INTERNATIONAL JOURNAL OF SOFTWARE ENGINEERING AND COMPUTER SYSTEMS
E-ISSN: 2180-0650
VOLUME 17, ISSUE 1, 2024, pp. 32–47
DOI: https://doi.org/10.15282/ijsecs.11.1.2025.3.0135

RESEARCH ARTICLE

The Relationship between the African Buffalo Optimization and Digital Humanities

Julius Beneoluchi Odili*

Institute of Digital Humanities, Anchor University Lagos, Ipaja, Lagos 112106, Nigeria

ABSTRACT - The African Buffalo Optimization (ABO) is a swarm optimization technique developed in 2015 by Odili and Kahar. It obtained its inspiration from the movement of cape buffalos across the wide African rainforests and savannah using basically two sounds: /waaa/ and /maaa/. Since its development, the algorithm has been applied to a wide range of applications ranging from numerical function optimization, tuning of the Peripheral, Integral, Derivative parameters of Automatic Voltage Regulators, Travelling salesman problem etc. Despite the enormous potential benefits of combining stochastic optimization algorithms and Digital Humanities (DH), there is a lack of research exploring this intersection. DH data, such as historical texts, images, and artifacts etc. often require complex analysis and visualization techniques. Traditional methods may not effectively capture the nuances and complexities of these data. In this study, the relationship between the African Buffalo Optimization and Digital Humanities is explored with the aim of attracting digital humanities researchers' attention to the enormous potentials available to the research community in swarm optimization algorithms like the ABO algorithm. Deploying Java programming language, this study implements the ABO algorithm to the various aspects of DH such as text analysis, digital pedagogy, cultural analytics, digital preservation, digital scholarship, digital museums and curation as well as digital edition and publishing. It was discovered that the ABO is good in optimizing topic modelling for literary analysis, sentiment analysis for historical text analysis and entity recognition for cultural trend analysis. In addition, the algorithm is effective in digital cultural analytics, digital pedagogy and scholarship. The successful implementation in this study proved to be a good contribution as it unravels the close relationship between the two disciplines that seemed to have nothing in common. In view of our findings, we recommend that global researchers in DH should explore the diverse opportunities inherent in swarm optimization algorithms to further DH scholarship and research.

ARTICLE HISTORY
Received : 17 October 2024
Revised : 9 December 2024
Accepted : 2 May 2025
Published : 6 May 2025

KEYWORDS
African Buffalo Optimization
Digital humanities
Cultural analytics
Digital Pedagogy
Text analysis

1.0 INTRODUCTION

Most research disciplines in modern times are interdisciplinary multi-disciplinary and transdisciplinary [1]. DH becomes a discipline of interest because it is not only multidisciplinary but also interdisciplinary and transdisciplinary [2]. Interdisciplinary research has been defined as such kinds of research deployed by individuals or teams that requires the integration of data, information, tools, techniques, theories, concepts and perspectives from two or more bodies cum disciplines of specialized knowledge to further fundamental capacity to provide solutions to problems whose real solutions are far beyond the immediate scope of a single discipline. As such, interdisciplinary research disciplines, rather than working independently, interacts with and works collaboratively with other disciplines [3]. Concomitantly, because interdisciplinary researches rely on shared knowledge, it has the capacity over time to birth novel interdisciplinary fields. For instance, nanoscience, neuroscience and biochemistry emerged as interdisciplinary disciplines which have eventually grown to become independent disciplines [4].

Meanwhile multidisciplinary research involves the assembly of experts from various disciplines, working together, with a view to solving a problem, with each expert drawing on his/her disciplinary knowledge. Therefore, while in interdisciplinary research projects, the team members aggregate each expert's knowledge in one integrated plan, in multidisciplinary research, team members remain within his/her discipline [5]. In summary, in an interdisciplinary research, team members are made up of experts from different disciplines working interdependently, interacting both formally and informally, while in a multidisciplinary research environment, a team is composed of experts from more than one professional discipline working independently and interacting formally On its part, transdisciplinary refers to an approach that goes beyond interdisciplinary collaboration, integrating multiple disciplines, methodologies and perspectives to address complex problems or issues. It involves:

1. Integrating knowledge and methods from various disciplines.
2. Involving stakeholders and practitioners from diverse backgrounds.
3. Focusing on real-world problems and applications.
4. Encouraging co-creation and collaboration.

***CORRESPONDING AUTHOR | J.B,Odili** | jodili@aul.edu.ng

5. Embracing complexity and uncertainty.
6. Seeking holistic and systemic understanding.
7. Developing innovative solutions and new knowledge [6].

DH is such a unique discipline because beyond interdisciplinary and multi-disciplinary, it is a transdisciplinary discipline that engages transdisciplinary approaches in arriving at solutions. The lack of research exploring the intersection of stochastic optimization algorithms like the Particle Swarm Optimization, Genetic Algorithm, Cuckoo Search, ABO and the DH discipline hinders the development of effective utilization of optimization techniques for DH data analysis. This knowledge gap limits the potential for innovative data analysis and visualization methods, ultimately affecting our understanding of human culture, literature and history. Our motivation in this paper is to explore the place of the ABO within the transdisciplinary scope of DH from the computational perspective. The ABO was chosen for this study because of the good results it has posted in the areas of its application.

Please note that transdisciplinary approaches are essential in addressing complex challenges like sustainability, social justice, and digital transformation, where multiple perspectives and expertise are required to develop comprehensive solutions [7]. In the context of ABO in DH, a transdisciplinary approach would involve combining computer science, humanities, social sciences and cultural studies to develop innovative digital methods and tools for analyzing, preserving, and disseminating cultural heritage [8]. This study, basically, is an attempt to explore the potential for interdisciplinary collaboration between digital humanists and computer scientists. It investigates how ABO can facilitate collaboration between digital humanists and computer scientists and examine the potential benefits of such collaboration.

The rest of this paper is organized in the following way: section two presents the review of relevant literature; section three examines the materials and methods of this study; section four explores the practical application areas of the ABO to DH; section five focusses on the results cum discussions and section six draw conclusions on the study.

2.0 RELATED WORKS

2.1 African Buffalo Optimization

The African Buffalo Optimization (ABO) algorithm is a nature-inspired optimization technique that mimics the behaviour of African buffalo herds [9]. It' is a metaheuristic algorithm used to solve optimization problems, particularly in engineering and computer science. Inspired by the buffalo's social structure and migration patterns, the ABO algorithm works as follows:

Algorithm 1. ABO algorithm

1. Initialization: A population of candidate solutions (buffalo) is randomly generated.
2. Migration: Buffalos move towards better solutions, influenced by their neighbours and the best solution found so far.
3. Grazing: Buffalo exploit resources (evaluate fitness) and update their positions.
4. Reproduction: Fitter buffalo reproduce, and their offspring replace weaker ones.
5. Termination: The algorithm stops when a termination condition is met (e.g., maximum iterations or satisfactory fitness) [10].

The ABO algorithm is presented:

1. Initialization: randomly place buffalos to nodes at the solution space;
2. Update the buffalos' exploitation using Equation:
3. $m_k' = m_k + lp1(bg - w_k) + lp2(bp_k - w_k)$
   where m_k and w_k represents the exploitation and exploration moves respectively of the k^{th} buffalo (k=1, 2…. N); lp1 and lp2 are learning factors; bg is the herd's best fitness and bp , the individual buffalo's best location
4. Update the location of buffalos using the Equation:
5. $w_k' = \frac{(w_k + m_k)}{\lambda}$
6. Is bg_{max} updating? Yes, go to 5. If No, go to 2
7. If the stopping criteria is not met, go back to algorithm step 2, else go to 6
8. Output best solution [11].

From the available application of the ABO, the algorithm offers a unique approach to optimization, with advantages like:

1. Robustness
2. Flexibility
3. Ability to handle complex problems

However, like other metaheuristics, ABO may require careful parameter tuning and may not always guarantee optimal solutions

2.2 Digital Humanities

Digital Humanities (DH) refers to the intersection of humanities research and digital technologies. It encompasses methods and tools from computer science, data analysis, and digital media to study and present humanities research. DH is an transdisciplinary field that combines humanities research with digital tools, methods, and media. It explores the intersection of technology and humanities, leveraging digital technologies to analyze, represent, and disseminate humanities research [12].

The major areas in Digital Humanities include:

1. Digital Scholarship: Using digital methods and tools to conduct research, analyze data, and present findings in humanities disciplines.'
2. Digital Pedagogy: Integrating digital technologies and methods into teaching and learning in humanities education.
3. Digital Cultural Heritage: Preserving, representing, and analyzing cultural heritage materials in digital formats.
4. Digital Editions: Creating digital versions of humanities texts, such as books, manuscripts, and archives.
5. Digital Art and Creativity: Using digital tools and media to create new forms of art, literature, and performance.
6. Human-Computer Interaction: Designing user interfaces and experiences for humanities research and applications.
7. DH Infrastructure: Developing and maintaining digital tools, platforms, and repositories for humanities research [13].

Since DH research and scholarship has become increasingly popular, especially since the turn of this century, its benefits are becoming more and more glaring. Some of these include: increased accessibility to humanities research and materials, new forms of analysis and representation, collaboration and community-building across disciplines and borders, innovative pedagogy and teaching methods as well as preservation and dissemination of cultural heritage materials [14]. In summary, this study aims to contribute to the development of innovative optimization techniques for cultural heritage data analysis, enhancing our understanding of human culture and history. The findings of this research will provide valuable insights for scholars, researchers, and practitioners in DH, optimization, and cultural heritage preservation.
.

3.0 METHODS AND MATERIAL

The African Buffalo Optimization is a swarm intelligence algorithm. Swarm optimization algorithms are a type of artificial intelligence inspired by the collective behaviour of biological systems, such as flocks of birds or schools of fish, movement of animals of diverse kinds in a swarm. These algorithms are designed to optimize complex problems by iteratively improving candidate solutions, mimicking the swarm's ability to adapt and converge on optimal solutions [15].

The ABO algorithm has exciting potential applications in DH, including text analysis, digital pedagogy, cultural analytics, digital preservation, digital scholarship, digital curation and museum studies, digital editions and publishing, social media and online communities [16]. The ABO Pseudocode for exploring the relationship between the algorithm and DH is presented:

Algorithm 2. ABO's pseudocode for exploring the relationship between the algorithm and DH

INPUT: DH dataset (text, images, videos)
OUTPUT: Optimized DH data analysis results

Step 1. Initialize ABO parameters:
- 1.1 Buffalo population size (N)
- 1.2 Number of iterations (T)
- 1.3 Learning rate (α)
- 1.4 Exploration-exploitation trade-off parameter (β)

Step 2. Initialize DH data analysis parameters:
- 2.1 Feature extraction method (e.g., TF-IDF, word embeddings)
- 2.2 Classification algorithm (e.g., SVM, random forest)

Step 3. Preprocess DH dataset:
- 3.1 Tokenize text data
- 3.2 Extract features from images and videos

Step 4. Apply ABO algorithm:
- 4.1 Initialize buffalo positions (x) and locations (v)
- 4.2 Evaluate fitness function (e.g., accuracy, precision, recall)
- 4.3 Update buffalo positions and locations using ABO equations
- 4.4 Repeat steps 1-4 until convergence or maximum iterations

Step 5. Analyze optimized DH data:
- 5.1 Extract features from optimized data
- 5.2 Apply classification algorithm to optimized data
- 5.3 Evaluate performance metrics (e.g., accuracy, precision, recall)

Step 6. Visualize results:
- 6.1 Use visualization tools (e.g., Tableau, Power BI, Gephi) to visualize optimized DH data analysis results

4.0 PRACTICAL APPLICATION AREAS OF THE ABO TO DH

4.1. Text Analysis

ABO, being a swarm optimization algorithm is useful in text analysis in digital humanities. In text analysis, the ABO can be used in optimizing topic modelling for literary analysis, sentiment analysis for historical text analysis and entity recognition for cultural trend analysis.

Moreover, ABO in text analysis can be used for tasks such as:

1. **Text clustering**: ABO can group similar text documents together based on their content, helping to identify patterns and topics.
2. **Feature selection**: ABO can select the most relevant features (words or phrases) in a text dataset, improving the performance of text classification or sentiment analysis models.
3. **Text summarization**: ABO can identify the most important sentences or phrases in a text document, generating a concise summary and
4. **Topic modelling**: ABO can discover hidden topics in a text corpus, revealing underlying themes and trends [17].
5. In summary, please note that to apply ABO to text analysis, the following steps needs to be taken to obtain the best outcome:
   - Preprocess the text data (e.g., tokenization, stop word removal)
   - Define the optimization problem (e.g., clustering, feature selection)
   - Initialize the ABO algorithm (e.g., population size, iteration count)
   - Run the ABO algorithm to search for the optimal solution
   - Evaluate and refine the results

A simple ABO Java code for text analysis is hereby presented:

Listing 1. ABO's Java code for text analysis

```
import java.util.*;
public class TextAnalysisABO {
    public static void main(String[] args) {
        // Text data
        String text = "This is a sample text for analysis.";
        // Pre-processing: Tokenization
        String[] tokens = text.split("\\s+");

        // ABO parameters
        int herdSize = 20;
        int iterations = 100;
        double stepSize = 0.1;

        // Initialize buffalo positions and velocities
        double[][] buffaloPositions = new double[herdSize][tokens.length];
        double[][] buffaloVelocities = new double[herdSize][tokens.length];

        // ABO algorithm
        for (int i = 0; i < iterations; i++) {
            // Update buffalo positions and velocities
            for (int j = 0; j < herdSize; j++) {
                for (int k = 0; k < tokens.length; k++) {
                    buffaloPositions[j][k] += buffaloVelocities[j][k];
                    buffaloVelocities[j][k] += stepSize * (Math.random() - 0.5);
                }
            }

            // Calculate fitness (e.g., term frequency)
            double[] fitness = new double[herdSize];
            for (int j = 0; j < herdSize; j++) {
                for (int k = 0; k < tokens.length; k++) {
                    fitness[j] += buffaloPositions[j][k] * tokens[k].length();
```

```
                    }
                }

                // Update best buffalo position
                int bestBuffalo = 0;
                for (int j = 1; j < herdSize; j++) {
                    if (fitness[j] > fitness[bestBuffalo]) {
                        bestBuffalo = j;
                    }
                }
                // Print best buffalo position (e.g., most frequent terms)
                System.out.println("Best buffalo position: " +
Arrays.toString(buffaloPositions[bestBuffalo]));
            }
        }
    }
```

4.2. Digital Pedagogy

ABO found relevance in DH subfields of digital pedagogy. In this area, personalized learning path optimization, adaptive assessment and feedback optimization and optimizing educational resource allocation. Specifically, ABO can be used to optimize learning processes, improve educational outcomes, and enhance the overall learning experience. Some potential ways to apply ABO in digital pedagogy:

1. **Personalized Learning:** ABO can be used to optimize personalized learning pathways for students, adapting to their individual learning styles, pace, and needs.
2. **Resource Allocation**: ABO can help optimize the allocation of digital resources, such as educational content, tools, and platforms, to maximize learning outcomes.
3. **Learning Scheduling**: ABO can be used to optimize learning schedules, ensuring that students have adequate time to learn and practice new concepts.
4. **Assessment and Feedback**: ABO can help optimize assessment and feedback processes, providing students with timely and relevant feedback to improve their learning.
5. **Virtual Learning Environments**: ABO can be used to optimize the design and organization of virtual learning environments, creating an immersive and engaging learning experience.
6. **Intelligent Tutoring Systems**: ABO can be applied to intelligent tutoring systems, providing real-time support and guidance to students.
7. **Learning Analytics**: ABO can help optimize learning analytics, identifying areas where students need extra support and improving overall learning outcomes [18].

From the foregoing, it is becoming clear that by applying ABO in digital pedagogy, educators and instructional designers can create more effective, efficient, and personalized learning experiences for students. A Java code for digital pedagogy is presented:

Listing 2. ABO's Java code for digital pedagogy

```
import java.util.*;
public class DigitalPedagogyABO {
    public static void main(String[] args) {
        // Learning objectives
        String[] objectives = {"Analyze digital texts", "Create digital artifacts",
"Design digital pedagogy"};

        // Student data
        int[][] studentSkills = {
            {3, 2, 1},  // Student 1 skills
            {2, 3, 2},  // Student 2 skills
            {1, 1, 3}   // Student 3 skills
        };

        // ABO parameters
        int herdSize = 20;
        int iterations = 100;
        double stepSize = 0.1;

        // Initialize buffalo positions and velocities
        double[][] buffaloPositions = new double[herdSize][objectives.length];
        double[][] buffaloVelocities = new double[herdSize][objectives.length];
```

```
            // ABO algorithm
            for (int i = 0; i < iterations; i++) {
                // Update buffalo positions and velocities
                for (int j = 0; j < herdSize; j++) {
                    for (int k = 0; k < objectives.length; k++) {
                        buffaloPositions[j][k] += buffaloVelocities[j][k];
                        buffaloVelocities[j][k] += stepSize * (Math.random() - 0.5);
                    }
                }

                // Calculate fitness (e.g., student skill match)
                double[] fitness = new double[herdSize];
                for (int j = 0; j < herdSize; j++) {
                    for (int k = 0; k < objectives.length; k++) {
                        fitness[j] += buffaloPositions[j][k] * studentSkills[k][j];
                    }
                }
                // Update best buffalo position
                int bestBuffalo = 0;
                for (int j = 1; j < herdSize; j++) {
                    if (fitness[j] > fitness[bestBuffalo]) {
                        bestBuffalo = j;
                    }
                }
                // Print best buffalo position (e.g., optimal learning path)
                System.out.println("Best buffalo position: " +
Arrays.toString(buffaloPositions[bestBuffalo]));
            }
        }
    }
```

4.3 Cultural Analytics

Cultural analytics is a major area of discussion in DH. In cultural analytics, the ABO is useful in network analysis optimization for social and cultural networks, data visualization optimization for cultural trend analysis and trend detection optimization for cultural and social phenomena. ABO can be used to optimize learning processes, improve educational outcomes, and enhance the overall learning experience. Some other ways ABO can be applied in cultural analytics includes:

1. **Cultural Trend Analysis**: ABO can help identify and track cultural trends, such as the spread of ideas, memes, or social movements, by analyzing social media data, online conversations, and cultural artifacts.
2. **Community Detection**: ABO can be used to identify and cluster cultural communities, such as online forums, social networks, or artistic groups, based on their shared interests, values, and behaviours.
3. **Cultural Influence Mapping**: ABO can help map the influence of cultural leaders, trendsetters, or opinion makers, and how they shape cultural narratives and trends.
4. **Cultural Evolution Modelling**: ABO can be applied to model the evolution of cultural phenomena, such as the development of languages, music genres, or artistic styles, over time.
5. **Cultural Sentiment Analysis**: ABO can help analyze the emotional tone and sentiment of cultural texts, such as social media posts, reviews or articles, to understand public opinion and cultural attitudes.
6. **Cultural Network Analysis**: ABO can be used to study the structure and dynamics of cultural networks, such as collaborations, citations, or references, to understand cultural exchange and innovation.
7. **Cultural Recommendation Systems**: ABO can help develop personalized cultural recommendation systems, suggesting music, movies, books, or art based on individual cultural preferences and tastes.

In conclusion, using ABO in cultural analytics assists researchers and analysts to gain a deeper understanding of cultural dynamics, trends and phenomena as well as develop innovative solutions for cultural institutions, marketing and policy-making [19]. Java code for ABO for cultural analytics is presented below:

Listing 3. Java code for ABO for cultural analytics

```
import java.util.*;
public class CulturalAnalyticsABO {
    public static void main(String[] args) {
        // Cultural data (e.g., texts, images, artifacts)
        String[] texts = {"Text1", "Text2", "Text3"};
        int[][] textFeatures = {
```

```
            {1, 2, 3},  // Text1 features
            {2, 3, 1},  // Text2 features
            {3, 1, 2}   // Text3 features
        };

        // ABO parameters
        int herdSize = 20;
        int iterations = 100;
        double stepSize = 0.1;

        // Initialize buffalo positions and velocities
        double[][] buffaloPositions = new double[herdSize][textFeatures[0].length];
        double[][] buffaloVelocities = new double[herdSize][textFeatures[0].length];

        // ABO algorithm
        for (int i = 0; i < iterations; i++) {
            // Update buffalo positions and velocities
            for (int j = 0; j < herdSize; j++) {
                for (int k = 0; k < textFeatures[0].length; k++) {
                    buffaloPositions[j][k] += buffaloVelocities[j][k];
                    buffaloVelocities[j][k] += stepSize * (Math.random() - 0.5);
                }
            }
            // Calculate fitness (e.g., cultural similarity)
            double[] fitness = new double[herdSize];
            for (int j = 0; j < herdSize; j++) {
                for (int k = 0; k < textFeatures.length; k++) {
                    fitness[j] += buffaloPositions[j][k] * textFeatures[k][j];
                }
            }
            // Update best buffalo position
            int bestBuffalo = 0;
            for (int j = 1; j < herdSize; j++) {
                if (fitness[j] > fitness[bestBuffalo]) {
                    bestBuffalo = j;
                }
            }
            // Print best buffalo position (e.g., cultural trends)
            System.out.println("Best        buffalo        position:        "        +
Arrays.toString(buffaloPositions[bestBuffalo]));
        }
    }

}
```

This code demonstrates ABO algorithm for cultural analytics, using text features and cultural similarity as a fitness function. The ABO algorithm iteratively updates the locations of "buffaloes" (agents) in a virtual landscape, where the fitness function represents the quality of the cultural analysis. In this example, the best buffalo position represents the cultural trends in the text data.

4.4. Digital Preservation

The ABO as a swarm intelligence technique is useful in digital preservation of artifacts as the algorithm can be deployed as a tool for optimizing digital preservation strategies for cultural artifacts, identifying optimal storage solutions for digital cultural heritage and in data compression optimization for digital preservation. [20]. Please find below the ABO code for digital preservation in Java programming language:

Listing 4. ABO code for digital preservation in Java programming language

```
import java.util.*;
public class DigitalPreservationABO {
    public static void main(String[] args) {
        // Digital artifacts (e.g., files, documents, images)
        String[] artifacts = {"Artifact1", "Artifact2", "Artifact3"};
        int[][] artifactFeatures = {
            {1, 2, 3},  // Artifact1 features (e.g., format, size, metadata)
            {2, 3, 1},  // Artifact2 features
            {3, 1, 2}   // Artifact3 features
        };
        // Preservation goals (e.g., accessibility, authenticity, usability)
        double[] goals = {0.8, 0.7, 0.9};
```

```
        // ABO parameters
        int herdSize = 20;
        int iterations = 100;
        double stepSize = 0.1;

        // Initialize buffalo positions and velocities
        double[][] buffaloPositions = new double[herdSize][artifactFeatures[0].length];
        double[][] buffaloVelocities = new double[herdSize][artifactFeatures[0].length];

        // ABO algorithm
        for (int i = 0; i < iterations; i++) {
            // Update buffalo positions and velocities
            for (int j = 0; j < herdSize; j++) {
                for (int k = 0; k < artifactFeatures[0].length; k++) {
                    buffaloPositions[j][k] += buffaloVelocities[j][k];
                    buffaloVelocities[j][k] += stepSize * (Math.random() - 0.5);
                }
            }
            // Calculate fitness (e.g., preservation quality)
            double[] fitness = new double[herdSize];
            for (int j = 0; j < herdSize; j++) {
                for (int k = 0; k < artifactFeatures.length; k++) {
                    fitness[j] += buffaloPositions[j][k] * artifactFeatures[k][j] * goals[k];
                }
            }
            // Update best buffalo position
            int bestBuffalo = 0;
            for (int j = 1; j < herdSize; j++) {
                if (fitness[j] > fitness[bestBuffalo]) {
                    bestBuffalo = j;
                }
            }
            // Print best buffalo position (e.g., optimal preservation strategy)
            System.out.println("Best buffalo position: " + Arrays.toString(buffaloPositions[bestBuffalo]));
        }
    }
}
```

Please note that this code demonstrates ABO algorithm for digital scholarship, using research data features and scholarship goals as a fitness function. The ABO algorithm iteratively updates the positions of "buffaloes" (agents) in a virtual landscape, where the fitness function represents the quality of the research strategy. In this example, the best buffalo position represents the optimal research strategy for the digital scholarship.

4.5. Digital Scholarship

Applying the ABO in digital scholarship subfield of DH is useful in optimizing research workflows and literature reviews, identifying optimal research questions and hypotheses and in optimizing data collection and analysis for digital scholarship. ABO's application in digital scholarship is fascinating as ABO can enhance digital scholarship in various ways. For instance, ABO is useful in:

1. **Optimizing Research Workflows**: ABO can streamline and optimize research workflows, automating tasks, and improving collaboration and productivity.
2. **Text Analysis**: ABO can be used for optimized text analysis, such as sentiment analysis, topic modelling, and named entity recognition, to uncover patterns and insights in large corpora of texts.
3. **Data Mining**: ABO can aid in data mining and knowledge discovery, identifying relevant patterns and relationships in large datasets.
4. **Digital Curation**: ABO can optimize digital curation processes, ensuring the long-term preservation and accessibility of digital research materials.
5. **Collaborative Research**: ABO can facilitate collaborative research by identifying optimal team compositions, workflows, and communication strategies.
6. **Research Impact Optimization**: ABO can help optimize research impact by identifying the most effective dissemination channels, audience engagement strategies, and metrics for evaluation.
7. **Digital Pedagogy**: ABO can optimize digital pedagogy by identifying the most effective digital resources, learning pathways, and assessment strategies for teaching and learning.

8. Therefore, by applying ABO in digital scholarship, researchers can accelerate discovery and innovation, enhance research quality and impact, improve collaboration and productivity, develop more effective research workflows and methodologies in addition to optimizing the use of digital tools and resources [21]. ABO code for digital scholarship in Java programming language is presented:

Listing 5. ABO code for digital scholarship in Java programming language

```
import java.util.*;
public class DigitalScholarshipABO {
    public static void main(String[] args) {
        // Research data (e.g., articles, books, datasets)
        String[] researchData = {"Data1", "Data2", "Data3"};
        int[][] dataFeatures = {
            {1, 2, 3},  // Data1 features (e.g., citations, authors, keywords)
            {2, 3, 1},  // Data2 features
            {3, 1, 2}   // Data3 features
        };
        // Scholarship goals (e.g., impact, relevance, accuracy)
        double[] goals = {0.9, 0.8, 0.7};

        // ABO parameters
        int herdSize = 20;
        int iterations = 100;
        double stepSize = 0.1;

        // Initialize buffalo positions and velocities
        double[][] buffaloPositions = new double[herdSize][dataFeatures[0].length];
        double[][] buffaloVelocities = new double[herdSize][dataFeatures[0].length];

        // ABO algorithm
        for (int i = 0; i < iterations; i++) {
            // Update buffalo positions and velocities
            for (int j = 0; j < herdSize; j++) {
                for (int k = 0; k < dataFeatures[0].length; k++) {
                    buffaloPositions[j][k] += buffaloVelocities[j][k];
                    buffaloVelocities[j][k] += stepSize * (Math.random() - 0.5);
                }
            }
            // Calculate fitness (e.g., scholarship quality)
            double[] fitness = new double[herdSize];
            for (int j = 0; j < herdSize; j++) {
                for (int k = 0; k < dataFeatures.length; k++) {
                    fitness[j]  +=  buffaloPositions[j][k]  *  dataFeatures[k][j]  *
goals[k];
                }
            }

            // Update best buffalo position
            int bestBuffalo = 0;
            for (int j = 1; j < herdSize; j++) {
                if (fitness[j] > fitness[bestBuffalo]) {
                    bestBuffalo = j;
                }
            }
            // Print best buffalo position (e.g., optimal research strategy)
            System.out.println("Best       buffalo       position:       "       +
Arrays.toString(buffaloPositions[bestBuffalo]));
        }
    }

}
```

4.6. Digital Curation and Museum Studies

When ABO is applied to digital curation and museum studies, it enhances the optimization of digital exhibit design and virtual museum experiences, helps in identifying optimal strategies for digital artifact curation and preservation as well as in optimizing digital museum collections and archives. Specifically, in digital curation and museum studies, ABO can be applied in various ways, including:

1. **Optimizing Digital Collections**: ABO can help optimize digital collections management, ensuring efficient storage, retrieval and preservation of digital artifacts.

2. **Digital Preservation**: ABO can aid in developing optimal digital preservation strategies, ensuring long-term accessibility and integrity of digital cultural heritage.
3. **Museum Digitalization**: ABO can optimize the digitization process of museum collections, ensuring efficient and effective digital capture and representation of cultural artifacts.
4. **Digital Exhibitions:** ABO can help design and optimize digital exhibitions, creating engaging and immersive experiences for online audiences.
5. **Collections Analytics**: ABO can aid in analyzing and visualizing collections data, providing insights into collection development, usage, and preservation.
6. **Digital Curation Workflows**: ABO can optimize digital curation workflows, streamlining tasks and improving collaboration among curators, conservators, and other stakeholders.
7. **Museum Online Presence**: ABO can help optimize museum online presence, ensuring effective digital engagement, marketing, and audience development.
8. **Digital Asset Management**: ABO can be useful in optimizing digital asset management, ensuring efficient storage, retrieval, and use of digital assets.
9. **Digital Conservation**: ABO can aid in developing optimal digital conservation strategies, ensuring the long-term preservation and accessibility of digital cultural heritage.
10. **Museum Community Engagement**: ABO can help optimize museum community engagement, ensuring effective digital outreach, participation, and engagement with diverse audiences [22].
11. From the foregoing, it is obvious that by deploying ABO in digital curation and museum studies, professionals can improve digital collections management, enhance online engagement, optimize digital exhibitions, and develop more effective digital curation workflows, among other benefits.

Listing 6: ABO's Java Code Listing for Digital Curation

```
import java.util.*;
public class DigitalCurationABO {
    public static void main(String[] args) {
        // Museum collection data (e.g., artifacts, metadata)
        String[] collectionData = {"Artifact1", "Artifact2", "Artifact3"};
        int[][] collectionFeatures = {
            {1, 2, 3},  // Artifact1 features (e.g., type, date, creator)
            {2, 3, 1},  // Artifact2 features
            {3, 1, 2}   // Artifact3 features
        };
        // Curation goals (e.g., preservation, accessibility, engagement)
        double[] goals = {0.8, 0.7, 0.9};

        // ABO parameters
        int herdSize = 20;
        int iterations = 100;
        double stepSize = 0.1;

        // Initialize buffalo positions and velocities
        double[][] buffaloPositions = new
double[herdSize][collectionFeatures[0].length];
        double[][] buffaloVelocities = new
double[herdSize][collectionFeatures[0].length];

        // ABO algorithm
        for (int i = 0; i < iterations; i++) {
            // Update buffalo positions and velocities
            for (int j = 0; j < herdSize; j++) {
                for (int k = 0; k < collectionFeatures[0].length; k++) {
                    buffaloPositions[j][k] += buffaloVelocities[j][k];
                    buffaloVelocities[j][k] += stepSize * (Math.random() - 0.5);
                }
            }
            // Calculate fitness (e.g., curation quality)
            double[] fitness = new double[herdSize];
            for (int j = 0; j < herdSize; j++) {
                for (int k = 0; k < collectionFeatures.length; k++) {
                    fitness[j] += buffaloPositions[j][k] * collectionFeatures[k][j] *
goals[k];
                }
            }

            // Update best buffalo position
```

```
            int bestBuffalo = 0;
            for (int j = 1; j < herdSize; j++) {
                if (fitness[j] > fitness[bestBuffalo]) {
                    bestBuffalo = j;
                }
            }

            // Print best buffalo position (e.g., optimal curation strategy)
            System.out.println("Best buffalo position: " +
Arrays.toString(buffaloPositions[bestBuffalo]));
        }
    }

}
```

This code demonstrates a basic ABO algorithm for digital curation and museum studies, using museum collection data features and curation goals as a fitness function. The ABO algorithm iteratively updates the positions of "buffaloes" (agents) in a virtual landscape, where the fitness function represents the quality of the curation strategy. In this example, the best buffalo position represents the optimal curation strategy for the digital collection. Note that this is a simplified example and can be extended to more complex digital curation and museum studies scenarios.

4.7. Digital Editions and Publishing

Another crucial area of DH where ABO can be of great benefit is in digital editions and publishing. In this area, ABO is useful in optimizing digital edition workflows and publication processes, identifying optimal formats and platforms for digital publications in addition to optimizing digital publishing workflows and distribution. Other areas that ABO can be help revolutionize in digital editions and publishing are:

1. **Optimizing digital text representation**: ABO can improve digital text representation, enabling efficient storage, retrieval, and display of text data.
2. **Enhancing digital edition design:** ABO can aid in creating optimal digital editions, considering layout, typography, and navigation, to enhance reader experience.
3. **Streamlining publishing workflows**: ABO can optimize publishing workflows, automating tasks, and improving collaboration among authors, editors and publishers.
4. **Maximizing digital publishing platforms**: ABO can optimize digital publishing platforms, ensuring efficient and effective publication and dissemination of digital content.
5. **E-book optimization**: ABO can optimize e-book formatting, ensuring optimal reading experiences on various devices and platforms.
6. **Digital scholarship enablement**: ABO can aid in optimizing digital scholarship, enabling researchers to analyze and visualize complex data and texts.
7. **Digital pedagogy enhancement**: ABO can optimize digital pedagogy, helping educators create effective digital learning materials and experiences.
8. **Accessibility improvement**: ABO can help optimize digital publications for accessibility, ensuring equal access for all readers, regardless of abilities.
9. **Digital rights management optimization**: ABO can aid in optimizing digital rights management, ensuring secure and efficient management of digital content and intellectual property.
10. **Digital preservation:** ABO can help optimize digital preservation strategies, ensuring long-term accessibility and integrity of digital publications [22].

Concomitantly, through the application of ABO in digital editions and publishing, professionals can create optimized digital texts, enhance reader experiences, and improve publishing workflows, among other benefits. Java implementation of the ABO algorithm in this area is presented:

Listing 7: ABO's Java Code Listing for Digital Editions

```
import java.util.*;
public class DigitalEditionsABO {
    public static void main(String[] args) {
        // Digital edition data (e.g., texts, images, metadata)
        String[] editionData = {"Text1", "Image1", "Metadata1"};
        int[][] editionFeatures = {
            {1, 2, 3},  // Text1 features (e.g., format, size, complexity)
            {2, 3, 1},  // Image1 features
            {3, 1, 2}   // Metadata1 features
        };
        // Publishing goals (e.g., readability, accessibility, discoverability)
```

```
        double[] goals = {0.9, 0.8, 0.7};

        // ABO parameters
        int herdSize = 20;
        int iterations = 100;
        double stepSize = 0.1;

        // Initialize buffalo positions and velocities
        double[][] buffaloPositions = new double[herdSize][editionFeatures[0].length];
        double[][] buffaloVelocities = new double[herdSize][editionFeatures[0].length];

        // ABO algorithm
        for (int i = 0; i < iterations; i++) {
            // Update buffalo positions and velocities
            for (int j = 0; j < herdSize; j++) {
                for (int k = 0; k < editionFeatures[0].length; k++) {
                    buffaloPositions[j][k] += buffaloVelocities[j][k];
                    buffaloVelocities[j][k] += stepSize * (Math.random() - 0.5);
                }
            }

            // Calculate fitness (e.g., edition quality)
            double[] fitness = new double[herdSize];
            for (int j = 0; j < herdSize; j++) {
                for (int k = 0; k < editionFeatures.length; k++) {
                    fitness[j] += buffaloPositions[j][k] * editionFeatures[k][j] *
goals[k];
                }
            }
            // Update best buffalo position
            int bestBuffalo = 0;
            for (int j = 1; j < herdSize; j++) {
                if (fitness[j] > fitness[bestBuffalo]) {
                    bestBuffalo = j;
                }
            }
            // Print best buffalo position (e.g., optimal edition strategy)
            System.out.println("Best buffalo position: " +
Arrays.toString(buffaloPositions[bestBuffalo]));
        }
    }

}
```

As in the other implementations, this code demonstrates ABO algorithm for digital editions and publishing with digital edition, data features and publishing goals as the fitness function. The ABO algorithm iteratively updates the positions of "buffaloes" (agents) in a virtual landscape, where the fitness function represents the quality of the edition strategy. In this example, the best buffalo position represents the optimal edition strategy for the digital publication.

4.8. Social Media and Online Communities

The current wave among the elite in the 21st century where DH is dominant is the social media and online communities across the globe. Here, the ABO can be readily available in optimizing social media analysis for cultural and social trends, identifying optimal strategies for online community engagement and participation in addition to optimizing social media content creation and curation. Other relevant application areas of ABO in social media and online communities include:

1. **Optimizing online engagement**: ABO can help optimize online engagement, identifying the most effective strategies for increasing user interaction and participation.
2. **Community building**: ABO can aid in building and optimizing online communities, ensuring effective moderation, user retention, and community growth.
3. **Social media management**: ABO can optimize social media management, automating tasks, and improving content creation and curation.
4. **Influencer identification**: ABO can help identify optimal influencers for online campaigns, ensuring maximum reach and impact.
5. **Content optimization**: ABO can optimize content creation and dissemination, ensuring effective messaging, timing, and audience targeting.
6. **Online reputation management**: ABO can aid in optimizing online reputation management, ensuring effective monitoring, response, and crisis management.

7. **User experience enhancement**: ABO can optimize user experience, ensuring intuitive interfaces, effective navigation, and personalized interactions.
8. **Social media analytics**: ABO can optimize social media analytics, providing actionable insights into user behaviour, engagement, and campaign effectiveness.
9. **Online marketing optimization**: ABO can aid in optimizing online marketing campaigns, ensuring effective targeting, ad placement, and conversion optimization.
10. **Community moderation**: ABO can help optimize community moderation, ensuring effective management of online discussions, user feedback, and conflict resolution.

Judging from the above discussions, it may be safe to say that applying ABO in social media and online communities, DH research community can create optimized online experiences, enhance user engagement, and improve community management, among other benefits [23]. The ABO implementation in Java programming language of social media and online communities is presented:

Listing 8. The ABO implementation in Java programming language of social media and online communities

```
import java.util.*;
public class SocialMediaABO {
    public static void main(String[] args) {
        // Social media data (e.g., posts, comments, likes)
        String[] socialMediaData = {"Post1", "Comment1", "Like1"};
        int[][] socialMediaFeatures = {
            {1, 2, 3},  // Post1 features (e.g., engagement, relevance, sentiment)
            {2, 3, 1},  // Comment1 features
            {3, 1, 2}   // Like1 features
        };

        // Community goals (e.g., engagement, growth, sentiment)
        double[] goals = {0.8, 0.7, 0.9};

        // ABO parameters
        int herdSize = 20;
        int iterations = 100;
        double stepSize = 0.1;

        // Initialize buffalo positions and velocities
        double[][] buffaloPositions = new
double[herdSize][socialMediaFeatures[0].length];
        double[][] buffaloVelocities = new
double[herdSize][socialMediaFeatures[0].length];

        // ABO algorithm
        for (int i = 0; i < iterations; i++) {
            // Update buffalo positions and velocities
            for (int j = 0; j < herdSize; j++) {
                for (int k = 0; k < socialMediaFeatures[0].length; k++) {
                    buffaloPositions[j][k] += buffaloVelocities[j][k];
                    buffaloVelocities[j][k] += stepSize * (Math.random() - 0.5);
                }
            }
            // Calculate fitness (e.g., community quality)
            double[] fitness = new double[herdSize];
            for (int j = 0; j < herdSize; j++) {
                for (int k = 0; k < socialMediaFeatures.length; k++) {
                    fitness[j] += buffaloPositions[j][k] * socialMediaFeatures[k][j] *
goals[k];
                }
            }
            // Update best buffalo position
            int bestBuffalo = 0;
            for (int j = 1; j < herdSize; j++) {
                if (fitness[j] > fitness[bestBuffalo]) {
                    bestBuffalo = j;
                }
            }
            // Print best buffalo position (e.g., optimal community strategy)
            System.out.println("Best buffalo position: " +
Arrays.toString(buffaloPositions[bestBuffalo]));
        }
```

```
        }
    }
```

This code is a basic demonstration of ABO algorithm for social media and online communities and uses social media data features and community goals as a fitness function. The ABO algorithm iteratively updates the positions of "buffaloes" (agents) in a virtual landscape, where the fitness function represents the quality of the community strategy. The best buffalo location in this code represents the optimal community strategy for the social media platform.

5.0 RESULTS AND DISCUSSION

The diverse results of Java implementation obtained from the exhaustive implementations, research on ABO has shown promising outcomes in various applications. Prominent among these are the fact that the ABO obtained very fast convergence. As such, hybridizing ABO with other algorithms will lead to greater efficiency and effectiveness. Moreover, the Java implementations showed improved problem-solving of complex problems like text analysis, digital scholarship and digital pedagogy. This is consistent with previous ABO applications to solve complex problems, such as the travelling salesman's problems, collision-avoidance in electric fish etc [9].

Moreover, ABO effectiveness in language relationship analysis is remarkable. ABO is quite useful in analyzing language relationships and migration patterns in digital humanities research. Again, the application of the ABO algorithm in DH reveals the algorithm's innate strength in optimizing digital humanities workflows: ABO could help optimize workflows in digital humanities projects, such as data cleaning, processing and visualization.

Further, ABO in DH research shows strength in term frequency analysis: The ABO Java code can identify the most frequent terms in the input text. In addition, effectiveness in optimized term extraction is strong. The ABO algorithm can optimize the extraction of relevant terms from the text.

As a result of the above, it may be safe to say that the ABO algorithm can effectively identify the most frequent terms in the input text of any size, including in the voluminous corpus of data. Lastly, the ABO can be applied to various text analysis tasks, such as keyword extraction, topic modelling, or sentiment analysis.

6.0 CONCLUSIONS

As can be seen from the foregoing discussions, the ABO has tremendous benefits when applied to DH. These applications demonstrate the potential of ABO to enhance and optimize various aspects of Digital Humanities, from text analysis and digital pedagogy to cultural analytics and digital preservation [24].

The application of the ABO) to DH has led to several conclusions:

1. Improved efficiency: ABO optimizes digital research methods, streamlining processes and reducing time consumption.
2. Enhanced analysis: ABO enables more accurate and in-depth analysis of complex data and texts.
3. Better digital curation: ABO ensures efficient storage, retrieval, and preservation of digital cultural heritage.
4. Effective digital pedagogy: ABO creates engaging and interactive digital learning materials and experiences.
5. Increased accessibility: ABO optimizes digital exhibits and online communities, ensuring equal access for all users.
6. Cultural insights: ABO provides new perspectives and understanding of cultural trends and patterns.
7. Digital preservation: ABO ensures long-term accessibility and integrity of digital cultural heritage.
8. Collaboration and community building: ABO facilitates collaboration and community engagement among researchers, scholars, and cultural institutions.
9. Innovative digital scholarship: ABO enables new forms of digital scholarship, pushing the boundaries of traditional research methods.
10. Interdisciplinary approaches: ABO fosters interdisciplinary collaboration, combining humanities, computer science, and social sciences.

In all, ABO's effectiveness to DH discipline is due to its noise tolerance, insensitivity to initial conditions, resilience to parameter variations, flexibility in handling diverse data types and adaptability to DH landscapes as well as easy integration with existing DH tools and frameworks. ABO is extremely robust against noisy or missing data, which is common in DH datasets. The algorithm can be easily integrated with existing DH tools and frameworks, thus facilitating seamless incorporation into existing workflows.

Finally, it is our belief that this study has been able to contribute to the development of innovative optimization techniques for DH data analysis, enhancing our understanding of human culture, education and history. The findings of this research provide useful insights for scholars, researchers, and practitioners in DH, optimization and cultural heritage preservation. By applying ABO to DH using appropriate high-level programming language such as Java (as used in this study), C, C++, C#, Python etc., researchers and scholars can harness the power of optimization techniques to enhance digital research, curation, and pedagogy, ultimately enriching our understanding and engagement with cultural heritage.

Similarly, leveraging the strengths of ABO, researchers and scholars can develop innovative solutions and improve the overall quality and impact of their work. Therefore, it is recommended that the impact of other swarm optimization algorithms should be explored in DH scholarship and research. Moreover, future research should focus on conducting thorough evaluations of ABO's performance in various DH applications such as the preservation of cultural heritage artifacts, cultural data analysis and interpretation as well as exploring the challenges of using ABO in digital humanities.

ACKNOWLEDGEMENTS

The author acknowledges the support received from the Management of Anchor University Lagos and specifically the Institute of Digital Humanities, Anchor University Lagos and the Department of Computing, Faculty of Science, Anchor University Lagos, Nigeria.

AUTHORS CONTRIBUTION

J,B. Odili is the sole writer of this manuscript from the conceptualization; formal analysis; methodology; data curation; and manuscript writing

CONFLICT OF INTEREST

No conflict of interest exists in the publication of this paper.

REFERENCES

[1] A. Shanableh, S. Aderibigbe, M. Omar, and A. Shabib, "Challenges and opportunities of multi-disciplinary, inter-disciplinary and trans-disciplinary research," *Higher Education in the Arab World: Research and Development*, pp. 311-325, 2022.

[2] A. Zuberogoitia, M. Madinabeitia, and D. Greenwood, "Transdisciplinary humanities for social transformation: The Global Digital Humanities degree at Mondragon University AS Fabrik," *Learning and Teaching*, vol. 17, pp. 1-23, 2024.

[3] K. Lieberknecht, H. Houser, A. Rabinowitz, S. A. Pierce, L. Rodríguez, F. Leite, *et al.*, "Creating meeting grounds for transdisciplinary climate research: the role of humanities and social sciences in grand challenges," *Interdisciplinary Science Reviews*, vol. 48, pp. 585-607, 2023.

[4] A. K. Dubey, A. Mukhopadhyay, and B. Basu, *Interdisciplinary Engineering Sciences: Concepts and Applications to Materials Science*: CRC Press, 2020.

[5] H. Singh and A. Singh, "ChatGPT: Systematic review, applications, and agenda for multidisciplinary research," *Journal of Chinese Economic and Business Studies*, vol. 21, pp. 193-212, 2023.

[6] M. G. Lawrence, S. Williams, P. Nanz, and O. Renn, "Characteristics, potentials, and challenges of transdisciplinary research," *One Earth*, vol. 5, pp. 44-61, 2022.

[7] F. Silva, F. Coward, K. Davies, S. Elliott, E. Jenkins, A. C. Newton, *et al.*, "Developing transdisciplinary approaches to sustainability challenges: The need to model socio-environmental systems in the longue durée," *Sustainability*, vol. 14, p. 10234, 2022.

[8] A. Călinescu, "The Future of Cultural Studies and the Humanities," *New Literaria*, vol. 3, pp. 37-52, 2022.

[9] J. B. Odili, A. Noraziah, B. Alkazemi, and M. Zarina, "Stochastic process and tutorial of the African buffalo optimization," *Scientific reports*, vol. 12, p. 17319, 2022.

[10] J. B. Odili, A. Noraziah, and R. M. Sidek, "Swarm intelligence algorithms' solutions to the travelling salesman's problem," in *IOP Conference Series: Materials Science and Engineering*, 2020, p. 012030.

[11] J. B. Odili and J. O. Fatokun, "The mathematical model, implementation and the parameter-tuning of the African buffalo optimization algorithm," in *2020 International Conference in Mathematics, Computer Engineering and Computer Science (ICMCECS)*, 2020, pp. 1-8.

[12] J. Luhmann and M. Burghardt, "Digital humanities—A discipline in its own right? An analysis of the role and position of digital humanities in the academic landscape," *Journal of the Association for Information Science and Technology*, vol. 73, pp. 148-171, 2022.

[13] R. Frost Davis, M. K. Gold, and K. D. Harris, "Network+ Publication+ Ecosystem: Curating Digital Pedagogy, Fostering Community," *Interdisciplinary Digital Engagement in Arts & Humanities (IDEAH)*, 2023.

[14] A. Hawkins, "Archives, linked data and the digital humanities: increasing access to digitised and born-digital archives via the semantic web," *Archival Science*, vol. 22, pp. 319-344, 2022.

[15] T. M. Shami, A. A. El-Saleh, M. Alswaitti, Q. Al-Tashi, M. A. Summakieh, and S. Mirjalili, "Particle swarm optimization: A comprehensive survey," *Ieee Access,* vol. 10, pp. 10031-10061, 2022.

[16] T. Giannini and J. P. Bowen, "Museums and digital culture: From reality to digitality in the age of COVID-19," *Heritage,* vol. 5, pp. 192-214, 2022.

[17] P. Verma, A. Verma, and S. Pal, "An approach for extractive text summarization using fuzzy evolutionary and clustering algorithms," *Applied Soft Computing,* vol. 120, p. 108670, 2022.

[18] R. Chatterjee, A. Bandyopadhyay, S. Chakraborty, and S. Dutta, "Digital education: the basics with slant to digital pedagogy-an overview," *Digital learning based education: transcending physical barriers,* pp. 63-80, 2023.

[19] S. Krämer, "Should we really 'hermeneutise'the Digital Humanities? A plea for the epistemic productivity of a 'cultural technique of flattening'in the Humanities," *Journal of Cultural Analytics,* vol. 7, 2023.

[20] E. Moraitou, Y. Christodoulou, and G. Caridakis, "Semantic models and services for conservation and restoration of cultural heritage: A comprehensive survey," *Semantic Web,* vol. 14, pp. 261-291, 2023.

[21] F. Quin, D. Weyns, M. Galster, and C. C. Silva, "A/B testing: a systematic literature review," *Journal of Systems and Software,* p. 112011, 2024.

[22] L. Tsipi, D. Vouyioukas, G. Loumos, A. Kargas, and D. Varoutas, "Digital Repository as a Service (D-RaaS): Enhancing Access and Preservation of Cultural Heritage Artifacts," *Heritage,* vol. 6, pp. 6881-6900, 2023.

[23] K. Skelton, R. Evans, and J. LaChenaye, "Hidden communities of practice in social media groups: Mixed methods study," *JMIR pediatrics and parenting,* vol. 3, p. e14355, 2020.

[24] J. Odili and C. N. Mabude, "African Buffalo Optimization to African Digital Culture and African Humanities," *Multilingual African Digital Semiotics and E-lit Journal (MADSEJ),* vol. 2, pp. 1-14, 2024.