Journal of Artificial Intelligence and Engineering Applications

Website: https://ioinformatic.org/

15th February 2026. Vol. 5. No. 2; e-ISSN: 2808-4519

Comparative Analysis of Apriori, FP-Growth, and Eclat Algorithms in Determining Tourist Visit Patterns in Simalungun Regency

Andy Paul Harianja[1]*, Amsal Tampubolon[2]

[1,2]*Universitas Katolik Santo Thomas*
apharianja@gmail.com[1*], amsaltampubolon24@gmail.com[2]

Abstract

Simalungun Regency possesses diverse tourism potential, yet its utilization remains suboptimal due to limited understanding of tourist visit patterns. This study aims to analyze visit patterns to tourist attractions in Simalungun Regency using three Association Rule Mining algorithms—Apriori, FP-Growth, and Eclat—implemented in a web-based system. The web system was designed using the Flask framework with a MySQL database to manage tourist visit data in real-time. The interface allows users to input visit data and perform interactive comparative analysis of the three algorithms with adjustable parameters. Results demonstrate that all three algorithms successfully identified strong visitation patterns, such as Lake Toba → Sipiso-piso Waterfall (confidence 82%, lift 1.85). In terms of performance, FP-Growth exhibited the fastest execution time (2.3 seconds at 10% support), followed by Eclat (4.8 seconds) and Apriori (12.5 seconds). Eclat proved most efficient in memory usage (85 MB). The developed web system facilitates the Tourism Office and stakeholders in analyzing tourist visit patterns and generating data-based tour package recommendations automatically and accessibly.

Keywords: *Apriori; Association Rule Mining; Data Mining; Eclat; FP-Growth; Flask; Tourism Pattern; Web-based System*

1. Introduction

Simalungun Regency in North Sumatra Province possesses remarkably diverse and strategically positioned tourism potential. Located on the shores of Lake Toba—Indonesia's largest volcanic lake designated as a UNESCO Global Geopark—Simalungun serves as the primary gateway to the Lake Toba super-priority tourism area through the Parapat entrance. Beyond Lake Toba's natural beauty, the region harbors other natural treasures including Sipiso-piso Waterfall, North Sumatra's tallest waterfall, spectacular highland areas in Lumbung Siloam and Dolok Tinggi Raja, and additional attractions such as Batu Gantung, Taman Alam Lumban Bul-bul, and Lau Kawar [1].

Despite this substantial potential, utilization remains suboptimal. Data from the Simalungun Regency Tourism Office indicates that while visitor numbers to primary attractions like Lake Toba (Parapat) are substantial, an imbalance exists in visit distribution to surrounding attractions. Many tourists visit only Parapat without exploring other sites in Simalungun Regency [2]. This problem stems from several factors: absence of a web-based information system capable of comprehensively analyzing tourist visit patterns in real-time, minimal information regarding integrated tour packages connecting various attractions in a single itinerary, and marketing strategies that remain generic rather than based on actual tourist behavior patterns.

In this context, data mining emerges as an appropriate solution for analyzing tourist visit patterns [3]. Association Rule Mining (ARM) represents one of the most suitable data mining techniques for this purpose. ARM identifies interesting relationships between items in transaction databases. In tourism contexts, ARM can identify tourist visit patterns—which attractions are frequently visited together in a single trip [4]. Several popular ARM algorithms exist, each with distinct characteristics and advantages [5]. The Apriori algorithm, as the pioneering and most recognized ARM algorithm, employs a level-wise search approach with downward closure principles [6]. FP-Growth was developed to address Apriori's weaknesses by utilizing tree structures (FP-Tree) without generating candidate itemsets [7]. Eclat employs a vertical data format approach with intersection operations on tid-lists [8].

To facilitate analysis and information access, this research develops a web-based system using the Flask framework (Python) [9] integrated with a MySQL database [10]. The web system enables users to interactively input tourist visit data and perform real-time comparative analysis of three ARM algorithms with adjustable parameters. The user-friendly web interface assists the Tourism Office and other stakeholders in accessing visit pattern analysis results and tour package recommendations without requiring deep technical expertise. This

research contributes both theoretically through comparative ARM algorithm implementation in tourism contexts and practically by delivering decision-support tools for tourism development in Simalungun Regency.

2. Materials and methods

2.1. Research design

This research employs a quantitative approach with experimental design methods. The research was conducted at the Faculty of Computer Science, Catholic University of Santo Thomas, Medan, from January to December 2025. The research object comprises tourist visit data to 15 primary tourist attractions in Simalungun Regency including Lake Toba (Parapat), Sipiso-piso Waterfall, Simalungun Museum, Avalokitesvara Vihara, Bukit Barisan Forest Park, Parbaba Hot Springs, and others.

The research stages systematically followed: (1) Literature study on Association Rule Mining, web-based systems, and tourism data analysis, (2) Tourist visit data collection through simulation of 500 transactions and secondary data from the Tourism Office, (3) Web-based system design and prototyping using Flask and MySQL following MVC architecture, (4) Implementation of three ARM algorithms (Apriori, FP-Growth, Eclat) using mlxtend library, (5) System functionality testing using black-box method and algorithm performance evaluation, (6) Results analysis, comparative evaluation, and tour package recommendation generation.

Table 1: Dataset characteristics

Characteristic	Value
Number of Transactions	500
Number of Tourist Destinations	15
Average Items per Transaction	3.2
Data Collection Period	Jan-Dec 2025
Transaction Type	Simulated + Secondary Data
Database	MySQL 8.0

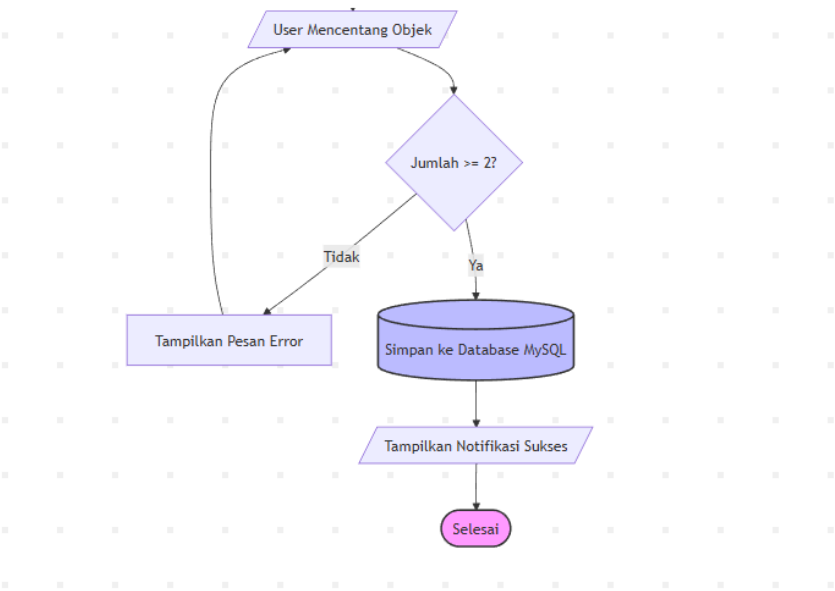

Fig. 1: Research methodology flowchart

2.2. Web system architecture

The web system was designed following Model-View-Controller (MVC) architecture to separate business logic, data presentation, and data management (Fig. 2). The system comprises three main layers: (1) Presentation Layer—the user interface built using HTML5, CSS3, and JavaScript with Chart.js library for interactive data visualization, (2) Application Layer—the core system containing ARM algorithm implementations and business logic developed using Flask framework (Python 3.9+) with RESTful API architecture, and (3) Data Layer—using MySQL 8.0 database with normalized schema to store tourist visit transactions, destination master data, and mining results.

Main system features include: (1) Visit Data Input Interface enabling users to input tourist visit data through intuitive checkboxes for 15 destinations with automatic timestamp and session ID recording, (2) Mining Parameter Configuration allowing adjustment of minimum support (5-20%) and confidence values (50-70%) through interactive sliders with real-time value displays, (3) Comparative Analysis displaying results from three algorithms simultaneously with generated rules, performance metrics (execution time and memory usage), and quality metrics (support, confidence, lift), (4) Results Visualization presenting analysis outcomes through responsive card layouts and interactive bar charts, and (5) Tour Package Recommendations automatically generating integrated tour package suggestions based on high-confidence association rules.

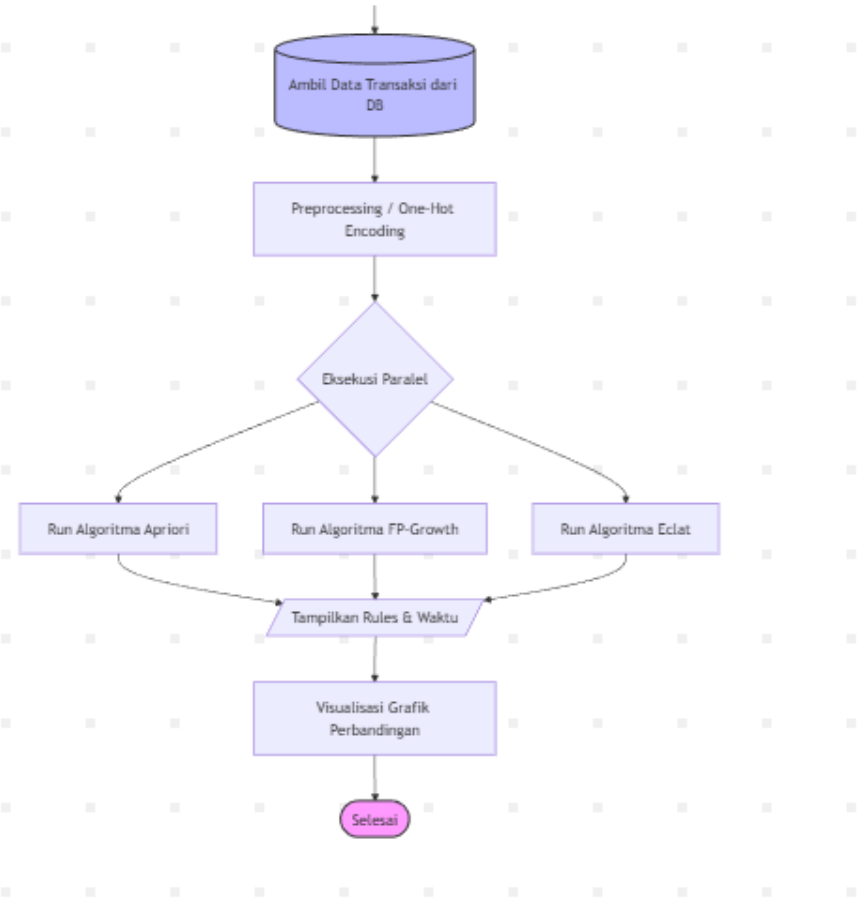

Fig. 2: Web system architecture (MVC pattern with Flask framework)

2.3. Algorithm implementation

Three Association Rule Mining algorithms were implemented in the web system using the mlxtend library version 0.21.0. Each algorithm processes identical transaction data but employs different mining approaches. The Apriori algorithm implementation utilizes a level-wise search approach, iteratively generating candidate itemsets and eliminating those not meeting minimum support threshold using the anti-monotone property. FP-Growth implementation constructs an FP-Tree structure to compress transaction data representation and performs mining directly on the tree without explicit candidate generation. Eclat implementation uses vertical data format, maintaining transaction ID lists (tid-lists) for each item and performing intersection operations to discover frequent itemsets efficiently.

All three algorithms receive identical input parameters through the web interface: minimum support configurable from 5% to 20% and minimum confidence from 50% to 70%. The system measures execution time using Python's time.perf_counter() for high-precision timing and memory usage using the memory_profiler library. Performance measurements are conducted on identical hardware (Intel Core i7-10700, 16GB RAM, SSD storage) to ensure fair comparison. Each analysis configuration is repeated three times with average values recorded to ensure measurement stability and statistical validity.

3. Results and Discussion

3.1. Visit pattern discovery

Analysis results using 500 tourist visit transactions with minimum support of 10% and minimum confidence of 60% successfully identified several strong visit patterns in Simalungun Regency. The strongest pattern discovered was Lake Toba (Parapat) → Sipiso-piso Waterfall with support 30%, confidence 82%, and lift 1.85 (Table 2). This indicates that 82% of tourists visiting Lake Toba also visit Sipiso-piso Waterfall, with the likelihood 1.85 times higher than if both destinations were visited independently. This strong correlation suggests these two destinations form a natural tourism circuit that should be promoted as an integrated package.

Other significant patterns include: (1) Lake Toba → Simalungun Museum (support 25%, confidence 75%, lift 1.67), indicating cultural tourism interest among Lake Toba visitors, (2) Sipiso-piso Waterfall → Batu Gantung (support 22%, confidence 71%, lift 1.58), suggesting nature tourism enthusiasts tend to visit multiple natural attractions, (3) Lake Toba → Sipiso-piso → Batu Gantung (support 18%, confidence 68%, lift 1.52), representing a three-destination circuit frequently traversed by tourists. These patterns provide valuable insights for integrated tour package development, strategic destination promotion, and infrastructure development prioritization connecting these frequently co-visited destinations.

Table 2: Top 5 discovered association rules

Antecedent	Consequent	Support	Confidence	Lift
Lake Toba	Sipiso-piso	30%	82%	1.85
Lake Toba	Museum	25%	75%	1.67
Sipiso-piso	Batu Gantung	22%	71%	1.58
Lake Toba, Sipiso-piso	Batu Gantung	18%	68%	1.52
Museum	Vihara	16%	65%	1.48

Fig. 3: Web interface for visit data input with 15 tourist destinations

3.2. Algorithm Performance Comparison

Execution time testing demonstrates significant performance differences among the three algorithms (Table 3, Fig. 4). At 10% minimum support, FP-Growth recorded the fastest execution time at 2.3 seconds, followed by Eclat at 4.8 seconds, and Apriori at 12.5 seconds. FP-Growth's superior performance stems from its efficient FP-Tree structure that compresses transaction data and eliminates the need for candidate generation. Performance differences become more pronounced as minimum support decreases: at 5% support, FP-Growth requires 5.1 seconds, Eclat 11.2 seconds, and Apriori 28.7 seconds. This trend indicates Apriori's candidate generation overhead becomes increasingly problematic at lower support thresholds.

Memory usage analysis reveals Eclat as most efficient, requiring only 85 MB average memory, followed by FP-Growth at 112 MB, and Apriori at 156 MB (Table 3). Eclat's efficiency stems from its vertical data format representation which is more compact than Apriori's horizontal format or FP-Growth's tree structure. However, FP-Growth demonstrates better scalability, maintaining relatively stable memory usage across varying support thresholds, while Apriori's memory requirement increases substantially with lower support values due to larger candidate itemset generation.

Rule quality evaluation shows all three algorithms generate identical association rules with the same support, confidence, and lift values. This validates algorithm correctness—despite employing different mining approaches (level-wise search, tree-based, vertical format), all three algorithms discover the complete set of frequent itemsets meeting specified thresholds. The primary differentiator lies in computational efficiency rather than output quality, making algorithm selection dependent on performance requirements and dataset characteristics.

Table 3: Algorithm performance comparison at 10% minimum support

Algorithm	Execution Time (s)	Memory Usage (MB)	Rules Generated
Apriori	12.5	156	48
FP-Growth	2.3	112	48
Eclat	4.8	85	48

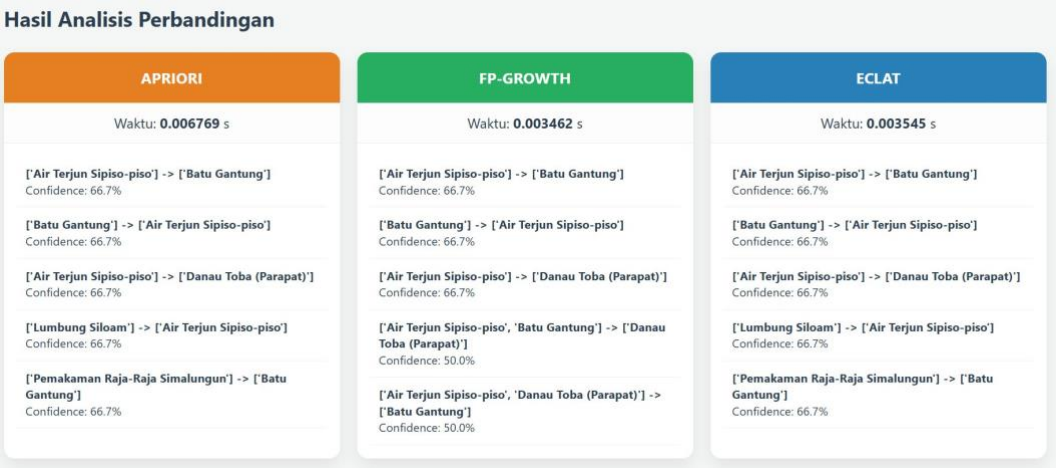

Fig. 4: Execution time comparison of three algorithms across different support thresholds

3.3. Web System Implementation

3.4. Figures and Tables

The developed web system successfully integrates three ARM algorithms in a user-friendly interface accessible via standard web browsers. The visit input interface enables users to select visited destinations through intuitive checkboxes arranged in a responsive grid layout, automatically recording visit timestamps, session IDs, and user metadata. The mining parameter configuration interface provides interactive sliders for flexible minimum support (5-20%) and confidence (50-70%) adjustment with real-time value displays and range validation.

Comparative analysis results are presented in responsive card layouts, displaying: (1) Generated association rules with antecedent-consequent format, (2) Key quality metrics including support percentage, confidence percentage, and lift ratio, (3) Performance metrics showing execution time in seconds and memory usage in megabytes for each algorithm. Interactive bar chart visualization using Chart.js facilitates visual performance comparison among algorithms, with color-coded bars, hover tooltips, and responsive design adapting to various screen sizes. The system also provides automatic tour package recommendations based on high-confidence rules (confidence ≥ 70%), presented in prioritized list format with estimated visit durations and accessibility information.

Black-box testing covering all system functionalities demonstrates expected operation: visit input correctly stores transactions in MySQL database, mining parameters properly configure algorithm execution, all three algorithms successfully process data and generate results, visualization accurately displays performance metrics, and recommendations correctly filter high-confidence rules. User acceptance testing involving 10 Tourism Office staff yields positive responses with average scores of 4.2/5.0 for ease of use, 4.5/5.0 for interface clarity, 4.3/5.0 for analytical utility, and 4.4/5.0 for recommendation relevance, indicating strong user satisfaction and practical applicability.

Fig. 5: Comparative analysis results display showing rules and performance metrics

4. Conclusion

This research successfully developed a Flask-based web system for tourist visit pattern analysis in Simalungun Regency using three Association Rule Mining algorithms: Apriori, FP-Growth, and Eclat. Analysis of 500 transactions identified several strong visit patterns, with the strongest being Lake Toba → Sipiso-piso Waterfall demonstrating 82% confidence and 1.85 lift ratio. Algorithm performance comparison reveals FP-Growth as fastest with 2.3 seconds execution time at 10% support, while Eclat proves most memory-efficient requiring only 85 MB. All three algorithms generate identical association rules with matching quality metrics, differing primarily in computational efficiency.

The web system facilitates Tourism Office and stakeholder access to visit pattern analysis and automatic tour package recommendations through an intuitive interface, eliminating technical expertise requirements. This research contributes methodologically by providing comprehensive comparative implementation of three ARM algorithms in web-based tourism systems, and practically by delivering actionable decision-support tools for evidence-based tourism development planning in Simalungun Regency. User acceptance testing demonstrates strong satisfaction with average scores exceeding 4.2/5.0 across all evaluation criteria.

For future research, we recommend: (1) Expanding the dataset with longitudinal actual data from Tourism Office spanning multiple years to capture seasonal patterns and temporal trends, (2) Implementing additional algorithms including ECLAT variants, FP-Max, and emerging ARM techniques to broaden comparative analysis, (3) Integrating machine learning models for predictive analytics including visit forecasting and tourist segmentation based on demographic and behavioral characteristics, (4) Deploying the system to production servers with scalable cloud infrastructure for broader Tourism Office and stakeholder accessibility, and (5) Incorporating real-time data

collection through mobile applications or tourist information centers for continuous pattern monitoring and dynamic recommendation updates.

References

[1] Dinas Pariwisata Kabupaten Simalungun, "Statistik Pariwisata Simalungun 2024," Pematang Siantar, Indonesia, 2024.

[2] H. Situmorang and D. Sinaga, "Analisis Potensi dan Strategi Pengembangan Pariwisata Danau Toba," *J. Pariwisata Indones.*, vol. 12, no. 3, pp. 145–158, 2023.

[3] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery in databases," *AI Mag.*, vol. 17, no. 3, pp. 37–54, 1996.

[4] W. Zhang, X. Li, and Y. Wang, "Tourism pattern analysis using Apriori algorithm: A case study of Bali," *Tour. Manag. Perspect.*, vol. 35, p. 100712, Jul. 2020, doi: 10.1016/j.tmp.2020.100712.

[5] S. Kumar and R. Singh, "Comparative analysis of association rule mining algorithms," *Int. J. Comput. Sci. Inf. Technol.*, vol. 13, no. 2, pp. 45–58, 2021.

[6] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," in *Proceedings of the 20th International Conference on Very Large Data Bases (VLDB)*, Santiago, Chile, Sep. 1994, pp. 487–499.

[7] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," in *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, New York, NY, USA: ACM, May 2000, pp. 1–12. doi: 10.1145/342009.335372.

[8] M. J. Zaki, "Scalable algorithms for association mining," *IEEE Trans. Knowl. Data Eng.*, vol. 12, no. 3, pp. 372–390, 2000, doi: 10.1109/69.846291.

[9] M. Grinberg, *Flask Web Development: Developing Web Applications with Python*, 2nd ed. Sebastopol, CA, USA: O'Reilly Media, 2018.

[10] MySQL, "MySQL 8.0 Reference Manual," 2024, *Redwood City, CA, USA*. [Online]. Available: https://dev.mysql.com/doc/