Cognition-Based Document Matching Within the Chatbot Modeling Framework

Sunu Jatmika[1], Syaad Patmanthara[2,*], Aji Prasetya Wibawa[3], Fachrul Kurniawan[4]

[1,2,3]*Department of Electrical Engineering and Informatics, Faculty of Engineering Universitas Negeri Malang, Indonesia*

[4]*Informatics Engineering, Universitas Islam Negeri Maulana Malik Ibrahim Malang, Indonesia*

(Received: February 1, 2024; Revised: March 10, 2024; Accepted: April 20, 2024; Available online: May 31, 2024)

Abstract

The aim of the study is to examine cognitive methods for document matching in a chatbot modeling framework by utilizing Euclidean Distance, Cosine Similarity, and BERT methodologies. Five primary indications are used to carry out evaluation in testing: document matching accuracy, document matching execution time, document search efficiency, consistency of document matching results, and the quality of the document representation in the matrix. Document matching accuracy is evaluated by precision; document matching execution time is measured from the beginning to the end of the document matching process; document search efficiency is measured through evaluation of execution time and matching accuracy; the consistency of document matching results is assessed by comparing method results when tested against the same or similar queries and the quality of document representation is assessed based on the method's ability to represent documents in a matrix or vector. The test findings offer a comprehensive understanding of how well the three approaches operate and exhibit their capacity to address the unique requirements of chatbot users. These results may contribute to the advancement of language technology applications, making it possible for chatbots to deliver pertinent information more rapidly and precisely. There are 1,755 labeled question samples in the dataset, which were split up into two sets: 60% for training (1,053 pieces), and 40% for testing (702 samples) to evaluate the model's performance. The test results show the accuracy of the three methods based on five measured evaluation indications, namely Euclidean Distance 0,45%, Cosine similarity 0,59%, and BERT 0,91%. By comprehending the benefits and drawbacks of each approach, this research strengthens contributions to the growth of chatbot systems to better serve user demands and opens the door for the creation of more complex human-machine interaction solutions.

Keywords: Document Matching, Chatbot Models, Evaluation Method, Method Performance, AI Chatbot

1. Introduction

Chatbots have become an integral component in various aspects of human interaction with computer systems [1]. The rapid growth in the use of chatbots underscores the importance of efficiency and accuracy in document matching to provide users with relevant and timely answers [2]. Therefore, the concept of cognition-based document matching has emerged as a promising approach within the chatbot modeling framework [3]. By utilizing cognitive principles, such as understanding natural language and semantic analysis [4], in the document matching process, it is expected to be able to improve the quality of interaction between users and chatbots. Cognition-based document matching improves chatbot performance, several things that need to be considered include: Understanding context, document-based cognition allows chatbots to better understand user context and goals. By understanding the content and topics in documents, chatbots can provide more relevant and appropriate responses. Example: The document "Aromatherapy for Relaxation" provides information about the benefits of aromatherapy in SPA. The chatbot uses this information to understand users' questions regarding aromatherapy and provide relevant answers.; Increased relevance, by using information from cognition-based documents, chatbots can tailor their responses to user needs and preferences. This helps increase the relevance and usefulness of the responses provided by the chatbot. Example: The document "Holistic Approaches to Wellness Retreats" describes a holistic approach to spas. The chatbot adjusts its recommendations based on this information to meet the user's preferences.; Entity and concept recognition, cognition-based documents contain information about entities and concepts relevant to a particular domain. By analyzing these documents, chatbots can recognize entities and concepts that are important for interacting with users. Example: The document "Innovative SPA

[*]**Corresponding author: Syaad Patmanthara (syaad.ft@um.ac.id)**
DOI: https://doi.org/10.47738/jads.v5i2.209

 ISSN 2723-6471

Technologies for Skin Rejuvenation" introduces skin care technologies. The chatbot uses this information to recognize terms and concepts related to spa technology in its interactions with users. The use of chatbots is growing in all fields, especially for service companies that are very helpful in the fields of marketing, decision-making, education, health, and entertainment [5]. Chatbot is the application of artificial intelligence to mimic human conversation [6].

The role of chatbots supported by artificial intelligence is very helpful for customers in getting information in real-time [7]. The development of artificial intelligence (AI) has played a central role in advancing chatbot technology significantly [8]. The integration of AI into chatbot development has resulted in tremendous progress in terms of human-machine interactions that are increasingly similar to interactions between humans [9],[10]. The use of natural language processing (NLP) techniques in chatbots has allowed the system to understand and respond to user requests and questions in a more natural and contextual way [11]. An illustration of utilizing the Long short-term memory (LSTM) algorithm in a chatbot to respond to inquiries about training registration details at BLK Surabaya [12],[13] [14]. Financial services chatbot based on BERT method [15]. AI technology has also enabled the development of more adaptive and personalized chatbots [16]. Overall, AI developments have opened up exciting new opportunities in the development of chatbot technologies, enabling them to become catalysts for more productive, intuitive, and immersive human-machine interactions [17]. By continuing to leverage innovation in the field of artificial intelligence, chatbots have the potential to continue to grow and provide increasingly rich and meaningful experiences for their users [18].

An important part of chatbots is likeness matching [19] documents relating to the provision of relevant and accurate information to users. By leveraging artificial intelligence techniques such as NLP and semantic analysis, chatbots can effectively match user requests or questions with the most relevant documents or content from existing data sources [20]. By training a model that understands patterns in relevant text data, chatbots can identify the information that best fits a user's request [21]. The cognitive approach to document matching in chatbots represents an interesting evolution in the development of human-machine interaction [22]. By incorporating cognitive elements such as context understanding, in-depth semantic analysis, and the ability to recognize user intentions and emotions, chatbots become better able to understand the true purpose of a user's question or request [23] [24]. Through the application of more sophisticated natural language processing techniques, chatbots are able to distinguish content that is similar in terms of concepts and meanings, not just in terms of words or phrases for document matching in chatbots represents an interesting evolution in the development of human-machine interaction [23]. By incorporating cognitive elements such as context understanding [25], in-depth semantic analysis, and the ability to recognize user intentions and emotions, chatbots become better able to understand the true purpose of a user's question or request [26]. Through the application of more sophisticated natural language processing techniques, chatbots can distinguish content that is similar in terms of concepts and meanings, not just words or phrases [27].

2. The Proposed Method

2.1. Documenting Matching Method

This method aims to identify the most relevant documents to a given query [28]. The primary objective is to provide users with information that best meets their needs so they may get the answers or solutions they need [29]. It involves gathering, examining, and contrasting text from several documents in order to identify the ones that most closely correspond to the user's query or request.

2.2. Chatbot Models

The primary objective of implementing a chatbot paradigm is to efficiently and successfully enable human-system interaction [29]. The goal of the chatbot model is to enhance the user experience while engaging with the system by responding to queries or instructions from users in a contextual and appropriate manner [30]. Chatbots, which utilize artificial intelligence and natural language processing, are made to respond to queries, offer information, and carry out other duties in a manner that mimics human communication, all with the goal of improving the user experience [31].

2.3. Evaluation Method

Evaluation techniques are employed to quantify and contrast the effectiveness of various document-matching strategies [32]. The goal is to gain an in-depth understanding of the strengths and weaknesses of each method, thereby enabling

the selection of the method that best suits specific needs and goals. This entails assessing how effectively a technique can yield answers that match a given query using suitable evaluation measures, such as accuracy, recall, precision, or F1-score.

2.4. Method Performance

Accurate and pertinent document-matching results for a given query are the aim of method performance analysis [33]. The evaluation of method performance is conducted by considering many factors, including accuracy, speed, and efficiency. The objective is to enhance the system's capability to furnish users with pertinent and valuable information [34]. In order to ensure that the results have a high information value for the user, it involves the creation and use of strategies and algorithms that can maximize the match between documents and queries.

2.5. Application Development

The goal of application development is to provide a complex and dependable language technology system to facilitate communication between computers and people [35]. The program is made to satisfy user demands by offering chatbot services that are dependable, responsive, and capable of giving users the information or help they want in a timely and correct manner. To improve user experience and satisfy a range of objectives, the produced apps must be able to incorporate several language technologies, such as speech recognition, natural language production, and natural language processing.

3. Method

In this study the stages were carried out as shown in figure 1 The Euclidean Distance, Cosine Similarity, and BERT methods were chosen because each has advantages in understanding and matching documents in a chatbot context. Euclidean Distance is simple and fast, while Cosine Similarity is more adaptive to variations in document length. BERT, with its deep understanding of context, is suited to dealing with semantic complexity. By utilizing these three methods, research can provide a comprehensive and adaptive solution for document matching in chatbots.

The three methods were chosen because of their ability to improve document matching in chatbots. Euclidean Distance provides fast results, Cosine Similarity is adaptive to document variations, and BERT understands context well. By combining the three, the research aims to provide more relevant and responsive answers to chatbot users.

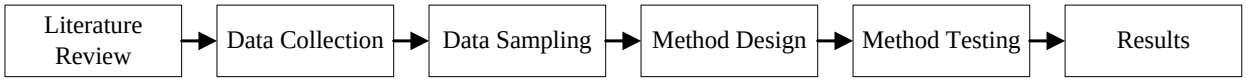

Figure 1. Research Stages

This literature review was carried out by collecting and reviewing research articles related to the document matching system to explore the latest methods that have been applied [36]. Chatbots use text mining to understand and respond appropriately to user texts [37]. The purpose of exploring this method is to gain knowledge about existing document-matching methods [38], understand the problems that exist in the document matching system, and get gaps from the research to be carried out. In addition to reviewing research articles, at this stage, a theoretical study of several literature related to the document matching system was also carried out to strengthen the theoretical basis that would be used to support the research. Data collection techniques, explain what techniques are most suitable for various types of research, so that one can easily decide which technique can be applied and is most suitable for his research. There are 1,755 labeled question samples in the dataset, which were split up into two sets: 60% for training (1,053 pieces), and 40% for testing (702 samples) to evaluate the model's performance in SPA Wellness documents. SPA Wellness is part of local wisdom [39]. At this stage, the process of designing a solution method that will be used in research is carried out. The design of this solution method aims to improve the performance of the document matching method that has been used, considering some of the weaknesses that existed in the previous method. In this study, the design of the solution method was carried out using the Euclidean Distance (ED), Cosine Similarity (CS), and Bi-directional Encoder Representations from the Transformers (BERT) approach. The process of testing the methods used and comparing the test results. In this study, tests were carried out to measure the degree of similarity and matching of documents and measure the results of Execution Time.

 ISSN 2723-6471

3.1. Method Design

This research method is divided into 3 parts for explanation as shown in Figure 2, namely input, process, and output. Inputs in data collection include initial information such as text documents, conversation transcripts, user data, and conversation context. This data is used as raw material for analysis and modeling, to develop a cognition-based document matching algorithm in a chatbot framework. The process, where the data that has been collected, goes through the Filtering stage for cleaning, followed by text processing such as tokenization and stemming. Then, the data is converted into a vector with TF-IDF. These vectors were tested using Euclidean Distance and Cosine Similarity to measure similarity, as well as BERT for more in-depth analysis. Output, representation of document matching result using cognition-bases Euclidean Distance, Cosine Similarity, and Bert methods.

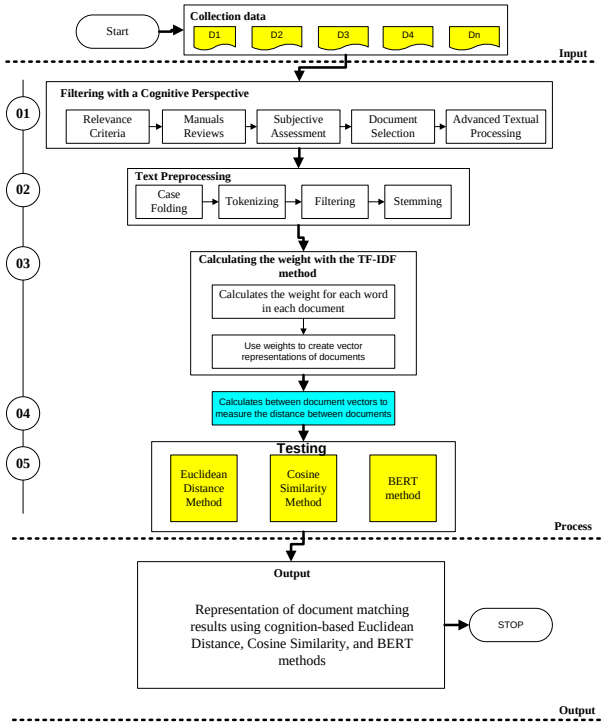

Figure 2. Method Design

The process begins by representing documents and user queries in a numerical vector space, where each dimension represents a specific feature of the text. Then, the Euclidean distance is calculated between the vector representations of the document and the query to determine how similar or different the document is to the query. The smaller the Euclidean distance between the document and query vectors, the more similar the two texts are.

Cosine Similarity is initially similar to Euclidean Distance, where documents and queries are represented in vector space. However, the difference lies in the similarity calculation. Cosine similarity measures the cosine of the angle between two vectors, reflecting the similarity in direction between them. The greater the cosine similarity value, the more similar the document is to the query

The use of BERT is more complex because it involves a pre-trained language model that has learned language representations from very large text data. First, document and query texts are fed into BERT to obtain a vector representation that describes the meaning and context of the text in depth. Next, the similarity between the document and query vectors is evaluated, similar to Cosine Similarity or other methods.

3.2. Sample Documents

In this survey, we conducted a comprehensive data collection study from the point of view of data management. Data collection mostly consists of data acquisition, data labeling, and refinement of existing data or models [40]. The goal of data acquisition is to find datasets that can be used to train machine learning models. Text data was collected from

 ISSN 2723-6471

various sources relevant to the topic of spa wellness. Table 1 shows examples of documents used in the SPA Wellness document.

Table 1. Simple Document SPA Wellness

Title	Description
Aromatherapy for Relaxation (d1)	This document explores the benefits of aromatherapy in promoting relaxation and stress relief. It discusses various essential oils, their therapeutic properties, and how they can enhance the spa wellness experience
Holistic Approaches to Wellness Retreats (d2)	his document delves into the concept of holistic wellness retreats, combining spa treatments with mindfulness practices, yoga, and nutritional guidance. It emphasizes the importance of a comprehensive approach to well-being
Innovative Spa Technologies for Skin Rejuvenation (d3)	Highlighting the latest advancements in spa technologies, this document explores innovative treatments and devices designed to rejuvenate and enhance skin health. It discusses the integration of technology into traditional spa practices
The Power of Hydrotherapy in Spa Treatments (d4)	This document focuses on the therapeutic benefits of hydrotherapy in spa settings. It covers various water-based treatments, such as hydro-massage, hot and cold plunge pools, and underwater jet massages, to promote relaxation and healing
Mindful Eating for Wellness (d5)	Exploring the connection between nutrition and wellness, this document discusses the implementation of mindful eating practices within spa environments. It emphasizes the role of balanced nutrition in supporting overall well-being.
Art and Creativity Workshops in Spa Retreats (d6)	This document explores the integration of art and creativity workshops into spa retreats to enhance the overall wellness experience. It discusses the therapeutic benefits of artistic expression in promoting relaxation and stress reduction
Nature-Inspired Spa Designs for Tranquility (d7)	Focusing on architectural and interior design aspects, this document showcases spa designs inspired by nature. It discusses how elements such as natural light, water features, and botanical aesthetics contribute to a tranquil spa atmosphere

The selection of examples of documents in Table 1 is because researchers have identified them to determine the main topics, they want to investigate in the field of SPA wellness, such as aromatherapy, spa technology, nutrition and healthy food, art and creativity in spas, etc. Next, researchers searched for documents relevant to these topics using trusted sources, such as scientific journals, books, research reports, or official spa and health websites and conducted interviews with SPA business people and customers. The search results of the documents found are then selected and assessed based on certain criteria, such as newness of information, diversity of topics, and quality of content. The most relevant and high-quality documents were then selected for inclusion in the research. To ensure the diversity of topics covered, researchers also considered the variety of topics that exist in the SPA wellness field, such as water therapy, nutrition, art, and technology. The selected documents can represent various aspects and dimensions of SPA wellness.

3.3. Calculating the Weight with the TF-IDF Method

Text preprocessing is the initial step in applying the TF-IDF technique to determine weights [41]. Case folding, tokenizing, filtering, and stemming are some of the steps that text processing entails in order to pick text data and make it more organized. Case folding This stage is almost always included when doing text preprocessing. Because the data held is not always structured and consistent in the use of capital letters, and numbers and eliminating empty characters in vector documents. Tokenizing is used to break sentences into words which are often called tokens. Filtering is used to extract important words from the token results. Common words that usually appear and have no meaning are called stop words. Stemming is a stage that is also needed to reduce the number of different indices of one data so that a word that has a suffix or prefix will return to its basic form.

Table 2. Example of Text Processing

Vector Document	Case Folding	Tokenizing	Filtering	Stemming
d_1= This document explores	this document explores aromatherapy's	['this', 'document', 'explores',	['document', 'explores', 'aromatherapy', "'s",	document explor aromatherapi benefit

 ISSN 2723-6471

aromatherapy's benefits for relaxation and stress relief in spa wellness, discussing essential oils and their therapeutic properties for enhanced relaxation.	benefits for relaxation and stress relief in spa wellness, discussing essential oils and their therapeutic properties for enhanced relaxation	'aromatherapy', "'s", 'benefits', 'for', 'relaxation', 'and', 'stress', 'relief', 'in', 'spa', 'wellness', ',', 'discussing', 'essential', 'oils', 'and', 'their', 'therapeutic', 'properties', 'for', 'enhanced', 'relaxation']	'benefits', 'relaxation', 'stress', 'relief', 'spa', 'wellness', ',', 'discussing', 'essential', 'oils', 'therapeutic', 'properties', 'enhanced', 'relaxation']	relax stress relief spa well , discuss essenti oil therapeut properti enhanc relax
d_2= This document explores innovative spa technologies designed to rejuvenate and enhance skin health, integrating technology into traditional spa practices	this document explores innovative spa technologies designed to rejuvenate and enhance skin health, integrating technology into traditional spa practices	['this', 'document', 'explores', 'innovative', 'spa', 'technologies', 'designed', 'to', 'rejuvenate', 'and', 'enhance', 'skin', 'health', ',', 'integrating', 'technology', 'into', 'traditional', 'spa', 'practices']	['document', 'explores', 'innovative', 'spa', 'technologies', 'designed', 'rejuvenate', 'enhance', 'skin', 'health', ',', 'integrating', 'technology', 'traditional', 'spa', 'practices']	document explor innov spa technolog design rejuven enhanc skin health , integr technolog tradit spa practic
q= Spa wellness	spa wellness	['spa', 'wellness']	['spa', 'wellness']	spa wellness

After the text processing process has been carried out, the next step is to calculate the Term Frequency (TF) of the appearance of each word (term). From the illustration in Table 2, the TF is obtained as in Figure 3.

	aromatherapi	benefit	design	discuss	document	enhanc	essenti	explor	health	innov	...	relax	relief	skin	spa	stress	technolog	therapeut	tradit	well	wellness
d1	1	1	0	1	1	1	1	1	0	0	...	2	1	0	1	1	0	1	0	1	0
d2	0	0	1	0	1	1	0	1	1	1	...	0	0	1	2	0	2	0	1	0	0
q	0	0	0	0	0	0	0	0	0	0	...	0	0	0	1	0	0	0	0	0	1

Figure 3. TF Value.

After the TF value is generated, the next step is to calculate the Inverse Document Frequency (IDF) with the formula: $\log \frac{n}{df}$, where n is the number of documents (d1, d2,…,di), and document frequency (df) is the number of frequency tokens appearing in all documents (d1, d2). For example, the "relax" token d1=2, d2=0 then the value df=1+0 so that df = $\log \frac{2}{1} = 0{,}3010$; the "explor" token d1=1, d2=1 then the value df=1+1 so that idf = $\log \frac{2}{2} = 0$. As a note, if the number of tokens appears more than 1 then only 1 is taken as in the "relax" token. The next step is to find the weight value for each token using the formula W= TF x IDF, An example of finding the weight value of the "relax" token is $W_{relax(d1)}$= 2 x 0,3010 = 0,6020, and $W_{relax(d2)}$=0x0,3010 = 0, and so on until all tokens get a weight value.

The results of the document-stemming process will become a dataset that will be calculated using TF-IDF [42],[43]. TF-IDF is used to analyze how important a word is in a document. TF, namely the higher the frequency value of the word appearing in a document, the higher the weight value for the word itself. Meanwhile, the IDF process is the opposite of the TF process. In IDF, the higher the frequency of occurrence of a word, the lower the weight value of the word itself will be. The next stage is weighting with the following formula:

$$W_{dt} = TF_{dt} \times IDF_t \quad (1)$$

d = document to d

t = the t-word of the keyword

W = the weight of the d-th document to the t-word

TF = number of words read

IDF = many documents contain the searched word

 ISSN 2723-6471

Term weighting is heavily influenced by the following: TF factor and IDF, where the Term Frequency (tf) factor is a factor in determining the weight of terms in documents, the number of values that appear in a word (term frequency) which is calculated in a weight to a word [44]. So, it can be concluded that the greater the term frequency in the document, the greater the document weight conformity value. While IDF is a reduction in the dominance of terms that often appear in various documents, this has an impact on the appearance of many terms in various documents and is considered a common term so its value is not important.

3.4. Euclidean Distance

Euclidean distance is a metric used to measure the distance or difference between two points in Euclidean space [45]. In two dimensions (for example, on a plane), the Euclidean distance between two points is calculated using the formula:

$$\mathrm{ED} = \sqrt{(\mathrm{x}_1 - \mathrm{y}_1)^2 + (\mathrm{x}_2 - \mathrm{y}_2)^2} \quad (2)$$

For more complex dimensions the Euclidean Distance between two points is calculated using the formula:

$$\mathrm{ED} = \sqrt{(\mathrm{x}_1 - \mathrm{y}_1)^2 + (\mathrm{x}_2 - \mathrm{y}_2)^2 + \ldots + (x_n - y_n)^2} \quad (3)$$

$(x_1, x_2, \ldots, x_n)$ represents the coordinates of the first point, $(y_1, y_2, \ldots, y_n)$ represents the coordinates of the second point. $(x_n - y_n)$ represents the difference in the $n - th$ coordinate of the two points.

When executing clustering, the Euclidean distance is considered, where if the interval between two points of the centroid is the same but the points are in the same or opposite direction, then they can also fall into the same cluster. Euclidean distance achieves very good results when data sets are organized into compact clusters. Despite the fact that Euclidean distances are exceptional in clustering, there is a drawback: when two entities do not have a uniform standard, their distances may differ compared to other pairs of entities that have similar attribute values.

Document matching begins with text processing, where the document is converted into a numerical representation. Next, the TF-IDF method is used to evaluate the importance of words in the document. Finally, the Euclidean distance is calculated between the document vectors to determine their similarity. This enables efficient and accurate document matching. For example, the value in Figure 2 is calculated using Euclidean distance with the help of the Python program as in Figure 4. The results are as in Figure 5.

```
ed=euclidean_distances(term_doc_matrix,term_doc_matrix)
df_ed=pd.DataFrame(ed,index=dataset.keys(),columns=dataset.keys())
df_ed
```

Figure 4. Example of a Euclidean Distance Calculation Program

From Figure 4 Euclidean Distance Calculation, it can be seen that document q has a closer Euclidean distance to document d_1 compared to document d_2. The Euclidean distance between documents q and d_1 is 4.242641, while the distance between documents q and d_2 is 4.123106. This shows that document q has a higher degree of similarity with document d_1 than with document d_2 based on the Euclidean Distance metric. However, these two distances are still quite significant, indicating that document q has quite a big difference from these two documents. This evaluation provides insight into the extent to which document q matches the documents in the dataset based on the difference in Euclidean distance. Although document q has closer similarities to document d1, further analysis is still needed to understand the deeper context and relevance of these documents.

	d1	d2	q
d1	0.000000	5.196152	4.242641
d2	5.196152	0.000000	4.123106
q	4.242641	4.123106	0.000000

Figure 5. Euclidean Distance Calculation

 ISSN 2723-6471

One of the limitations is when the data does not have a uniform distribution or when the features are not on the same scale. For example, if a document is high in dimensionality or has attributes that differ greatly in scale, Euclidean distance may no longer be an ideal metric because the difference in scale may dominate the contribution of the other attributes. Therefore, potential strategies to mitigate this limitation include normalizing the data before calculating the Euclidean distance, or using other methods that are more suitable for complex data, such as non-linear mapping methods or the use of kernels in classifier algorithms.

3.5. Cosine Similarity

After the data is weighted, the next step is the data will be equated with the cosine similarity method [46], [47]. Cosine Similarity is two vectors that have a measure of similarity in dimensional space, which is obtained from the product of the two vectors being compared. cosine of 0^0 is 1, then the similarity value of two vectors is said to be similar if the cosine similarity value is 1. Documents with known weights will be calculated using the cosine formula for the length of the vector. Cosine similarity data can be seen as follows:

$$CS = \cos \emptyset = \frac{A \,.B}{\|A\| \quad \|B\|} = \frac{\sum_{i=1}^{n}(A_i x\, B_i)}{\sqrt{\sum_{i=1}^{n}(Ai)^2} \quad x \quad \sqrt{\sum_{i=1}^{n}(Bi)^2}} \tag{4}$$

A = Vector A: which will be compared with the word similarity

B = Vector B: which will be compared with the word similarity

Ai = Term Weight (word)i in blocks Ai

Bi = Term Weight (word)i in blocks Bi

i = Number of terms (word) in documents/sentences

n = Vector sum

D = Documents

For example, the value in figure 2 is calculated using Cosine. Similar results as in figure 6.

	d1	d2	q
d1	1.000000	0.155952	0.074892
d2	0.155952	1.000000	0.154431
q	0.074892	0.154431	1.000000

Figure 6. Cosine Similarity Calculation

If calculated using cosine similarity for document matching, the result is as in Figure 5, It can be seen that the cosine similarity value between documents d1 and d2 is 0.156. This shows that the two documents have a fairly low level of similarity, which could mean that the content or topics discussed in the two documents are relatively different. However, when compared with the cosine similarity value between other documents, such as between documents d1 and q (0.075) or between documents d2 and q (0.154), the cosine similarity value between d1 and d2 is relatively higher. However, in general, the cosine similarity values recorded in the table indicate that there are quite significant differences between the vector representations of the documents in question. This can be caused by variations in topic or focus of discussion in each document so that the level of similarity between these documents varies.

3.6. Bidirectional Encoder Representation from Transformers (BERT)

BERT is a system by which Google's algorithm uses pattern recognition to better understand how human beings communicate so that it can return more relevant results for users [48]. The development of Indonesian NLP technology can be said to be minimal when compared to English NLP [49]. This also became a challenge when I wanted to develop an Indonesian NLP model [50].

 ISSN 2723-6471

On the other hand, the development of NLP in recent years has been very rapid, especially after the Attention-based approach was discovered [51]. This approach underlies the new deep learning method in NLP, namely Transformers, which degrades the traditional LSTM approach [52]. Now there are so many variants of the Transformers model architecture. One of the well-known is BERT [29]. Figure 7 is the BERT source code Python algorithm.

BERT vector representation of a text t can be obtained by taking the output of a particular layer in the BERT transformer architecture. For example, we can use the output of the final layer before the SoftMax layer for [CLS] tokens in text, which are generally used for classification tasks. When using the BERT model, the term [CLS] refers to a special token that is added at the beginning of each text input. Let's assume that BERT vector representation of the text t is given by $\mathrm{BERT}(\mathrm{t}) = \mathrm{v_t}$, with $\mathrm{v_t}$ is a vector that has the length d, where d is BERT vector representation dimensions

```
procedure BERT_Main(text, task_data):
        tokens = Tokenize(text)
        embeddings = Embed_Tokens(tokens)
        attended_embeddings = Self_Attention(embeddings)
        encoded_embeddings = Encoder_Layers(attended_embeddings)
        pretraining_loss = Compute_Pretraining_Loss(encoded_embeddings)
        Update_Parameters(pretraining_loss)
        task_specific_layers = Select_Task_Specific_Layers(encoded_embeddings)
        task_loss = Compute_Task_Loss(task_specific_layers, task_data)
        Update_Parameters(task_loss)
        new_task_data = Load_New_Task_Data()
        inference_result = Inference(encoded_embeddings, new_task_data)
        return inference_result
```

Figure 7. BERT algorithm

BERT is a language model that uses a transformer architecture to better understand the context of words in sentences. Compared with previous models, BERT is able to take into account word context from both directions in a sentence, resulting in richer and more accurate word representation. This makes BERT a breakthrough NLP and improves the performance of chatbot systems in understanding and responding to user questions better.

BERT makes a significant contribution to document matching within a chatbot framework by enabling deeper context understanding, complex language handling, better text representation, and adaptability to new contexts. With these capabilities, BERT enhances chatbots' ability to provide more relevant and meaningful answers to users, opening up new opportunities in the development of more sophisticated and effective chatbots in supporting more dynamic and contextual human-machine interactions.

4. Result and Discussion

Using an SPA wellness document, the test measures word similarity between 1500 to 2000 words, search accuracy (on a scale of 0 to 1, 1 being the best), execution duration in seconds, and Euclidean distance (on a similarity scale of 0 to 1, 1 being the most similar). Five indicators for testing include document matching accuracy, document matching execution time, efficiency in document search, consistency in document matching results, and quality of document representation in the matrix. Each indicator is tested with different variables as in Table 3.

Table 3. Indicator Variables

Indicator	Variable
Document matching accuracy	X1= percentage of documents relevant to the query. X2= precision, recall and f1-score values from matching results.
Document matching execution time	X3= time in seconds (document matching) X4= algorithm complexity
Efficiency in document search	X5= The number of steps required to search for relevant documents X6= Usage of system resources such as CPU, RAM, or storage space
Consistency in document-matching results	X7= The degree of variation in document matching results when the same query is used in repeated tests

 ISSN 2723-6471

	X8= The standard deviation value or the coefficient of variation of the matching results
Quality of document representation in the matrix	X9= Euclidean distance between document representation vectors X10= Cosine similarity between document representation vectors X11= Scores or values generated by the BERT model for each pair of documents and queries

BERT method: The highest search accuracy value was obtained from the BERT method, with a value of 0.86. Even though it shows high accuracy, BERT requires a longer execution time, namely 30 seconds. This is due to the greater complexity of the BERT model and a more complicated computational process. Cosine Similarity: The cosine similarity method provides fast results with an execution time of only 5 seconds. However, although the execution time is shorter, the search accuracy is slightly lower compared to BERT, with a value of 0.91. Cosine similarity is a simple method that measures the angular similarity between text representation vectors, but its ability to understand the context and meaning of the text may be limited. Euclidean distance shows potential as a fast method with an execution time of 7 seconds. However, the accuracy and relevance values are not given in this table, making it difficult to directly evaluate how good this method is at document matching.

Table 4 represents a similarity matrix, comparing different data points (labeled as "d1" through "d10") and a query ("q"). Each value indicates the similarity score between pairs of documents or between a document and the query.

Table 4. Cosine Similarity Measurement Results

VD	d1	d2	d3	d4	d5	d6	d7	d8	d9	d10	q
d1	1,000	0,192	0,181	0,192	0,238	0,000	0,489	0,238	0,072	0,000	0,527
d2	0,192	1,000	0,174	0,185	0,228	0,000	0,192	0,228	0,069	0,000	0,593
d3	0,181	0,174	1,000	0,174	0,215	0,000	0,181	0,215	0,065	0,000	0,195
d4	0,192	0,185	0,174	1,000	0,228	0,303	0,192	0,228	0,069	0,000	0,207
d5	0,238	0,228	0,215	0,228	1,000	0,000	0,238	0,282	0,085	0,000	0,256
d6	0,000	0,000	0,000	0,303	0,000	1,000	0,000	0,000	0,000	0,000	0,000
d7	0,489	0,192	0,181	0,192	0,238	0,000	1,000	0,238	0,072	0,000	0,527
d8	0,238	0,228	0,215	0,228	0,282	0,000	0,238	1,000	0,085	0,000	0,256
d9	0,072	0,069	0,065	0,069	0,085	0,000	0,072	0,085	1,000	0,000	0,077
d10	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	0,000	1,000	0,000
q	0,527	0,593	0,195	0,207	0,256	0,000	0,527	0,256	0,077	0,000	1,000

Table 5 compares different similarity measures for a set of data points (d1 to d10). Euclidean Distance quantifies the straight-line distance between two points in a multi-dimensional space. Cosine Similarity assesses the similarity between two vectors by measuring the cosine of the angle between them. Higher values indicate greater similarity. BERT (Bidirectional Encoder Representations from Transformers) captures semantic context from text, and its similarity values also indicate how similar the data points are. Each measure serves a different purpose: Euclidean distance focuses on geometric distance, cosine similarity on direction, and BERT on semantic context/

Table 5. Comparison Results of Three Methods

z	Euclidean Distance	Cosine Similarity	BERT
d1	0	0,527	0,746
d2	0	0,593	0,910
d3	0	0,195	0,533
d4	0,225	0,207	0,558
d5	0,255	0,256	0,558
d6	0,414	0,000	0,422
d7	0,414	0,527	0,747

 ISSN 2723-6471

d8	0,732	0,256	0,740
d9	0,871	0,077	0,610
d10	1	0,000	0,505

Figure 8. Graph comparison: The graph compares three different metrics: Euclidean Distance, Cosine Similarity, and BERT. Each metric is represented by a distinct line with different colors and patterns. The x-axis ranges from "d1" to "d10," while the y-axis values range from 0 to 1.2. Metric interpretation: Euclidean Distance (Orange Line), measures the straight-line distance between two points in a multi-dimensional space. Lower values indicate greater similarity. Cosine Similarity (Gray Line), measures the cosine of the angle between two vectors. Higher values indicate greater similarity. BERT (Blue Line), refers to Bidirectional Encoder Representations from Transformers. A powerful language model for natural language understanding. Its line represents some comparison or similarity score. Observations: Each metric's line fluctuates, showing variations across different "d" values. The graph provides insights into how these metrics perform relative to each other.

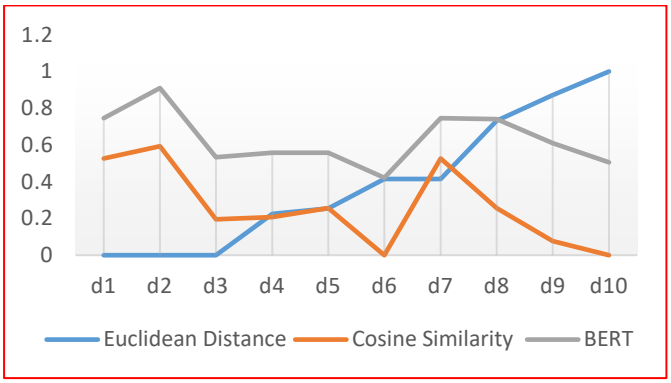

Figure 8. Test Graph

Table 6 BERT, Cosine Similarity, and Euclidean Distance are compared based on execution time, search accuracy, and Euclidean distance scores. BERT takes the longest execution time (29 units), but achieves the highest accuracy (0.91). Meanwhile, Cosine Similarity has the shortest execution time (5 units) but lower accuracy (0.59). The Euclidean Distance scores vary across methods.

Table 6. Measurement Results

Method	Execution Time	Search Accuracy	Euclidean Distance
BERT	29	0,91	0,79
Cosine Similarity	5	0,59	0,67
Euclidean Distance	7	0,45	0,74

The balance between accuracy and efficiency of execution time is crucial in developing chatbot systems. Although methods like BERT may offer high accuracy in document matching, the time required to perform this process may be longer compared to other methods such as cosine similarity. For example, when a chatbot user requests information that requires a quick document search, such as a venue's hours of operation, the priority may be more about speed of response than absolute accuracy. However, for questions that require highly relevant and in-depth answers, such as information about specific spa treatments, accuracy becomes a more important factor even though the document-matching process takes longer. Therefore, it is important to consider these two factors simultaneously and choose the document-matching method that best suits the specific needs of the chatbot application.

5. Conclusion

In the context of document matching utilizing the BERT, cosine similarity, and Euclidean distance approaches, the BERT method exhibits good search accuracy, with a value of 0.91, even with a longer execution time, specifically 29

seconds, according to the table 6. The cosine similarity approach, on the other hand, yields somewhat lower accuracy (0.91) but faster results (just 5 seconds to execute). While the presented Euclidean distance value of 0.74 suggests that this approach may be quick, more research is required to fully assess its correctness and applicability. Therefore, the selection of document-matching methods should consider the balance between accuracy and execution time efficiency based on the specific needs of the application.

In the future, further research could explore ways to integrate this document-matching method with other aspects of chatbot systems, such as emotion recognition or service personalization, to achieve a more advanced and more effective level of interaction between humans and machines

The selection of a document-matching method must be based on a deep understanding of the specific needs of the chatbot application being built. Some factors to consider include the type of information the user will be searching for, the level of accuracy required, the desired speed of response, and the complexity of the search. For example, if a chatbot application aims to provide general information and quick responses to frequently asked questions, a simpler document matching method such as cosine similarity may be more appropriate. However, if a chatbot application aims to provide highly relevant and in-depth answers to more complex questions, such as spa treatment recommendations tailored to individual needs, more advanced methods such as BERT may be more effective. Therefore, the choice of document matching method must be adjusted to the specific objectives and characteristics of the chatbot application being built.

6. Declarations

6.1. Author Contributions

Conceptualization: SJ and FK; Methodology: FK; Software: SJ; Validation: SJ, FK; Formal Analysis: SJ, FK; Investigation: SJ; Resources: FK; Data Curation: FK; Writing Original Draft Preparation: SJ and FK; Writing Review and Editing: FK and SJ; Visualization: SJ. All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.3. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

6.4. Institutional Review Board Statement

Not applicable.

6.5. Informed Consent Statement

Not applicable.

6.6. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Rapp, L. Curti, and A. Boldi, “The human side of human-chatbot interaction: A systematic literature review of ten years of research on text-based chatbots,” *Int. J. Hum. Comput. Stud.*, vol. 151, no. March, p. 102630, 2021.

[2] R. K. Ibrahim, S. R. M. Zeebaree, and K. F. S. Jacksi, “Survey on semantic similarity based on document clustering,” *Adv. Sci. Technol. Eng. Syst.*, vol. 4, no. 5, pp. 115–122, 2019, doi: 10.25046/aj040515.

[3] Mele, T. Russo Spena, V. Kaartemo, and M. L. Marzullo, “Smart nudging: How cognitive technologies enable choice architectures for value co-creation,” *J. Bus. Res.*, vol. 129, no. March 2019, pp. 949–960, 2021.

[4] A. Rajput, “Natural language processing, sentiment analysis, and clinical analytics,” *Innov. Heal. Informatics A Smart Healthc. Prim.*, pp. 79–97, 2019, doi: 10.1016/B978-0-12-819043-2.00003-4.

[5] E. Adamopoulou and L. Moussiades, “An Overview of Chatbot Technology,” in *IFIP Advances in Information and*

Communication Technology, 2020, vol. 584 IFIP, pp. 373–383. doi: 10.1007/978-3-030-49186-4_31.

[6] M. Dahiya, "A Tool of Conversation: Chatbot, International Journal of Computer Sciences and Engineering, Volume-5, Issue-5 E-ISSN: 2347-2693," *Int. J. Comput. Sci. Eng.*, pp. vol. 5, no. December, 2017, [Online]. Available: https://www.researchgate.net/publication/321864990_A_Tool_of_Conversation_Chatbot

[7] Y. Cheng and H. Jiang, "Customer–brand relationship in the era of artificial intelligence: understanding the role of chatbot marketing efforts," *J. Prod. Brand Manag.*, vol. 31, no. 2, pp. 252–264, 2022, doi: 10.1108/JPBM-05-2020-2907.

[8] F. Aslam, "The Impact of Artificial Intelligence on Chatbot Technology: A Study on the Current Advancements and Leading Innovations," *Eur. J. Technol.*, vol. 7, no. 3, pp. 62–72, 2023, doi: 10.47672/ejt.1561.

[9] E. S. Cross and R. Ramsey, "Mind Meets Machine: Towards a Cognitive Science of Human-Machine Interactions," *Trends Cogn. Sci.*, vol. 25, no. 3, pp. 200–212, 2021, doi: 10.1016/j.tics.2020.11.009.

[10] S. Hawanti and K. M. Zubaydulloevna, "AI chatbot-based learning: alleviating students' anxiety in English writing classroom," *Bull. Soc. Informatics Theory Appl.*, vol. 7, no. 2, pp. 182–192, 2023, doi: 10.31763/businta.v7i2.659.

[11] J. Purohit, A. Bagwe, R. Mehta, O. Mangaonkar, and E. George, "Natural language processing based jaro-the interviewing chatbot," *Proc. 3rd Int. Conf. Comput. Methodol. Commun. ICCMC 2019*, no. Iccmc, pp. 134–136, 2019, doi: 10.1109/ICCMC.2019.8819708.

[12] Y. S. H. Langgeng, E. I. Setiawan, S. Imron, and J. Santoso, "Long Short-Term Memory-Based Chatbot for Vocational Registration Information Services," *J. Appl. Data Sci.*, vol. 4, no. 4, pp. 414–430, 2023, doi: 10.47738/jads.v4i4.128.

[13] T. Lalwani, S. Bhalotia, A. Pal, S. Bisen, and V. Rathod, "Implementation of a Chat Bot System using AI and NLP," *Int. J. Innov. Res. Comput. Sci. Technol.*, vol. 6, no. 3, pp. 26–30, 2018, doi: 10.21276/ijircst.2018.6.3.2.

[14] A. Sulistyo, A. P. Wibawa, D. D. Prasetya, and F. A. Ahda, "LSTM-Based Machine Translation for Madurese-Indonesian," *J. Appl. Data Sci.*, vol. 4, no. 3, pp. 190–199, 2023, doi: 10.47738/jads.v4i3.113.

[15] S. Yu, Y. Chen, and H. Zaidi, "AVA: A Financial Service Chatbot Based on Deep Bidirectional Transformers," *Front. Appl. Math. Stat.*, vol. 7, no. August, pp. 1–11, 2021, doi: 10.3389/fams.2021.604842.

[16] J. Feine, S. Morana, and A. Maedche, "Leveraging Machine-Executable Descriptive Knowledge in Design Science Research – The Case of Designing Socially-Adaptive Chatbots," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2019, vol. 11491 LNCS, pp. 76–91. doi: 10.1007/978-3-030-19504-5_6.

[17] Y. K. Dwivedi *et al.*, "Artificial Intelligence (AI): Multidisciplinary perspectives on emerging challenges, opportunities, and agenda for research, practice and policy," *Int. J. Inf. Manage.*, vol. 57, no. July, pp. 0–1, 2021.

[18] S. B. Lee, "Chatbots and Communication: The Growing Role of Artificial Intelligence in Addressing and Shaping Customer Needs," *Bus. Commun. Res. Pract.*, vol. 3, no. 2, pp. 103–111, 2020, doi: 10.22682/bcrp.2020.3.2.103.

[19] J. Wang and Y. Dong, "Measurement of text similarity: A survey," *Inf.*, vol. 11, no. 9, pp. 1–17, 2020, doi: 10.3390/info11090421.

[20] Y. Kang, Z. Cai, C. W. Tan, Q. Huang, and H. Liu, "Natural language processing (NLP) in management research: A literature review," *J. Manag. Anal.*, vol. 7, no. 2, pp. 139–172, 2020, doi: 10.1080/23270012.2020.1756939.

[21] Z. Dai and J. Callan, "Deeper text understanding for IR with contextual neural language modeling," *SIGIR 2019 - Proc. 42nd Int. ACM SIGIR Conf. Res. Dev. Inf. Retr.*, pp. 985–988, 2019, doi: 10.1145/3331184.3331303.

[22] J. Chen, O. Agbodike, and L. Wang, "Memory-based deep neural attention (mDNA) for cognitive multi-turn response retrieval in task-oriented chatbots," *Appl. Sci.*, vol. 10, no. 17, pp. 1–11, 2020, doi: 10.3390/app10175819.

[23] Z. Drus and H. Khalid, "Sentiment analysis in social media and its application: Systematic literature review," *Procedia Comput. Sci.*, vol. 161, pp. 707–714, 2019, doi: 10.1016/j.procs.2019.11.174.

[24] F. T. Admojo, A. Lajis, and H. Nasir, "Systematic Literature Review on Ontology-based Indonesian Question Answering System," *Knowl. Eng. Data Sci.*, vol. 6, no. 2, p. 129, 2023, doi: 10.17977/um018v6i22023p129-144.

[25] W. N. Hidayat, S. Patmanthara, R. K. Sari, and T. A. Sutikno, "Cognitive ability improvement in learning resource development course through implementation of life-based learning models using LMS," *J. Phys. Conf. Ser.*, vol. 1193, no. 1, 2019, doi: 10.1088/1742-6596/1193/1/012034.

[26] Y. Yijun, G. Ruiyuan, L. Yu, L. Qiuxia, and X. Qiang, "MixDefense: A Defense-in-Depth Framework for Adversarial

Example Detection Based on Statistical and Semantic Analysis," *arXiv*, vol. 2, 2021, [Online]. Available: http://arxiv.org/abs/2104.

[27] P. Suta, X. Lan, B. Wu, P. Mongkolnam, and J. H. Chan, "An overview of machine learning in chatbots," *Int. J. Mech. Eng. Robot. Res.*, vol. 9, no. 4, pp. 502–510, 2020, doi: 10.18178/ijmerr.9.4.502-510.

[28] H. K. Azad and A. Deepak, "Query expansion techniques for information retrieval: A survey," *Inf. Process. Manag.*, vol. 56, no. 5, pp. 1698–1735, 2019, doi: 10.1016/j.ipm.2019.05.009.

[29] Vazquez-Ingelmo, F. J. Garcia-Penalvo, and R. Theron, "Information Dashboards and Tailoring Capabilities-A Systematic Literature Review," *IEEE Access*, vol. 7, pp. 109673–109688, 2019, doi: 10.1109/ACCESS.2019.2933472.

[30] K. F. Haugeland, A. Følstad, C. Taylor, and C. Alexander, "Understanding the user experience of customer service chatbots: An experimental study of chatbot interaction design," *Int. J. Hum. Comput. Stud.*, vol. 161, no. January, 2022, doi: 10.1016/j.ijhcs.2022.102788.

[31] P. D. Larasati, A. Irawan, S. Anwar, M. F. Mulya, M. A. Dewi, and I. Nurfatima, "Chatbot helpdesk design for digital customer service," *Appl. Eng. Technol.*, vol. 1, no. 3, pp. 138–145, 2022, doi: 10.31763/aet.v1i3.684.

[32] Z. Kastrati, A. S. Imran, and S. Y. Yayilgan, "The impact of deep learning on document classification using semantically rich representations," *Inf. Process. Manag.*, vol. 56, no. 5, pp. 1618–1632, 2019, doi: 10.1016/j.ipm.2019.05.003.

[33] Boukhari and M. N. Omri, "Approximate matching-based unsupervised document indexing approach: application to biomedical domain," *Scientometrics*, vol. 124, no. 2, pp. 903–924, 2020, doi: 10.1007/s11192-020-03474-w.

[34] D. Wahyono, K. Asfani, M. M. Mohamad, A. Aripriharta, A. P. Wibawa, and W. Wibisono, "New smart map for tourism using artificial intelligence," *EECCIS 2020 - 2020 10th Electr. Power, Electron. Commun. Control. Informatics Semin.*, pp. 213–216, 2020, doi: 10.1109/EECCIS49483.2020.9263435.

[35] B. Shneiderman, "Human-Centered Artificial Intelligence: Reliable, Safe & Trustworthy," *Int. J. Hum. Comput. Interact.*, vol. 36, no. 6, pp. 495–504, 2020, doi: 10.1080/10447318.2020.1741118.

[36] B. P. Manuaba, I. W. B. Sentana, I. N. G. A. Astawa, I. W. Suasnawa, and I. P. B. A. Pradnyana, "Social Media Mining with Fuzzy Text Matching: A Knowledge Extraction on Tourism After COVID-19 Pandemic," *Knowl. Eng. Data Sci.*, vol. 5, no. 2, p. 143, 2022, doi: 10.17977/um018v5i22022p143-149.

[37] F. Kurniawan and A. P. Wibawa, "Text Mining Techniques for Identify Islamophobic Conversation Language by Selecting Preprocessing Feature," *Res. Sq.*, pp. 1–9, 2021, [Online]. Available: https://www.researchsquare.com/article/rs-1105114/latest.pdf

[38] M. Yang, Q. Qu, Y. Shen, Z. Zhao, X. Chen, and C. Li, "An Effective Hybrid Learning Model for Real-Time Event Summarization," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 32, no. 10, pp. 4419–4431, 2021, doi: 10.1109/TNNLS.2020.3017747.

[39] S. Jatmika, S. Patmanthara, A. P. Wibawa, and F. Kurniawan, "THE MODEL OF LOCAL WISDOM FOR SMART WELLNESS TOURISM WITH OPTIMIZATION MULTILAYER," *J. Theor. Appl. Inf. Technol.*, vol. 102, no. 2, pp. 640–652, 2024.

[40] Y. Roh, G. Heo, and S. E. Whang, "A Survey on Data Collection for Machine Learning: A Big Data-AI Integration Perspective," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 4, pp. 1328–1347, 2021, doi: 10.1109/TKDE.2019.2946162.

[41] S. Negara and D. Triadi, "Topic modeling using latent dirichlet allocation (LDA) on twitter data with Indonesia keyword," *Bull. Soc. Informatics Theory Appl.*, vol. 5, no. 2, pp. 124–132, 2021.

[42] P. Wibawa and M. N. Hakim, "Stemming Bahasa Jawa Menggunakan Damerau Levenshtein Distance (Dld)," *J. Tek. Inform.*, vol. 14, no. 1, pp. 22–27, 2021, doi: 10.15408/jti.v14i1.15010.

[43] Wang, W. Xu, W. Yan, and C. Li, "Text similarity calculation method based on hybrid model of LDA and TF-IDF," *ACM Int. Conf. Proceeding Ser.*, pp. 1–8, 2019, doi: 10.1145/3374587.3374590.

[44] Chen, Z. Zhang, J. Long, and H. Zhang, "Turning from TF-IDF to TF-IGM for term weighting in text classification," *Expert Syst. Appl.*, vol. 66, pp. 1339–1351, 2016, doi: 10.1016/j.eswa.2016.09.009.

[45] S. P. Patel and S. H. Upadhyay, "Euclidean distance based feature ranking and subset selection for bearing fault diagnosis," *Expert Syst. Appl.*, vol. 154, 2020, doi: 10.1016/j.eswa.2020.113400.

[46] P. Y. Ristanti, A. P. Wibawa, and U. Pujianto, "Cosine Similarity for Title and Abstract of Economic Journal Classification," *Proceeding - 2019 5th Int. Conf. Sci. Inf. Technol. Embrac. Ind. 4.0 Towar. Innov. Cyber Phys. Syst. ICSITech 2019*, no.

July, pp. 123–127, 2019, doi: 10.1109/ICSITech46713.2019.8987547.

[47] Wibowo, C. Quix, N. S. Hussien, H. Yuliansyah, and F. D. Adhinata, "Similarity Identification of Large-scale Biomedical Documents using Cosine Similarity and Parallel Computing," *Knowl. Eng. Data Sci.*, vol. 4, no. 2, p. 105, 2022.

[48] C. Sur, "RBN: enhancement in language attribute prediction using global representation of natural language transfer learning technology like Google BERT," *SN Appl. Sci.*, vol. 2, no. 1, pp. 1–15, 2020, doi: 10.1007/s42452-019-1765-9.

[49] P. Wibawa *et al.*, "Deep Learning Approaches with Optimum Alpha for Energy Usage Forecasting," *Knowl. Eng. Data Sci.*, vol. 6, no. 2, p. 170, 2023, doi: 10.17977/um018v6i22023p170-187.

[50] Kurniawati, E. M. Yuniarno, and Y. K. Suprapto, "Deep Learning for Multi-Structured Javanese Gamelan Note Generator," *Knowl. Eng. Data Sci.*, vol. 6, no. 1, p. 41, 2023, doi: 10.17977/um018v6i12023p41-56.

[51] Prasetya, F. Andika, A. Bella, and P. Utama, "Deep learning in education : a bibliometric analysis," *Bull. Soc. Informatics Theory Appl.*, vol. 6, no. 2, pp. 151–157, 2022.

[52] Y. Sujatna *et al.*, "Stacked LSTM-GRU Long-Term Forecasting Model for Indonesian Islamic Banks," *Knowl. Eng. Data Sci.*, vol. 6, no. 2, p. 215, 2023, doi: 10.17977/um018v6i22023p215-250.