Matrik: Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer
Vol. 24, No. 2, March 2025, pp. 259∼272
ISSN: 2476-9843, accredited by Kemenristekdikti, Decree No: 200/M/KPT/2020
DOI: 10.30812/matrik.v24i2.4598

Comparison of Text Representation for Clustering Student Concept Maps

Reni Fatrisna Salsabila[1], Didik Dwi Prasetya[1], Triyanna Widiyaningtyas[1], Tsukasa Hirashima[2]
[1]Universitas Negeri Malang, Malang, Indonesia
[2]Hiroshima University, Hiroshima, Japan

Article Info

Article history:

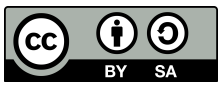
Received November 11, 2024
Revised December 18, 2024
Accepted February 11, 2025

Keywords:

Bidirectional Encoder Representations from Transformers;
Term Frequency-Inverse Document Frequency;
Clustering;
Concept Map.

ABSTRACT

This research aims to address the critical challenge of selecting a text representation method that effectively captures students' conceptual understanding for clustering purposes. Traditional methods, such as Term Frequency-Inverse Document Frequency (TF-IDF), often fail to capture semantic relationships, limiting their effectiveness in clustering complex datasets. This study compares TF-IDF with the advanced Bidirectional Encoder Representations from Transformers (BERT) to determine their suitability in clustering student concept maps for two learning topics: Databases and Cyber Security. The method used applies two clustering algorithms: K-Means and its improved variant, K-Means++, which enhances centroid initialization for better stability and clustering quality. The datasets consist of concept maps from 27 students for each topic, including 1,206 concepts and 616 propositions for Databases, as well as 2,564 concepts and 1,282 propositions for Cyber Security. Evaluation is conducted using two metrics Davies-Bouldin Index (DBI) and Silhouette Score, to assess the compactness and separability of the clusters. The result of this study is that BERT consistently outperforms TF-IDF, producing lower DBI values and higher Silhouette Scores across all clusters (k= 2 - k=10). Combining BERT with K-Means++ yields the most compact and well-separated clusters, while TF-IDF results in overlapping and less-defined clusters. The research concludes that BERT is a superior text representation method for clustering, offering significant advantages in capturing semantic context and enabling educators to identify student misconceptions and improve learning strategies.



Corresponding Author:

Didik Dwi Prasetya, +6281297000116,
Department of Electrical and Informatics Engineering,
Universitas Negeri Malang, Malang, Indonesia,
Email: didikdwi@um.ac.id.

How to Cite:
R. Fatrisna Salsabila, D. Dwi Prasetya, T. Widyaningtyas, and T. Hirashima, "Comparison of Text Representation for Clustering Student Concept Maps", *MATRIK: Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer*, Vol.24, No.1, pp. 259-272, March, 2025.

1. INTRODUCTION

The development of information technology has brought significant changes in various fields, including education. In this digital era, the amount of information in the form of text has increased exponentially, so text-based data management has become increasingly complex[1, 2]. One method that can help handle this large text data is clustering, which aims to group data based on specific similarities [3, 4] Clustering can map text data into more structured groups, facilitating better analysis and decision-making in various fields, including education. One area that can greatly benefit from clustering is concept map representation, particularly in open-ended concept maps, where clustering helps reveal patterns and insights in students' understanding of complex topics. In the context of education, data grouping is very relevant, especially in assessing students' understanding of the material being taught. For example, educators can use concept maps to map students' understanding of a specific topic, such as databases. This concept map helps visualize how students connect key concepts, which is useful for evaluating the effectiveness of learning and identifying areas where students may experience misconceptions [5, 6]. Clustering is particularly valuable for open-ended concept maps, as it groups similar conceptual patterns, providing deeper insights into students' comprehension and helping educators tailor instructional strategies more effectively.

The text representation technique is a key factor in producing optimal clustering results. One commonly used method is Term Frequency-Inverse Document Frequency (TF-IDF). TF-IDF calculates the frequency of words in a document and adjusts their weights based on the occurrence of those words across the dataset. This method has been proven effective in many applications, but it has limitations in capturing the context of words more deeply. Because of this, this method may not always be ideal for tasks that require a better understanding of semantics [7]. Along with advances in Natural Language Processing (NLP), word embeddings-based methods such as BERT (Bidirectional Encoder Representations from Transformers) are increasingly used due to their ability to better capture word context. BERT can understand words bidirectionally, both from left to right and vice versa, allowing this model to produce richer text representations. This makes BERT particularly suitable for tasks that require a deep understanding of text, such as clustering [6] Previous research has shown that BERT consistently outperforms TF-IDF in a variety of text grouping metrics, such as K-Means, EFCM (Enhanced Fuzzy C-Means), DEC (Deep Embedded Clustering), and IDEC (Improved Deep Embedded Clustering) [8, 9]. In addition, some studies integrate BERT with LDA (Latent Dirichlet Allocation), which has successfully improved the accuracy of topic modeling when used in conjunction with the K-Means algorithm [10]. Word embedding-based approaches such as BERT have also been shown to be able to significantly improve clustering performance, especially on large datasets, as demonstrated in studies using WEClustering (Word Embedding Clustering) techniques [11] Based on these findings, this study aims to evaluate the performance of BERT and TF-IDF as a text representation method in the clustering task of student concept maps on database topics.

This study evaluates the performance of BERT and TF-IDF as text representation methods in clustering tasks using K-Means and K-Means++ algorithms. K-Means is a popular clustering algorithm due to its simplicity and ability to minimize the distance between the data and the cluster center (centroid) [12] This algorithm divides the data into k clusters based on the Euclidean distance between the data and the centroid, iterating until the centroid's position does not significantly change. K-Means++ is a variant that improves the centroid initialization process, resulting in more stable clustering and reduced sensitivity to random initialization [13]. Performance evaluation is carried out using two main metrics, namely the Silhouette Score and the Davies-Bouldin Index (DBI) [14]. The Silhouette Score is used to measure how well objects in a cluster are grouped by comparing the distance of those objects to other clusters, while the Davies-Bouldin Index (DBI) evaluates the quality of separation between clusters with lower values indicating better clustering quality [15]. These metrics provide quantitative insights into clustering quality but are not without limitations. The DBI, which measures the ratio of intra-cluster cohesion to inter-cluster separation, may inaccurately reflect clustering performance in datasets with highly variable cluster sizes or densities. For instance, clusters with large spreads or irregular shapes can result in deceptively low DBI values, suggesting an artificial improvement in separation. Similarly, the Silhouette Score compares the average intra-cluster distance to the nearest cluster's inter-cluster distance, which becomes less reliable in high-dimensional datasets where distances between data points converge [14]. Given these limitations, this study employs a complementary approach by combining these metrics with qualitative analysis, such as visualization and inspection of cluster findings, to comprehensively evaluate clustering results. This enables a more nuanced understanding of the strengths and weaknesses of BERT and TF-IDF in producing meaningful clusters, particularly in educational datasets.

This research contributes significantly to understanding how different text representations influence clustering outcomes, particularly in grouping students' concept maps. Gaps remain in prior research, particularly the limited exploration of comparing contextual (BERT) and non-contextual (TF-IDF) text representation methods for clustering educational datasets, specifically student concept maps. While BERT has been shown to outperform TF-IDF in various clustering applications, its use in mapping student understanding through concept maps has been underexplored, as these maps require a nuanced analysis of semantic relationships. Furthermore, previous studies have not adequately addressed the role of advanced clustering algorithms like K-Means++ in enhancing clustering stability and quality for educational datasets. This research aims to evaluate the performance of BERT and TF-IDF as

text representation methods in clustering student concept maps on database topics using K-Means and K-Means++ clustering algorithms. This study explicitly focuses on educational datasets to identify student comprehension patterns through optimal clustering techniques. It provides a detailed comparison of clustering outcomes using the Davies-Bouldin Index and Silhouette Score, offering a more robust performance evaluation. The findings offer valuable insights for educators in understanding students' conceptual comprehension, enabling them to identify learning difficulties or misconceptions and tailor effective teaching strategies. Beyond education, this approach highlights broader applicability in fields like healthcare, business, and social media, offering an efficient solution for managing the growing complexity of text data by uncovering meaningful patterns and insights.

2. RESEARCH METHOD

This research adopts a systematic methodology divided into five key stages: data collection, text extraction, data preprocessing, text representation, and clustering modeling. Each stage plays a vital role in ensuring the analysis's quality, reliability, and relevance. The process begins with data collection, followed by extracting and preprocessing the textual information to ensure consistency and accuracy. Text representation transforms the data into a structured format suitable for clustering, which is the central focus of the modeling stage. The flowchart in Figure 1 outlines the sequential steps, providing a visual representation of the transition from data collection to clustering evaluation. Through this comprehensive approach, the research seeks to deliver meaningful insights into clustering student concept maps, ensuring the results are both robust and applicable.

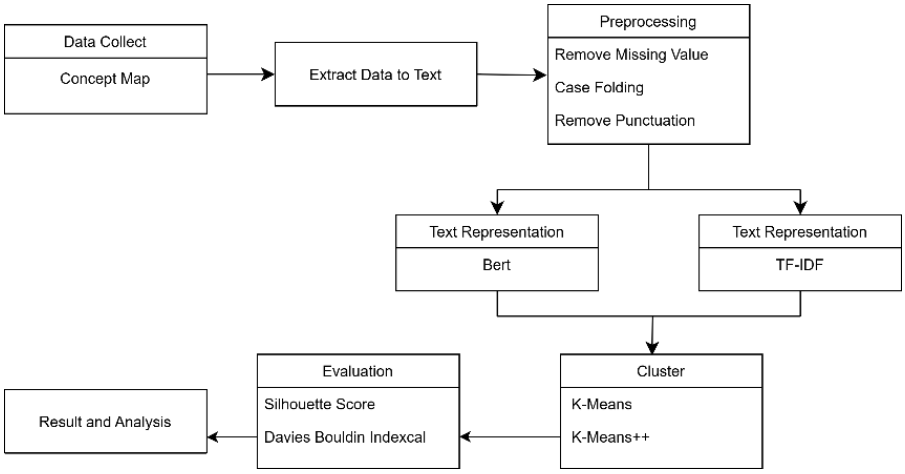


Figure 1. Flowchart of the research process

2.1. Data Collection

Data collection in this study was conducted through concept maps produced by 27 students. The concept maps included 1,206 concepts and 616 relationship nodes, which were chosen as the main instrument due to their ability to effectively illustrate students' understanding of the relationships between concepts in the database topic. Each student produced a concept map containing various important elements, such as entities, attributes, and relations, which provided an in-depth visual representation of how students connect key ideas in databases. Each element in the concept map, such as an entity, represents an object or thing identified in the database topic, while attributes describe the characteristics of the entity [16]. Relations, on the other hand, describe the interrelationships between entities and how they are established in the context of the material being taught. This data covered a wide variety of understanding, with some students able to describe complex relationships between the concepts while others showed simpler relationships. Through these concept maps, data collection focused on understanding students' understanding of database concepts and how they relate to the various elements. The data collected will be used as a basis for further analysis, such as clustering, to identify patterns in students' understanding and find areas that require more attention in the teaching process [17]. A sample of the concept maps from a few students is shown in Table 1. The data collected will be used for further analysis, such as clustering, to identify patterns in students' understanding and find areas that require more attention in the teaching process.

Table 1. Sample Data Concept Maps

Students	Concept Map
Student 1	Relational databases use a ”two-dimensional table” structure. Two-dimensional tables consist of ”rows or tuples.” Two-dimensional tables consist of ”columns or . . .
Student 2	relational database advantage ”easy to perform data operations” relational database advantage ”simple” relational database definition ”two-dimensional table” relational database has ”domains” two-dimensional tables are called ”relations” relational database has ”cardinality” relational database has ”attributes” . . .
Student 3	a relational database consists of ”tuples,” relational database key, ”relational key,” relational key, ”super key key,” relational key key, ”candidate key,” relational key key ”primary key” relational key key ”alternate key” relational key key ”foreign key” relational rules ”integrity rules” super key is ”primary key” primary key is ”primary key” foreign key is ”guest key” relational database has ”language” language called ”SQL” . . .

These maps were chosen to illustrate students' understanding of the relationships between concepts in the database topic. Each concept map includes entities, attributes, and relations, showing how students connect key ideas in databases. Figure 2 provides an example of a concept map, showing multiple nodes and the relationships between them. Propositions are formed by combining nodes with relationships, such as “An entity has attributes” or “A relational key is associated with a primary key.” This visual representation highlights how students' understanding of database concepts is structured and interconnected.

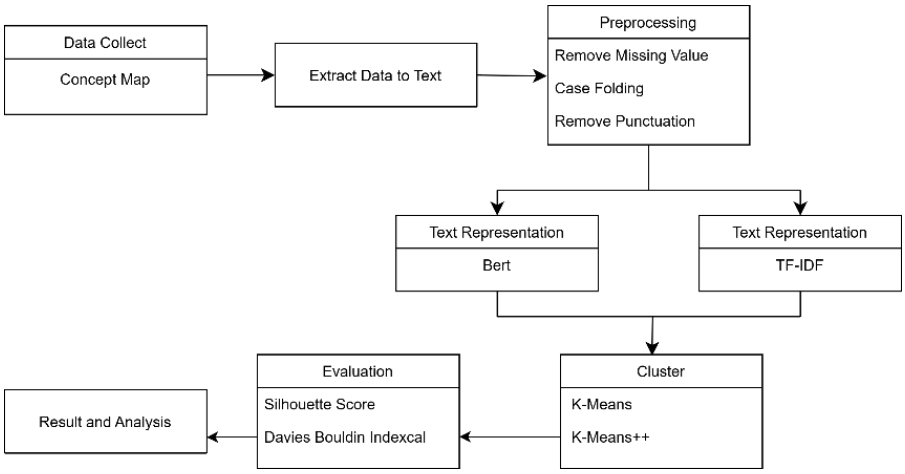


Figure 2. Visualization of collecting proposition of concept map

2.2. Data Extraction

This research systematically converts visual maps into textual data to facilitate more comprehensive and automated analysis. The conversion into text allows for easier manipulation and processing using computational methods. Once the text is generated, it includes key terms derived from the concept maps and the relationships between these terms. This textual data becomes the foundation for further analysis using NLP techniques. NLP offers a wide range of tools for analyzing language, including tokenization, part-of-speech tagging, and semantic analysis, allowing a more in-depth understanding of the student's comprehension. According to recent studies, NLP has proven effective in interpreting and evaluating large-scale textual data making it a powerful tool for assessing students' understanding based on the extracted text. In this process, data initially stored in an SQL database is extracted and converted into a plain text format. Extracting data from SQL into text ensures a simpler, more flexible structure for subsequent analyses. After the text conversion, it is exported into a CSV (Comma-Separated Values) file, which is commonly used for data analysis due to its straightforward tabular format. This step is crucial as it prepares the data for clustering techniques, which require organized datasets for identifying patterns or groups within the data. Clustering methods, such as K-means or hierarchical clustering, can then be applied to the CSV file, allowing researchers to group similar concepts or key terms together based on their relationships and frequency of occurrence. Combining concept maps, SQL data extraction, NLP techniques, and clustering allows for a holistic analysis of student comprehension. By leveraging these tools, educators and researchers can identify distinct patterns in student understanding, enabling more targeted interventions and improved educational strategies. Moreover, using CSV format makes integrating data with various machine learning models easier for further clustering and classification analysis. Figure 3 illustrates the process of converting the

concept map data from an SQL database into plain text and then exporting it into a CSV (Comma-Separated Values) file. In the figure, arrows represent the flow from concept maps (stored in SQL) to conversion into text and, ultimately, CSV format for easier data manipulation and clustering analysis.

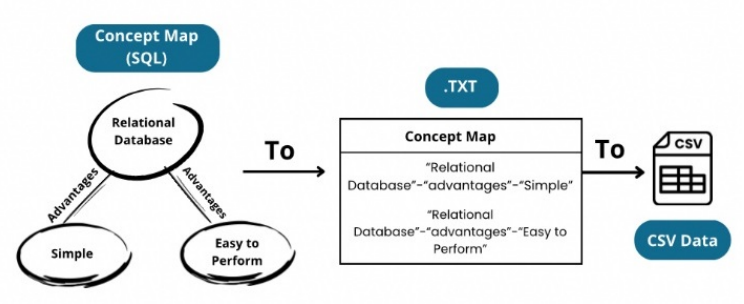


Figure 3. Visualization of data extraction

2.3. Data Preprocessing

Once the text has been extracted from the students' concept maps, the next step is to preprocess the data to ensure better quality of text representation before further analysis. The first stage in this preprocessing is to remove missing values. All empty text data containing NaN (Not a Number) values are converted into empty strings to avoid errors or interference in the subsequent analysis process. This step makes the data more consistent and prevents missing values. Next, punctuation cleanup is performed using regular expressions. The text is clean of non-alphabetic symbols such as periods, commas, question marks, and other irrelevant characters for analysis. This cleaning aims to reduce noise in the data so that the focus remains on meaningful and relevant content. The last stage is case folding or converting uppercase letters to lowercase, which aims to maintain consistency in the data. All text is converted to lowercase, so there is no difference in the analysis between the same words, but they are written in uppercase or lowercase. For example, the words "Database" and "database" will be treated as the same entity, preventing bias in the analysis regarding word frequency. Overall, this preprocessing helps to prepare cleaner, more uniform data for further analysis, such as in clustering or other text analysis processes. With optimally processed data, the quality of analysis results will be more accurate and reliable and provide more meaningful results for research. Figure 4 shows an example of the data preprocessing workflow. The figure illustrates how CSV data is first processed to remove missing values, then undergoes case folding where uppercase words are converted to lowercase. Finally, the punctuation is removed to focus on the relevant content.

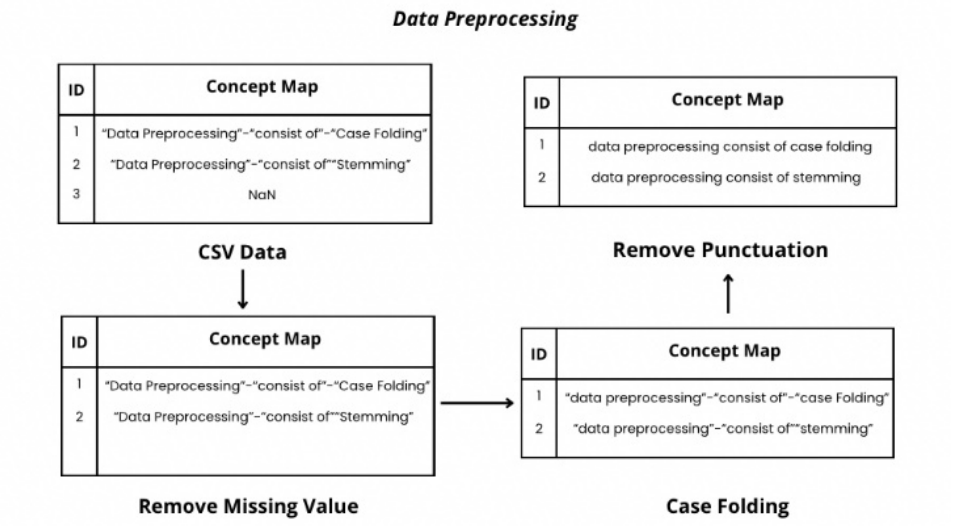


Figure 4. Visualization of data preprocessing

2.4. Modelling

The modeling stage involves two main parts: text representation and data grouping.

2.4.1. Text Representation

This research uses two methods for text represent, ation: BERT and TF-IDF. These two methods were chosen to compare modern and traditional approaches to representing text data.

1. BERT is used to produce contextual representations of text, capturing relationships between words in sentences from two directions (bidirectional). BERT can capture deeper nuances of meaning, making it perfect for identifying conceptual relationships in students' concept maps.BERT's input consists of token embeddings, segment embeddings, and positional embeddings [18]. The input sentence is tokenized where each word is split into subword units. The token embeddings $E(t_i)$ for each token (t_i) are computed by Equation 1.

$$E(t_i) = TokenEmbedding(t_i) + SegmentEmbedding(t_i) + PositionalEmbedding(t_i) \tag{1}$$

This composite embedding allows BERT to capture both the lexical meaning of the tokens and their positions within the sentence. After processing through BERT, a contextual embedding is generated for each token. For text representation, we often use the [CLS] token embedding, which is added at the beginning of every input sentence and is designed to capture the overall sentence meaning. This embedding serves as the representation of the entire text document as in Equation 2.

$$H_{CLS} = BERT\_Output_{CLS} \tag{2}$$

2. TF-IDF is a traditional text representation method, and TF-IDF weighs words based on how often they appear within the document and across the dataset using Equation 3. This approach allows the identification of the most important words in each text. The TF-IDF formula is used to give weight to words based on their frequency in the document [19]. Where TF (t, d) is the frequency of the occurrence of the word t in the document d. IDF(t) is the inverse logarithm of the number of documents containing the word t divided by the total number of documents in the corpus.

$$TF - IDF(t, d) = TF(t, d) \times IDF(t) \tag{3}$$

2.4.2. Clustering

After the text is represented, the clustering process uses several algorithms, namely K-Means, K-Means++, and Agglomerative Hierarchical Clustering (AHC).

1. K-Means divides data into k clusters based on the distance between the data and the cluster center (centroid). To improve the stability and accuracy of results, K-Means is a simple and widely used clustering algorithm to divide data into a number of k clusters. This algorithm works by calculating the Euclidean distance between the data and the center of the cluster (centroid) and then grouping the data into clusters with the nearest centroid. This process is repeated until the centroid no longer changes. The formula for calculating the Euclidean distance between two points is shown in Equation 4 [20]. Where, x1 and x2 are used as a reference to determine the distance between the data and the centroid. d_{ij} similarity calculation distance, n = number of vectors, x_{ik} = input image vector, x_{jk} = comparison image vector. In addition, the objective functions minimized in the K-Means algorithm to get the optimal cluster using Equation 5 [21]. Where, Ci is a set of data in cluster i, and μ_i is the centroid of cluster i. This algorithm is suitable for grouping data on a large scale and has a clear cluster structure.

$$Distance(x_1.x_(2)) = \sqrt{\sum_{i=1}^{n}(x_{1i-x_{2i}})^2} \tag{4}$$

$$J = \sum_{i=1}^{k} \sum_{x\sigma Ci} ||x - \mu_i||^2 \tag{5}$$

2. To improve K-Means' performance, K-Means++ is applied in the initial centroid selection process. K-Means++ solves the sensitivity problem to random initialization that often occurs in standard K-Means by selecting centroids based on probabilities proportional to the distance from the previous centroid. Thus, centroid initialization is more stable, and clustering results are more accurate [22]. The formula for selecting the new centroid in K-Means++ is shown in Equation 6. Where, (dx_i) is the distance between the data point x_i and the nearest centroid that has been selected. This algorithm accelerates convergence and reduces the risk of getting poor clustering results.

$$P(x_1) = \frac{d_(xi)^2}{\sum_{xj\sigma x}^{d(xj)^2}} \tag{6}$$

2.5. Evaluation

Evaluation of clustering results is carried out using several commonly used evaluation metrics, including:

1. Davies-Bouldin Index (DBI): This metric measures how well a cluster is formed, with lower values indicating better clustering outcomes. DBI is calculated as the ratio of the distance between clusters to the width of the cluster using Equation 7 [23]. Where, σ_i is the average distance between the elements in the cluster and the centroid cluster, $d(c_i, c_j)$ is the distance between the centroid cluster i and j,k is the total number of clusters.

$$DBI = \frac{1}{k}\sum_{i=1}^{k} max_{j\neq i}\left(\frac{\sigma_i + \sigma_j}{d_{ij}}\right) \tag{7}$$

2. Silhouette Score: This metric assesses how closely data in one cluster is to data in another (See Equation 8), indicating how well the data is grouped. Used to measure how well an object in the same cluster compares to an object in another cluster [24]. Where, $\alpha(i)$ is the average distance in the cluster and b(i) is the distance to the nearest cluster. Higher silhouette scores (closer to 1) suggest better-defined and well-separated clusters.

$$s(i)\frac{b(i) - \alpha(i)}{max(\alpha(i), b(i))} \tag{8}$$

3. Compute Time: In addition to the quality of clustering, compute time is also evaluated to assess the efficiency of each text representation method and clustering algorithm. This metric is particularly relevant for large datasets, where time efficiency is crucial for method selection. Lower compute times indicate more efficient algorithms, which are desirable for handling substantial volumes of data in practical applications.

3. RESULT AND ANALYSIS

In this section, the results of the clustering experiment will be explained by comparing two text representation methods, BERT and TF-IDF, and two clustering algorithms, K-Means and K-Means++, with the number of clusters (c) ranging from 2 to 10. The evaluation was carried out using two main metrics, the Davies-Bouldin Index (DBI) and the Silhouette Score, to assess the quality of the clusters produced.

3.1. BERT and TF-IDF for Text Representation

After the data processing stage, where the raw text is prepared for analysis, the text is transformed into word embeddings using TF-IDF and BERT models. Word embeddings provide a numerical text representation, capturing its semantic meaning and context. This approach allows for a more in-depth understanding of the relationships between words within sentences and documents, enabling sophisticated analysis of textual data. Table 2 shows TF-IDF represents text numerically by focusing on the frequency of words in a document and their rarity across the corpus. This approach assigns higher weights to significant words within a specific document but less frequently in other documents, helping to identify key terms. For instance, in the text "Teacher relational database advantage easy to perform data operations relational database advantage simple relational database definition two-dimensional table," TF-IDF generates a vector such as [-1.0884764, 0.30652913, -0.5568338, 0.22192995, -0.7475788, . . .]. Each value in the vector corresponds to a term in the text, with its weight reflecting the term's importance in the document relative to the entire corpus. Similarly, for the

text ”Student 1 relational database uses a two-dimensional table structure two-dimensional tables consist of rows or tuples two-dimensional tables consist of columns or attributes,” the TF-IDF representation produces a vector like [-0.76811856, -0.31373638, -0.6640695, -0.017617596, . . .]. These numerical vectors, while straightforward, are effective for identifying term significance but lack the ability to capture semantic meaning or context.

Table 2. Text representation of TF-IDF model

Text	Text Representation
Teacher relational database advantage easy to perform data operations relational database advantage simple relational database definition two-dimensional table . . .	[-1.0884764,0.30652913,-0.5568338,0.22192995,-0.7475788,0.050853185,0.43495926,0.524072,-0.6448173,-0.10213829,-1.0481268,-0.31979442,-0.9449612,0.87223846,0.000991097,-0.17455451,0.6454823,0.15461184,-0.15041155,0.25662586,-0.09411203,-0.38996008,-0.5381783,-0.05705671,0.8738157, . . .]
Student 1 relational database use a two-dimensional table structure two dimensional tables consist of rows or tuples two dimensional tables consist of columns or attributes $ldots$	[-0.76811856,-0.31373638,-0.6640695,-0.017617596,0.7831697,-0.041491713,0.48265025,0.49603134,-0.43335092,-0.1678787,-0.4972308,-0.09873922,-1.1433048,0.46140924,0.26342595,-0.18301694,0.4730455,0.41408986,0.02495681,0.002344119,-0.54013574,-0.046343658,-0.47672573,-0.12486508,. . .]

However, TF-IDF also has notable limitations. It treats words independently, ignoring the relationships and context in which they appear, which means it lacks the ability to capture semantic meaning. Furthermore, it is insensitive to synonyms, as different words with the same meaning are treated as entirely separate entities. This approach also struggles with scalability for large corpora, as the computational complexity can increase significantly with the size of the dataset. Despite its limitations, TF-IDF remains a valuable tool in scenarios where a simple, frequency-based text analysis is sufficient. Unlike advanced models like BERT, which capture deep semantic relationships and contextual meanings. Table 3 shows that BERT generates these semantic representations through its multi-layered architecture, which considers the entire context of a sentence in both directions-left-to-right and right-to-left. This bidirectional understanding produces embeddings that are rich in semantic details. Each embedding consists of two primary components: a score that reflects the significance of a particular word or phrase within the text and a multi-dimensional vector that represents the semantic characteristics of the text.

Table 3. The text representation of the BERT model

Text	Text Representation
Teacher relational database advantage easy to perform data operations relational database advantage simple relational database definition two-dimensional table. . .	(array ([[0.7782415]], dtype=float32), array ([[-0.23834857, 0.9095637, 0.23737827, . . . , 0.57109797, -0.58448243, -0.3526079], [0.10946223, 1.6046929, -0.41067538, . . . , 0.71168053, -0.14333607, 0.03262531]], dtype=float32))
Student 1 relational database use a two-dimensional table structure two dimensional tables consist of rows or tuples two dimensional tables consist of columns or attributes	(array ([[0.8976546]], dtype=float32), array ([[0.1094626, 1.6046932, -0.4106756, . . . , 0.7116804, -0.14333594, 0.03262523], [0.10946223, 1.6046929, -0.41067538, . . . , 0.71168053, -0.14333607, 0.03262531]], dtype=float32))

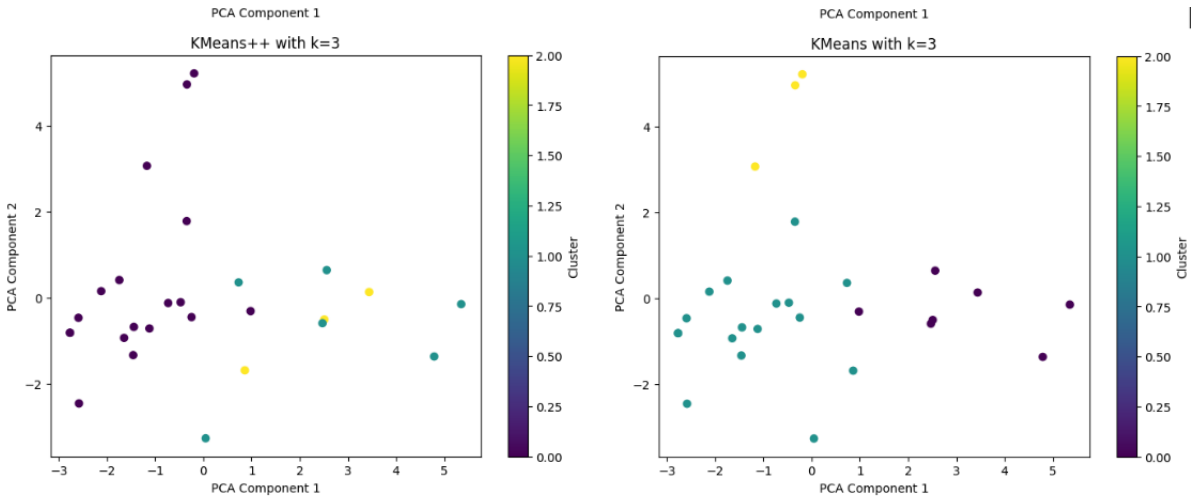
For example, the embedding for the word ”Guru” includes a primary score of 0.7782415, which indicates the relative importance of the word in its context. Its accompanying embedding vector contains hundreds of dimensions, each representing specific semantic attributes. Values within this vector, such as -0.23834857 and 0.9095637, depict the word’s nuanced meaning and its relationship to other words in the dataset. Similarly, the embedding for the term ”Siswa 1” has a primary score of 0.9999999, highlighting its high contextual weight. The corresponding embedding vector for ”Siswa 1” also includes numerous dimensions, with values like 0.1094626 and 1.6046932, offering a distinct semantic representation based on the word’s context. These embeddings allow for a deeper, more accurate representation of the text, making it suitable for advanced analytical tasks.

By utilizing word embeddings from BERT, text data is transformed into a semantic numerical format that machine learning algorithms can easily process. This transformation enhances tasks like clustering or classification, where understanding the context and meaning of the text is critical. Unlike traditional methods such as TF-IDF, which focus on word frequency, BERT embeddings

provide a comprehensive semantic perspective, enabling extracting meaningful patterns and insights from complex textual data. This makes BERT an invaluable tool for sophisticated text analysis and semantic understanding.

3.2. Clustering Results Using BERT and TF-IDF

Building upon the explanation of text representation, this section presents the clustering results obtained with BERT and TF-IDF representations, further analyzed through Principal Component Analysis (PCA) visualizations.PCA is a dimensional reduction technique that projects data into two-dimensional space, making it easier to understand how data is distributed and grouped. The selection of k=3 for PCA analysis in this research was based on the results of the clustering evaluation, which showed that the number of clusters provided optimal results. Evaluation using metrics such as DBI (Davies-Bouldin Index) and Silhouette Score shows that for BERT text representation, the lowest DBI value and the highest Silhouette Score are achieved at k=3, indicating that the data is well grouped, and the separation between clusters is quite clear. In addition, at k=3, clustering creates three groups that represent different themes or characteristics in the data, which is particularly relevant for text data analysis. This selection also helps provide simpler and more understandable visualizations, as it reduces complexity by capturing the most significant variability of the data. Thus, k=3 not only results in optimal cluster separation but also ensures the stability and reliability of the results obtained, especially when using advanced BERT models to understand the context of the data. The visualization of the PCA results in Figure 5 (a) shows the data distribution grouped using K-Means++ for TF-IDF at k=3. The figure shows that the cluster generated by TF-IDF still shows a significant overlap, reflecting a high DBI value and a low Silhouette Score. This indicates that even though K-Means++ uses better centroid initialization, the TF-IDF representation is still incapable of generating an optimal cluster. In Figure 5 (b), which shows the clustering results with K-Means for TF-IDF at k=3, the overlapping between clusters is seen more compared to Figure 5. This shows that K-Means produces worse clusters in terms of separation than K-Means++. This difference reflects the important role of better centroid initialization in K-Means++, especially for the data represented by TF-IDF. Therefore, K-Means++ provides improved cluster separation quality for TF-IDF, although the results are still not optimal.



(a) (b)

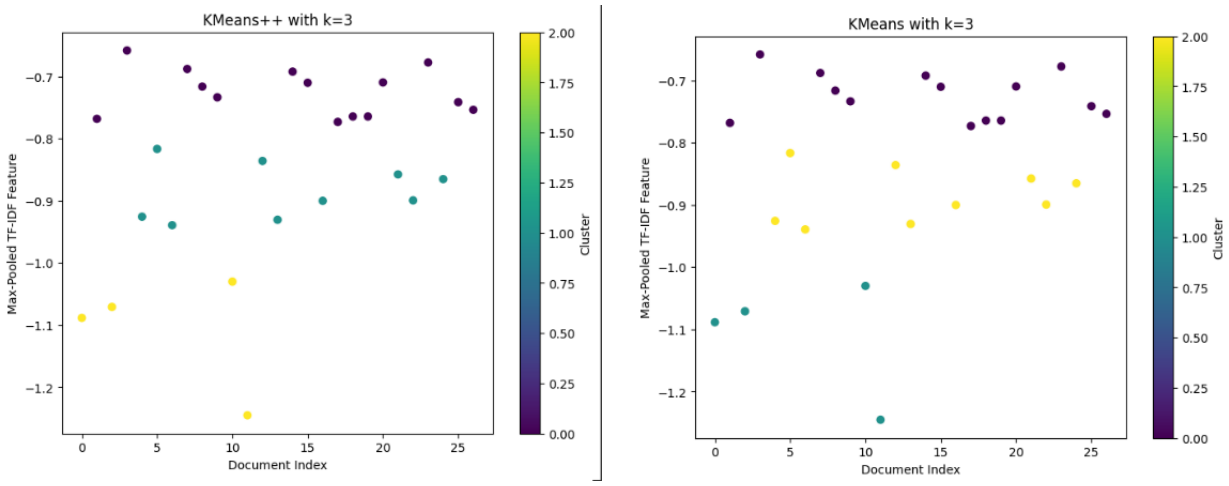
Figure 5. TF-IDF PCA Results : (a) PCA K-Means++ (b) PCA K-Means

In contrast, Figure 6 shows clustering results with BERT for K-Means and K-Means++ at k=3. In both images, it can be seen that the cluster generated by BERT is much more separate and clear than TF-IDF. However, the difference between K-Means and K-Means++ in BERT is insignificant, both in visualization and quantitative metrics such as DBI and Silhouette Score. These results show that BERT as a context-based text representation is already powerful enough to produce compact and discrete clusters, so the better centroid initialization of K-Means++ does not provide a significant improvement compared to the standard K-Means. The findings of this research are that embedding-based text representation methods, such as BERT, consistently outperform word frequency-based methods like TF-IDF in clustering tasks. BERT achieves significantly lower Davies-Bouldin Index (DBI) values and higher Silhouette Scores across all cluster variations, indicating more compact and distinct clusters. In contrast, TF-IDF produces higher DBI values and lower Silhouette Scores, demonstrating poorer clustering performance with more overlapping clusters. These results confirm

the robustness of BERT in capturing contextual relationships within text, making it a superior choice for tasks requiring semantic understanding.

These results are consistent with previous studies emphasizing BERT's ability to capture text context. Research by [25] and [8]. similarly, BERT outperforms traditional methods like TF-IDF in clustering tasks, particularly due to its bidirectional understanding of language. In this research, K-Means++ was used to improve the clustering performance of TF-IDF by reducing overlap between clusters. However, TF-IDF's performance remains inferior to BERT's even with this enhancement. The role of K-Means++ in optimizing centroid initialization, thus reducing sensitivity to random initialization, has been well documented in previous research, such as in [13] This variant of the K-Means algorithm is particularly useful for simpler text representation methods like TF-IDF. However, BERT's ability to produce cohesive clusters with minimal dependence on initialization has been demonstrated in other studies, such as [10], which suggested that embedding-based models are more suitable for clustering tasks on complex textual data.

Regarding computation time, BERT takes slightly longer than TF-IDF, but its clustering quality justifies the additional time required. While TF-IDF is faster, the superior clustering results provided by BERT, as shown in this study, outweigh the time tradeoff. Previous research by [10] suggested that the computational cost of BERT is justified in large-scale text analysis because it leads to more accurate clustering results, offering a clear advantage when high-quality clustering is essential. These findings have significant implications for various fields that require text analysis, such as education, healthcare, and business. Embedding-based text representations such as BERT enable more in-depth and accurate data analysis. In education, for example, better quality clustering can help educators map students' understanding of certain concepts and identify areas that require special attention [6]. In the healthcare sector, better text representations are also useful for clustering patients based on medical records, which enables a more comprehensive analysis of patients' health conditions [9]. In the business sector, BERT-based representations can aid in more accurate customer segmentation, making it easier for companies to identify groups of customers based on their behavioral patterns. The findings support embedding methods for complex text analysis tasks, with BERT models showing consistently higher performance than TF-IDF.



(a) (b)

Figure 6. BERT PCA Results : (a) PCA K-Means++ (b) PCA K-Means

3.3. Comparison of Clustering Algorithms

When applied to the BERT and TF-IDF text representations, a comparison is made between the performance of two clustering algorithms, K-Means, and K-Means++. This comparison underscores the influence of centroid initialization on the clustering quality produced by the text representation methods discussed previously. K-Means is a widely used clustering algorithm for its simplicity; however, it has a drawback in its reliance on random centroid initialization. K-Means++ was developed to address this issue by randomly selecting the first centroid and the following centroids based on their distance from the existing centroids, reducing the risk of suboptimal clustering results. Table 4 shows the clustering performance using K-Means on both text representation methods. This table shows that the DBI (Davies-Bouldin Index) value for BERT is consistently lower compared to TF-IDF across all clusters (k) tested, which ranges from 2 to 10. For example, at k=3, the DBI for BERT reached 0.458, while TF-IDF remained high at 1.771. A

lower DBI value on BERT indicates that the resulting cluster is more compact and separate. In addition, the Silhouette Score, which measures how well each piece of data is grouped in its cluster compared to other clusters, also shows better results for BERT. The Silhouette Score value for BERT ranged from 0.604 at k=2 to 0.648 at k=3, the highest value across the entire number of clusters. On the other hand, TF-IDF shows a lower Silhouette Score value, ranging from 0.145 at k=2 and 0.159 at k=3, which indicates that the data in the TF-IDF cluster is more overlapping and less separate.

Table 4. The Performance with K-Means

CLUSTER	BERT			TF-IDF		
	DBI	SILHOUETTE	TIME	DBI	SILHOUETTE	TIME
C2	0.564260	0.604004	0.007104	2.172.047	0.145681	0.154367
C3	0.458293	**0.648553**	0.014404	1.771.313	**0.158873**	0.101930
C4	0.458518	0.572952	0.010196	1.726.094	0.115568	0.029113
C5	0.334928	0.603757	0.011221	1.550.980	0.122767	0.029587
C6	0.357469	0.617612	0.010370	1.694.438	0.080349	0.032192
C7	0.304513	0.619093	0.010614	1.303.231	0.098461	0.025379
C8	0.326798	0.583010	**0.010181**	1.207.111	0.142489	**0.022083**
C9	0.351529	0.572483	0.010473	1.275.461	0.082539	0.046873
C10	**0.314800**	0.614212	0.009667	**1.163.148**	0.092459	0.060076

Table 5 shows the clustering performance using K-Means++. K-Means++ is an algorithm that improves centroid initialization by probabilistically selecting the initial centroid, thereby reducing the risk of getting poor clustering results due to random initialization. In this table, the results for BERT with K-Means++ still show lower DBI values and higher Silhouette Score compared to TF-IDF. For example, at k=3, the DBI value for BERT remains at 0.458, while for TF-IDF, it increases to 1.884, which indicates that despite improvements, the clusters generated by TF-IDF are still less than optimal. Although K-Means++ provides a significant improvement in clustering results for TF-IDF compared to K-Means, with DBI decreasing from 1.884 to 1.566, the Silhouette Score value for TF-IDF still shows poor performance, which is only 0.121 at k=3. The analysis shows that BERT-based text representation has a significant advantage in producing better clustering than TF-IDF. Lower DBI scores and higher Silhouette Scores on BERT indicate that this method captures semantic relationships between words more effectively. This makes the clusters generated by BERT more compact and more separate, which is highly desirable in clustering analysis. Meanwhile, although K-Means++ slightly improves cluster separation for TF-IDF, the results still show that this method is not effective enough to produce quality clusters, especially when compared to BERT. Execution times are also recorded in the table, where BERT shows relatively stable and slightly longer times, especially at k=3. In contrast, TF-IDF exhibits more variable times but is generally faster than BERT, especially on smaller cluster counts. In conclusion, the selection of the right text representation method has a great influence on the clustering results obtained. As a context-based method, BERT provides significant advantages in terms of better cluster separation over word frequency methods such as TF-IDF.

Table 5. The Performance Using K-Means++

CLUSTER	BERT			TF-IDF		
	DBI	SILHOUETTE	TIME	DBI	SILHOUETTE	TIME
C2	0.564260	0.604004	**0.009396**	1.566.123	**0.156705**	0.045804
C3	0.458293	**0.648553**	0.015797	1.884.220	0.121956	**0.027458**
C4	0.328536	0.645751	0.009986	1.995.527	0.067511	0.053268
C5	0.341342	0.577760	0.011914	1.848.760	0.075282	0.030125
C6	0.357469	0.617612	0.015682	1.615.653	0.075439	0.036518
C7	0.378361	0.580959	0.012871	1.607.500	0.092515	0.033019
C8	0.345419	0.552306	0.015368	1.321.948	0.112939	0.039163
C9	0.309251	0.558326	0.015672	1.225.661	0.104211	0.028651
C10	**0.298479**	0.587758	0.021866	**1.153.897**	0.106570	0.045636

When comparing these results with the broader body of work in the field, it is clear that BERT consistently outperforms TF-IDF in clustering quality and robustness. Studies like [25] and [8] have reported similar findings, reinforcing BERT's ability to outperform TF-IDF in various clustering applications due to its advanced language model. Furthermore, even with K-Means++, which helps optimize TF-IDF clustering, BERT's performance remains superior. These observations underline the importance of using sophisticated embedding models, such as BERT, for clustering tasks that require a nuanced understanding of textual data.

Overall, the results of this study confirm that BERT provides superior clustering results compared to TF-IDF, both with K-Means and K-Means++. Although K-Means++ can improve TF-IDF clustering results by reducing cluster overlap, it is still not as good as BERT. This emphasizes the importance of using sophisticated embedding models to achieve better clustering quality in text analysis. In the future, exploring the combination of BERT with more complex clustering algorithms, such as DBSCAN or agglomerative clustering, is recommended to see if there is further improvement potential. This combination is expected to result in more optimal clustering quality, especially in clustering complex and high-dimensional data.

4. CONCLUSION

The results of this study confirm that BERT consistently outperforms TF-IDF in clustering students' concept maps, particularly in the context of database topics. Across all tested cluster configurations (k=2 to k=10), BERT demonstrated significantly lower Davies-Bouldin Index (DBI) values and higher Silhouette Scores, indicating superior clustering quality. These results reflect BERT's ability to capture semantic relationships between concepts more effectively than TF-IDF, which relies solely on word frequency. In particular, at k=3, BERT produced the most compact and well-separated clusters, suggesting that it is better equipped to group related concepts meaningfully and interpretably. While K-Means++ improved the stability of clustering results by addressing issues related to centroid initialization, its effect on TF-IDF was not sufficient to overcome the inherent limitations of this traditional text representation method. Even when paired with K-Means++, TF-IDF still resulted in overlapping and less distinct clusters than those produced by BERT. This highlights the challenge of using frequency-based methods like TF-IDF for clustering complex and semantically rich data, where understanding the context of words plays a critical role in distinguishing between closely related concepts. These findings underline the importance of using advanced text representation methods, like BERT, for clustering tasks that require a deeper understanding of semantic relationships. BERT's superior performance demonstrates its potential to provide more accurate insights into complex datasets, such as student concept maps, which are crucial for understanding students' comprehension and knowledge structures.

5. ACKNOWLEDGEMENTS

The Acknowledgments section is optional. Research sources can be included in this section.

6. DECLARATIONS

AUTHOR CONTIBUTION

Reni Fatrisna Salsabila, the first author, conceived and designed the research, conducted data collection and analysis, and drafted the initial manuscript. Didik Dwi Prasetya, the second author, contributed to the experimental design, performed statistical analysis, and provided critical revisions to improve the manuscript. Triyanna Widiyaningtyas, the third author, provided critical revisions to improve the manuscript. Tsukasa Hirashima, the fourth author, assisted in refining the analysis and contributed critical insights to enhance the manuscript's theoretical framework.

FUNDING STATEMENT

This study was conducted without any financial support.

COMPETING INTEREST

The authors declare no conflict of interest regarding the publication of this article.

REFERENCES

[1] Y. Bilan, O. Oliinyk, H. Mishchuk, and M. Skare, "Impact of information and communications technology on the development and use of knowledge," vol. 191, p. 122519, https://doi.org/10.1016/j.techfore.2023.122519.

[2] A. Löwstedt, "Developmental Stages of Information and Communication Technology," vol. 31, no. 4, pp. 758–778, https://doi.org/10.1093/ct/qtaa015.

[3] S. Wang, A. Beheshti, Y. Wang, J. Lu, Q. Z. Sheng, S. Elbourn, and H. Alinejad-Rokny, "Learning Distributed Representations and Deep Embedded Clustering of Texts," vol. 16, no. 3, p. 158, https://doi.org/10.3390/a16030158.

[4] V. Adu, M. D. Adane, and K. Asante, "Similarity Measure Algorithm for Text Document Clustering, Using Singular Value Decomposition," pp. 8–25, https://doi.org/10.9734/cjast/2021/v40i2231475.

[5] S. M. Dol and P. M. Jawandhiya, "Classification Technique and its Combination with Clustering and Association Rule Mining in Educational Data Mining — A survey," vol. 122, p. 106071, https://doi.org/10.1016/j.engappai.2023.106071.

[6] G. Aşıksoy, "Computer-Based Concept Mapping as a Method for Enhancing the Effectiveness of Concept Learning in Technology-Enhanced Learning," vol. 11, no. 4, p. 1005, https://doi.org/10.3390/su11041005.

[7] Z. Labd, S. Bahassine, K. Housni, F. Z. A. Hamou Aadi, and K. Benabbes, "Text classification supervised algorithms with term frequency inverse document frequency and global vectors for word representation: A comparative study," vol. 14, no. 1, p. 589, https://doi.org/10.11591/ijece.v14i1.pp589-599.

[8] A. Subakti, H. Murfi, and N. Hariadi, "The performance of BERT as data representation of text clustering," vol. 9, no. 1, p. 15, https://doi.org/10.1186/s40537-022-00564-9.

[9] E. C. Garrido-Merchan, R. Gozalo-Brizuela, and S. Gonzalez-Carvajal, "Comparing BERT Against Traditional Machine Learning Models in Text Classification," vol. 2, no. 4, pp. 352–356, https://doi.org/10.47852/bonviewJCCE3202838.

[10] L. George and P. Sumathy, "An integrated clustering and BERT framework for improved topic modeling," vol. 15, no. 4, pp. 2187–2195, https://doi.org/10.1007/s41870-023-01268-w.

[11] V. Mehta, S. Bawa, and J. Singh, "WEClustering: Word embeddings based text clustering technique for large datasets," vol. 7, no. 6, pp. 3211–3224, https://doi.org/10.1007/s40747-021-00512-9.

[12] C. Wu, B. Yan, R. Yu, B. Yu, X. Zhou, Y. Yu, and N. Chen, K -Means Clustering Algorithm and Its Simulation Based on Distributed Computing Platform," vol. 2021, no. 1, p. 9446653, https://doi.org/10.1155/2021/9446653.

[13] J. Y. K. Chan, A. P. Leung, and Y. Xie, "Efficient High-Dimensional Kernel k-Means++ with Random Projection," vol. 11, no. 15, p. 6963.

[14] A. Naghizadeh and D. N. Metaxas, "Condensed silhouette: an optimized filtering process for cluster selection in K-means,Procedia Computer Science, vol. 176, pp. 205–214, 2020.

[15] Q. Li, S. Yue, Y. Wang, M. Ding, and J. Li, "A New Cluster Validity Index Based on the Adjustment of Within-Cluster Distance," vol. 8, pp. 202 872–202 885, https://doi.org/10.1109/ACCESS.2020.3036074.

[16] D. D. Prasetya and T. Hirashima, "Associated Patterns in Open-Ended Concept Maps within E-Learning," vol. 5, no. 2, p. 179, https://doi.org/10.17977/um018v5i22022p179-187.

[17] D. D. Prasetya, A. Pinandito, Y. Hayashi, and T. Hirashima, "Analysis of quality of knowledge structure and students' perceptions in extension concept mapping," vol. 17, no. 1, p. 14, https://doi.org/10.1186/s41039-022-00189-9.

[18] Q. Zhang, Y. Sun, L. Zhang, Y. Jiao, and Y. Tian, "Named entity recognition method in health preserving field based on BERT," vol. 183, pp. 212–220, https://doi.org/10.1016/j.procs.2021.03.010.

[19] T. Kwon, J. Myung, J. Lee, K.-i. Kim, and J. Song, "A Network Packet Analysis Method to Discover Malicious Activities," vol. 0, no. S, pp. 143–153, https://doi.org/10.1633/JISTAP.2022.10.S.14.

[20] R. Suwanda, Z. Syahputra, and E. M. Zamzami, "Analysis of Euclidean Distance and Manhattan Distance in the K-Means Algorithm for Variations Number of Centroid K," vol. 1566, no. 1, p. 012058, https://doi.org/10.1088/1742-6596/1566/1/012058.

[21] F. S. Mukti, A. Junikhah, P. M. A. Putra, A. Soetedjo, and A. U. Krismanto, "A Clustering Optimization for Energy Consumption Problems in Wireless Sensor Networks using Modified K-Means++ Algorithm," vol. 15, no. 3, pp. 355–365, https://doi.org/10.22266/ijies2022.0630.30.

[22] J. Meng, Z. Yu, Y. Cai, and X. Wang, "K-Means++ Clustering Algorithm in Categorization of Glass Cultural Relics," vol. 13, no. 8, p. 4736, https://doi.org/10.3390/app13084736.

[23] C. D. Gutiérrez, J. N. Ruiz, S. C. Salazar, J. P. G. López, J. D. Zapata, and J. F. Botía, "Performance of Hybrid Clustering-Classification Approach for Dual-Band System in a Mode-Locked Fiber Laser," vol. 12, pp. 104 115–104 125, https://doi.org/10.1109/ACCESS.2024.3409565.

[24] M. Shutaywi and N. N. Kachouie, "Silhouette Analysis for Performance Evaluation in Machine Learning with Applications to Clustering," vol. 23, no. 6, p. 759, https://doi.org/10.3390/e23060759.

[25] P. Charoenkwan, C. Nantasenamat, M. M. Hasan, B. Manavalan, and W. Shoombuatong, "BERT4Bitter: A bidirectional encoder representations from transformers (BERT)-based model for improving the prediction of bitter peptides," vol. 37, no. 17, pp. 2556–2562, https://doi.org/10.1093/bioinformatics/btab133.