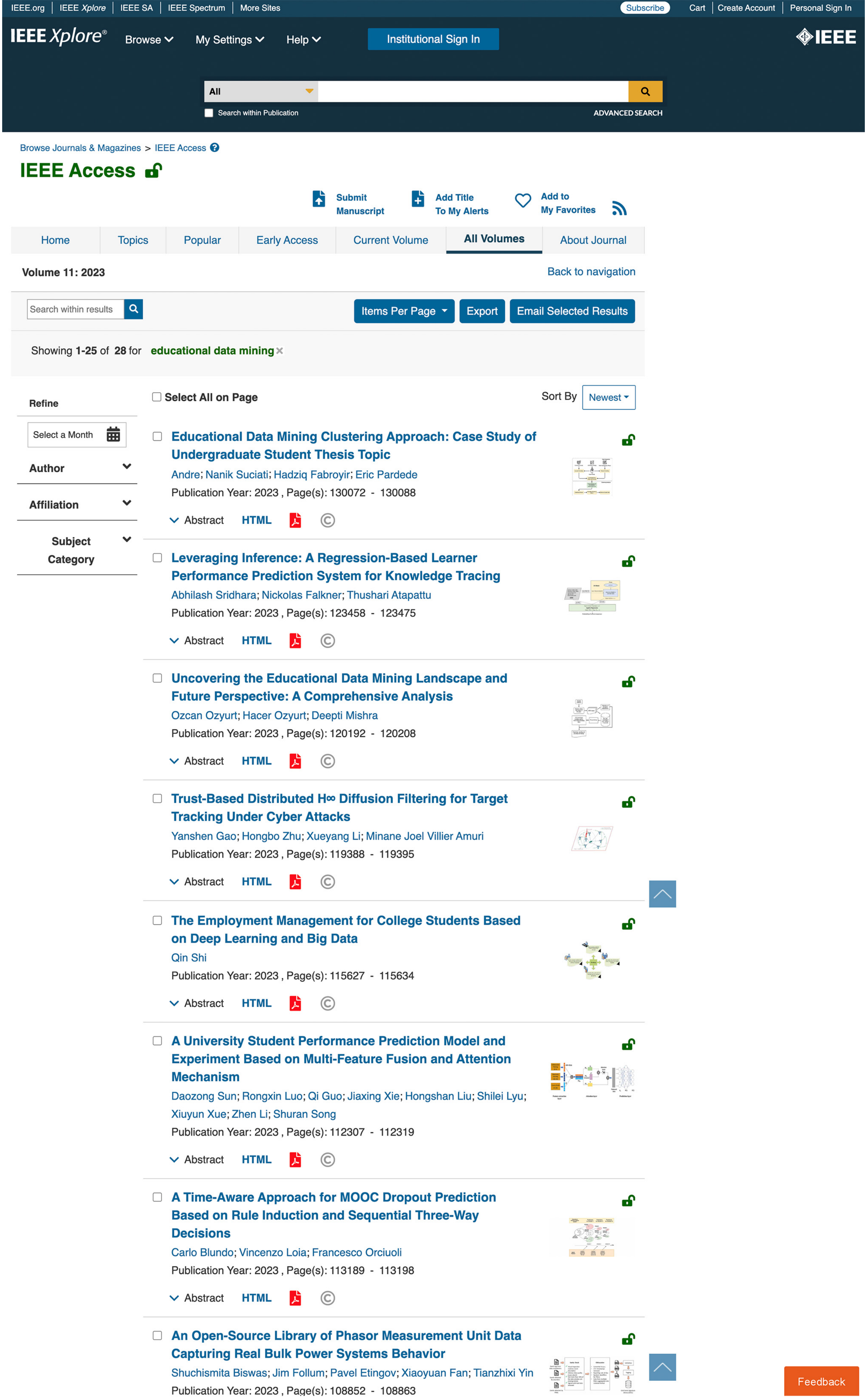

IEEE.org | IEEE *Xplore* | IEEE SA | IEEE Spectrum | More Sites
Subscribe
Cart | Create Account | Personal Sign In
IEEE *Xplore*®
Browse
My Settings
Help
Institutional Sign In
IEEE
All
□ Search within Publication
ADVANCED SEARCH

Browse Journals & Magazines > IEEE Access

IEEE Access

Submit Manuscript
Add Title To My Alerts
Add to My Favorites
Home | Topics | Popular | Early Access | Current Volume | All Volumes | About Journal

Volume 11: 2023

Showing **1-25** of **28** for **educational data mining** ×

□ **Educational Data Mining Clustering Approach: Case Study of Undergraduate Student Thesis Topic**
Andre; Nanik Suciati; Hadziq Fabroyir; Eric Pardede
Publication Year: 2023 , Page(s): 130072 - 130088
Abstract HTML

□ **Leveraging Inference: A Regression-Based Learner Performance Prediction System for Knowledge Tracing**
Abhilash Sridhara; Nickolas Falkner; Thushari Atapattu
Publication Year: 2023 , Page(s): 123458 - 123475
Abstract HTML

□ **Uncovering the Educational Data Mining Landscape and Future Perspective: A Comprehensive Analysis**
Ozcan Ozyurt; Hacer Ozyurt; Deepti Mishra
Publication Year: 2023 , Page(s): 120192 - 120208
Abstract HTML

□ **Trust-Based Distributed H∞ Diffusion Filtering for Target Tracking Under Cyber Attacks**
Yanshen Gao; Hongbo Zhu; Xueyang Li; Minane Joel Villier Amuri
Publication Year: 2023 , Page(s): 119388 - 119395
Abstract HTML

□ **The Employment Management for College Students Based on Deep Learning and Big Data**
Qin Shi
Publication Year: 2023 , Page(s): 115627 - 115634
Abstract HTML

□ **A University Student Performance Prediction Model and Experiment Based on Multi-Feature Fusion and Attention Mechanism**
Daozong Sun; Rongxin Luo; Qi Guo; Jiaxing Xie; Hongshan Liu; Shilei Lyu; Xiuyun Xue; Zhen Li; Shuran Song
Publication Year: 2023 , Page(s): 112307 - 112319
Abstract HTML

□ **A Time-Aware Approach for MOOC Dropout Prediction Based on Rule Induction and Sequential Three-Way Decisions**
Carlo Blundo; Vincenzo Loia; Francesco Orciuoli
Publication Year: 2023 , Page(s): 113189 - 113198
Abstract HTML

□ **An Open-Source Library of Phasor Measurement Unit Data Capturing Real Bulk Power Systems Behavior**
Shuchismita Biswas; Jim Follum; Pavel Etingov; Xiaoyuan Fan; Tianzhixi Yin
Publication Year: 2023 , Page(s): 108852 - 108863

Educational Data Mining Clustering Approach: Case Study of Undergraduate Student Thesis Topic

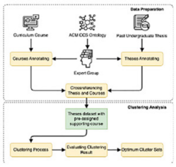
Publisher: IEEE   Cite This   PDF

Andre ; Nanik Suciati ; Hadziq Fabroyir ; Eric Pardede   **All Authors**

Abstract:

This study aims to investigate the potential of educational data mining (EDM) in addressing the issue of delayed completion in undergraduate student thesis courses. Delayed completion of these courses is a common issue that affects both students and higher education institutions. This study employed clustering analysis to create clusters of thesis topics. The research model was constructed using expert labeling to assign each thesis title to a computer science ontology standard. Cross-referencing was employed to associate supporting courses with each thesis title, resulting in a labeled dataset with three supporting courses for each thesis title. This study analyzed five different clustering algorithms, including K-Means, DBScan, BIRCH, Gaussian Mixture, and Mean Shift, to identify the best approach for analyzing undergraduate thesis data. The results demonstrated that k-means clustering is the most efficient method, generating five distinct clusters with unique characteristics. Furthermore, this study investigated the correlation between educational data, specifically GPA, and the average grades of courses that support a thesis title and the duration of thesis completion. Our investigation revealed a moderate correlation between GPA, thesis-supporting course average grades, and the time to complete the thesis, with higher academic performance being associated with shorter completion times. These moderate results indicate the need for further studies to explore additional factors beyond GPA and the average grades of thesis-supporting courses that contribute to delays in thesis completion. This study contributes to the understanding and evaluation of educational outcomes within study programs, as defined in the curriculum, particularly concerning the design and implementation of thesis topics. Additionally, the clustering results serve as a foundation for future research and offer valuable insights into the potential of EDM techniques to assist in selecting appropria...

(Show More)



Student Thesis Topic Clustering System Architecture.

Published in: IEEE Access (Volume: 11)

Page(s): 130072 - 130088   **DOI:** 10.1109/ACCESS.2023.3332818

Date of Publication: 14 November 2023   **Publisher:** IEEE

Electronic ISSN: 2169-3536

Funding Agency:

More Like This

An Outcome Based Analysis on Heart Disease Prediction using Machine Learning Algorithms and Data Mining Approaches
2022 IEEE World AI IoT Congress (AIIoT)
Published: 2022

Scalable and parallel machine learning algorithms for statistical data mining - Practice & experience
2015 38th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)
Published: 2015

Feedback

SECTION I.
Introduction

Educational Data Mining (EDM) involves data mining, machine learning, and statistical methodologies to extract valuable insights from educational datasets [1].

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2023.3332818

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Educational Data Mining Clustering Approach: Case Study of Undergraduate Student Thesis Topic

ANDRE[1,2], NANIK SUCIATI[1] (Member, IEEE), HADZIQ FABROYIR[1](Member, IEEE), and ERIC PARDEDE[3] (Senior Member, IEEE)

[1]Department of Informatics, Faculty of Intelligent Electrical and Informatics Technology, Institut Teknologi Sepuluh Nopember, Surabaya 60111, Indonesia

[2]Department of Informatics, Faculty of Engineering, Universitas Surabaya, Surabaya 60293, Indonesia

[3]Department of Computer Science and Information Technology, La Trobe University, Bundoora, VIC 3086, Australia

Corresponding author: Nanik Suciati (e-mail: nanik@if.its.ac.id).

This work was supported by Ministry of Education, Culture, Research, and Technology, Indonesia under grant of doctoral dissertation research year 2022.

ABSTRACT This study aims to investigate the potential of educational data mining (EDM) to address the issue of delayed completion in undergraduate student thesis courses. The problem of delayed completion of these courses is a common issue that impacts both students and higher education institutions. The study employed clustering analysis to create clusters of thesis topics. The research model was constructed by using expert labeling to assign each thesis title to a computer science ontology standard. Cross-referencing was employed to associate supporting courses with each thesis title, resulting in a labeled dataset with three supporting courses for each thesis title. This study analyzed five different clustering algorithms, including K-Means, DBScan, BIRCH, Gaussian Mixture, and Mean Shift, to identify the best approach for analyzing undergraduate thesis data. The results demonstrated that K-Means clustering was the most efficient method, generating five distinct clusters with unique characteristics. Furthermore, this research investigated the correlation between educational data, specifically GPA and the average grades of courses that support a thesis title and the duration of thesis completion. Our investigation revealed a moderate correlation between GPA, thesis-supporting course average grades, and the time to complete the thesis, with higher academic performance associated with shorter completion times. These moderate results indicate the need for further studies to explore additional factors beyond GPA and the average grades of thesis-supporting courses that contribute to thesis completion delays. This study contributes to understanding and evaluating the educational outcomes within study programs as defined in the curriculum, particularly concerning the design and implementation of thesis topics. Additionally, the clustering results serve as a foundation for future research and offer valuable insights into the potential of using EDM techniques to assist in selecting appropriate thesis topics, thereby reducing the risk of delayed completion.

INDEX TERMS computing classification system, undergraduate thesis, clustering analysis, k-means, ontology

I. INTRODUCTION

Educational Data Mining (EDM) involves data mining, machine learning, and statistical methodologies to extract valuable insights from educational datasets [1]. These datasets are often obtained from Learning Management Systems (LMS) and include detailed information such as assignment submission dates, LMS access logs, and social interaction within the platform. Additionally, less detailed data, such as student transcript historical data, which contains information on courses attended and grades received, can also be used in EDM. By analyzing these datasets, EDM can identify trends, patterns, and relevant information that may not be immediately apparent, allowing for a deeper understanding of educational processes and outcomes.

In the field of education, the use of EDM has become increasingly important for both learners and educators. Recent research has focused on applying advanced EDM techniques to analyze large datasets and extract meaningful patterns. EDM can be used to predict students' learning behavior [2], discover hidden information through clustering approaches [3]–[5], analyze the impact of a learning method, and advance scientific understanding. The technique can be applied to anomaly detection, association rule problems, clustering, classification, regression, and summary problems.

Clustering technique is an unsupervised learning method to find information from a large dataset and study the relationships and patterns between data. With clustering, data is grouped based on the proximity or similarity of their attributes. Studies found that clustering technique in education delivers benefits for the learner by delivering better recommendations such as adjusting learning styles, material selection, educator selection, and other benefits to improve stakeholder performance [4]–[6]. The clustering technique is also applicable in EDM. Romero and Ventura's taxonomy [7] provides insight into numerous research in this area. For example, the problems of determining significant contributors that affect learner performance [8], [9]; the development of student learning profiles based on learner behavior data [6], [10], [11]; and the prediction of miscellaneous academic outcomes (student dropout, learner performance, learner behavior) [12]–[15].

Several ways and points of views identify and investigate the issues pertaining to the educational domain [16]. Universities in Indonesia officially refer to the government regulation about undergraduate thesis duration, which is a six-month timeframe, as specified in the course syllabus. This course type has different characteristics from regular courses. The learning process involves mentoring between supervisors/advisors and students, working on real-world topics, and requiring the students' cognitive abilities. Students must tackle challenges critically, creatively, and independently for the undergraduate thesis to succeed. Nevertheless, delay in completing the undergraduate thesis is one of the issues where students, on average, finish the thesis in more than six months or two semesters.

In Educational Data Mining (EDM), researchers often use clustering methods for various tasks, such as classifying courses, predicting student behavior, and creating course recommendation systems. However, there is still a gap in the application of EDM techniques for categorizing undergraduate thesis topics. The selection of a thesis topic is of great importance to students as it can affect their academic performance and time management, especially in their final year of study. As a result, we proposed to investigate the most effective clustering results using historical data from undergraduate theses. What sets our research apart is the data preparation techniques we employed to generate high-quality clusters of undergraduate student theses. This contributes to a better understanding and assessment of the educational outcomes defined within study programs related to the design and execution of thesis topics. Furthermore, our study explores the correlation between students' GPAs and the average grades in thesis-supporting courses concerning the time required to complete their theses. The results of this research can be beneficial for further studies related to topic or thesis title recommendation systems, considering students' academic transcripts. Therefore, the outcomes of this further research can assist students in selecting the right thesis topic or title, thereby minimizing delays in completing their thesis.

The data labeling process focuses on expert judgment using the ACM Computing Classification System (CCS) as the standard of domain knowledge ontology in computer science. The technique uses expert judgment to select at least three courses contributing to the undergraduate thesis title. However, the data preprocessing method based on domain knowledge derived from ontology, in our best knowledge has never been proposed. Therefore, this research's urgency lies in the fact that the correct topic of the undergraduate thesis will significantly determine student success. The primary objective of this research is to examine how EDM can be applied in the analysis of previous undergraduate student thesis titles to uncover patterns and structures, leading to the identification of appropriate and accurate thesis topics through cluster analysis.

II. LITERATURE REVIEW

This section describes the literature on the benefits and applications of EDM, clustering-based approach, clustering algorithms and some evaluation metrics.

A. EDUCATIONAL DATA MINING

EDM is a cutting-edge paradigm for establishing ways to evaluate atypical sources of evidence that occur in educational environments [17]. Furthermore, employing those ways to comprehend students and their learning settings properly is essential. Classification, association, clustering, regression, forecasting, sequencing, and descriptive data mining techniques are still used and exploited in EDM [18]. In addition, EDM uses statistics and machine learning to enhance its effectiveness. Nonetheless, the fundamental goal of EDM is to discover knowledge from a collection of educational data that will benefit its stakeholders, primarily educators and learners [7].

EDM is widely used to assess and understand student motivation, attitudes, and behavior. For example, Rohlíkov [16] conducted research assessing student attitudes toward quizzes on Moodle LMS. The dataset comprised of 610 student activities from five Moodle quizzes. This research identified the reliability of the process mining method used in detecting student attitudes during quizzes. EDM can predict motivational deficits in the classroom by paying attention to the relationship between learning attitudes and student performance [19]. The questionnaire in the study was delivered to 180 students from 48 different courses at six different universities. It generates a motivation index that divides students into three categories: autonomous (those who learn through their activities in the LMS), controlled (students who update their data and information in the LMS), and e-learning driven (those who learn through their activities in the LMS, such as forums). The findings revealed a direct relationship between student performance (student results), autonomous groups, and e-learning motivation.

Another advantage of EDM is that it allows for creating a student profile model. Adaptive learning refers to a learning method that adjusts to the characteristics of each student and is more efficient, and has a more significant impact than conventional learning [20]. Dutt et al. [11] tested multiple clustering approaches on a dataset of 600 students and used the clustering technique to group students based on their historical data and learning behavior. The student profiles were developed with the clustering process and used as the basis of the construction of personalized e-learning. On the other hand, Miranda et al. [21] revealed that EDM can be used to predict student dropouts. This study included a total of 3,362 students with 51 features that consist of student academic records, family data, school characteristics and admission process. It implemented data-driven adjustments to the educational environment to better map and classify students. The study revealed the profile of at-risk students, which can be used to make early intervention. In conjunction with the student's perspective, the construction of a teacher profile has also been investigated in previous research. For example, Tondeur et al. [22] studied teacher profiling using the questionnaire and association rule methodologies to identify characteristics of effective teachers. The study found that trained teachers with more positive views placed a greater emphasis on collaboration, whereas those with negative attitudes placed a greater emphasis on feedback.

B. CLUSTERING APPROACH IN EDM

Cluster analysis is a data mining technique to group entities that share common traits compared to other entities belonging to other groups. The application is widely known for pattern recognition, multimedia retrieval, machine learning, and statistics and applicable in many subjects. Although there exists many clustering algorithms, one of the most popular and often used is the k-means algorithm. This clustering technique groups objects into clusters with the nearest mean. The k-mean algorithm iteratively divides the dataset into a k number of clusters so that each node will have a minimum sum of the squared distance to its respective centroid.

The common application of the clustering technique with k-means is a segmenting system prevalent in the education domain. Using e-learning data, Rawat and Dwivedi demonstrated how the clustering technique combined with the k-means algorithm can categorize students based on their features and behaviors [5]. The research used Moodle to collect students' usage data on assignments and quizzes. The students' interaction can be retrieved from several sources such as forums, chat, and messaging while doing quizzes and assignments in Moodle. This data was generated as log files of the Moodle server, which was later extracted and pre-processed. Their model generated three clusters that depict student profiles: non-active, average, and active. The number of clusters was then verified using the elbow and silhouette evaluation, a heuristic metric to determine the number of optimum clusters. They then created a course recommendation system on the Moodle platform that produced results based on the cluster of student profiles. Finally, they implemented the statistical metrics to evaluate the results, such as the root mean squared error (RMSE), precision, recall, and F1. The conclusion stated that future research should explore extracting implicit ratings from Moodle server log files to enrich the user-item rating matrix. This solution will help overcome the challenge of sparse data from users not providing detailed ratings due to a lack of motivation or incentives. Domain knowledge can also be extracted and integrated into the recommendation process to enhance the learner profile further and improve the quality of recommendations.

Additionally, one of the main areas for improvement in building a course recommendation system using a k-means clustering algorithm with data extracted from LMS is the need for more personalization in the recommendations. The reason is that the algorithm relies solely on clustering patterns in the data and does not consider individual students' unique preferences and needs. Furthermore, the quality of the recommendations is highly dependent on the quality of the data, which can be affected by various factors such as incomplete or inaccurate student profiles, biased or outdated data, and limited data available for specific user groups.

Aher and Lobo's combined k-means clustering with the association rule algorithm to provide optimal course selection recommendations[23]. The dataset used for the analysis consisted of course enrolment data from 100 distance learning students, which was processed through the k-means clustering technique to form n-clusters. The experiments used three different clustering methods: Simple K-means clustering, Farthest First clustering, and Expectation Maximization clustering algorithm. The association rule algorithm was then employed to determine the relationship between courses within the same cluster. The algorithm demonstrated that courses are more likely to be taken together and can be modeled through association rules. Furthermore, the result indicated that the Simple K-means clustering and Apriori association rule algorithm combination did not require the data preparation stage, and it produced more association rules, which increased the strength of the association rule. Future work includes exploring other combinations of data mining techniques for course recommendations in distance learning, integrating the system with existing e-learning platforms, and potentially applying the system to MOOCs. While combining k-means and association rules yields better candidate courses than using association rules alone, one of the key disadvantages of the association rule algorithm is the lack of context in the correlations found. Association rule algorithms only focus on finding correlations between items and need to consider the context in which these items occur, potentially leading to incorrect conclusions and rules that could be more meaningful. To address this issue, incorporating ontologies of knowledge into the analysis can provide a more contextual approach and better account for the context in which the correlations occur, leading to more accurate conclusions and meaningful rules. In this case, the researcher could have considered the curriculum structure since some courses may have prerequisites before the student can enroll.

Moubayed's study [4] emphasized the importance of analyzing student engagement levels on e-learning platforms through clustering. The research was based on 486 undergraduate science students and their activities recorded in the student log. The researchers established an engagement meter to quantify student involvement by measuring interaction and effort. Metrics related to interaction described how often the student engaged with the course content on the learning platform. In contrast, metrics related to effort described the level of exertion the student put in to finish the course assignments. The parameters are event date, type, location, start time, end time, and student ID. The optimal number of clusters was determined through evaluation, ranging from 2 to 5, based on prior literacy studies on engagement level classification. The study was then analyzed using silhouette methods, and it found that the number of two clusters representing low and high engagement levels was considered the best result. The future works are to test the model on a different course/semester to investigate its generalizability, collect and evaluate total time spent and average time per session to better gauge students' engagement, examine the impact of engagement metrics on student performance, and explore qualitative-based data analysis to modify course content based on student preferences. Another work currently under preparation explores identifying weak students based on their course performance. However, it is essential to note that before students enroll in a course, the researcher should thoroughly investigate the potential disadvantages of their engagement levels. Choosing the wrong course can lead to demotivation and a negative learning experience. Clustering student's prior study can help determine the best courses for each student, considering their engagement levels and learning styles. This personalization can lead to more personalized and compelling learning experiences and increase the likelihood of success. Therefore, students must make informed decisions about their course selection to ensure optimal engagement and success.

Another research work on learning behavior, in particular, examines student migration patterns regarding the conformity of courses taken with curriculum guidelines [24]. The clustering technique is based on a limited set of educational and academic records such as grades, courses, IP, and timestamps. These preferences require the factors influencing student migration patterns comprehensively. Additionally, using k-means algorithms to cluster similar objects in the education domain may not be the most appropriate method to effectively capture the complex relationships between student migration patterns and curriculum conformity. The proposed P-CEA method for analyzing dynamic educational data could be improved by integrating demographic data, conducting social network analysis, developing a predictive model for identifying at-risk students, exploring cross-disciplinary applications, and refining the method by adjusting weightings or incorporating additional clustering algorithms. These future works could enhance the understanding of factors contributing to student success or failure and provide early warnings and counseling to prevent students from dropping out.

Additionally, the study could benefit from incorporating ontologies in computer science to address these limitations. Ontologies provide a structured and standardized representation of knowledge that can be used to capture the complex relationships between different entities effectively. By incorporating ontologies, the study could better capture the factors influencing student migration patterns, such as student background, academic interests, and socio-economic factors. Additionally, using ontologies would provide a more comprehensive representation of the data, making it easier to analyze and interpret the results. This technique leads to a more in-depth understanding of the relationship between student migration patterns and curriculum conformity.

Different clustering analysis research on EDM has been extensively conducted in recent years. However, more work still needs to be done on the undergraduate thesis dataset. An undergraduate thesis is a comprehensive research project completed by students in their final year of undergraduate studies. It typically involves an in-depth investigation of a research question or topic of the student's choice and the analysis and interpretation of data to conclude. Cluster analysis is a powerful data mining technique that can be used to identify patterns and similarities in large datasets. Applying cluster analysis to undergraduate thesis projects can reveal insights that may only be apparent through traditional analysis methods, such as identifying common themes or trends across multiple thesis projects. This information can inform future undergraduate students' curriculum design and research topics and identify potential areas for further research. Additionally, cluster analysis can help students better understand the broader context of their research and how it relates to similar research in their field. Ultimately, using cluster analysis to analyze undergraduate thesis projects can help identify valuable insights and inform future research directions.

C. CLUSTERING ALGORITHM

Clustering is a machine-learning technique that groups data items according to similarity or distance. The purpose of clustering algorithms is to split data into groups or clusters, where each group contains comparable data points and is unique. Clustering algorithms facilitate the exploration and comprehension of detailed information by grouping similar data points. They can compress big datasets, detect anomalies, assist with recommendation systems, segment images, and identify market segments. Clustering techniques provide non-obvious insights into patterns, trends, and correlations within data. They can be utilized in numerous fields, including data mining, machine learning, image processing, and marketing.

There are numerous clustering methods include centroid-based, hierarchical, density-based, and distribution-based methods[25]. Centroid-based clustering is a clustering algorithm that combines similar data points based on their proximity to the centroid, which serves as the cluster's representative point. In this clustering, the algorithm allocates each data point to the nearest centroid after randomly selecting K centroids (where K is the desired number of clusters). The program then calculates the new centroids as the mean of all the data points in each cluster and repeats the process of assignment and recalculation until convergence is reached.

K-means clustering and the Mean-shift algorithm are two of the most used methods for clustering based on centroid. K-means aims to minimize the sum of squared distances between each data point and its assigned centroid. The algorithm updates the centroids and reassigns the data points iteratively until convergence. K-means is computationally efficient, making it suited for big datasets, and it has been implemented in several applications, such as picture segmentation, document clustering, and customer segmentation. Mean-shift algorithm is another centroid-based clustering algorithm that shifts each data point iteratively toward the most significant density of data points until it reaches a convergence point, which acts as the cluster's centroid. The algorithm estimates the thickness of the data points using a kernel density function, and the shifting procedure moves each data point in the direction of the steepest climb of the density function. Mean-shift is good at identifying clusters of different forms and sizes, making it appropriate for clustering applications like picture segmentation and object tracking.

Both the K-means and Mean-shift algorithms have advantages and disadvantages. K-means is sensitive to the initial selection of centroids, resulting in various clusters. On the other hand, Mean-shift is computationally more expensive than K-means, rendering it unsuitable for large datasets. Moreover, Mean-shift may yield an arbitrary number of clusters, and determining the appropriate number of clusters can be challenging. Despite these drawbacks, centroid-based clustering is widespread due to its relative efficiency and effectiveness in discovering clusters in high-dimensional data.

Hierarchical clustering is a technique that groups data points with similar characteristics based on proximity. With this type of clustering, the algorithm generates a hierarchy of clusters, beginning with individual data points as the initial clusters and merging them iteratively until all data points belong to the same cluster. The two primary types of hierarchical clustering are agglomerative and divisive. Agglomerative begins with each data point as its cluster and continues by combining the most comparable clusters until all data points belong to a single cluster. In contrast, divisive clustering begins with all data points in a single cluster and recursively divides them into smaller groups. The algorithm determines the similarity between clusters or data points using a distance metric in all hierarchical clustering methods. Several metrics, such as Euclidean distance, cosine distance, and correlation distance, can be used to calculate the distance between two points.

BIRCH (Balanced Iterative Reduction and Clustering) is a common hierarchical clustering technique developed to cluster massive datasets effectively. BIRCH uses a tree-based data structure to represent data points and clusters, allowing it to progressively create and update the clustering model as new data points are introduced. Additionally, the technique employs a clustering mechanism that compresses the data points, reducing the memory requirement and enabling BIRCH to handle big datasets efficiently. The branching factor, the threshold number, and the number of clusters are three critical factors that can be modified to maximize BIRCH's clustering performance. The branching factor sets the maximum number of child nodes associated with each internal node in a tree. The threshold value defines the maximum number of data points an internal node can carry before splitting into two child nodes. Finally, the number of clusters determines the desired number of clusters for the output.

Density-based clustering is a technique that clusters densely packed data points together while isolating less dense regions. This clustering technique is excellent for detecting clusters of arbitrary shape and can handle noise and outliers. DBSCAN is the most prevalent density-based clustering algorithm (Density-Based Spatial Clustering of Applications with Noise). Identifying dense regions of data points and allocating them to the same cluster is how DBSCAN operates. The algorithm requires two parameters: the minimum number of data points needed to build a dense region (called minPts) and a distance measure that determines the radius surrounding each data point within which other data points are considered neighbors.

DBSCAN begins by randomly selecting an unvisited data point and determining if it has at least minPts neighbors within a distance measure-defined radius. If the point has sufficient neighbors, it is placed in a new cluster. If not, the point is labeled as noise or a boundary point, and the algorithm continues to the next unvisited point. Next, DBSCAN checks the neighbors of each newly added data point to a cluster and adds them to the same cluster if they have sufficient neighbors within the radius. The procedure is repeated until all dense sections of data points have been allocated to clusters, and all noise or boundary points have been found.

Data noise and outliers may be handled using DBSCAN and other density-based clustering techniques, which is an advantage. In addition, they may recognize clusters of arbitrary shape, which is challenging for existing clustering algorithms that assume clusters are spherical or have a specific shape. However, a downside of DBSCAN is that it requires careful parameter adjustment to produce optimal results, and the clustering outcome can be sensitive to the distance measure and minPts value. In addition, the technique may perform poorly on datasets or clusters with drastically varied and fluctuating densities.

Distribution-based clustering algorithm assumes data points are created from a probability distribution and employs statistical methods to discover data groups. This clustering technique is excellent for detecting clusters that follow a specific distribution, such as the Gaussian or Poisson distribution. Gaussian Mixture Model is a popular approach for distribution-based clustering (GMM). GMM implies that the data points are derived from a mixture of Gaussian distributions, each cluster representing a different Gaussian component. Using an iterative technique such as Expectation-Maximization, the process estimates the Gaussian mixture model's parameters, such as the mean and covariance of each element (EM).

The GMM algorithm begins by randomly initializing the Gaussian mixture model's parameters. Using Bayes' rule, the computer iteratively calculates the likelihood that each data point corresponds to each component of the Gaussian mixture model. Based on these probabilities, the algorithm modifies the parameters of the Gaussian mixture model to match the data better. The procedure is repeated until the algorithm reaches a solution. GMM and other distribution-based clustering methods may need to perform better on datasets with irregularly sized or shaped clusters. In addition, they may necessitate careful parameter tweaking and be sensitive to the number of components used for the mixture model. In general, distribution-based clustering is a robust technique that can handle various data kinds and applications, especially when the data points follow a particular distribution.

D. CLUSTERING PERFORMANCE EVALUATION

Unlike supervised approaches, where ground truth is used as an indicator of clustering performance evaluation, as an unsupervised approach the clustering results obtained using k-means do not have a specific evaluation measure associated with them. In this case, because the number of clusters depends on the initial input, a particular approach for evaluating the model's performance based on the number of clusters is required. They include the Elbow method and metric evaluation such as Silhouette analysis, Calinski-Harabasz, and Davies-Bouldin score index.

The Elbow method indicates the optimum number of clusters based on the sum calculation of the squared distance between the data points and the cluster centroid. The results of the calculation are then plotted onto a diagram, which resembles an "elbow" shape. A heuristic rule of thumb is that the optimal number of selected clusters is reached when the graph exhibits diminishing returns. Then, the graph moves approximately in a straight line parallel to the X-axis. The K value that corresponds to this point is the optimal K value or the ideal number of clusters.

Silhouette analysis metric can identify the quality and performance of cluster results. The silhouette coefficient determines the degree to which clusters are separated from one another. The formula for calculating coefficients is as shown in (4).

$$s(o) = \frac{b(0) - a(0)}{\max\{a(0), b(0)\}} \tag{4}$$

With $a(0)$ denotes the average distance between point o and all other data points within its cluster. The $b(0)$ is the average distance between o and all clusters to which o does not belong, expressed as a minimum average distance. A coefficient close to -1 indicates that the number of clusters is not optimal. While a value close to 0 indicates overlapping clusters. As a result, to construct the best cluster, it is often desired that the coefficient be significant and close to 1.

Silhouette analysis evaluates the clustering quality of each data point by calculating the distance between the data point and the other points in its cluster, as well as the distance between the data point and the points in the following neighboring cluster. Higher silhouette scores indicate superior cluster quality. Conversely, a high silhouette score suggests that a data point is well-matched to its cluster and poorly matched to nearby clusters, which suggests that the clustering is effective.

The Calinski-Harabasz index is another cluster evaluation metric [26]. This clustering validation calculates the ratio of the sum distribution of data points within and between clusters. The Calinski-Harabasz calculation formula can be seen in (5)

$$s = \frac{tr(B_k)}{tr(W_k)} \times \frac{n_E - k}{k - 1} \tag{5}$$

Where s is the Calinski-Harabasz score resulting from the division of the dispersion ratio between clusters $tr(B_k)$ with the dispersion ratio within the cluster $tr(W_k)$. $n_E$refers to the number of data, and k is the number of clusters. Calinski-Harabasz evaluates the ratio between cluster variation to within-cluster variance, reflecting how effectively the clusters are separated. A high Calinski-Harabasz score suggests that the clusters have unique patterns and a wide gap between their means.

Davies-Bouldin metric is another clustering evaluation method that calculates the average similarity for each cluster compared to other similar clusters [27]. Davies-Bouldin calculates the average similarity between each cluster and its most similar cluster by calculating the distance between the cluster means and the cluster sizes. As opposed to Calinski-Harabasz , a low Davies-Bouldin index suggests that clusters are well-separated and distinct, with limited overlap or similarity. The Davies-Boulding equation can be seen in (6)

$$S_{i,j} = \frac{P_i + P_j}{D_{i,j}} \tag{6}$$

P_i is the average distance from the data point to the centroid in cluster i. The same also applies to P_j. Meanwhile, $D_{i,j}$ shows the distance between the cluster centroids i and j respectively. So, the ratio between the average distance between the two clusters i and j and the distance between the clusters is shown in the $S_{i,j}$ similarity value.

In conclusion, Silhouette metric examines the quality of the clustering of individual data points, Calinski-Harabasz method analyzes the separation and distinctness of the clusters as a whole, and Davies-Bouldin metric evaluates the similarity and overlap between the clusters. Depending on the unique objectives and characteristics of the clustered data, each indicator can provide valuable insights into the performance of clustering methods.

III. METHODOLOGY

This section is broken up into distinct parts. The first part describes some underlying problems that contribute to late completion of student thesis. The second half presents clustering architecture, and the last part explains the clustering validation with different techniques.

A. UNDERGRADUATE THESIS PROBLEMS

It is envisaged that the final undergraduate thesis will be completed within six months, as specified in most course syllabus. Based on statistics, students take more than six months to complete the thesis. For example, in a case study conducted at the Informatics Engineering Department of the University of Surabaya, 300 students completed their undergraduate thesis during the graduation period of 2016-2021. The average amount of time required to finish the thesis is 8.5 months. It takes the shortest (3.13 months) and longest (24.36 months) amount of time. Students typically spend two semesters working on their undergraduate thesis.

Thesis completion time affects the study duration as one contributing factor that determines the quality of the university as quantified by the standard accreditation score. In addition, students who finish their studies on time have benefits in terms of study costs, scholarship consideration, and others.

The determinants of the delayed completion of the undergraduate thesis are motivation, cognitive abilities, and the supervisor's role [2], [28], [29]. Three factors contribute to the low motivation:

a) low autonomy: students do not like the topic of their final assignment, or students do not have room to make decisions [14], [30];

b) low usefulness: students feel that the topics they are working on have no impact or are less useful; and

c) general/academic procrastination: students procrastinate which has a small but cumulative impact on late completion [31]–[33].

The above factors contribute to low motivation in working on student theses would affect the delay in finishing their thesis on schedule. Therefore, as precautionary steps to minimizing the problems, the student should be able and allowed to choose their appropriate thesis topic. In doing so, the student will benefit by having a range of suitable topics as their consideration to choose one as their preference. However, providing a range of relevant topics for students is challenging because the breadth and complexity of potential topics can be overwhelming, and the scope may differ among universities. To tackle this problem, this study uses EDM techniques to analyze historical undergraduate thesis data and uncover hidden patterns. The goal is to determine suitable clusters of thesis topics that students can choose based on their interests and proficiency. This analysis utilizes the standard Computing Classification System (CCS) ontology, which categorizes fields in computer science into 13 primary knowledge domains with branches up to 4 levels of depth. By mapping each undergraduate thesis title to multiple knowledge domains, this study aims to provide valuable information for students to make informed decisions about their thesis topic.

B. THE CLUSTERING SYSTEM ARCHITECTURE

We present a clustering system architecture of students' undergraduate thesis with main novelties focused on the involvement of computer science ontology to determine the thesis base supporting knowledge and curated by experts. Our methodology begins with data preparation that involves expert decisions to annotate past undergraduate theses and supporting courses. The clustering process can then start with the pre-annotated dataset. Finally, we investigated the proper clustering algorithm configuration within the clustering process step to produce the optimum clusters. Fig 1 visualizes our system architecture.

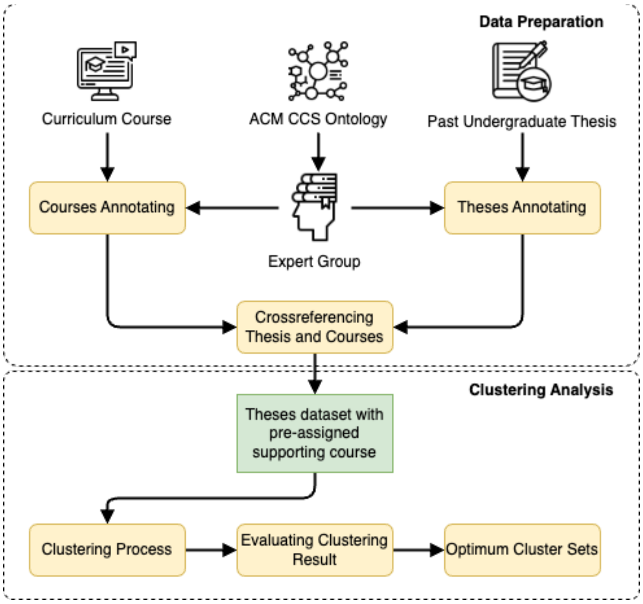


FIGURE 1. Student Thesis Topic Clustering System Architecture

1) DATA PREPARATION

Our clustering system architecture begins with data preparation. Three data sources are involved: the curriculum courses, the CCS ontology, and the past undergraduate thesis. The curriculum courses source highly depends on each institution's curriculum design; however, they should support the student's undergraduate thesis as the penultimate course within a degree. In our case, we employ 72 courses, consisting of 23 compulsory courses; the rest are elective courses. Our research only chooses limited mandatory courses because we hypothesize that students had already mastered the introductory course upon reaching the final semester. On the contrary, elective courses include more advanced subject matters and frequently cover multiple disciplines or competencies simultaneously.

The formal ontologies to establish the ground knowledge of the forthcoming annotation use CCS, accessible online at https://dl.acm.org/ccs. CCS is used to classify research publications in computer science. We see that ontology is relevant to the needs of our system architecture and that the Computer Science curriculum in most universities worldwide has also adopted the CCS in their curriculum design.

The final data source is students' undergraduate thesis records. We extract 300 past student undergraduate theses from the class of 2016–2021 in Informatics degree, Universitas Surabaya. Our dataset can be downloaded from this repository: https://github.com/scancampy/student-thesis-dataset. In this study, all 300 titles contain different computer

TABLE 1
COMPUTING METHODOLOGIES CCS ONTOLOGY SNIPPET

Top-level	2nd tier	3rd tier	4th tier
Computing methodologies			
↳	Symbolic and algebraic manipulation		
	↳	Symbolic and algebraic algorithms	
		↳	Combinatorial algorithms, Algebraic algorithms, Nonalgebraic algorithms . . .
		Computer algebra systems	
		. . .	
	Parallel computing methodologies		
	↳	Parallel algorithms	
		↳	MapReduce algorithms, Self-organization, Shared memory algorithms
		Parallel programming languages	
	Artificial intelligence		
	↳	Natural language processing	
		↳	Information extraction, Machine translation, Discourse, dialogue and pragmatics
		Knowledge representation and reasoning	
		. . .	
	. . .		

science knowledge areas such as Information Management, Computational Science, Intelligent Systems, Software Engineering, Graphics and Visualization, Human-Computer Interaction, etc. We aim to cluster the undergraduate computer science topics and highlight each cluster's insight and characteristics.

Following the data collection, the data preparation involves forming an expert group for analysis, as illustrated in Fig 1. The group comprises of the laboratory head, supervisor, and curriculum design team. The experts undertake three activities, beginning with a study of the computer science domain hierarchy derived from the CCS ontology. The relation between the experts and the CCS data source is shown in Fig 1. The CCS hierarchy consists of knowledge area ontologies that extend up to the fourth level. Each level contains knowledge areas of computer science, with the top-level hierarchies representing general knowledge areas and deeper levels showing more specialized knowledge areas. Table 1 offers an example of the Computing Methodologies ontologies hierarchy, one of the top-level hierarchies of the CCS ontology, and provides more specialized knowledge areas at a deeper level.

We assign formal ontologies to each course to ensure alignment between courses and thesis requirements. Experts manually annotate each course by examining the terminology in the syllabus, lesson plans, and other relevant documents to arrive at appropriate decisions. We use the CCS ontology to label each course, as it contains a large amount of material for each general topic. Multiple ontologies may be related to each course, and we use coefficients to determine their contributions. For example, in the Big Data Analytics course shown in Table 2, the Design and Analysis of Algorithms ontology has the highest coefficient, followed by the Visualization and Machine Learning ontologies. By cross-referencing the thesis and course ontologies, we can identify courses that match the thesis requirements and decide which ones students should master. The coefficients indicate how extensively a particular ontology contributes to the course content.

After annotating all 72 courses, experts continue with annotation of the thesis. To annotate thesis titles with related CCS ontologies, experts evaluate each thesis document's title, abstract, and keywords to choose relevant ontologies. For example, in table 3, the thesis titled "Development of Decision Supporting Systems Using the Weighted Product Methodologies for Credit Installment of Vehicle Sales" is annotated with three contributing ontologies based on their relevance to the title (high, medium, and low). Next, the expert selects the deepest branch in the ontology structure. As shown in table 3, the contributing knowledge areas are Operation Research, Information System Application, and Software Notation and Tools. The three deepest and most relevant ontologies are multi-criteria optimization and decision-making, decision support systems, and frameworks, the 3rd tier components rooted in operations research. Expert involvement strengthens the accuracy and reliability of labeling outcomes. Utilizing this ontology, we look at the thesis topic's relevance to the main deepest ontology branch. The deeper the selected ontology, the more accurate the classification process.

2) ONTOLOGIES CROSSREFERENCING

We annotate the dataset using CCS ontologies to identify the knowledge areas relevant to a particular thesis. We match the ontologies of courses and thesis titles using a cross-referencing process, as shown in Figure 1. We develop expert annotation tools, a web-based system, to help experts conduct the annotation process for courses and theses [34].

TABLE 2
COMPUTING METHODOLOGIES ACM CCS ONTOLOGY SNIPPET

Course	ACM CCS	Coefficients
Modeling and Simulation	Modeling and Simulation	0.6
	Probability and statistics	0.3
	Mathematical analysis	0.1
Decision Support Systems	Operation Research	1
Big Data Analytics	Design and analysis of algorithms	0.5
	Visualization	0.3
	Machine learning	0.2
Artificial Intelligence for Game	Theory and algorithms for application domains	0.5
	Artificial intelligence	0.4
	Cross-computing tools and techniques	0.1

Our expert annotation tools aid the annotation process, making it semi-automatic. This means the tools automatically select the three courses that contribute the most to a given thesis title based on the cross-referencing and coefficients. However, experts can still review and manually adjust the results if needed. Figure 2 shows a screenshot of the annotation tool, displaying the top three courses for a specific thesis title. Experts first annotate each thesis with relevant ontologies to determine the courses that contribute to a thesis title. We then use cross-referencing to identify all courses that share the same ontologies with the thesis. Each course contains different ontologies with varying coefficients indicating their degree of contribution to the course content. The annotation tool sorts these courses in descending order of their coefficient values to determine the most significant contributors to the thesis. The tool then automatically selects the top three courses with the highest coefficients, which experts can review and manually adjust if needed. However, because we involve more than one expert in annotating a single thesis, it may be common to appear that there are disagreements among the experts in selecting the courses. Our tools can highlight the dispute by providing an easy interface and facilitating the experts to make vote [34]. In conclusion, our streamlined approach enables experts to identify the most relevant courses for a given thesis title. We obtained a dataset of annotated past thesis titles, each with three contributing courses, which we can use for clustering analysis.

To facilitate the upcoming clustering analysis, we have selected the three courses that support each thesis title as the dataset features. These features are critical in ensuring that the resulting clusters accurately reflect the knowledge areas covered in each thesis. We choose this because the thesis is a crucial component of a student's academic career, allowing them to showcase their knowledge and skills. To excel in their thesis, students must have a solid understanding of various supporting theories and concepts. By taking advanced and specialized courses, students can deepen their knowledge and skills beyond the introductory level [35]. For instance, in artificial intelligence, a student may take advanced courses on in-depth algorithms, such as genetic algorithms, or deep learning courses that focus on artificial neural networks.

TABLE 3
ANNOTATION RESULT DATA SNIPPE

Priority	1st tier	2nd tier	3rd tier	4th tier
High	**Operations research**			
	↳	Decision analysis		
		↳	Multi-criterion optimization and decision-making	
Medium	**Information systems applications**			
	↳	Decision support systems		
Low	**Software notations and tools**			
	↳	General programming languages		
		↳	Language features	
				Frameworks

In addition, most universities offer elective courses that students can take to explore their interests and expand their knowledge. For example, at Universitas Surabaya, where the case study was conducted, students typically take 3-5 electives. Based on these reasons, we have determined that each undergraduate thesis should have at least three supporting courses to enhance the fluency and comprehensiveness of the thesis. This approach ensures that students have a solid foundation in the relevant knowledge areas and can produce high-quality work.

3) IMPLEMENTATION OF CLUSTERING TECHNIQUES

Clustering analysis, especially an algorithm that uses distance metrics, requires a numerical representation of each data point, commonly achieved using encoding methods that assign specific values to each data point. However, a label encoding method is needed for categorical data, such as our dataset that refers to courses supporting individual thesis titles. Label encoding assigns a unique numerical value to each category in the dataset, allowing the categorical data to be represented numerically for clustering analysis. Our study organizes the courses based on knowledge areas using the ontology from CCS. Table 4 depicts the snippet of each course's encoding label to numerical representation associated with knowledge areas (root ontology). Our study demonstrates that the choice of encoding is not critical if it preserves the relative distance between data points, and the clustering results should be similar.

In the following steps, we investigate the best clustering algorithm to deliver an optimal cluster set to extract the features that are concealed from view. We have selected five clustering algorithms: k-means, Mean-shift, DBScan, BIRCH, and Gaussian Mixture.

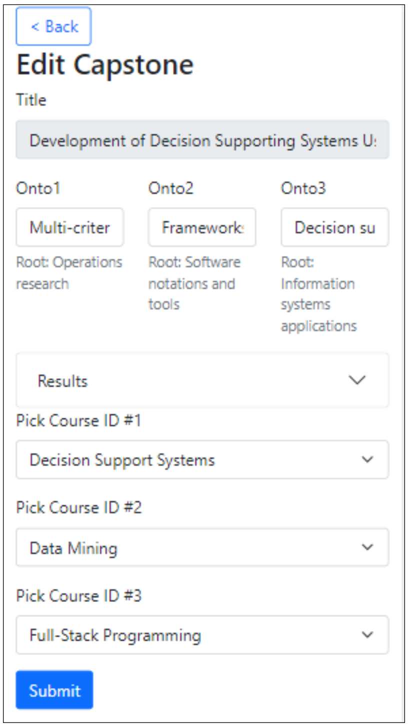
TABLE 4
SNIPPET OF COURSE ENCODING

Encoding	Root Ontology	Course
1	Software and Its Engineering	Web Programming
2		Web Framework Programming
3		Full-Stack Programming
	. . .	
13	Networks	Computer Network
14		Distributed Programming
15		Advanced Computer Network
	. . .	
18	Human Centered Computing	Human Computer Interaction
19		Mixed Reality
20		Immersive Computing
	. . .	
53	Computing Methodologies	AI Fundamental
54		Machine Learning
55		Modeling and Simulation
	. . .	

K-means and Mean-shift are based on the centroid, while the rest are based on density, hierarchy, and distribution.

Different clustering methods reveal a variety of distinguishing traits. A clustering method known as density-based clustering groups data points that are concentrated in an area with a high density. This cluster technique does not consider outliers and works to ensure that the cluster center point is located at the clustered data point. When doing distribution-based clustering, careful consideration is given to the probability that the algorithm will include a data point in the cluster. The further away a data point is located from the cluster's epicenter, the lower the possibility that the algorithm will include it in the cluster. Calculating the squared distance from the predefined centroid is how each data point in the centroid-type cluster is created. Adjustments are made to determine the new centroid's location at the end of each iteration until the convergence criterion is met. Finally, hierarchical clustering is a subtype designed solely for use with hierarchical datasets.

Among those popular centroid-based clustering algorithms, k-means and Mean-shift have been established as the solid algorithms that produce optimum cluster sets. In k-means, we decided the number of K clusters the algorithm process and deliver. However, K is not the best possible choice. We use the elbow technique, a heuristic approach to determine the scoring index that defines the quality of clusters' results. The degree of variance in each cluster number can be determined using the elbow approach, which involves calculating the square distance that separates each point from the cluster's center. The steps for the k-means clustering process are as follows. First, the data encoding process converts the dataset value into a numeric representation. The clustering method can only read numeric data. Encoding values are organized into groups according to the extent of their underlying scientific basis. For example, data science and artificial intelligence courses will use an encoding value of 1–50. And software engineering and enterprise systems courses use an encoding value of 50–100. This procedure is applied to all fields. Secondly, we determine the number of clusters. This number is expressed as the optimal number, as proven by the elbow method in the results and discussion section. Finally, the clustering process is conducted. The algorithm runs iteratively until the convergence is accomplished and all data points have been appointed to the nearest optimum cluster center.



Edit Capstone

Title

Development of Decision Supporting Systems U:

Onto1	Onto2	Onto3
Multi-criter	Framework	Decision su
Root: Operations research	Root: Software notations and tools	Root: Information systems applications

Results

Pick Course ID #1

Decision Support Systems

Pick Course ID #2

Data Mining

Pick Course ID #3

Full-Stack Programming

FIGURE 2. Expert Annotation Tools Automatically Determined Three Supporting Courses

The Mean-Shift technique, which is another centroid-style clustering algorithm, is an additional alternative to the k-means technique. This algorithm is unsupervised learning without the need for any parameters. This algorithm works by first computing the mean of the dataset, and then shifting each data point to the area of the cluster mean that is closest to the center of that mean. Shifting this value does not change the original value but only keeps the label. In most cases, mean-shift performs admirably for image datasets [36].

As in case of density-based DBScan algorithm, the two most important factors are the eps and the minimum data point (minPts). The eps parameter is used to configure the maximum distance that can exist between two data points before those points are no longer considered to be part of the neighborhood. In contrast, the minPts parameter specifies the least amount of data points that must be present in a cluster in order for it to be considered valid. Some research proposed an automatic method to determine those parameters[37]–[39].

Furthermore, the BIRCH clustering method is known to be effective for large amounts of data. This method condenses the data set into a succinct summary while maintaining as much of the information as feasible. To reduce the amount of time needed to complete the operation, the clustering procedure is applied to the compact dataset version. The branching factor, the threshold, and the number of clusters are all examples of BIRCH parameters. The branching factor is the maximum number of CF sub-clusters that can be found on each individual node. The maximum number of data points that can be contained within a sub-cluster of the CF Tree's leaf node is referred to as the threshold. While n cluster is the anticipated total number of target clusters that will exist once the BIRCH algorithm has been run to completion. The BIRCH parameter can be determined automatically [40].

The Gaussian Mixture Technique is a clustering algorithm that uses a distribution-based approach. This algorithm performs a clustering process similar to k-means. Gaussian Mixture differs from k-Means in that it considers the distribution as well as the covariance of the data distribution. This allows for the visual shape of the algorithm outputs to change, as opposed to k-Means, which often produces circular output. Both hard and soft clustering can be

TABLE 5
CLUSTERING TECHNIQUE CONFIGURATION

Technique	Based	Configuration
K-Means	Centroid	`Number of cluster = 5, kmeans++`
DBScan	Density	`Eps = 7, minPts = 30`
BIRCH	Hierarchical	`Branching factor = 50, threshold =7, cluster =5`
Mean-Shift	Centroid	`N/A`
Gaussian Mixture	Distribution	`n_component = 9`

accomplished with the help of this approach. In contrast to hard clustering, soft clustering assigns a probability to each data point regarding whether it belongs to a cluster. The parameter known as n component is used by the Gaussian Mixture algorithm, and it specifies the number of clusters that will be produced by the method. To determine the number of clusters, we can use the Akaike Information Criterion (AIC), and Bayesian Information Criterion (BIC), which evaluate the complexity of the dataset.

IV. RESULT AND DISCUSSIONS

This section is divided into two parts. The first part explains the results of clustering that were obtained using popular clustering techniques. The second part presents the results of clustering based on ontology content and the correlation between clustering results and GPA as well as supporting course grades.

A. EXPERIMENTING WITH CLUSTERING TECHNIQUES

The results of data preprocessing are 300 thesis titles ready to be processed further through the clustering algorithm. The clustering process uses Python, the Sci-kit library, and the Google Collab Notebook platform. The dataset synthesizes 300 thesis titles that have gone through the data preprocessing. We perform the optimal configuration for each clustering technique in conducting the clustering process. Before the process of clustering, we ensure that we are using the most effective configuration for each different type of clustering. The configuration of each different clustering method is presented in Table 4.

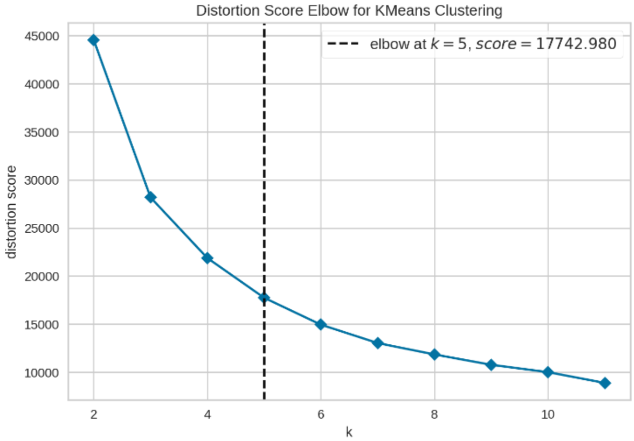


FIGURE 3.Elbow Metric Graph Present the Number of Optimum Cluster

We conduct the first experiment with the k-means clustering with the dataset that had been annotated by three different supporting courses. As the result, our k-mean clustering algorithm identifies five as the cluster. The confirmed value can be demonstrated with complete confidence using the elbow method of measurement, as depicted in Fig 3.

This is due to the fact that the cut-off points for any number of clusters above five is regarded to have converged and increasing the number of clusters does not significantly alter the results. In addition, another centroid-based clustering algorithm called Mean-Shift doesn't require configuration because it is a non-parametric unsupervised learning algorithm and doesn't account for any cluster or feature shapes.

For DBScan we use two parameters: MinPts and eps. First, we conduct trial and error by experimenting with various MinPts and eps values to produce the best possible clustering result. MinPts indicate the minimum number of data points required to determine a cluster. After deciding the MinPts, a range of values for eps is tested to find the best clustering results. Using a MinPts value of 30 and an eps value of 7 resulted in the best clustering performance. We also apply this manual testing method to the BIRCH algorithm and found that a branching factor of 50, a threshold of 7, and a total of 5 clusters produced the best results.

The Gaussian Mixture Technique is a distribution-based method that needs the cluster number to be set up at the

TABLE 6
RESULT OF CLUSTERING TECHNIQUE PERFORMANCE

Technique	Num. of Cluster	Execution Time (seconds)	Silhouette score	Calinski-Harabasz score	Davies Bouldin score
K-Means	5	0.03181781769	**0.4206**	**190.1684**	0.858
BIRCH	5	**0.01628289223**	0.3480041781	133.9241321	**0.6055691236**
Gaussian Mix	9	0.06254787445	0.405	140.626	0.901
DBScan	2	0.1276900768	0.103	11.198	3.738
Mean-Shift	3	1.660001612	0.395	98.309	0.946

beginning. This number can then be determined by the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC), which measure the complexity of the dataset and provide the ideal number of clusters, which in our case is 5.

We utilize three commonly used quality metrics to measure the performance of different clustering methods on the undergraduate student thesis dataset: Silhouette score, Calinski-Harabaz index, and Davies-Bouldin index. Silhouette score measures the similarity of a point to its cluster compared to other clusters. The score ranges from -1 to 1, where a score of 1 indicates that the point is well-matched to its cluster and poorly matched to neighboring clusters. A score of 0 indicates that the point is equally similar to neighboring clusters to its own cluster, and a negative score means that the point is more identical to neighboring clusters than its own. Calinski-Harabaz index measures the ratio of between-cluster variance to within-cluster variance. Higher values indicate better-defined clusters, with larger separations between the clusters and more minor variances within each cluster. The Davies-Bouldin index measures the average similarity between each cluster and its most similar cluster. Lower values indicate better clustering, with tighter and more separate clusters.

We have presented the results of our performance and evaluation metric tests in Table 6. Our analysis of various clustering methods found that K-Means performed the best in the Silhouette score, Calinski-Harabasz, and Davies Bouldin metrics. K-Means is a distance-based clustering algorithm that works well when the clusters have a spherical or circular shape, and the data points are well separated. On the other hand, BIRCH was the fastest algorithm, with an execution time of 0.01628 seconds. This is because BIRCH is an algorithm that constructs a tree-based data structure to represent the data distribution and perform clustering on a condensed version rather than the entire dataset. Likewise, the Gaussian Mixture algorithm produced a relatively large number of clusters (9) compared to the other algorithms. The reason is that the Gaussian Mixture models are flexible and can model complex shapes of clusters. Hence, they fit the data better when the underlying distributions are complex or have multiple modes.

The clustering results indicate that both density-based algorithms: DBSCAN and Mean-Shift, produced relatively poor results compared to the other clustering algorithms. This is because our dataset has varying densities or irregularly shaped clusters and is not concentrated. In datasets with varying densities, choosing appropriate values for the algorithm's parameters may be difficult, such as the eps and the minPts. Specifically, DBSCAN has a Silhouette score of 0.103, a Calinski-Harabasz score of 11.198, and a Davies Bouldin score of 3.738, indicating that the clusters are not well-separated and are overlapping. The Mean-Shift produces a Silhouette score of 0.395, Calinski-Harabasz score of 98.309, and Davies Bouldin score of 0.946 with three clusters, but is the slowest clustering algorithm with an execution time of 1.6600 seconds. The Mean-Shift algorithm is a density-based clustering algorithm and can be computationally expensive when dealing with large datasets. The algorithm's slowness can be attributed to factors such as the bandwidth parameter, convergence criteria, and dataset size.

Short execution time is crucial when selecting an algorithm to ensure optimal performance. As more data is added regularly, an efficient algorithm becomes imperative to ensure fast and accurate results. It is important to consider this factor in the algorithm selection process. Following the preceding discussion, we conclude that, in our case, computer science students' undergraduate thesis dataset would benefit from applying the k-means clustering technique.

However, this experiment shows that when applying clustering analysis, we must consider four aspects: dataset characteristics, understanding goals/research questions, using evaluation metrics, and scalability. Understanding the data's structure, size, and distribution can help select a suitable clustering algorithm. Additionally, understanding the

TABLE 7
STATISTICAL SUMMARY OF CLUSTER BASED ON ACM CCS ROOT ONTOLOGY

	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5	Mean
Software Engineering	**48.00%**	**37.28%**	**46.67%**	31.01%	0.00%	**35.32%**
Networks	0.44%	3.23%	0.00%	0.00%	0.00%	1.10%
Human Computer Interaction	8.44%	2.51%	20.00%	1.55%	0.00%	4.64%
Theory of Computation	9.78%	32.26%	0.00%	0.78%	2.38%	12.80%
Mathematics of Computing	0.00%	0.72%	5.00%	1.55%	8.33%	1.77%
Information System	18.22%	24.01%	18.33%	**37.60%**	38.10%	27.37%
Computer System & Organization	0.44%	0.00%	0.00%	3.10%	7.14%	1.66%
Computing Methodologies	10.22%	0.00%	1.67%	18.22%	**41.67%**	11.70%
Applied Computing	3.56%	0.00%	8.33%	2.71%	2.38%	2.43%
Hardware	0.00%	0.00%	0.00%	3.10%	0.00%	0.88%
Security	0.89%	0.00%	0.00%	0.39%	0.00%	0.33%

research question can determine the most appropriate clustering algorithm type. For example, density-based clustering algorithms such as DBScan may be more suitable if the goal is to identify outliers or anomalies. The clustering results should be evaluated using appropriate metrics such as the Silhouette, Calinski-Harabasz, and Davies Bouldin scores. The performance of the clustering algorithm should be compared against other algorithms, and the results should be interpreted in the context of the research question. Some clustering algorithms may not be suitable for large datasets due to their computational complexity and memory usage. Therefore, the scalability of the algorithm should be taken into consideration.

One advantage of clustering this dataset is the ability to examine the features of the thesis subjects chosen by students. In future research, we can use the results of this cluster as part of the components of a recommendation system. The expert already performs the annotation process using CCS ontologies as references and preserves the basics of determining courses supporting the thesis title. The recommendation system allows students who want to undergo thesis topics to choose courses from their transcript as a vital input recommendation system.

B. CLUSTER RESULTS

Table 7 displays the statistical distribution of computer science ontology in 5 clusters, categorized based on the ontology used in encoding the previous dataset labels. The percentage of each ontology per cluster is calculated by dividing the number of ontologies found in the cluster by the total number of ontologies. As can be seen, not all root ontologies have been satisfied. In this instance, the case study contains a thesis title whose substance cannot map to a specific root ontology. For instance, students infrequently choose the title of a thesis relevant to networks. In addition, most students are interested in titles associated with information systems. Nevertheless, every cluster has its own distinct set of traits. Clusters 1, 2, and 3 focus on software engineering thesis topics. Cluster 1 applies software engineering to information system products, such as personal health assistant applications, crowd reporting applications, and hospital logistics information systems. Cluster 2 combines software engineering with the scientific theory of computation, computational algorithms, and intelligent systems, such as Digital Whiteboard, smart e-catering applications, and the multiplayer game Nonogram. Finally, cluster 3 combines software engineering with aspects of human-computer interaction in the products produced, such as life simulation games, intuitive bowling game applications, and virtual reality physics simulation. The substance described by cluster 3 is distinct from those described by the other clusters. Most of the titles in cluster 3 are focused on educational topics, video games, and the connection between humans and computers.

Cluster 4 predominantly covers information system ontology, with e-commerce websites for small businesses, e-government applications, and job recommendation information systems as examples. This cluster is more towards the title of software engineering, which is applied to information systems projects with examples of leaf ontologies such as development frameworks, compilers, and software maintenance tools. Finally, cluster 5 focuses on computing methodologies that produce an information system output. For example, the information system for the logistical needs of victims of natural disasters, sentiment analysis, and a decision support system for purchasing goods. In addition to this, the majority of the systems in cluster 5 mainly belong to intelligent system topics. These systems include information retrieval, machine learning, and artificial intelligence.

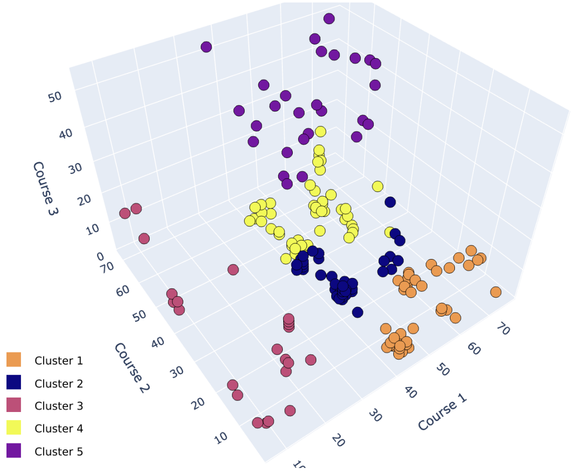


FIGURE 4. **3D Plotting K-Means Clustering Result**

It is possible to draw the conclusion, given that cluster 2 is the largest cluster, that most titles pertaining to information systems are gathered in this cluster with leaf ontologies, including enterprise computing, data management systems, and information system applications. Software engineering ontology dominates all clusters with a percentage of 35.32%, followed by information system ontology with 27.37%. The remaining ontology is distributed evenly, with the theory of computation at 12.80% and applied computing at 11.70%. The curriculum team can use the results of this cluster analysis to evaluate and improve the study program curriculum. For instance, a study program can determine the direction of its research by examining the distribution of ontologies in the clusters. Additionally, the cluster analysis results can serve as a reference in developing prediction systems, decision support systems, and thesis recommendation systems.

Figure 4 displays a 3D plot representing the k-means clustering outcome. The plot illustrates the separation of 300 data points into five distinct clusters. Each axis corresponds to the encoding of supporting courses associated with each data point, while each data point represents a thesis. As depicted in Fig 4, the distribution of data points is dependent on the encoding of three supporting courses outlined in Table 4. For instance, data points 0-10 belong to the Software Engineering ontology, whereas data points 53-63 belong to the Computing Methodologies ontology. Based on the visual inspection of the 3D plot, the dataset exhibits non-uniform

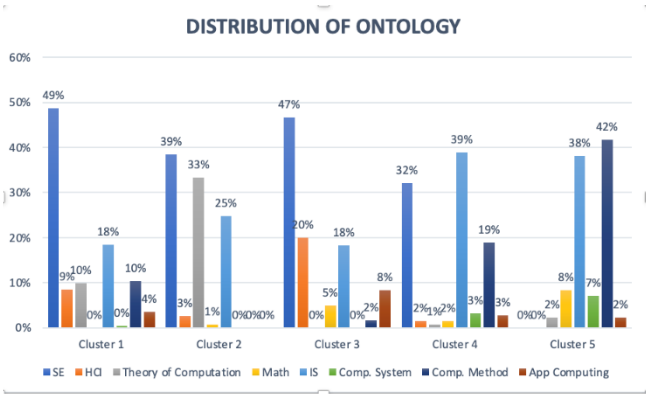


FIGURE 5. Distribution of Ontology

clustering with varied shapes. Consequently, density-based algorithms yield suboptimal results.

Figure 5 showcases our research in a visually appealing way, presenting the scientific content of various thesis titles. Through our analysis, we identified the top five ontologies as software engineering, information systems, human-computer interaction, computing methodologies, and applied computing. By counting the ontologies in each cluster and calculating the percentage distribution, we were able to highlight the predominant ontology and its distribution within the dataset. Each cluster is also associated with a course, such as "Intelligent Information Retrieval" in Cluster 0, "Applied Database" in Cluster 1, "Software Engineering" in Cluster 2, "Enterprise System Implementation" in Cluster 3, and "Human-Computer Interaction" in Cluster 4. This information can be used to develop a recommendation system for thesis topics and titles, taking into account the relevance of courses to students' abilities.

TABLE 8
CORRELATION ANALYSIS SUMMARY

Cluster #	Σ	µD (months)	µ GPA	ρ (GPA, D)	ρ (C, D)
1	30	6.68	3.63	-0.3242469726	-0.3246881753
2	90	9.02	3.23	-0.3865122321	-0.3661417629
3	70	9	3.36	-0.5156894356	-0.4460672125
4	66	8.03	3.55	-0.528283729	-0.420287703
5	46	9.28	3.31	-0.4853516519	-0.4037355567

We examined how GPA and the duration of a thesis are linked, as well as how the average grades of thesis-supporting courses and the duration of a thesis are related. We took each thesis data point and charted the student's GPA and the average grades thesis-supporting courses. We utilized a Pearson correlation analysis to determine the correlation between GPA, the average grades of thesis-supporting courses, and the duration of the thesis. Using a scatter plot, Figure 6 illustrates the distribution of data points based on completion time, GPA, and the average grades of thesis-supporting courses. The trendline for completion time demonstrates a connection between GPA, the average grades of thesis-supporting courses, and the duration of the thesis. The higher the GPA and the average grades of thesis-supporting courses, the shorter the duration of the thesis.

The results of the Pearson correlation analysis can be found in Table 8. The Σ symbol represents the total number of data points in a cluster, while the symbol µD denotes the average time, in months, taken to complete a thesis for a specific cluster, and µGPA signifies the average GPA for the same cluster. The symbol ρ(GPA, D) shows the correlation test outcome between GPA and the duration of thesis completion in a specific cluster, while ρ(C, D) represents the correlation test outcome between the average grades of thesis-supporting courses and the duration of thesis completion. The correlation values suggest that there is a moderate correlation between GPA and grades in supporting courses with the length of thesis completion. Negative values indicate an inverse correlation, meaning that lower GPA values correspond to longer thesis completion times for students, while higher GPA values correspond to faster thesis completion times. This trend is also observed in the correlation between grades in the average grades of thesis-supporting courses and thesis completion duration.

Upon analysing the clusters of theses, it was revealed that Cluster 1 boasted the shortest average completion time of 6.68 months. This grouping comprised 30 theses that focused on software engineering in information system products. Interestingly, students with higher GPAs completed their theses faster in this cluster, indicating a negative correlation of -0.3242 between GPA and thesis duration. In contrast, Cluster 5 had the longest average completion time of 9.28

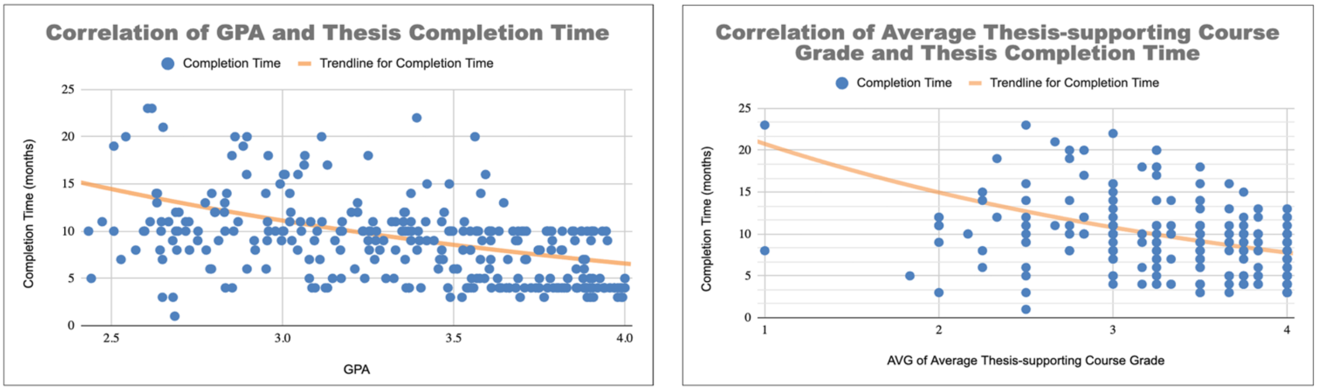


FIGURE 6. Scatter Plot Diagram of Correlation Between GPA, Supporting Course Grade and Thesis Completion Time

months. This cluster consisted of 46 theses that focused on intelligent systems. Here, a moderately negative correlation of -0.4854 between GPA and thesis duration was observed, signifying that students with higher GPAs completed their theses more quickly in this cluster, paralleling the results seen in Clusters 1 and 2. Furthermore, we observed that Cluster 5, which took the longest to complete, featured a considerable amount of ontological content pertaining to Mathematics, as depicted in Figure 5. Mathematics is a significant domain in the computer science program, and further research is required to determine whether proficiency and comprehension of mathematics contribute to the smooth progression of thesis work. These findings underscore the importance of selecting a suitable thesis topic to ensure timely completion.

Regarding certain clusters, Cluster 1 indicates that GPA and average grades in thesis-supporting courses have a minimal effect on thesis completion time within this cluster. However, Cluster 4 exhibits the highest correlation between GPA and thesis duration, emphasizing the significance of academic performance in expediting thesis completion. Similarly, Cluster 3 displays the strongest correlation between average grades in thesis-supporting courses and thesis duration, highlighting the importance of good performance in these courses for timely thesis completion. Nevertheless, the moderate correlation results suggest that factors beyond GPA and average grades in supporting courses contribute to the duration of thesis completion. Further studies are necessary to explore additional influences, such as student motivation in thesis work. This investigation could delve into areas like procrastination, confidence levels, and students' autonomy in selecting appropriate topics. Moreover, it should scrutinize the impact of supervisory guidance styles, supervisor reputations, and alignment between student-selected topics and advisors' expertise. Lastly, it should investigate the influence of academic abilities reflected in students' academic transcripts and the number of repeated courses. These findings offer valuable insights for future research. We can examine differences in correlation between clusters to better understand how diverse academic programs prepare students for their theses. This information can guide us in evaluating and improving our academic programs to ensure they adequately equip students for their research endeavours.

However, it is essential to note that clustering results may vary when using datasets from other universities. Our study showcases the effectiveness of k-means clustering to map each study program's knowledge composition and distribution pattern. This provides valuable insights for future research in higher education and aids in developing topic recommendation procedures. Our study can serve as a benchmark for future research in this area.

V. CONCLUSIONS

In this study we investigated the EDM dataset to discover concealed data and operational patterns among 300 titles from thesis course at the University of Surabaya. Students must apply the skills and knowledge gained during their education by working on thesis course. During their work, as part of the requirements, students also demonstrate the ability to think critically, creatively, and independently while receiving guidance from a supervisor or mentor. Regrettably, delays in finishing the undergraduate thesis are common at universities. This matter is a concern because delays in completing the thesis might also negatively affect the student's grade and the institution's accreditation. One of the most common reasons is that students select thesis topics that are not well-suited to their competencies. Therefore, a suitable thesis topic based on a student's academic records could solve the thesis delay problem.

Secondly, we investigated which clustering techniques are practical and efficient for the problem. We prepared the dataset extracted from past undergraduate thesis and annotated it with three supporting courses. Based on the comparison of clustering techniques, we conclude that k-means is an effective and efficient algorithm for this dataset. The clustering process produces five clusters. Furthermore, we concluded that applying k-means technique to other university datasets is possible and should deliver different insights and cluster patterns. Through a series of experiments and processes, this experiment significantly contributes to the understanding and evaluation of the learning outcomes of study programs defined in the curriculum following the design and implementation of thesis topics. Similarly, the clustering results are essential as a building block for the future work. Eventually, this study will benefit higher education as a reference in formulating the study program research roadmap.

There is room for improvement in future works. This includes the data preparation step outcomes depend highly on the expert's judgment, which means that annotation mistakes are still possible. Another annotation method that we should consider is crowdsourced annotation, which is more cost-effective. In this case, we consider employing a group of lecturers and students in specific topics of expertise as crowdsourced in labeling our dataset. Secondly, the manual annotation process conducted by an expert can benefit significantly if the students thesis supervisor is involved. This supervisor should be more familiar with the factual content of the thesis than the independent experts. Therefore, it can reduce errors of mislabeling during the annotation process. Finally, our k-means uses a non-unweighted dataset. It means that all three features of the supporting course considered have the same proportion and influence in supporting the content of the student undergraduate thesis. For example, some courses may have a dominating influence on the thesis title compared to other courses. By implementing weighted clustering, we can annotate additional information about the features' weight on each undergraduate thesis.

Research on EDM data has yielded significant results, particularly in the area of student thesis clustering. This research has the potential to be improved and maintained for even better outcomes. The findings can be used to develop a tailor-made method for suggesting topics for undergraduate theses based on individual preferences and interests. This research can also be applied to other fields of study by using a standard ontology that aligns with the domain knowledge of those fields. Moreover, the research indicates that factors beyond GPA and average grades in supporting courses affect the time it takes to complete a thesis. Future studies should explore other influences, such as student motivation, procrastination, confidence levels, and autonomy in topic selection. Additionally, it is crucial to investigate the impact of supervisory guidance style, supervisor reputation, student-topic alignment, academic ability reflected in academic transcripts, and the number of repeated courses.

REFERENCES

[1] C. Romero and S. Ventura, "Educational data mining: A survey from 1995 to 2005," *Expert Syst. Appl.*, vol. 33, no. 1, pp. 135–146, 2007, doi: 10.1016/j.eswa.2006.04.005.

[2] I. O. Pappas, M. N. Giannakos, L. Jaccheri, and D. G. Sampson, "Assessing student behavior in computer science education with an fsQCA approach: The role of gains and barriers," *ACM Trans. Comput. Educ.*, vol. 17, no. 2, 2017, doi: 10.1145/3036399.

[3] M. Durairaj and C. Vijitha, "Educational Data mining for Prediction of Student Performance Using Clustering Algorithms," vol. 5, no. 4, pp. 5987–5991, 2014.

[4] A. Moubayed, M. Injadat, A. Shami, and H. Lutfiyya, "Student Engagement Level in an e-Learning Environment: Clustering Using K-means," *Am. J. Distance Educ.*, vol. 34, no. 2, pp. 137–156, 2020, doi: 10.1080/08923647.2020.1696140.

[5] B. Rawat and S. K. Dwivedi, "Discovering learners' characteristics through cluster analysis for recommendation of courses in e-learning environment," *Int. J. Inf. Commun. Technol. Educ.*, vol. 15, no. 1, pp. 42–66, 2019, doi: 10.4018/IJICTE.2019010104.

[6] V. Efrati, C. Limongelli, and F. Sciarrone, "A Data Mining Approach to the Analysis of Students' Learning Styles in an e-Learning Community: A Case Study," in *Universal Access in Human-Computer Interaction. Universal Access to Information and Knowledge*, 2014, pp. 289–300.

[7] C. Romero and S. Ventura, "Educational data mining: A review of the state of the art," *IEEE Trans. Syst. Man Cybern. Part C Appl. Rev.*, vol. 40, no. 6, pp. 601–618, 2010, doi: 10.1109/TSMCC.2010.2053532.

[8] L. Khanna, S. N. Singh, and M. Alam, "Educational data mining and its role in determining factors affecting students academic performance: A systematic review," in *2016 1st India International Conference on Information Processing (IICIP)*, 2016, pp. 1–7. doi: 10.1109/IICIP.2016.7975354.

[9] K. T. Sanvitha Kasthuriarachchi, S. R. Liyanage, and C. M. Bhatt, "A Data Mining Approach to Identify the Factors Affecting the Academic Success of Tertiary Students in Sri Lanka," in *Software Data Engineering for Network eLearning Environments: Analytics and Awareness Learning Services*, S. Caballé and J. Conesa, Eds. Cham: Springer International Publishing, 2018, pp. 179–197. doi: 10.1007/978-3-319-68318-8_9.

[10] H. Jeong and G. Biswas, "Mining Student Behavior Models in Learning-by-Teaching Environments".

[11] S. Kausar, X. Huahu, I. Hussain, Z. Wenhao, and M. Zahid, "Integration of Data Mining Clustering Approach in the Personalized E-Learning System," *IEEE Access*, vol. 6, pp. 72724–72734, 2018, doi: 10.1109/ACCESS.2018.2882240.

[12] F. Del Bonifro, M. Gabbrielli, G. Lisanti, and S. P. Zingaro, "Student Dropout Prediction," in *Artificial Intelligence in Education*, 2020, pp. 129–140.

[13] I. Lykourentzou, I. Giannoukos, V. Nikolopoulos, G. Mpardis, and V. Loumos, "Dropout prediction in e-learning courses through the combination of machine learning techniques," *Comput. Educ.*, vol. 53, no. 3, pp. 950–965, 2009, doi: https://doi.org/10.1016/j.compedu.2009.05.010.

[14] F. Yang and F. W. B. Li, "Study on student performance estimation, student progress analysis, and student potential prediction based on data mining," *Comput. Educ.*, vol. 123, pp. 97–108, 2018, doi: https://doi.org/10.1016/j.compedu.2018.04.006.

[15] N. Tomasevic, N. Gvozdenovic, and S. Vranes, "An overview and comparison of supervised data mining techniques for student exam performance prediction," *Comput. Educ.*, vol. 143, p. 103676, 2020, doi: https://doi.org/10.1016/j.compedu.2019.103676.

[16] S. M. Razaulla, M. Pasha, and M. U. Farooq, "Integration of Machine Learning in Education: Challenges, Issues and Trends," in *Machine Learning and Internet of Things for Societal Issues*, C. Satyanarayana, X.-Z. Gao, C.-Y. Ting, and N. B. Muppalaneni, Eds. Singapore: Springer Singapore, 2022, pp. 23–34. doi: 10.1007/978-981-16-5090-1_2.

[17] "International Educational Data Mining Society," 2022. https://educationaldatamining.org/ (accessed Mar. 09, 2022).

[18] A. Dutt and M. A. Ismail, "Logical Review on Educational Data Mining," *Int. J. Comput. Commun. Netw.*, vol. 9, no. 3, pp. 39–42, 2020, doi: 10.30534/ijccn/2020/01932019.

[19] M. Muñoz-organero, P. J. Muñoz-merino, C. D. Kloos, and S. Member, "Student Behavior and Interaction Patterns With an LMS as Motivation Predictors in E-Learning Settings," vol. 53, no. 3, pp. 463–470, 2010.

[20] C. H. Su, "Designing and developing a novel hybrid adaptive learning path recommendation system (ALPRS) for gamification mathematics geometry course," *Eurasia J. Math. Sci. Technol. Educ.*, vol. 13, no. 6, pp. 2275–2298, 2017, doi: 10.12973/EURASIA.2017.01225A.

[21] J. Miranda, D. Olaya, V. Jonathan, and W. Verbeke, "Redefining profit metrics for boosting student retention in higher education," vol. 143, no. August 2020, 2021, doi: 10.1016/j.dss.2021.113493.

[22] J. Tondeur, S. K. Howard, and J. Yang, "One-size does not fit all: Towards an adaptive model to develop preservice teachers' digital competencies," *Comput. Human Behav.*, vol. 116, p. 106659, 2021, doi: https://doi.org/10.1016/j.chb.2020.106659.

[23] S. B. Aher and L. M. R. J. Lobo, "Combination of machine learning algorithms for recommendation of courses in E-Learning System based on historical data," *Knowledge-Based Syst.*, vol. 51, pp. 1–14, 2013, doi: 10.1016/j.knosys.2013.04.015.

[24] S. A. Priyambada, M. Er, B. N. Yahya, and T. Usagawa, "Profile-Based Cluster Evolution Analysis: Identification of Migration Patterns for Understanding Student Learning Behavior," *IEEE Access*, vol. 9, pp. 101718–101728, 2021, doi: 10.1109/ACCESS.2021.3095958.

[25] A. Saxena *et al.*, "A review of clustering techniques and developments," *Neurocomputing*, vol. 267, pp. 664–681, 2017, doi: 10.1016/j.neucom.2017.06.053.

[26] K. H. Tie, A. Senawi, and Z. L. Chuan, "An Observation of Different Clustering Algorithms and Clustering Evaluation Criteria for a Feature Selection Based on Linear Discriminant Analysis," in *Enabling Industry 4.0 through Advances in Mechatronics*, 2022, pp. 497–505.

[27] M. Mughnyanti, S. Efendi, and M. Zarlis, "Analysis of determining centroid clustering x-means algorithm with davies-bouldin index evaluation," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 725, no. 1, 2020, doi: 10.1088/1757-899X/725/1/012128.

[28] R. Romaniuc and C. Bazart, "Intrinsic and Extrinsic Motivation," *Encycl. Law Econ.*, pp. 1–4, 2015, doi:

10.1007/978-1-4614-7883-6_270-1.

[29] A. Petersen, M. Craig, J. Campbell, and A. Tafliovich, "Revisiting why students drop CS1," *ACM Int. Conf. Proceeding Ser.*, pp. 71–80, 2016, doi: 10.1145/2999541.2999552.

[30] J. Nouri, K. Larsson, and M. Saqr, *Identifying Factors for Master Thesis Completion and Non-completion Through Learning Analytics and Machine Learning*, vol. 11722 LNCS. Springer International Publishing, 2019. doi: 10.1007/978-3-030-29736-7_3.

[31] T. O'Donoghue and M. Rabin, "Procrastination on long-term projects," *J. Econ. Behav. Organ.*, vol. 66, no. 2, pp. 161–175, 2008, doi: 10.1016/j.jebo.2006.05.005.

[32] E. Irrazabal, M. A. Mascheroni, C. Greiner, and G. Dapozo, "Procrastination at the conclusion of the master's thesis: Results from a survey on computer science students in Northeast Argentina," *2017 43rd Lat. Am. Comput. Conf. CLEI 2017*, vol. 2017-Janua, pp. 1–6, 2017, doi: 10.1109/CLEI.2017.8226391.

[33] E. H. Seo, "The relationships among procrastination, flow, and academic achievement," *Soc. Behav. Pers.*, vol. 39, no. 2, pp. 209–218, 2011, doi: 10.2224/sbp.2011.39.2.209.

[34] A. Andre, N. Suciati, and H. Fabroyir, "Expert Annotation Tools for Labeling Student Capstone Project based on ACM CCS Ontology," in *2022 11th Electrical Power, Electronics, Communications, Controls and Informatics Seminar (EECCIS)*, 2022, pp. 345–350. doi: 10.1109/EECCIS54468.2022.9902931.

[35] C. Lipson, *How to write a BA thesis: A practical guide from your first ideas to your finished paper*. University of Chicago Press, 2018.

[36] Z. Liu *et al.*, "Segmentation of white blood cells through nucleus mark watershed operations and mean shift clustering," *Sensors (Switzerland)*, vol. 15, no. 9, pp. 22561–22586, 2015, doi: 10.3390/s150922561.

[37] A. Karami and R. Johansson, "Choosing DBSCAN Parameters Automatically using Differential Evolution," *Int. J. Comput. Appl.*, vol. 91, no. 7, pp. 1–11, 2014, doi: 10.5120/15890-5059.

[38] Z. Falahiazar, A. Bagheri, and M. Reshadi, "Determining the parameters of DBSCAN automatically using the multi-objective genetic algorithm," *J. Inf. Sci. Eng.*, vol. 37, no. 1, pp. 157–183, 2021, doi: 10.6688/JISE.202101_37(1).0011.

[39] N. Rahmah and I. S. Sitanggang, "Determination of Optimal Epsilon (Eps) Value on DBSCAN Algorithm to Clustering Data on Peatland Hotspots in Sumatra," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 31, no. 1, 2016, doi: 10.1088/1755-1315/31/1/012012.

[40] B. Lorbeer, A. Kosareva, B. Deva, D. Softić, P. Ruppel, and A. Küpper, "A-BIRCH: Automatic threshold estimation for the BIRCH clustering algorithm," *Adv. Intell. Syst. Comput.*, vol. 529, no. April 2018, pp. 169–178, 2017, doi: 10.1007/978-3-319-47898-2_18.

ANDRE received the bachelor's degree in computer science from the Universitas Surabaya, in 2006, and the M.Sc. degree in digital media technology from the Nanyang Technology University, in 2012 He is currently pursuing the Ph.D. degree. He has also been a Lecturer with Universitas Surabaya, since 2007. His research interests include game design and development, mobile development, XR development and human computer interface.

NANIK SUCIATI (Member, IEEE) received the master's degree in computer science from the University of Indonesia, in 1998, and the Dr.Eng. degree in information engineering from the University of Hiroshima, in 2010. She is currently an Associate Professor with the Department of Informatics, Institut Teknologi Sepuluh Nopember. She has published more than 50 journal articles and conference papers on computer science. Her research interests include computer vision, computer graphics, and artificial intelligence.

HADZIQ FABROYIR (Member, IEEE) received his Doctor of Computer Science and Information Engineering from the National Taiwan University of Science and Technology. In 2020, he joined the faculty of the Department of Informatics, Institut Teknologi Sepuluh Nopember as an assistant professor. His research interests include Human-Computer Interaction focusing on virtual navigation and extended reality.

ERIC PARDEDE (Senior Member, IEEE) received the master's degree in information technology and the Ph.D. degree in computer science from La Trobe University, Melbourne, Australia. He is currently an Associate Professor with La Trobe University. He has published more than 150 publications in international journals, conference proceedings and books. His research interests include data analytics, IT education, and entrepreneurship.