System generalizability: Evaluating the proposal performance by applying various documents' domains/types helps determine and identify limitations.

c) Robustness to noise and variability: Improving the IRS performance against the data noise and variability efficiently.

d) Consider more evaluating metrics (or integrating external knowledge): This helps to understand system performance, such as the F1-score, normalized discounted cumulative gain, and mean average precision. For external knowledge, ontologies or semantic networks could be incorporated

e) User-based assessment: It is worth evaluating system usability with user experience/interaction to achieve more enhancement.

f) Expanding to more retrieval tasks: Besides document retrieval, further retrieval work may expand to include question answering and entity/multimedia retrieval.

g) Evaluating more evolutionary algorithms: To improve system impact, it is recommended that further evolution (parameter-tuning) strategies/techniques be investigated.

h) Flexibility and adaptivity to updates: Environmental dynamism necessitates proposing new mechanisms for real-time changes to comply with changes in the document, user, architecture, or platform changes.

Addressing these aspects in future research could further enhance the proposal's effectiveness, efficiency, and usability in practical information retrieval scenarios.

7. Conclusions

CA was adopted in IRSs to overcome two technical issues that the users still suffer from current IRS while retrieving the relevant documents. The first issue represents the irrelevance of the first displayed retrieved documents. The second issue is the non-retrievable related documents. This paper developed and modified algorithms to contribute two main outcomes. Firstly, to obtain clean and prepared data stored in the database with less storage space and minimum response time. Secondly, modified evolutionary algorithms with the main concept being a population and manipulation of chromosomes according to the proposed FF with a communication protocol between belief space and population space to improve the retrieval of the user queries. The experimental results provide remarkable performance in terms of precision, recall, response time, and storage space. The results of the tested queries showed interesting values of 99% precision and recall with 0.8965 ms response time, while the needed memory space was only 18 MB. These results outperform considered similar work. The retrieval results are stored in a look-up table; this table works as cache memory. Additionally, the modified CA algorithm demonstrated its capacity to adapt and function well with the created operators, including the proposed FF, random selection mechanism, and tournament mechanism to achieve parent selection. The suggested system's optimal outcome was obtained with a population size of 80 after ten iterations, finding the user query's most pertinent documents without the need for rating which is the step that consumes longer time in other IRS systems. Additionally, because the documents are ranked according to their fitness values, this approach speeds up the ranking phase and ensures that the user query is satisfied with the most relevant information. For future work, the proposed system should be tested with larger datasets.

Reference

[1] D.N Mhawi and S.H Hashem, Proposed Hybrid Correlation Feature Selection Forest Panalized Attribute Approach to advance IDSs, Karbala Int J Mod Sci **7**, 405–420 (2021).

[2] S. Ceri, A. Bozzon, M. Brambilla, E. Della Valle, P. Fraternali, S. Quarteroni, An Introduction to Information Retrieval. In *Web Information Retrieval*; 3–11 (2013).

[3] M. Sarrouti and S. Ouatik El Alaoui, A passage retrieval method based on probabilistic information retrieval and UMLS concepts in biomedical question answering, J Biomed Inform **68**, 96–103 (2017).

[4] R. Kumar and S.C Sharma, Hybrid optimization and ontology-based semantic model for efficient text-based information retrieval, J Supercomput (2022).

[5] H.W Oleiwi, N. Saeed, H.L Al-taie, D.N Mhawi, Evaluation of Differentiated Services Policies in Multihomed Networks Based on an Interface-Selection Mechanism, Sustainability **14**, 1–12 (2022).

[6] A. Karim Abdul Hassan and D. Enteesha mhawi, A Proposed Method for Documents Indexing, Diyala J Pure Sci **13**, 43–56 (2017).

[7] A. Karim Hassan and D. Enteesha mhawi, Enhance Inverted Index Using in Information Retrieval, Eng Technol J **34**, 302–310 (2016).

[8] A.H Mhawi, and D.N. Information Retrieval Using Modified Genetic Algorithm, Al mansour *J* **72**, 15 (2017).

[9] R. Lakshmana Kumar, N. Kannammal, S. Krishnamoorthy, S. Kadry, Y. Nam, Semantics based clustering through cover-kmeans with ontovsm for information retrieval, Inf Technol Control **49**, 370–380 (2020).

[10] S. Zhang, L. Yao, A. Sun, Y. Tay, Deep learning based recommender system: A survey and new perspectives, ACM Comput Surv **52** (2019).

[11] M. Erritali, A. Beni-Hssane, M. Birjali, Y. Madani, An approach of semantic similarity measure between documents based on big data, Int J Electr Comput Eng **6** (2016).

[12] S. Chen, S. Xie, Q. Chen, Integrated embedding approach for knowledge base completion with CNN, Inf Technol Control **49**, 622–642 (2020).

[13] B. Kulzer, How do people with diabetes benefit from big data and artificial intelligence?, Diabetologe **17** (2021).

[14] D. Vatansever, J. Smallwood, E. Jefferies, Varying demands for cognitive control reveals shared neural processes supporting semantic and episodic memory retrieval, Nat Commun **12** (2021).

[15] D.S Jabonete and M.M De Leon, Development of an Automatic Document to Digital Record Association Feature for a Cloud-Based Accounting Information System. In Proceedings of the Lecture Notes in Networks and Systems; Vol. 283 (2022).

[16] I.W Ghindawi, M.S Kadhm, D.N Mhawi, The Weighted Feature Selection Method. J Coll Oeducation **2018**, *no.3.*

[17] W. Iqbal, W.I Malik, F. Bukhari, K.M Almustafa, Z. Nawaz, Big data full-text search index minimization using text summarization, Inf Technol Control **50**,

375–389 (2021).

[18] D.N Mhawi and S.H Hashim, Proposed Hybrid EnsembleLearninig algorithms for an Efficient Intrusion Detection System, IJCCE **22**, 73–84 (2022).

[19] N. El-Bathy, G. Azar, M. El-Bathy, G. Stein, Intelligent information retrieval lifecycle architecture based clustering genetic algorithm using SOA for modern medical industries. In Proceedings of the IEEE International Conference on Electro Information Technology; (2011).

[20] P. Zhang, H. Gao, Z. Hu, M. Yang, D. Song, J. Wang, Y. Hou, B. Hu, A bias–variance evaluation framework for information retrieval systems, Inf Process Manag **59** (2022).

[21] S. Bhardwaj and S. Sharma, An automated framework for incorporating fine-grained news data into S&P BSE SENSEX stock trading strategies, Indian J Sci Technol **9** (2016).

[22] H.W Oleiwi and H. Al-Raweshidy, Cooperative SWIPT THz-NOMA/6G Performance Analysis, Electron **11** (2022).

[23] K.A Hambarde and H. Proenca, Information Retrieval: Recent Advances and Beyond, 1–26 (2023).

[24] F. Wang, J. Liu, H. Wang, Sequential Text-Term Selection in Vector Space Models, J Bus Econ Stat **39**, 82–97 (2021).

[25] L. Chen, D. Kulasiri, S. Samarasinghe, A novel data-driven boolean model for genetic regulatory networks, Front Physiol **9** (2018).

[26] D.N Mhawi, H.W Oleiwi, N.H Saeed, H.L Al-Taie, An Efficient Information Retrieval System Using Evolutionary Algorithms, Network **2**, 583–605 (2022).

[27] H.W Oleiwi and H. Al-Raweshidy, SWIPT-Pairing Mechanism for Channel-Aware Cooperative H-NOMA in 6G Terahertz Communications, Sensors **22**, 6200 (2022).

[28] H.W Oleiwi, N. Saeed, Al-Raweshidy, H.S. A Cooperative SWIPT-Hybrid-NOMA Pairing Scheme considering SIC imperfection for THz Communications. *Proc - 2022 IEEE 4th Glob Power, Energy Commun Conf GPECOM 2022*, 638–643 (2022).

[29] H.W Oleiwi, D.N Mhawi, H. Al-Raweshidy, MLTs-ADCNs: Machine Learning Techniques for Anomaly Detection in Communication Networks, IEEE Access **10**, 91006–91017 (2022).

[30] H.W Oleiwi, H.L Al-Taie, N. Saeed, D.N Mhawi, A Comparative Investigation on Different QoS Mechanisms in Multi-Homed Networks, Iraqi J Ind Res **9**, 1–11 (2022).

[31] N. Al-ufoq, Company, N.A. *The 3rd International Scientific Conference of Computer Sciences (3SCCS2021)*; ISBN 9789922945552 (2022).

[32] B. Xu, H. Lin, Y. Lin, K. Xu, L. Wang, J. Gao, Incorporating semantic word representations into query expansion for microblog information retrieval, Inf Technol Control **48**, 626–636 (2019).

[33] F.S Al-Anzi and D. AbuZeina, Toward an enhanced Arabic text classification using cosine similarity and Latent Semantic Indexing, J King Saud Univ - Comput Inf Sci **29**, 189–195 (2017).

[34] M. Toro, A general overview of formal languages for individual-based modelling of ecosystems, J Log Algebr Methods Program **104**, 117–126 (2019).

[35] S. Katoch, S.S Chauhan, V. Kumar, A review on genetic algorithm: past, present, and future, Multimed Tools Appl **80**, 8091–8126 (2021).

[36] M.S Kadhm, An Accurate Diabetes Prediction System Based on K-means Clustering and Proposed Classification Approach, Int J Appl Eng Res **13**, 4038–4041 (2018).

[37] H.W Oleiwi, D.N Mhawi, H. Al-Raweshidy, A Meta-Model to Predict and Detect Malicious Activities in 6G-Structured Wireless Communication Networks, Electron **12** (2023).

[38] T. Back, U. Hammel, H.P Schwefel, Evolutionary computation: Comments on the history and current state, IEEE Trans Evol Comput **1**, 3–17 (1997).

[39] H.A Sameer, A.H Mutlag, S.K Gharghan, Journal of techniques **4**, 24–32 (2022).

[40] N. Rychtyckyj and R.G. Reynolds, Using Cultural Algorithms to Improve Knowledge Base. 1405–1412 (1998).

[41] M. Ohsaki, An input method using discrete fitness values for interactive GA, J Intell Fuzzy Syst **6**, 131–145 (1998).

[42] D.N Mhawi, A. Aldallal, S. Hassan, Advanced Feature-Selection-Based Hybrid Ensemble Learning Algorithms for Network Intrusion Detection Systems, Symmetry (Basel) **14**, 1461 (2022).

[43] J. Gu, R. Xiang, X. Wang, J. Li, W. Li, L. Qian, G. Zhou, C.R Huang, Multi-probe attention neural network for COVID-19 semantic indexing, BMC Bioinformatics **23** (2022).

[44] J. Andrés, A.H Toselli, E. Vidal, Search for Hyphenated Words in Probabilistic Indices: A Machine Learning Approach. In Proceedings of the Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); Vol. 14187 LNCS, pp. 269–285 (2023).

[45] I. Gil-Leiva, P.D Ortuño, R.F Corrêa, Automatic indexing of scientific articles on Library and Information Science with SISA, KEA and MAUI, Rev Esp Doc Cient **45** (2022).

[46] J. Rāts and I. Pede, Using a Topic Based Model to Automate Indexing and Routing of Incoming Enterprise Documents, Balt J Mod Comput **10**, 545–559 (2022).

[47] A. Briciu, A.D Călin, D.L Miholca, C. Moroz-Dubenco, V. Petrașcu, G. Dascălu, Machine-Learning-Based Approaches for Multi-Level Sentiment Analysis of Romanian Reviews, Mathematics **12** (2024).

[48] X. Wang, R.E Mercer, F. Rudzicz, KenMeSH: Knowledge-enhanced End-to-end Biomedical Text Labelling. In Proceedings of the Proceedings of the Annual Meeting of the Association for Computational Linguistics; Vol. 1, pp. 2941–2951 (2022).

[49] H. Liu, S. Carini, Z. Chen, S. Phillips Hey, I. Sim, C. Weng, Ontology-based categorization of clinical studies by their conditions, J Biomed Inform **135** (2022).

[50] R. Kumar and S.C Sharma, Hybrid optimized query expansion strategy for semantic information retrieval using spatial bound whale and binary moth flame optimization algorithm, Concurr Comput Pract Exp **34** (2022).

[51] H. Viltres-sala and V. Estrada-sentí, Information Retrieval Model with Query Expansion and User Preference Profile, INGENERA *32*, 0–3 (2023).

[52] Q. Xu, Y. Huang, S. Wu, C. Nugent, Clustering-based fusion for medical information retrieval, J Biomed Inform **135** (2022).

[53] J.Y Lee and S.B Cho, Sparse fitness evaluation for reducing user burden in interactive genetic algorithm. In Proceedings of the IEEE International Conference on Fuzzy Systems; Vol. 2 (1999).

[54] M. Shirakawa and M. Arakawa, Multi-objective optimization system for plant layout design (3rd report, Interactive multi-objective optimization technique for pipe routing design), J Adv Mech Des Syst Manuf **12** (2018).

[55] X. Sun and D. Gong, Surrogate model-assisted interactive genetic algorithms with individual's fuzzy and stochastic fitness, J Control Theory Appl **8**, 189–199 (2010).

[56] R.T Liaw, A cooperative coevolution framework for evolutionary learning and instance selection, Swarm Evol Comput **62** (2021).

[57] Y. Zhang, Research of web search based on cultural algorithm new framework, Procedia Eng **29**, 3641–3645 (2012).

Biography:

Doaa N. Mhawi is a Ph.D. Researcher at Computer Science Department, University of Technology, Baghdad, Iraq. She received a B.Sc. degree from the Department of Computer Science, University of Technology, Baghdad, in 2013 and an M.Sc. in Artificial Intelligence from the Computer Science Department, University of Technology, Baghdad, Iraq, in 2016. She is currently working at Middle Technical University, Baghdad, Iraq.

Haider W. Oleiwi is a Ph.D. researcher at Brunel University London's Electronic and Electrical Engineering Department. He received the B.Sc. degree from the Department of Electrical and Electronic Engineering, Electronic and Communications Division, University of Technology, Baghdad, in 2006, and the M.Sc. in Wireless Communication Systems from the Electronic and Electrical Engineering Department, Brunel University London, in 2008. He has worked in various positions for several academic, research centers, government, non-governmental organizations, and industrial establishments. He is currently a Wireless Networks and Communications Group (WNCG) member at Brunel University London.

Ammar AlDallal, Associate Professor, Chairperson of Telecommunications Engineering Department, Ahlia University. He holds a Bachelor's and Master's degrees in computer engineering from Kuwait University and obtained a Ph.D. in computing and systems information from Brunel University London in 2012. He taught more than 20 courses for undergraduate and postgraduate students related to software development, programming, algorithms, machine learning, bioinformatics, artificial intelligence, and computer security. He supervised a vast number of graduation projects, master dissertations, and PhD theses. AlDallal has more than 35 papers published in international journals and conference proceedings discussing several aspects of information retrieval, computer security, and machine learning. He serves as a peer reviewer in several international journals and conferences. He received the Best Paper and Best Presenter awards at several international conferences. He is a member of the Bahrain Society of Engineers and a senior member of IEEE. He is recognized by Brunel University London as a Ph.D. academic supervisor. Dr. AlDallal received a fellowship in teaching and learning from the Higher Education Academy-UK. He worked as a System Programmer in International Turnkey Systems-Kuwait (1998–2004) and as a Senior System Analyst in International Turnkey Systems in the Kingdom of Bahrain (2004–2012).