A Survey On Frequent Pattern And Association Rule Mining Techniques

[1]Devika B. Gadhavi
[1]Lecturer
[1]Information Technology Department
[1]R. C. Technical Institute, Ahmedabad, India

[2]Darshna M Trivedi
[2]Lecturer
[2]Information Technology Department
[2]R. C. Technical Institute, Ahmedabad, India

Abstract: Frequent pattern mining and association rule mining are foundational in data mining, enabling the discovery of hidden relationships and patterns within large datasets. This comprehensive survey, provides an expanded overview of the key algorithms, their evolution, comparative strengths and weaknesses, and major research trends. The report covers traditional, advanced, and emerging methods, with a focus on both static and dynamic data environments, scalability, utility-based mining, parallelization, and domain-specific adaptations. Real-world applications, open challenges, and future research directions are also discussed.

1.Introduction

Frequent pattern mining (FPM) and association rule mining (ARM) are pivotal in extracting valuable insights from vast data repositories. The primary goal is to identify itemsets, sequences, or substructures that occur frequently, and to derive association rules that reveal dependencies among data items. These techniques are widely applied in market basket analysis, bioinformatics, web usage mining, intrusion detection, and more.

1.1 Motivation The exponential growth of data in diverse fields such as retail, healthcare, finance, and social networks has intensified the need for robust and efficient data mining techniques. FPM and ARM help organizations uncover hidden trends, optimize operations, and make data-driven decisions.

1.2 Historical Perspective Since the introduction of the Apriori algorithm in the early 1990s, the field has witnessed rapid advancements, with a focus on improving efficiency, scalability, and applicability to various data types and domains.

2.Fundamental Concepts

o **Frequent Itemset:** Sets of items, subsequences, or substructures that appear together in a dataset with frequency above a user-defined threshold.
o **Association Rule**: An implication of the form A => B, indicating that the presence of itemset A implies the presence of itemset B with certain support and confidence.

2.1 Key Measures

o Support: Frequency of occurrence of an itemset in the dataset.
o Confidence: Likelihood that itemset B appears when A appears.
o Lift: Ratio of observed support to expected support if A and B were independent.
o Conviction, Leverage, and Other Measures: Used for more nuanced evaluation of rule interestingness.

2.2 Problem Formulation Given a dataset of transactions, the main tasks are: - Mining all frequent itemsets that satisfy a minimum support threshold. - Generating association rules from these itemsets that satisfy a minimum confidence threshold.

3. Classical Algorithms

3.1 Apriori Algorithm Proposed by Agrawal and Srikant in 1994 [1], Apriori is a foundational algorithm based on the downward closure property. It generates candidate itemsets level-wise and prunes infrequent ones. It is simple and interpretable but suffers from high computational costs due to multiple database scans and large candidate sets.

3.2 FP-Growth Algorithm Introduced by Han et al. in 2000 [3], FP-Growth uses a pattern-growth approach with a compact tree structure (FP-tree) to avoid candidate generation. It is faster and more memory-efficient than Apriori but can be complex in tree construction.

3.3 ECLAT Algorithm Proposed by Zaki in 2000 [2], ECLAT employs a vertical data format and depth-first search. It efficiently mines frequent itemsets via TID-set intersections, suitable for dense datasets but may use more memory in sparse data.

3.4 Other Notable Algorithms Early methods like AIS and SETM focused on candidate generation, while Relim introduced recursive elimination. RARM, DIC, and Partition offer improvements targeting performance bottlenecks.

4. Advanced and Specialized Techniques

4.1 Vertical and Horizontal Data Layouts - Horizontal Layout: Used in Apriori and FP-Growth, represents each transaction as a row. - Vertical Layout: Used in ECLAT, associates items with transaction ID lists.

4.2 Maximal and Closed Itemset Mining - Maximal Frequent Itemsets: Largest frequent itemsets with no frequent supersets. - Closed Frequent Itemsets: No super-itemset has the same support. - Algorithms include MAFIA, GenMax, CHARM, and DCI-Closed.

4.3 Algorithms for Data Streams Real-time mining requires algorithms such as CPS-tree, variable sliding window (VSW), and Tmoment to adapt to concept drift and memory limits.

4.4 Utility-Based and Constraint-Based Mining - Utility Mining: Incorporates item importance or value (e.g., profit). - Constraint-Based Mining: Integrates user-defined constraints to focus the mining process.

4.5 Parallel and Distributed Mining Mining large-scale datasets is enabled by parallel adaptations like Parallel FP-Growth and MapReduce implementations (e.g., PFP [7]), leveraging platforms such as Hadoop and Spark.

4.6 Incremental and Dynamic Mining Algorithms like FUP, FUP2, and SSR efficiently update patterns in dynamic databases. Support thresholds can adapt to data evolution.

5. Comparative Analysis

Table 5.1: Comparative analysis

Algorithm	Data Layout	Search Strategy	Strengths	Weaknesses	Reference
Apriori	Horizontal	BFS	Simple, interpretable	Multiple scans, high candidate count	[1]
FP-Growth	Horizontal	DFS	Fast, low memory, 2 scans	Tree construction overhead	[3]
ECLAT	Vertical	DFS	Suitable for dense data	High memory for sparse data	[2]
MAFIA	Vertical	BFS	Maximal itemset mining	Post-pruning needed	[14], [15]
GenMax	Vertical	BFS	Efficient maximal mining	Complex implementation	[14]
CPS-tree	Horizontal	DFS	Efficient for streams	Tree restructuring overhead	[5], [6]

6. Applications

- ○ **Market Basket Analysis**: Identifying product combinations.
- ○ **Bioinformatics**: Discovering gene/protein associations.
- ○ **Web Usage Mining**: Analyzing navigation patterns.
- ○ **Intrusion Detection**: Identifying unusual network behaviors.
- ○ **Healthcare**: Detecting disease co-occurrence.

7. Key Research Trends

7.1 Efficiency Improvements Focus on reducing memory and computation, using data compression and optimized structures.
7.2 Incremental and Dynamic Mining Development of algorithms for real-time, evolving data with attention to concept drift.
7.3 Utility and Constraint-Based Mining Expanding focus beyond frequency to incorporate value-based and user-guided pattern discovery.
7.4 Parallel and Distributed Mining Growing adoption of parallelism and distributed environments like Spark and Hadoop.
7.5 Domain-Specific Adaptations Advancements in mining temporal, spatial, and graph data, and enhancing privacy-preservation.

8. Challenges and Open Issues

- Scalability: Managing big data volume and dimensionality.
- Pattern Explosion: Handling large output sizes.
- Dynamic Data: Ensuring efficiency in real-time mining.
- Interpretability: Making results actionable.
- Privacy and Security: Safe mining of sensitive data.
- Integration with ML: Enhancing analytics through hybrid approaches.

9. Future Directions

- Deep Learning Integration: Use of neural networks to enrich pattern discovery.
- Automated Pattern Selection: Filtering based on relevance.
- Explainable AI: Improved pattern interpretability.
- Edge and Real-Time Mining: Algorithms for IoT and edge devices.
- Ethical and Fair Mining: Ensuring unbiased, responsible insights.

10. Conclusion

Frequent pattern and association rule mining have evolved significantly, with numerous algorithms tailored for various data types and application needs. While Apriori and FP-Growth remain foundational, ongoing research addresses scalability, efficiency, and adaptability for modern data challenges. The field continues to expand into new domains, including real-time analytics, utility-based mining, and privacy-preserving techniques. Future research will likely focus on integrating ARM with advanced machine learning, ensuring interpretability, and addressing ethical concerns.

REFERENCES

[1] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules," VLDB, 1994.

[2] M.J. Zaki, "Scalable Algorithms for Association Mining," IEEE Transactions on Knowledge and Data Engineering, vol. 12, no. 3, pp. 372-390, 2000.

[3] J. Han, J. Pei, and Y. Yin, "Mining Frequent Patterns without Candidate Generation," SIGMOD, 2000.

[4] Y. Liu, W. Liao, and A. Choudhary, "A Fast High Utility Itemsets Mining Algorithm," UBDM Workshop, 2005.

[5] C. F. Ahmed, S. K. Tanbeer, B. S. Jeong, and Y.-K. Lee, "Efficient Tree Structures for High Utility Pattern Mining in Incremental Databases," IEEE Transactions on Knowledge and Data Engineering, vol. 21, no. 12, pp. 1708–1721, 2009.

[6] J. Han, H. Cheng, D. Xin, and X. Yan, "Frequent-Pattern Stream: Mining Frequent Patterns in Data Streams," VLDB, 2003.

[7] M. Li, D. Lee, H. Wang, and J. Y. Park, "PFP: Parallel FP-Growth for Query Recommendation," ACM RecSys, 2008.

[8] A. Meenakshi, "Survey of Frequent Pattern Mining Algorithms in Horizontal and Vertical Data Layouts," IJACST, 4(4), April 2015.

[9] Sivarani Jalmanayani, T. Subramanyam, "A Survey on Data Mining Association Rules By Effective Algorithms," IJISRT, 2(7), July 2017.

[10] Jagmeet Kaur, Neena Madan, "Association Rule Mining: A Survey," International Journal of Hybrid Information Technology, vol. 8, no. 7, 2015.

[11] S. Maurya, P. Gupta, A. Gupta, "A Survey on Mining Frequent Itemsets over Data Streams," IJCA, vol. 179, no. 8, December 2017.

[12] Shahana, K. Bhavya, R. M. Suresh, "A Review on Different Association Rule Mining Algorithms," IJERCS, July 2017.

[13] P. Fournier-Viger, A. Gomariz, T. Gueniche, A. Soltani, C. Wu, and V. S. Tseng, "SPMF: A Java Open-Source Pattern Mining Library," Journal of Machine Learning Research, vol. 15, pp. 3389–3393, 2014.

[14] B. Goethals, "Survey on Frequent Pattern Mining," Technical Report, Helsinki Institute for Information Technology, 2003.

[15] C. Lucchese, S. Orlando, and R. Perego, "Mining Top-K Patterns from Binary Datasets in Presence of Noise," IEEE Transactions on Knowledge and Data Engineering, vol. 19, no. 4, pp. 408–425, 2007.

[16] A. Gyenesei, "Mining Weighted Association Rules for Fuzzy Items," ACM SIGMOD Record, vol. 32, no. 2, pp. 93–98, 2003.