Journal of Information Systems and Informatics
Vol. 6, No. 2, June 2024 e-ISSN: 2656-4882 p-ISSN: 2656-5935
DOI: 10.51519/journalisi.v6i2.716 Published By DRPM-UBD

Assessing the Accuracy Level of University-Based Website-Based Search Engines Using F-Measure and Hellinger

Irfan Santiko[1,*], Gerry Andriana[2]

[1]Information System Departement, Amikom Purwokerto University, Banyumas, Indonesia
[2]Informatics Departement, Amikom Purwokerto University, Banyumas, Indonesia
Email: [1]irfan.santiko@amikompurwokerto.ac.id, [2]gerry_an@gmail.com

Abstract

Websites are an information medium that is becoming something that is needed in this era. Including the media within the campus environment. The problem is that the campus as a forum or place for student learning is considered less than optimal in presenting information on student learning activities. For example, library reference information, administration, important announcements, and other similar information. The current solution is that universities use social media platform communication media which are considered accurate, which actually adds to problems when the media is used not in accordance with its function, such as promotions, hoax information and irrelevant information. This causes the information to become too massive so that the level of accuracy and relevance is reduced. The author's solution is to optimize the search engine on the campus website platform to be used as an absolute information medium. So the information obtained will be more targeted and accurate. Starting from measuring the level of accuracy to the impact of the results will be discussed in this article. The technique used to measure accuracy is a quantitative technique consisting of the F-Measure and the Hellinger Method. As a result, the campus will know that to distribute related news, the campus can find out keywords that are considered strategic in every report on the media website.

Keywords: Search Engine, Website, Keyword Measuring, News Accuracy

1. INTRODUCTION

A similarity detection tool that can show the degree of variance between text documents was developed for Indonesian text documents in an effort to reduce plagiarism [1]. The document similarity detection model uses the Hellinger algorithm technique in an effort to achieve high accuracy [2], [3]. Hellinger's pre-calculation method finished the tokenization, filtering, and word root construction processes. Word root formation is done using the Indonesian rule basis stemmer approach, whereas stop-word tuning is employed for filtering [4], [5]. A perception test is performed utilizing an extensive Indonesian lexicon to determine the relevance and irrelevantness of the terms being searched, prior to evaluating the

similarity identification technique. The results of testing the document similarity finding tool with the Hellinger technique [6].

There are many words that have been put together in a sentence in a document that shares some similarities with other documents, or with many other documents that already exist. The search results with visible results show how similar the documents are to one another. Search engines operate in accordance with their purpose, which is to give users access to pertinent information [7], [8]. Due to the employment of various algorithms, search engines generate information with varying outcomes.

Textual documentation can be particularly helpful if it is easily accessed and handled within an integrated database. There are not many actors or data-interested parties that have implemented integrated documents [9]. Prefixes, inserts, and suffixes are what make Indonesian unique as a text document object that may be utilized for a variety of purposes, including data processing. Periodically, the volume of data grows; if it is not adequately handled, it will not be able to be utilized to its fullest extent. The same material in data will keep repeating and piling up. As a result, in order to store and locate data with ease, we require a data storage pattern. It is possible to facilitate efficient information archiving and retrieval with search engine models [10], [11].

In order to facilitate the usage of data and serve as a tool for identifying document similarities within an information system or during data collection, document search engines must be developed [12]. When creating search engines, various factors are taken into account, including material and categories that are applicable to daily life and accurately and rapidly accessible information within text documents [13]. Table 1 displays previous research on plagiarism. The goal of this research is to develop a search engine that leverages the Hellinger method to detect plagiarism in text content.

Tabel 1. The table form which used, table font is adjusting

Sam, Caitlin (2020) [14]	Mithun, Ahamed. (2020) [8]	Yu, Dongjin (2020) [15]	Rath, Mamata. (2021) [16]
Research focuses on efficient and effective approaches to constructing and labeling such networks. This network-based topic visualization model proves to be a powerful means for exploring,	This research focuses on measuring correlation heterogeneity and is defined as the distance between the correlation matrix and the correlation matrix	This research focuses its attention on identifying some key parameters that will help identify plagiarism in a better way. Techniques used in semantic	The research focuses on the design of a text document search engine that is used to measure or see similarities between documents in a

Sam, Caitlin (2020) [14]	Mithun, Ahamed. (2020) [8]	Yu, Dongjin (2020) [15]	Rath, Mamata. (2021) [16]
characterizing, and summarizing large collections of unstructured text documents.	with a constant value of 0.	modeling of information allow for deeper levels of accuracy.	database or corpus.

Using a search engine that applies the Hellinger Method will help solve this issue by providing highly accurate information search results and a quick search procedure.

2. METHODS

The author used the following actions to create keywords and evaluate search accuracy for news concerning campuses:

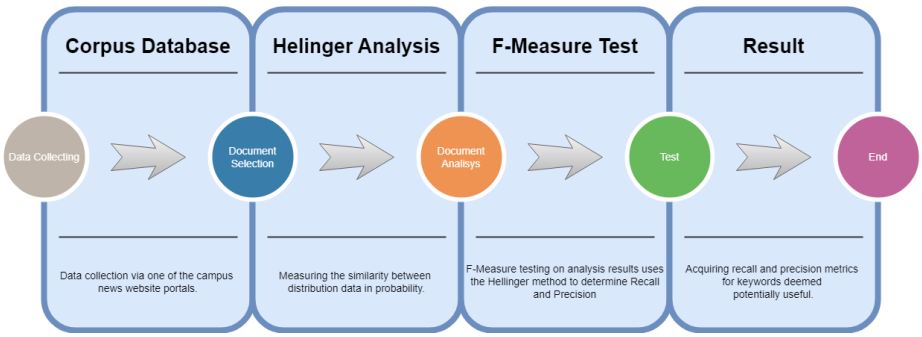

Figure 1. Steps in the process of fixing a problem.

A search engine that compares or measures the similarity between documents in a database was created as a result of this research. The information used came from the university's technology-focused website portal or news sources covering the IT industry, with a particular emphasis on the most recent developments in the field. The research object was a technology-based university website, as this news portal regularly distributes the most recent information about IT and focuses on IT-related themes. A search engine that compares or measures similarities between documents in a database will be created as a result of this research. Following that, news from the university website is collected and kept in a database for similarity analysis. In order to make it simple for users to locate a document when they need to, the database saving procedure begins with detecting existing documents or generating clusters of existing documents and assigning them a document code.

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

2.1. Corpus Databases

Essentially, any grouping of multiple texts can be referred to as a corpus (McEnery and Wilson, 2001). The contents of every text can be referred to as the corpus, as the word corpus is Latin for body. However, in a contemporary linguistic context, the term "corpus" has a more precise meaning. For the several languages it represents, a corpus serves as a standard reference. This requires that it be widely accessible to other scholars. A publicly accessible corpus has the benefit of offering a standard against which research can be conducted.
Using a query model, this software was developed to assist in determining how similar each document is to other texts in the corpus. The detector is designed with a search engine display model in mind, and the output will be search results ranked according to the Hellinger technique.

2.2. Hellinger Method

Hellinger is a method used to calculate the level of similarity between two objects. For set notation use Equation 1.

$$= \left(2 \sqrt{1 - \sum_{i=1}^{d} \sqrt{P_i Q}\, i} \right) \quad (1)$$

where p and q are different documents. pi is term i in document p qi is term i in document q. To calculate the similarity between two documents using the Hellinger method, the following steps can be followed:

1) Document Pre-processing: Pre-process the document text to clean and prepare it before calculations. This step includes steps such as removing punctuation, converting all text to lowercase, removing linking words, and performing stemming or lemmatization if necessary.
2) Vector Representation: Convert document text to vector representation. One commonly used method is the vector space or bag-of-words (BoW) model. In this model, each document is represented as a vector in N-dimensional space, where N is the number of unique words in the corpus. Each element of the vector represents the frequency with which a word appears in the document.
3) Compute Histogram: For each document, compute a histogram of the vector representation. A histogram depicts the frequency distribution of words in a document.
4) Histogram Normalization: Normalize the histogram by dividing each element by the total number of elements in the histogram. This ensures that the histogram represents a valid probability distribution.

5) Calculating Hellinger Distance: Calculate the Hellinger distance between two histograms using the following formula:

$$Hellinger\ Distance = sqrt(0.5 * sum((sqrt(hist1[i]) - sqrt(hist2[i]))^2)) \quad (2)$$

Where hist1 and hist2 are the histograms of the two documents to be compared. This calculation involves calculating the difference between the square roots of the histogram elements and calculating the distance of the square roots of these differences.

6) Interpretation of Results: The smaller the Hellinger distance value, the more similar the documents are.

2.3. Hellinger Method Testing

The Hellinger approach was used for document similarity detection due to its high document accuracy. The degree of accuracy of the developed tool is then assessed using recall and precision tests on the document similarity detection tool. The recall test will yield data regarding the quantity of documents that are found during a document search. The accuracy level of the tool will be determined by the precision test based on the search results. A perception analysis method based on the Big Indonesian Dictionary is used in this testing phase.

3. RESULTS AND DISCUSSION

3.1. Document Detector Implementation

Users will be able to enter queries into this interface's query column. The appearance of a search engine informs the design of the document similarity detection model. Users will find it easy to utilize a display that is straightforward and simple to operate. The user uses the search box to input a document, or a duplicate of the document, to implement this detection tool. The user can then click the search button once the document to be compared has been entered.

In this research used document data sourced from news using Indonesian. Therefore, the results that will be presented in the table are keywords originating from Indonesian language news as shown in Figure 2.

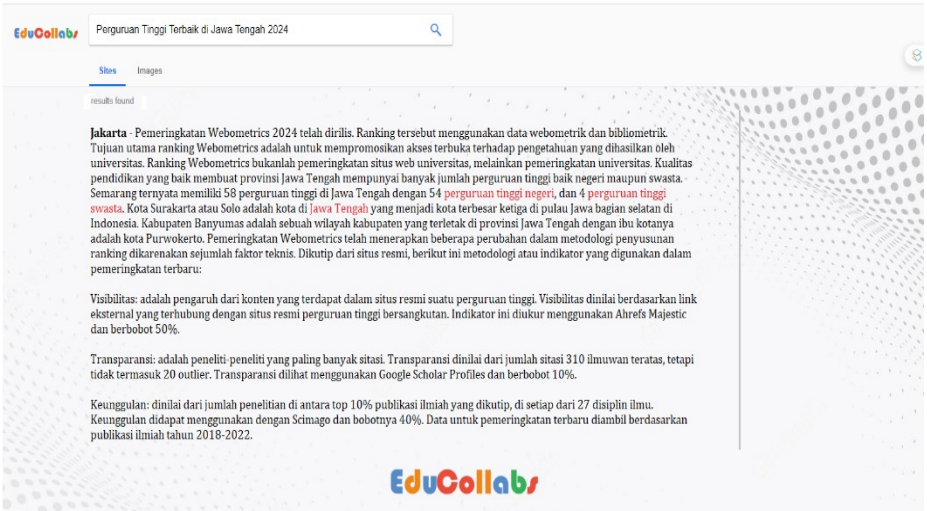

Figure 2. Sample of Discovery of search text results.

3.2. Documentation of data into a corpus

Documents downloaded from the news portal detik dot com are stored in a different database and assigned a unique identifier for each document. News articles are entered into the corpus using the detik dot net website. First, the news is categorized and numbered before being added to the corpus of papers. In order to help with analysis, topic grouping is required. Document numbering is also necessary in order to locate documents inside the corpus. The corpus table view is present in Table 4. Because the examination would refer to Indonesian university documents, the author used an example of news articles written in the language.

Table 2. Corpus documents

ID	Title	Content	Documents
1	*TOP 3 Daerah dengan Perguruan Tinggi Terbanyak di Jawa Tengah, Wong Jateng Pasti Tahu Siapa Juaranya.* (Source from: https://www.ayosemarang.com/)	*Kualitas pendidikan yang baik membuat provinsi Jawa Tengah mempunyai banyak jumlah perguruan tinggi baik negeri maupun swasta. Semarang ternyata memiliki 58 perguruan tinggi di Jawa Tengah dengan 54 perguruan tinggi negeri, dan 4 perguruan tinggi swasta. Kota Surakarta atau Solo adalah kota di Jawa Tengah yang menjadi kota terbesar ketiga di pulau Jawa bagian selatan di Indonesia. Kabupaten Banyumas adalah sebuah wilayah kabupaten yang terletak di provinsi Jawa Tengah dengan ibu kotanya adalah kota Purwokerto.*	HTML
2	*20 Kampus Terbaik di Jawa*	*Perguruan Tinggi Negeri (PTN) dan Perguruan Tinggi Swasta (PTS) ternama dengan berbagai pilihan jurusan atau program studi (prodi). Tidak hanya universitas, tetapi ada politeknik,*	HTML

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

Tengah Versi Webometrics 2023. (Source from: https://tekno.tempo.co/)	*sekolah tinggi hingga institut yang menyediakan pembelajaran teori dan praktik bagi mahasiswa. Situs analitik Webometrics merilis daftar peringkat perguruan tinggi terbaik di seluruh dunia secara berkala (realtime). Sebanyak lebih dari 31 ribu kampus, baik negeri maupun swasta yang ikut diurutkan. Di Indonesia, terdapat 3.381 kampus yang dinilai berdasarkan indikator tertentu.*

3.3. Tokenization

The process of dividing text or words into smaller pieces, known as "tokens," is known as tokenization. In natural language processing, tokens are fundamental units that typically take the form of words, sentences, or symbols. When developing a method for document similarity detection, tokenization is done initially. Tokenization is accomplished by writing computer code that splits sentences into individual words, which are subsequently input into tables. Words that have undergone tokenization appear exactly as they are, all lowercase. Following its separation from the phrase, each word is subsequently saved with its corresponding document code or number. The tokenization results are displayed in Table 3.

Table 3. Implementation of the Tokenization Process in the token Result Table.

Title	Term	Documents
Perguruan Tinggi terbanyak yang di minati diwilayah jawa tengah	*perguruan*	HTML
Perguruan Tinggi terbanyak yang di minati diwilayah jawa tengah	*tinggi*	HTML
Perguruan Tinggi terbanyak yang di minati diwilayah jawa tengah	*kampus*	HTML
Perguruan Tinggi terbanyak yang di minati diwilayah jawa tengah	*semarang*	HTML
Perguruan Tinggi terbanyak yang di minati diwilayah jawa tengah	*favorit*	HTML
Perguruan Tinggi terbanyak yang di minati diwilayah jawa tengah	*unggul*	HTML
Perguruan Tinggi terbanyak yang di minati diwilayah jawa tengah	*lulusan*	HTML

3.4. Filtering

Words that have already been divided during the token process are filtered. The following words are compared or examined to determine if they are part of the list of stopwords. The system will eliminate a word from the data or data table if it appears in a stop word or is similar to another term. Applying stopword tuning data to filter. The screening process's outcomes are displayed in Table 4.

Table 4. Results of the screening process.

Title	Term	Documents
Perguruan Tinggi terbanyak yang di minati diwilayah jawa tengah	*kampus*	HTML
Perguruan Tinggi terbanyak yang di minati diwilayah jawa tengah	*favorit*	HTML
Perguruan Tinggi terbanyak yang di minati diwilayah jawa tengah	*akreditasi*	HTML
Perguruan Tinggi terbanyak yang di minati diwilayah jawa tengah	*semarang*	HTML
Perguruan Tinggi terbanyak yang di minati diwilayah jawa tengah	*terbaik*	HTML
Perguruan Tinggi terbanyak yang di minati diwilayah jawa tengah	*unggul*	HTML
Perguruan Tinggi terbanyak yang di minati diwilayah jawa tengah	*lulusan*	HTML

Title	Term	Documents
Perguruan Tinggi terbanyak yang di minati diwilayah jawa tengah	*kerja*	HTML
Perguruan Tinggi terbanyak yang di minati diwilayah jawa tengah	*mudah*	HTML
Perguruan Tinggi terbanyak yang di minati diwilayah jawa tengah	*PTN*	HTML
Perguruan Tinggi terbanyak yang di minati diwilayah jawa tengah	*PTS*	HTML

3.5. Steming

A stemming programming code generator is used to create word roots, also known as stemming. The rule base stemmer method is used to create word roots. The way the system operates is by eliminating suffixes, inserts, and prefixes. In this stemming procedure, a constructed word will be transformed into a base term. Stemmer results are displayed in Table 5.

Table 5. STEM Result.

Title	Term	Frequency	Rank Frequency
Kampus terbaik favorit anda di jawa tengah?	*kampus*	40	80
Kampus terbaik favorit anda di jawa tengah?	*favorit*	20	40
Kampus terbaik favorit anda di jawa tengah?	*akreditasi*	6	12
Kampus terbaik favorit anda di jawa tengah?	*PTN*	2	4
Kampus terbaik favorit anda di jawa tengah?	*PTS*	2	4
Kampus terbaik favorit anda di jawa tengah?	*unggul*	1	1
Kampus terbaik favorit anda di jawa tengah?	*lulusan*	1	1

3.6. Hellinger Count

In the discussion in this paper, the author tries to provide an overview of the similarities in the documents. The case study is on searches from various platforms with similar keywords. The case study on this search engine application uses Indonesian language text documents. The query entered into the search engine is a keyword with 2 terms, namely "*Kampus Terbaik*", "*Perguruan Tinggi*", "*Jawa Tengah*", 3 terms "*Perguruan Tinggi Terbaik*", "*Kampus Jawa Tengah*", "*Unggulan PTS PTN*", 4 terms "*Perguruan Tinggi Jawa Tengah*", "*Kampus Terbaik PTN PTS*". 5 terms "*Perguruan Tinggi Terbaik di Jawa Tengah*".
The recall value is calculated using the equation (3):

$$R = \frac{Number\ of\ relevant\ items\ retrieved}{Total\ number\ of\ relevant\ items\ in\ collection} \quad (3)$$

With R being recall, the R value is obtained by comparing the number of relevant items retrieved with the total number of relevant items in collection. Recall is a document that is called from the document similarity detector according to a user request that follows the pattern of the document similarity detector. The greater the recall value, it is not enough to judge whether a document similarity detector is good or not (4).

$$P = \frac{Number\ of\ relevant\ items\ retrieved}{Total\ number\ of\ items\ retrieved} \quad (4)$$

Where P is Precision. then the P value is obtained by comparing the Number of relevant items retrieved with the Total number of items retrieved. Precision is the number of documents retrieved from the relevant database after the user assesses them with the required information. The greater the precision value of a document similarity detector, the document similarity detector can be said to be good. The results of the Recall calculation for the keyword " *Kampus Terbaik* " are as follows (5):

$$R = \frac{Number\ of\ relevant\ items\ retrieved}{Total\ number\ of\ relevant\ items\ in\ collection} \quad (5)$$

Recall = 0.47
Precision calculation results for the keyword " *Kampus Terbaik* " are as follows (6):

$$P = \frac{Number\ of\ relevant\ items\ retrieved}{Total\ number\ of\ items\ retrieved} \quad (6)$$

Precision = 0.89
Complete results can be seen in table 8. The average calculation results for Recall and precision are as follows;
Average Recall = 0.31
Average Precision = 0.71

Table 6. Recall and Precision calculation results.

#	Query	Recall	Precision
1	*Kampus Terbaik*	0,47	0,89
2	*Perguruan Tinggi*	0,55	0,84
3	*Jawa Tengah*	0,37	0,79
4	*Perguruan Tinggi Terbaik*	0,29	0,50
5	*Kampus Jawa Tengah*	0,47	0,89
6	*Unggulan PTS PTN*	0,12	0,20
7	*Perguruan Tinggi Jawa Tengah*	0,12	0,67
8	*Kampus Terbaik PTN PTS*	0,12	0,67
9	*Perguruan Tinggi Terbaik di Jawa Tengah*	0,29	0,83
10	*Kampus Unggulan Favorit di Jawa Tengah*	0,15	0,62

After obtaining the calculation results, for more details they will be changed to the following chart:

p-ISSN: **2656-5935** http://journal-isi.org/index.php/isi e-ISSN: **2656-4882**

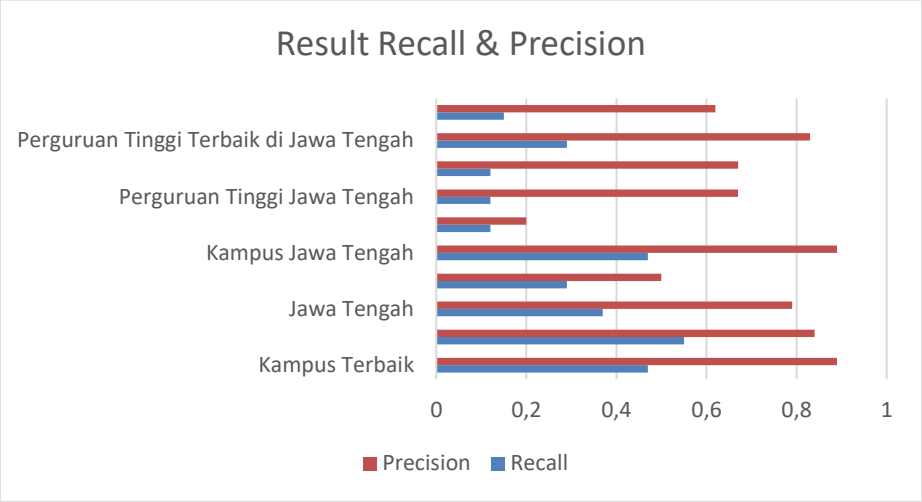

Figure 3. Infographic results from recall and precision calculations for document keywords.

3.7. Discussion.

The author wants to talk about the potential for improving keyword accuracy and how it can affect how long news stays relevant. The media on campus websites plays a significant role in helping the campus perform in terms of sharing information about its events. A good campus will, based on the signals, give prospective students and the broader public accurate information when they register.

As is well known, information media is used by campuses in Indonesia and around the world to compete with one another. Regretfully, not every university information department is aware of the best practices and approaches to sharing critical information. This has an impact on who is targeted for registration among potential students. Currently, the only required media on campuses is a website that every user can view. By adopting Hellinger's news keyword accuracy assessment technique, information departments on campus will be able to create more effective campus marketing initiatives.

4. CONCLUSION

Finding similarities between documents the advantage of the Hellinger method's development was its capacity to identify documents with precise document similarity results (precision = 0.89). The Hellinger method's text document search

yields results with an average recall of 0.31 and an average precision of 0.71 when it comes to document similarity, according to the recall and precision tests. Therefore, it can be said that the Hellinger method is appropriate for application in the analysis of information documents found on university websites. This will have an impact on how accurately users of website-based information systems at tertiary education institutions assimilate information. Similar to what was discussed before, Hellinger's analysis's findings will be able to assist campus information managers in developing and setting up strategic in news planning.

REFERENCES

[1] Y. Bai, L. Zhao, Z. Wang, J. Chen, and P. Lian, "Entity Thematic Similarity Measurement for Personal Explainable Searching Services in the Edge Environment," *IEEE Access*, vol. 8, pp. 146220 – 146232, 2020, doi: 10.1109/ACCESS.2020.3014185.

[2] Q. Wang, X. Liu, W. Liu, A.-A. Liu, W. Liu, and T. Mei, "MetaSearch: Incremental Product Search via Deep Meta-Learning," *IEEE Trans. Image Process.*, vol. 29, pp. 7549 – 7564, 2020, doi: 10.1109/TIP.2020.3004249.

[3] F.-C. Yuan and C.-H. Lee, "Intelligent sales volume forecasting using Google search engine data," *Soft Comput.*, vol. 24, no. 3, pp. 2033 – 2047, 2020, doi: 10.1007/s00500-019-04036-w.

[4] A. A. Jalal, "Text mining: Design of interactive search engine based regular expressions of online automobile advertisements," *Int. J. Eng. Pedagog.*, vol. 10, no. 3, pp. 35 – 48, 2020, doi: 10.3991/IJEP.V10I3.12419.

[5] P. Stolarski, W. Lewoniewski, and W. Abramowicz, "Cryptocurrencies perception using wikipedia and google trends," *Inf.*, vol. 11, no. 4, 2020, doi: 10.3390/INFO11040234.

[6] S. Venkatachalam, L. P. Subbiah, R. Rajendiran, and N. Venkatachalam, "An ontology-based information extraction and summarization of multiple news articles," *Int. J. Inf. Technol.*, vol. 12, no. 2, pp. 547 – 557, 2020, doi: 10.1007/s41870-019-00367-x.

[7] J. Sun, S. Hu, X. Nie, and J. Walker, "Efficient Ranked Multi-Keyword Retrieval with Privacy Protection for Multiple Data Owners in Cloud Computing," *IEEE Syst. J.*, vol. 14, no. 2, pp. 1728 – 1739, 2020, doi: 10.1109/JSYST.2019.2933346.

[8] A. M. Mithun and Z. A. Bakar, "Empowering information retrieval in semantic web," *Int. J. Comput. Netw. Inf. Secur.*, vol. 12, no. 2, pp. 41 – 48, 2020, doi: 10.5815/ijcnis.2020.02.05.

[9] P. Ježek, J. L. Teeters, and F. T. Sommer, "NWB Query Engines: Tools to Search Data Stored in Neurodata Without Borders Format," *Front. Neuroinform.*, vol. 14, 2020, doi: 10.3389/fninf.2020.00027.

[10] G. Amato, F. Carrara, F. Falchi, C. Gennaro, and L. Vadicamo, "Large-scale instance-level image retrieval," *Inf. Process. Manag.*, vol. 57, no. 6, 2020, doi: 10.1016/j.ipm.2019.102100.

[11] W. Darmalaksana, C. Slamet, W. B. Zulfikar, I. F. Fadillah, D. S. Maylawati, and H. Ali, "Latent semantic analysis and cosine similarity for hadith search engine," *Telkomnika (Telecommunication Comput. Electron. Control.*, vol. 18, no. 1, pp. 217 – 227, 2020, doi: 10.12928/TELKOMNIKA.V18I1.14874.

[12] C. Li, "Research on an enhanced web information processing technology based on ais text mining," *Recent Adv. Electr. Electron. Eng.*, vol. 14, no. 1, pp. 29 – 36, 2021, doi: 10.2174/2352096513999201026224357.

[13] S. Soltani, S. A. H. Seno, and R. Budiarto, "Developing Software Signature Search Engines Using Paragraph Vector Model: A Triage Approach for Digital Forensics," *IEEE Access*, vol. 9, pp. 55814 – 55832, 2021, doi: 10.1109/ACCESS.2021.3071795.

[14] C. Sam, N. Naicker, and M. Rajkoomar, "Meta-analysis of artificial intelligence works in ubiquitous learning environments and technologies," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 9, pp. 603 – 613, 2020, doi: 10.14569/IJACSA.2020.0110971.

[15] D. Yu, L. Zhang, C. Liu, R. Zhou, and D. Xu, "Automatic Web service composition driven by keyword query," *World Wide Web*, vol. 23, no. 3, pp. 1665 – 1692, 2020, doi: 10.1007/s11280-019-00742-5.

[16] M. Rath, J. J. P. C. Rodrigues, and G. S. Oreku, "Applications of cognitive intelligence in the information retrieval process and associated challenges," *Int. J. Cogn. Informatics Nat. Intell.*, vol. 15, no. 1, pp. 26 – 38, 2021, doi: 10.4018/IJCINI.2021010103.