IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 11, November 2026

DOI: 10.17148/IJARCCE.2026.1511177

PTKOA: Interpretable Knowledge Prioritisation of Multidimensional Temporal Association Rules under Cross-Cohort and Cross-Organization Shift

Mr B. B. L. V. Prasad[1], Dr. Eedi Hemalatha[2]

Research Scholar, Department of Computer Science and Engineering, Jawaharlal Nehru Technological University Hyderabad (JNTUH), Hyderabad, Telangana-500085, India[1]

Assistant Professor, Department of Computer Science and Engineering, Sri Chaitanya Institute of Technology & Research (SCIT), Khammam, Telangana-507001, India[1]

Professor, Department of Computer Science and Engineering, Jawaharlal Nehru Technological University Hyderabad (JNTUH), Hyderabad, Telangana 500085, India[2]

Abstract: As behaviour evolves across cohorts or from demographic groups to entire organisations, knowledge-based decision systems are prone to inherit hundreds of valid association rules without an understanding of which rules will remain useful ultimately. In this paper, we introduce the Prioritised Temporal Knowledge Optimisation Approach (PTKOA), an interpretable post-mining framework for minimal multidimensional temporal association rules. PTKOA embeds these scores in addition to normalising common features: support, confidence, log-lift, a Historical Relationship Score (HRS) which captures persistence through logs of past observations on the associations captured, transaction-weighted coverage and the stability of each score within defined time windows. Weights are learned from source-development cutoffs and frozen before future or cross-environment evaluation scenarios. Experiments on Online Retail II and the orthogonal ARHyMST archive include character-tied product-context rules, powerful recency baselines, and learned baselines, as well as LOEO transfer, ablation, sensitivity, and bootstrap analyses of our results; mindest bootstraps each variety of machine learning-based heterogeneity runtime enumeration of total tree levels, errors diagnostics+ from regression trees. On the primary 30-dimensional bidirectional Online Retail II transfer, PTKOA executed NDCG@100 = 0.5132 ±0.1156 and Precision@100 = 0.3286 ±0.1081, as opposed to 0.3882 ±0.1413 and 0.2521 ±0.0917 for a similarly tuned traditional ensemble of ensembles (won at top-13) PTKOA trained on Online-Retail achieved NDCG@100 = 0.8393 over ARHyMST but reverse transfer in favor of recency-aware scoring. This means PTKOA is primarily an auditable knowledge-prioritisation layer, with the historical part useful if a recurrence structure can be ported and should subserve recency under stronger shift.

Keywords: knowledge-based systems; multidimensional temporal association rules; knowledge prioritisation; temporal association-rule mining; interpretable ranking; cross-organisation transfer; learning to rank; robustness

I. INTRODUCTION

Mined rules are based on knowledge from interpretable evidence in the decision systems of these sources. ARM and large transactional databases mining introduced support and confidence for discovering co-occurrence rules [1]; this became practical with Apriori [2]. However, discovery does not actually dictate priority: a valid rule can be provisional, inefficient, or simply irrelevant after cohort or organisational changes. And so, the problem we discuss here is which of the rules that are already valid should be surfaced first, then used for other purposes later on.

Support and confidence do not encode recurrence, exposure-weighted persistence (i.e. in the past), stability or recency. The coexistence of product predicates together with country, channel, basket-size, customer-frequency, day-period and weekday/weekend context is particularly relevant when it comes to multidimensional temporal rules. The ranking layer should differentiate between knowledge arising from permanence and ephemeral strength, while maximising interpretability.

This is a direct result of the two glass-box layers that PTKOA uses; HRS abstracts recurring, exposure-weighted coverage and stability, while PKS integrates normalised support confidence and log-lift within a simplex constraint from HRS

 This work is licensed under a Creative Commons Attribution 4.0 International License

DOI: 10.17148/IJARCCE.2026.1511177

output. Freezing the weights source-selection—and before target-future evaluation—results in a leak-free, testable, and portable approach. The study makes five contributions:

- An interpretable formulation for knowledge-prioritisation. This decomposes the quality of global rules from source selection based on frequency and stability citation patterns, and also has a term to define source-only weights for improved Top-K runtime efficiency.
- A benchmark for annual-transferral (30 days) with two Online Retail II queries, in addition to a self-contained inter-organisational stress test of ARHyMST using optimised temporal decay, RayGrange, or LambdaMART instead of simple weak single-measure benchmarks.
- Type multidimensional temporal rules having served as context specifications for the product-oriented consequence and leakage-safe predicate: Experiments across all three environments.
- Assessments of robustness and reproducibility, via fine-grid simplex search, bootstrapping weight stability, spaced cutoffs, threshold- and window-sensitivity analysis, leave-one-environment-out evaluation (LOEO), moving-block bootstrap backing up similar outcomes even when arbitrarily cropping data on time so that the models were not overfitting noise and error prediction diagnostics and interpretability.
- Deployment-oriented initialisation: The study does not argue that any model class is better than the others. Earlier historical recurrence has been useful if based on a structure of recurrence that can migrate. Thus, scoring with respect to recency is a more prudent route, particularly under larger domain changes.

Novelty statement. PTKOA is ultra low-cap: no new miner, an auditable conjunction of historical provenance with arbitrarily frozen rank and unit transfer test across cohorts, organisations and typed d-dimensional rules. This paper shows that external experiments expose rather than excuse systematic history transfer pitfalls.

This paper presents a chronological review of works related to this post-mining objective.

II. RELATED WORK

The original formulation of association-rule mining dates back to 1993 [1] in combination with the Apriori algorithm [2]. Multilevel rules [3] (multilevel association rule mining), frequent episode [4], multidimensional data-cube rules [5], alternative itemset generation [6], multiple minimum supports[7], further multilevel mining[8]; scalable association mining[9]; FP-growth algorithm for repeated and caching.[10] Temporal rules of emergence in sequence mining; PrefixSpan[11]

Temporal predictions and graded ranking using sequential-learning perspectives [13], NDCG [14], calendar-based temporal rules [15], standard association-analysis methods [16], inter-transaction mining [17], relational learners or relational learning-based models, LambdaMART[19]; broad treatments of mining[20],[21],[22],[23] and xgboost[24].

More recent works include the temporal-ARM survey [25], up-to-date and soft-set temporal rules [26], [27], multivariate and graph-sequence formulations [28], [29], [30], application-oriented temporal ARM, dynamic knowledge-graph reasoning [32]and various types of structural-relational (fact/logic/motif) rules [33],[34], fuzzy graph-sequence rules. While these studies enhance the degree of discovery or presentation, they do not test Multidimensional temporal rules prioritisation after a source-frozen post-mining model on unseen environments directly. This paper shows how to close this narrower gap with strong recency/learned baselines, dependence-aware resampling, typed rules, cross-organisation transfer, LOEO testing and failure diagnosis. The resulting methodological position is summarised in Table 1.

TABLE I HIGHLIGHTS THE REMAINING DISTINCTION: PTKOA IS EVALUATED AS A SOURCE-FROZEN POST-MINING PRIORITY LAYER RATHER THAN AS ANOTHER RULE-DISCOVERY ALGORITHM.

Table 1. Positioning of PTKOA as a post-mining knowledge-prioritisation framework			
Method family	**Temporal / multidim. evidence**	**Post-mining priority**	**Frozen transfer**
Classical ARM	Static; limited context	No	No
Temporal/sequential ARM	Explicit time; task-limited context	Mostly discovery	Rare
Multidimensional/relational	Partial time; rich predicates	Mostly discovery	Rare

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 11, November 2026

DOI: 10.17148/IJARCCE.2026.1511177

Table 1. Positioning of PTKOA as a post-mining knowledge-prioritisation framework

Method family	Temporal / multidim. evidence	Post-mining priority	Frozen transfer
Learning-to-rank	Feature dependent; arbitrary features	Yes	Possible
Recent temporal reasoning	Explicit time; relational/multivariate	Task specific	Task specific
PTKOA	HRS + recency; typed rules	Yes	Yes

III. PTKOA KNOWLEDGE-PRIORITIZATION METHODOLOGY

A. Overall Architecture

Fig. 1 first summarises the full PTKOA pipeline so that the later equations and algorithms can be read in context.

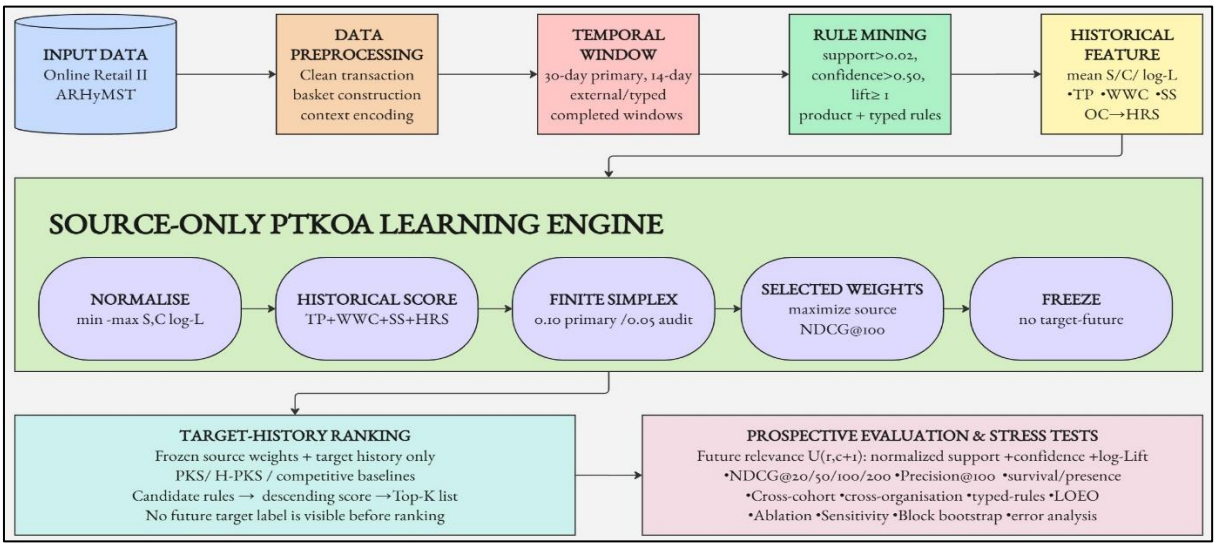


Fig. 1 PTKOA knowledge-prioritisation architecture from data preparation and temporal rule mining to source-only tuning, frozen transfer, and prospective decision-support evaluation.

Figure 1 is read left to right, then down. The top row takes the two transactional sources and transforms them into cleaned, temporally windowed rule histories. The ordinary PTKOA engine normalises conventional quality, builds up HRS, queries the source-only simplex, and freezes certain weights. The bottom row applies these frozen weights to the target history before revealing any future label. Finally, Top-K future utility is then measured under cross-cohort, cross-organisation, typed-rule, LOEO, sensitivity, bootstrap and error-analysis stress tests in the evaluation block. That is, arrows encode information flow and the anti-leakage boundary; as shown in Fig 1, evidence from future targets should only be included after ranking is complete.

B. Data Preparation

The data stage specifies the shared format used by all methods. We leverage Online Retail II [36] and the independent ARHyMST archive [37] for our experiments. The Online Retail II data is pre-processed by eliminating missing (sub)customer identifiers, perfect duplicates, cancellations and records with zero or negative quantity/price; the ARHyMST was not changed (i.e., null and self-duplicates were removed) before creating baskets per person-date-channel. Typed rules only capture context that is observable through history: Online Retail II incorporates predicates for country, basket-size, past-only customer-frequency, day-period, and weekday/weekend, while ARHyMST uses basket-size, past-only customer-frequency, and channel. Each retained typed rule has at least a context predicate and one product (with a product-oriented consequent).

This ensures that the contextual relevance is preserved for learning rules on multiple dimensions, without any leakage of future target information into feature construction.

 This work is licensed under a Creative Commons Attribution 4.0 International License



International Journal of Advanced Research in Computer and Communication Engineering
Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 11, November 2026
DOI: 10.17148/IJARCCE.2026.1511177

The cleaned, context-aware transactions are then divided into non-overlapping temporal windows to define the information boundary available at each cutoff.

C. Temporal Window Construction
Equation (1) defines the temporal boundary used at every cutoff.

$$W_t = \{ x_i \in D \mid (t-1)\Delta \leq \tau_i - \tau_0 < t\Delta \} \dots\dots\dots (1)$$

In Eq. (1), W_t is window t, x_i is transaction i, τ_i its timestamp, τ_0 the origin, and Δ is the window length; the half-open interval prevents overlap. The default benchmark references end-to-end 30-day windows, while extrinsic/typed tests are based on complete 14-day windows at the micro-lesson terminals (i.e., excluding partial terminal windows)

Likewise, each method reads the same conditions mined from the same historical windows and thresholds.

D. Frequent Rule Mining
Frequent itemsets of length ≤ 3 need support > 0.02 in each window, while candidate rules need confidence>0.50 and lift ≥ 1; see Eq. (2), which describes these measures for r: X → Y.

$$supp_t(r) = \frac{N_t(X \cup Y)}{N_t},\ conf_t(r) = \frac{N_t(X \cup Y)}{N_t(X)},\ lift_t(r) = \frac{conf_t(r)}{supp_t(Y)} \dots\dots\dots (2)$$

In Eq. (2), N_t is the window size,$N_t(X \cup Y)$counts co-occurrence, and $N_t(X)$ counts the antecedent. Support measures contemporary (ex-post) prevalence, confidence measures conditional reliability, and lift measures deviation from Y's marginal prevalence. It merges equivalent rules across windows, while preserving their histories, and uses the same thresholds for typed vs. In ARHyMST, executed typed runs generated 7,998 instances (780 unique rules), in OR09 12,097 instances (7,705 unique rules) and in OR10 8,111 instances (3,206 unique rules).

Then PTKOA gives a concrete description of how each metarule acts on the fixed history.

E. Historical Relationship Score
For rule r, Eq. (3) defines temporal persistence (TP), transaction-weighted coverage (WWC), and support stability (SS).

$$TP_r = \frac{|O_r|}{T}, WWC_r = \frac{\sum_{t \in O_r} N_t}{\sum_{t=1}^{T} N_t},\ SS_r = \frac{1}{1+\frac{\sigma(s_r)}{\mu(s_r)}} \dots\dots\dots\dots (3)$$

TP is a measure of recurrence, while WWC weights recurrence based on transaction exposure and SS penalises support variability. This is because TP and WWC are correlated occurrence measurements, as shown by Eq. (4). First, average them into OC, then combine OC with SS to get HRS.

$$OC_r = \frac{TP_r + WWC_r}{2}, HRS_r = \frac{OC_r + SS_r}{2} \dots\dots\dots (4)$$

HRS lies in [0,1], with larger values indicating recurrent, exposure-supported, stable rules; Algorithm 1 implements this calculation.

Algorithm 1. Historical Relationship Feature Extraction for Candidate Rules
Input: Temporal rule sets $R(1), R(2), \dots\dots, R(c)$; **transaction counts** $(N(1), N(2), \dots\dots, N(c)$; **historical cutoff** c
Output: Historical feature vector $z(r) = [\bar{S}_r, \bar{C}_r, \bar{L}_r, HRS_r]$**for every candidate rule** r
BEGIN
1. Construct the candidate-rule universe $C \leftarrow \cup_{t=1}^{c} R(t)$
2. FOR each rule $r \epsilon C$ **DO**
3. Identify the occurrence-window set $O_r \leftarrow \{t | r \epsilon R(t), 1 \leq t \leq c\}$
4. Collect the historical sequences of support, confidence, and lift for r.
5. Compute Temporal Persistence: $TP_r = |O_r|/c$

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 11, November 2026

DOI: 10.17148/IJARCCE.2026.1511177

6. Compute Transaction-Weighted Coverage: $WWC_r = \sum_{t \in O_r} N_t \,/\, \sum_{t=1}^{T} N_t$
7. Compute Support Stability: $SS_r = \frac{1}{1+\sigma(s_r)/\mu(s_r)}$
8. Compute the occurrence component: $OC_r = TP_r + WWC_r/2$
9. Compute the Historical Relationship Score: $HRS_r = OC_r + SS_r/2$
10. Compute historical mean support: $\bar{S}_r \leftarrow means\ (S_r)$
11. Compute historical mean confidence: $\bar{C}_r \leftarrow means(C_r)$
12. Compute historical mean log-lift: $\bar{L}_r \leftarrow means(\log(L_r))$
13. Construct the feature vector: $z(r) = [\bar{S}_r, \bar{C}_r, \bar{L}_r, HRS_r]$
14. END FOR
15. RETURN $z(r): r \epsilon C$
END

Complexity: $O(RT)$ for direct temporal-history construction, where R is the number of candidate rules and T is the number of historical windows.

Its only input to PKS is the mean support, mean confidence, mean log-lift, and HRS returned by Algorithm 1 for each candidate. Direct history assembly is O(RT) (the input sequence length embodied in the same unit as R), while four-component scoring is O(R) after features have been established.

The next step is to merge this temporal insight with classical rule quality.

F. Prioritised Knowledge Score

Equation (5) min-max normalises each non-degenerate conventional feature to [0,1].

$$\tilde{x}_r = \frac{x_r - min(x)}{max(x) - min(x)} \ldots\ldots\ldots \quad (5)$$

After that, the normalised support, confidence, and log-lift are combined with HRS using Eq. (6), as all edges are set to be non-negative and thus directly interpretable, with the additional requirement of binarity.

$$PKS_r = w_s \tilde{S}_r + w_c \tilde{C}_r + w_l \tilde{L}_r + w_h HRS_r,\ w_j \geq 0, \sum_j w_j = 1 \ldots\ldots \quad (6)$$

The 0.05 grid left (for OR09, selected PKS weights of (0.10, 0.00, 0.15, 0.75); for OR10, (0.25, 0.05, 0.10, 0.60); and for ARHyMST, a structure of (0.20; 020;025;035), with the bootstrap medians preserving dominant patterns.]

We define the recency comparator in equation (7): feature values are exponentially decayed with a half-life of h = 3 windows before source-only optimisation.

$$D_x(r,c) = \frac{\sum_{t \leq c} 2^{-(c-t)/h}\, x_{r,t}}{\sum_{t \leq c} 2^{-(c-t)/h}}, h = 3 \ldots\ldots\ldots (7)$$

H-PKS merges static quality, HRS and decay into a diagnostics variant intended to down-weight history when recency is favoured by source validation.

Future relevance is defined as follows, where we assign 0 to a rule not present in window c + 1 and otherwise take an average across the normalised future support, confidence and log-lift (equation (8)).

$$U_{r,c+1} = \begin{cases} 0, & r \notin R_{c+1}, \\ \dfrac{\tilde{S}_r^f + \tilde{C}_r^f + \tilde{L}_r^f}{3}, & r \in R_{c+1}. \end{cases} \ldots\ldots\ldots\ldots (8)$$

This has been used as a graded utility in NDCG@K [14], and treated future presence as ideal relevance for Precision@K; other than the candidate set and future target, all methods are identical.

Source Only Tuning and Frozen Transfer are given by Algorithm 2.

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940



International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 11, November 2026

DOI: 10.17148/IJARCCE.2026.1511177

G. PTKOA Ranking Algorithm

The source-to-target approach is carefully designed to ensure that future labels do not leak into model selection.

Algorithm 2. Source-Only PTKOA Optimisation and Frozen Knowledge Transfer
Input: Source-development cutoffs C_s; target cutoff c_t; simplex weight grid G; requested Top-K
Output: Source-selected frozen weight vector $w^*$and ranked target Top-K rule list
BEGIN
1. FOR each source-development cutoff $c \epsilon C_s$ DO
2. Apply Algorithm 1 to source windows $1, \ldots, c$ to obtain historical feature vectors $z_c(r)$.
3. Min-max normalise mean support, mean confidence, and mean log-lift using the source history.
4. END FOR
5. FOR each candidate weight vector $w = (w_s, w_c, w_l, w_h) \epsilon G$ DO
6. FOR each source-development cutoff $c \epsilon C_s$ DO
7. Compute the Prioritised Knowledge Score for every source candidate rule: $PKS_r(w) = w_s\tilde{S}_r + w_c\tilde{C}_r + w_l\tilde{L}_r + w_h HRS_r$
8. Rank source candidate rules in descending order of $PKS_r(w)$
9. Evaluate the ranking using next-window source NDCG@100.
10. END FOR
11. Compute the mean source-development NDCG@100 for the weight vector w.
12. END FOR
13. Select the source-optimal weight vector $w^* \leftarrow \arg max_{w \epsilon G}$ MeanNDCG@100(w)
14. FREEZE w^*.
Do not use target-future information for further optimisation.
15. Apply Algorithm 1 to target-history windows $1, \ldots\ldots, c_t$.
16. Normalise target-history support, confidence, and log-lift using only information available up to c_t.
17. FOR each target candidate rule r DO
18. Compute $PKS_r(w^*)$
19. END FOR
20. Sort all target candidate rules in descending order of $PKS_r(w^*)$.
21. Select the first K rules as the final prioritised knowledge set.
22. Only after the ranking is fixed, reveal target next-window relevance and compute prospective evaluation metrics.
TURN $(w^*)$and the ranked Top-K rule list.
END

Complexity: For four score components, the simplex contains 286 weight vectors at step 0.10 and 1,771 vectors at step 0.05. Target scoring is $O(Rd)$, and full ranking requires $O(R \log R)$.

Algorithm 2 freezes w^* before target evaluation; only after the Top-K list is produced is future utility revealed. The 0.10 and 0.05 simplex grids contain 286 and 1,771 vectors; target scoring is O(Rd), and full sorting is O(R log R).

H. Theoretical Properties

Since the score has five mathematical features that are relevant to audits, but do not claim validity or common transfer, we looked at how it behaves numerically.

Bounded: As each component is in [0,1] and the total weights are non-negative and sum up to 1, PKS is a convex combination of these components and so lies in [0,1].

Monotonicity: For any normalised component $z_j, \partial PKS/\partial z_j = w_j \geq 0$; increasing one proof component while keeping the others unchanged cannot lower the priority score.

Perturbation stability: Since $|w^{\mathrm{T}}(z - z')| \leq ||z - z'||\infty$, PKS is 1-Lipschitz under the infinity norm; a pairwise allowance greater than 2ε cannot reverse when each score changes by at most ε.

 This work is licensed under a Creative Commons Attribution 4.0 International License

DOI: 10.17148/IJARCCE.2026.1511177

Scale invariance: A positive-affine transformation $x' = ax + b\ (a > 0)$ does not change non-degenerate min–max normalised values and thus causes the PKS component invariant (constant).

Finite–grid optimality: The chosen vector is the exact data-driven maximiser over the fully specified source simplex grid; no guarantees are made about unsearched regions of the continuous simplex.

Computational complexity: Scoring is $O(Rd)$ when all historical features are executed; full ranking is O(R log R). Before sorting, the post-execution score is linear with respect to grid resolution for fixed d.

These properties form the mathematical description; experiments test what might be called effectiveness, robustness, and transferability, as well as where something fails**.**

IV. EXPERIMENTAL RESULTS AND DISCUSSION

A. Experimental Design

The evaluation is organised by five research questions: primary future ranking (RQ1), cross-organisation transfer (RQ2), typed multidimensional transfer (RQ3), robustness and unseen-environment behaviour (RQ4), and failure conditions (RQ5). Online Retail (II) — 2 annual cohorts ARHyMST — 14.9 million raw product records Tuning stays source-only in both full 30-day primary and 14-day ext/typed windows.

To avoid any interpretations arising from re-liking parameters, Table 3 fixes the executed windows, thresholds, source-only search spaces and ranking results along with dependence-aware re-evaluation before evaluating results.

B. Parameter Settings

Table 2. Experimental settings and source-only evaluation protocol.

Item	Executed setting
Windows	30-day primary; 14-day external/typed; complete windows only
Rule thresholds	support > 0.02; confidence > 0.50; lift ≥ 1; max length = 3
PKS search	source-only simplex: 0.10 primary; 0.05 audit (1,771 vectors)
Temporal decay	half-life = 3; decayed S, C, log-lift, presence
Learned baselines	Ridge α∈{0.01,0.1,1,10}; LambdaMART depth 2/3, 100/200 trees, lr=0.05
Ranking outcomes	NDCG@20/50/100/200; Precision@100; survival/presence

The competition assesses the competing methods using common mining thresholds and evaluation definitions. Statistical inference is performed using paired Wilcoxon tests and moving-block bootstrap (block length = 3; 10,000 resamples).

C. Primary Comparison and Ablation

This leads us to the following research question: RQ1: Does using historical evidence improve prospective Top-K ranking? The base ordering from the previous 30 days (bidirectional) can be viewed in Figure 2.

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 11, November 2026

DOI: 10.17148/IJARCCE.2026.1511177

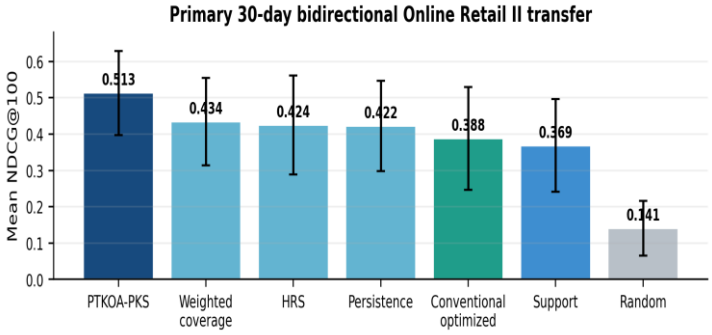


Fig. 2 Mean NDCG@100 across fourteen bidirectional Online Retail II annual-transfer cutoffs; error bars show between-cutoff variation.

Figure 2 indicates the mean NDCG@100, and we use error bars to represent variation between cutoffs. PTKOA obtains an NDCG@100 of 0.5132 ± 0.1156 and a Precision@100 of 0.3286 ± 0.1081, beating the optimised conventional composite, which achieved scores as follows: NDCG@100 = 0.3882±0.1413, Precision@100 = 0.2521±0.0917, winning in total from a total of 14 cutoffs only in 1. Ignoring HRS reduces the NDCG to 0.3844, ignoring support reduces the NDCG to only 0.4130, and using only HRS yields 0.4245.

Hence, traditional prevalence and historical recurrence/stability provide complementary information with the primary regime.

To make it easier to see the transfer direction from favourable to unfavourable, Table 3 presents both the primary and principal external/typed results in one table.

Table 3. Main quantitative results for item-based and typed transfer

Evaluation setting	PTKOA / PKS	Comparator	Δ
Primary 30-day OR	0.5132 (0.3286)	Conventional 0.3882 (0.2521)	+0.1250
OR → ARHyMST	0.8393 (0.9523)	Decay 0.7757 (0.9369)	+0.0636
ARHyMST → OR09	0.0323 (0.0180)	Decay 0.4876 (0.4410)	−0.4553
ARHyMST → OR10	0.1282 (0.0935)	Decay 0.4874 (0.4370)	−0.3592
Typed OR09 → AR	0.8523 (0.9762)	Decay 0.8269 (0.9638)	+0.0254
Typed OR10 → AR	0.8081 (0.9092)	Decay 0.9557 (0.9731)	−0.1476

The main asymmetry is already shown in the table: PTKOA can manage some forward directions, but decay beats many reverse or typed transfers.

D. Sensitivity, Scalability and Runtime

Figure 3 shows the wins and losses of PTKOA for each of the nine threshold configurations transferred over their variable windows by using blue to indicate wins and orange to represent losses from these transfers. We find that PTKOA wins in 23 of the 27 settings, winning all eighteen 15-day and 30-day settings, while losing four times among the nine sparser configurations with a 60-day window. It separates a strong dense-window edge from the long-window barrier rather than averaging that barrier away.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 11, November 2026

DOI: 10.17148/IJARCCE.2026.1511177

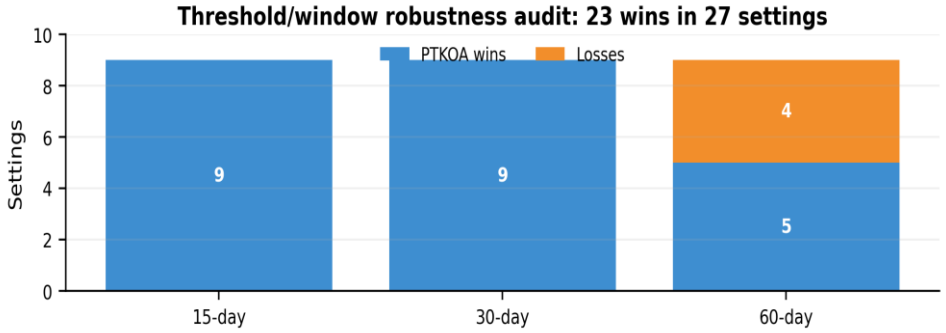


Fig. 3 Threshold/window robustness: PTKOA wins every 15-day and 30-day setting and 5 of 9 60-day settings.

ARHyMST has 14.58M cleaned records and 2.64M baskets, broken down into 19 windows, producing a total of 16,020 item-based rule instances and 7,998 typed instances. The Online Retail standard results returned 12,097 and 8,111 rows. Given d = 4, the scoring of PTKOA before sorting is O(R), so the ranking overhead is low compared to mining.

For the Online Retail dataset, ARHyMST window preparation time required on average 2.07 s (Maximum: 2.53 s; Total: 39.39 s), and for the typed miner it took 0.162/0.152 s per Online Retail window. These are traces local to an implementation that studiously avoids cross-machine speed claims (ie, excluding raw file ingestion time).

E. Robustness and Predictive Evaluation

In ARHyMST, a (0.20, 0.20, 0.25, 0.35) vector would be supported for every variable; however, median HRS weights were preferred in both the results from OR09 (median: 0.75; IQR: 0.70–0.85) and OR10 (median: 0.60; IQR: 00).50 to 00.) This asymmetry was left nearly intact in the every-third-cutoff analyses, and so they concluded the source was robust but not portable.

It includes NDCG@20/50/100/200, Precision@100, support-only and three-window evaluation settings (cuts) of survival relevance, cumulative support and MAP & ROC AUC. The macro NDCG@100 values for optimised decay, Ridge, LambdaMART, HRS and original PTKOA are 0.5956, 0.4832, 0.4636, 0.4272 and 0.3701 across six external directions, respectively.

F. Cross-Dataset and Strong-Baseline Transfer

RQ2 investigates organisation shift. Using grouped bars to compare Original PKS, H-PKS, temporal decay and HRS across three source→target directions in Figure 4 (larger bars are better mean NDCG@100). Indeed, PTKOA (trained on ARHyMST) is performing worse than PKS (trained on Online Retail) for both targets – Please refer NDCG@100 [0.8393 vs 0.0323] and Precision@100 [0.9523 vs 0.1282] Results for H-PKS show a temporal decay at 0.4875/0.4860 as well, which is nearly in line with the former (note H-PKS reweights toward recency). Hence the portability of recurrence is directional, not universal.

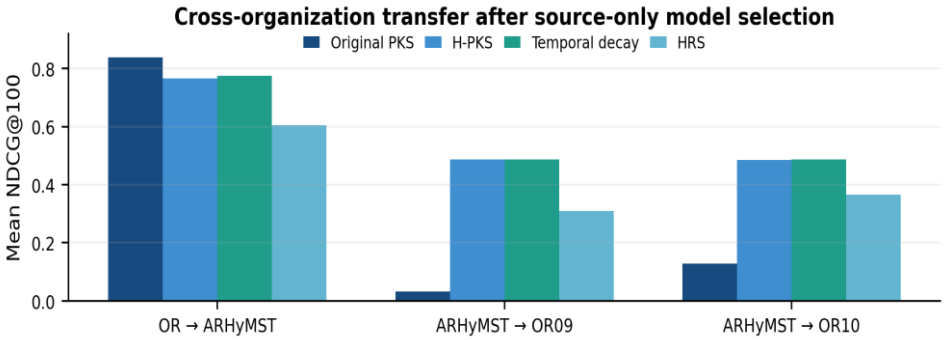


Fig. 4 Organisation-level transfer after source-only model selection

.

In Table 4, we again test against our optimised temporal decay as well as LambdaMART and Ridge for the same set of external directions.

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 11, November 2026

DOI: 10.17148/IJARCCE.2026.1511177

Table 4. Strong-baseline source-cohort external stress test (mean NDCG@100).				
Source → target	Original PKS	H-PKS	Temporal decay	LambdaMART / Ridge
OR09 → ARHyMST	0.8395	0.8493	0.8719	0.8643 / 0.6220
OR10 → ARHyMST	0.7406	0.7526	0.7253	0.8220 / 0.7399
ARHyMST → OR09	0.0394	0.4952	0.4952	0.0374 / 0.3108
ARHyMST → OR10	0.1465	0.4909	0.4909	0.2564 / 0.3736

Table 4 shows the difference in performance across strong baselines depends on the direction, ruling out a weak-comparator explanation and supporting the recurrence-versus-recency boundary.

G. Typed Multidimensional and LOEO Evaluation

RQ3 asks whether the transfer boundary survives true typed multidimensional rules. PTKOA and temporal-decaying bars are juxtaposed (side-by-side) in Fig. 5 for all six source→target directions to maintain visibility on the direction-specific gains and failures. For other directions, PTKOA achieves 0.8523 whereas OR09→ARHyMST only achieves 0.8269; however, for two other directions such as OR10→ARHyMST (0.9557 versus 0.8081) and ARHyMST→OR09, we see temporal decay dominating even more strongly (0.4676 versus 0.0632). Thus, the inclusion of context makes for a stronger representation of the rule but does not remove the recurrence-versus-recency portability boundary.

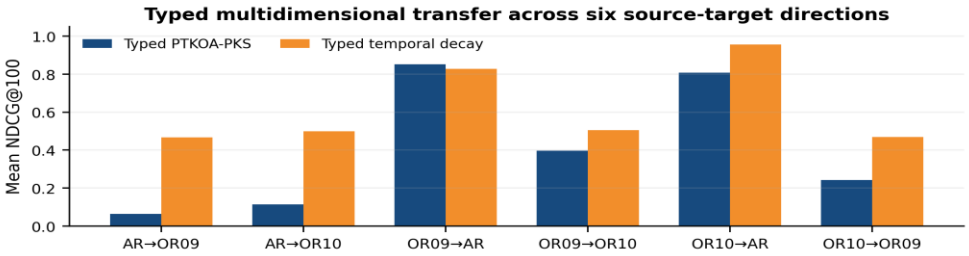


Fig. 5 Multi-source Boxed from local reference methodology across six transfer directions.

The PTKOA increases to 0.8523 as opposed to 0.8269 for OR09→ARHyMST, but decay is much higher on a few directions: OR10 →ARHyMST (0.9557 vs 0.8081) and ARHyMST→OR09 (0.4676 vs 0.0632). This shows that context alone does not eliminate the boundary of portability.

Similarly, the LOEO assessment leads to a conditional result: PKS (0.8419 vs decay 0.7588) is favoured when ARHyMST is held out, and holding out OR09 or OR10 favours decay (0.4946 vs 0.2404 and 0.4899 vs 0.3432, respectively). So, portability is not fundamentally solved by source-splitting.

H. Statistical Significance and Error Analysis

Building on the basic idea of using paired tests, rolling cutoffs are based on interval estimates from a moving-block bootstrap (block length 3). The no-difference reference is represented as a vertical dashed line at zero (Fig. 6): intervals entirely to the right promote PTKOA, intervals entirely to the left favour temporal decay, and those that cross zero indicate uncertainty. The output from the direct economy-wide comparison of PTKOA versus the benchmark using the standard approach yields one-sided Wilcoxon p = 0.000427, Holm-corrected p = 0.001709, and rank-biserial effect size (low scores better) = 0.9238. These pacifiers for outer PTKOA-to-decay comparisons unsuccessfully inform future external comparisons between dependence-aware intervals.

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 11, November 2026

DOI: 10.17148/IJARCCE.2026.1511177

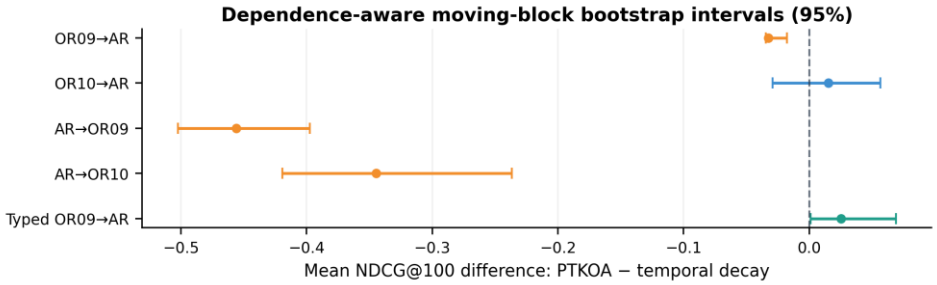


Fig. 6 99.8 NDCG@100 (1 pt) 95% intervals for PTKOA minus optimised temporal-decay (block length 3)

Both of these are mirrors of OR09→ARHyMST, which has a mean Δ = −0.0324 (95% decay block interval: [−0.0347, −0.0179]) while ARHyMST→OR09 and ARHyMST→OR10 are −0.4558 and −0.3445 respectively Note that the confidence intervals for OR10→ARHyMST is +0.0153 and overlap zero; therefore we fail to reject the null hypothesis here as well. For all 13 cutoffs, the adjusted range for OR09→ARHyMST (an indication of two influences: [+0.06886,+0.00095]) produces p = 0.1272 (two-sided Wilcoxon). If an incomplete form is avoiding one-to-one judgement of uncertainty where the cutoff dependence and sample size differ, two summaries can be reported instead.

RQ5 What observable conditions explain the PTKOA–decay gap? The conducted correlations are reported in Table 5 before the interpretation on the level of direction.

Table 5. Error-analysis diagnostics associated with the PTKOA–decay performance gap.

Diagnostic	Spearman ρ	p-value	Meaning
Future presence rate	+0.7201	3.34×10^{-18}	Better when more historical candidates survive
Candidate rules (log)	−0.6290	5.17×10^{-13}	Relative performance declines as candidate space expands
Future-present rules	+0.5307	4.89×10^{-9}	More surviving rules reduce the historical-ranking gap
Mean JS shift	−0.1178	0.2291	Simple marginal shift is not significant

Disaggregating these diagnostics shows that the improvement in PTKOA is due to the survival of more historical candidates, but its relative performance declines as the candidate universe grows; even a simple marginal shift based on Jensen–Shannon between news/candidates is not significant. ARHyMST→OR09 fails 16,353 candidate rules with a future-presence rate of 0.0114 and mean gap of −0.4558, while OR10→ARHyMST contains ∼1,252 candidates (future-presence rate = 0.6403; mean gap = +0.0153).

I. Discussion

A caveat for this assertion is that while PTKOA does deliver this quality in the main within-retailer dataset, it can also be transferred from Online Retail to ARHyMST. On the other hand, reverse transfers, macro averages, and a class of more sophisticated rules that can extend LOEO show that getting back to the bloom cannot be warbled. The temporal decay may be more effective than either original PKS or nonlinear learned ranker, preferably under a stronger shift (as in Stimulus II).

The PTKOA is an audit layer between rule discovery and downstream decision or explanation modules, for knowledge-based systems. H-PKS uses its historical evidence to control this contradictory filtering, while the parts of H-PKS visible to users explain why a rule appears. This needs to remain up-to-date (in comparison, or as an alternative) during deployment.

V. CONCLUSION AND FUTURE WORK

The model PTKOA was considered an interpretable knowledge-prioritisation framework that avoids the direct utilisation of upcoming target labels during the selection stage. It obtains NDCG@100 = 0.5132 and Precision@100 = 0.3286 in the main Online Retail II benchmark, does better than an equally optimised conventional composite, and wins for 13 out

DOI: 10.17148/IJARCCE.2026.1511177

of 14 cutoffs by this measure. Ablation, sensitivity, fine-grid search, and bootstrap stability confirm the historical contribution in this regime.

Independent ARHyMST transfer, strong baselines, LOEO, moving-block bootstrap & typed rules show the core boundary: can-transfers-stay-transfers avoid decay or remain stable to map back over durations that happen to arrange, but opposite transfers may lose by finding all classes across an AGC under a different kind of mapping? This is why PTKOA should be regarded as an auditable post-mining layer in which the accurate history of its contributions must also be verified.

Limitations include the requirement of two public retail organisations, sensitivity to window size, difference in typed-predicate granularity, no independent business outcome, an uninformative simple Jensen–Shannon shift diagnostic, and implementation-local runtimes. Additional future work could extend PTKOA to streaming and privacy-preserving distributed settings, as well as develop source-only recurrence-versus-recency diagnostics and test in additional organisations and settings by enriching the leakage-safe context while evaluating downstream utility.

A. Data and repository availability

Only Online Retail II [36] and ARHyMST [37] are public. It is essentially a catch-all of files that consists of fine-grid searches which bootstrap/LOEO and spaced-cutoff audits, as well as typed-rule outputs/cross-dataset results/error-analysis files/moving-block bootstrap outputs/SHA-256 manifests. Only executed evidence is reported.

ACKNOWLEDGMENT

I express my sincere gratitude to my research supervisor, Dr Eedi Hemalatha, for the invaluable guidance, encouragement, and continuous support throughout this research. My deepest and most heartfelt appreciation goes to my beloved wife, whose unwavering love, patience, sacrifices, and constant encouragement have been my greatest source of strength. This achievement would not have been possible without her support. I also thank my colleagues and friends for their encouragement, cooperation, and valuable support during this research journey. Finally, I extend my gratitude to everyone who contributed, directly or indirectly, to the successful completion of this work.

REFERNCES

[1] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," *ACM SIGMOD Record*, vol. 22, no. 2, pp. 207–216, 1993, doi: 10.1145/170036.170072.

[2] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules in large databases," in *Proc. 20th Int. Conf. Very Large Data Bases (VLDB)*, 1994, pp. 487–499.

[3] J. Han and Y. Fu, "Discovery of multiple-level association rules from large databases," in *Proc. 21st Int. Conf. Very Large Data Bases (VLDB)*, 1995, pp. 420–431.

[4] H. Mannila, H. Toivonen, and A. I. Verkamo, "Discovery of frequent episodes in event sequences," *Data Mining and Knowledge Discovery*, vol. 1, no. 3, pp. 259–289, 1997, doi: 10.1023/A:1009748302351.

[5] M. Kamber, J. Han, and J. Y. Chiang, "Metarule-guided mining of multi-dimensional association rules using data cubes," in *Proc. 3rd Int. Conf. Knowledge Discovery and Data Mining (KDD)*, 1997, pp. 207–210.

[6] C. C. Aggarwal and P. S. Yu, "A new framework for itemset generation," in *Proc. ACM Symp. Principles of Database Systems (PODS)*, 1998, pp. 18–24, doi: 10.1145/275487.275490.

[7] B. Liu, W. Hsu, and Y. Ma, "Mining association rules with multiple minimum supports," in *Proc. ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 1999, pp. 337–341, doi: 10.1145/312129.312274.

[8] J. Han and Y. Fu, "Mining multiple-level association rules in large databases," *IEEE Trans. Knowledge and Data Engineering*, vol. 11, no. 5, pp. 798–805, 1999, doi: 10.1109/69.806937.

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

[10] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," in *Proc. ACM SIGMOD Int. Conf. Management of Data*, 2000, pp. 1–12, doi: 10.1145/342009.335372.

[11] J. M. Ale and G. H. Rossi, "An approach to discovering temporal association rules," in *Proc. ACM Symp. Applied Computing*, 2000, pp. 294–300, doi: 10.1145/335603.335770.

[12] J. Pei *et al.*, "PrefixSpan: Mining sequential patterns efficiently by prefix-projected pattern growth," in *Proc. IEEE Int. Conf. Data Engineering (ICDE)*, 2001, pp. 215–224, doi: 10.1109/ICDE.2001.914830.

[13] T. G. Dietterich, "Machine learning for sequential data: A review," in *Structural, Syntactic, and Statistical Pattern Recognition*, Lecture Notes in Computer Science, vol. 2396, 2002, pp. 15–30, doi: 10.1007/3-540-70659-3_2.

 This work is licensed under a Creative Commons Attribution 4.0 International License

DOI: 10.17148/IJARCCE.2026.1511177

[14] K. Järvelin and J. Kekäläinen, "Cumulated gain-based evaluation of IR techniques," *ACM Trans. Information Systems*, vol. 20, no. 4, pp. 422–446, 2002, doi: 10.1145/582415.582418.

[15] Y. Li, P. Ning, X. S. Wang, and S. Jajodia, "Discovering calendar-based temporal association rules," *Data & Knowledge Engineering*, vol. 44, no. 2, pp. 193–218, 2003, doi: 10.1016/S0169-023X(02)00135-0.

[16] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*. Pearson, 2006.

[17] A. J. T. Lee and C.-S. Wang, "An efficient algorithm for mining frequent inter-transaction patterns," *Information Sciences*, vol. 177, no. 17, pp. 3453–3476, 2007, doi: 10.1016/j.ins.2007.03.007.

[18] L. De Raedt, *Logical and Relational Learning*. Springer, 2008, doi: 10.1007/978-3-540-68856-3.

[19] C. J. C. Burges, "From RankNet to LambdaRank to LambdaMART: An overview," Microsoft Research, Tech. Rep. MSR-TR-2010-82, 2010.

[20] J. Han, M. Kamber, and J. Pei, *Data Mining: Concepts and Techniques*, 3rd ed. Morgan Kaufmann, 2012, doi: 10.1016/C2009-0-61819-5.

[21] W. Fan and A. Bifet, "Mining big data: Current status, and forecast to the future," *SIGKDD Explorations*, vol. 14, no. 2, pp. 1–5, 2013, doi: 10.1145/2481244.2481246.

[22] C. C. Aggarwal and J. Han, Eds., *Frequent Pattern Mining*. Springer, 2014, doi: 10.1007/978-3-319-07821-2.

[23] C. C. Aggarwal, *Data Mining: The Textbook*. Springer, 2015, doi: 10.1007/978-3-319-14142-8.

[24] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2016, pp. 785–794, doi: 10.1145/2939672.2939785.

[25] A. Segura-Delgado, M. J. Gacto, R. Alcalá, and J. Alcalá-Fdez, "Temporal association rule mining: An overview considering the time variable as an integral or implied component," *WIREs Data Mining and Knowledge Discovery*, vol. 10, no. 4, Art. no. e1367, 2020, doi: 10.1002/widm.1367.

[26] Q. Zhao, Q. Li, D. Yu, and Y. Han, "TSARM-UDP: An efficient time series association rules mining algorithm based on up-to-date patterns," *Entropy*, vol. 23, no. 3, Art. no. 365, 2021, doi: 10.3390/e23030365.

[27] X. Liu, F. Feng, Q. Wang, R. R. Yager, H. Fujita, and J. C. R. Alcantud, "Mining temporal association rules with temporal soft sets," *Journal of Mathematics*, vol. 2021, Art. no. 7303720, 2021, doi: 10.1155/2021/7303720.

[28] X. Du and F. Yu, "A fast algorithm for mining temporal association rules in a multi-attributed graph sequence," *Expert Systems with Applications*, vol. 192, Art. no. 116390, 2022, doi: 10.1016/j.eswa.2021.116390.

[29] L. Wang, L. Gui, and P. Xu, "Incremental sequential patterns for multivariate temporal association rules mining," *Expert Systems with Applications*, vol. 207, Art. no. 118020, 2022, doi: 10.1016/j.eswa.2022.118020.

[30] O. Bahri, P. Li, S. Filali Boubrahimi, and S. M. Hamdi, "Shapelet-based temporal association rule mining for multivariate time series classification," in *Proc. IEEE Int. Conf. Big Data*, 2022, pp. 242–251, doi: 10.1109/BIGDATA55660.2022.10020478.

[31] M. R. H. Iman, P. Chikul, G. Jervan, H. Bahsi, and T. Ghasempouri, "Anomalous file system activity detection through temporal association rule mining," in *Proc. 9th Int. Conf. Information Systems Security and Privacy*, 2023, pp. 733–740, doi: 10.5220/0011805100003405.

[32] K. Liang *et al*., "A survey of knowledge graph reasoning on graph types: Static, dynamic, and multi-modal," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 46, no. 12, pp. 9456–9478, 2024, doi: 10.1109/TPAMI.2024.3417451.

[33] R. Liu, X. Wang, S. Xie, X. Luo, and H. Su, "FL-Evo: Jointly modeling fact and logic evolution patterns for temporal knowledge graph reasoning," *Expert Systems with Applications*, vol. 286, Art. no. 128081, 2025, doi: 10.1016/j.eswa.2025.128081.

[34] M. Milella, G. Pagliarini, G. Sciavicco, and I. E. Stan, "Temporal association rules from motifs," in *Proc. 32nd Int. Symp. Temporal Representation and Reasoning (TIME)*, Leibniz Int. Proc. Informatics (LIPIcs), vol. 355, 2025, pp. 19:1–19:7, doi: 10.4230/LIPIcs.TIME.2025.19.

[35] F. Li, Y. Zhao, and X. Wang, "Mining weighted temporal association rules in dynamic complex systems via non-attributed graph sequence with fuzzy structure," *Systems*, vol. 14, no. 7, Art. no. 735, 2026, doi: 10.3390/systems14070735.

[36] D. Chen, "Online Retail II," UCI Machine Learning Repository, 2012, doi: 10.24432/C5CG6D. [Online]. Available: https://archive.ics.uci.edu/dataset/502/online+retail+ii

[37] V. Đurišić, S. Vujošević, L. Kašćelan, and S. Vuković, "ARHyMST anonymized retail transaction dataset," Zenodo, 2026, doi: 10.5281/zenodo.20788609. [Online]. Available: https://doi.org/10.5281/zenodo.20788609.

 This work is licensed under a Creative Commons Attribution 4.0 International License

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Impact Factor 8.471 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 15, Issue 11, November 2026

DOI: 10.17148/IJARCCE.2026.1511177

BIOGRAPHY



Mr B.B.L.V.Prasad holds a B.E. & M.Tech (CSE) from OU and is pursuing a PhD at JNTUH, Hyderabad. He began his career at RAM Informatics Ltd. (CMM Level 5) and ProArch IT Solution (MNC), Hyderabad. He served as Assistant Professor at BIET and AIET, HOD I/C, and guided several B.Tech and M.Tech projects. Certified in Personality Development (Vivekananda Institute), Soft Skills, and Wipro Mission 10X at BIET, he received the Best Teacher Award and attended several workshops and FDPs. Associate Professor (CSE), Website Coordinator & TPO at SKEC, he is crrently an Associate Professor (CSE) at SCIT, Khammam. His expertise includes DevOps, Neural Networks, Deep Learning, DAA, Java, C, Python, DBMS, FLAT, Compiler Design, and more. He has participated in AICTE, JNTUH, NBA, NAAC & Spot Inspections, published 10+ international journal articles (including a Scopus-indexed paper), and holds a patent. He authored "Artificial Intelligence: Principles & Applications" and runs PRASAD CAREERIZATION on YouTube Channel:**http://www.youtube.com/c/prasadcareerization**



Dr Eedi Hemalatha completed her B.Tech and M.Tech in Computer Science and Engineering at JNTUH with Distinction. Her PhD in Parallel Graph Algorithms, Indian Institute of Technology Hyderabad (IITH). She is currently working as a Professor at JNTUH. Her areas of research interest include Parallel, Concurrent and Distributed Computing, HPC, Grid & Cloud Computing and Parallel Graph Algorithms. She has published in 10+ International Journals, 3+ International Conferences, and publications in national Journals, National Conferences, and Workshops. Teaching interests include Computer Programming, Data Structures, Operating Systems, Linux Internals, Scripting Languages, Parallel Computing, HPC, and Cloud Computing.