When Do Unsupervised Regimes Help? A Downstream Utility Study for Sensor Time Series

Ebrahim Behrouzian Nejad[1] and Inês Dutra[1,2]

[1] Department of Computer Science, Faculty of Sciences, University of Porto, Porto, Portugal
Behrouzian.e@fc.up.pt
[2] CINTESIS@RISEHealth, Porto, Portugal
dutra@fc.up.pt

Abstract. Unsupervised regime detection in time series is usually evaluated as a clustering problem: are the discovered regimes compact, stable, and interpretable? For operational sensor time series, this is not enough. A detected regime is useful only if it becomes a reusable pattern vocabulary. This paper studies that question. Starting from previously detected regimes in six settings — S1–S3, hierarchical S3c, Power, and WRM — we evaluate five downstream tasks from the same regime artifacts: prototype-based annotation, regime-conditioned motif discovery, transition-probability anomaly detection, Markov regime forecasting, and prototype- or regime-mean reconstruction. Utility is task- and data-dependent. On the synthetic settings, annotation reaches 92.2–100%, anomaly detection AUC 0.775–1.000, forecasting gains up to 55.6 percentage points, reconstruction $R^2 = 0.812$ to near-perfect, and S1–S3 motif distances are near-zero. On Power and WRM, annotation reaches 80.7% on Power, anomaly detection reaches AUC 0.748 on Power and 0.707 on WRM, Markov forecasting improves over the majority baseline by 24.4 percentage points on Power, and regime-mean reconstruction explains $R^2 = 0.718$ of Power variance. They are less useful when stochastic within-regime variation dominates, as in WRM reconstruction ($R^2 = 0.134$) and WRM annotation (49.1%). We identify two label-free predictors of downstream success: the prototype discrimination ratio and within-regime variance. The paper therefore reframes regime detection as pattern vocabulary construction: discovered regimes should be judged not only by whether they cluster well, but by when their structure transfers to downstream analytical tasks.

Keywords: Sensor time series · Regime detection · Pattern mining · Downstream utility · Unsupervised learning

1 Introduction

Sensor data streams from buildings, facilities, and industrial systems frequently move through recurring operating modes. Unsupervised regime detection seeks

to recover those modes without labelled examples, usually by segmenting the signal and clustering the resulting segments. Recent work in change-point detection, time-series clustering, and motif discovery provides useful tools for this setting, including offline segmentation methods, elastic distances, matrix profile search, and temporal clustering models [1–5]. Yet a practical question remains under-explored: once regimes have been discovered, do they actually help with downstream sensor analytics?

This question matters because a clustering can be internally stable and still provide little operational value. Conversely, a regime vocabulary may be imperfect as a final segmentation but still be useful as a compact state-space abstraction for later tasks. In sensor operations, this abstraction can serve several roles: prototypes can label new segments, regime-conditioned subseries can expose motifs, transition matrices can flag unusual regime transitions, regime sequences can support forecasting, and regime means can reconstruct the dominant signal level.

This paper evaluates when unsupervised regimes help in this downstream sense. We deliberately avoid re-presenting the full regime-detection pipeline as the main contribution. Instead, we treat the output of such a pipeline as a *regime vocabulary*: a set of prototypes, segment labels, transition probabilities, and per-segment quality scores. The novelty is the evaluation lens: we ask which downstream tasks benefit from those regime artifacts and which structural signals predict success before deployment.

A regime constitutes a useful mined pattern only when it can be used, interpreted, and trusted in subsequent analysis. The paper is organized around three claims:

- regimes should be evaluated not only as clusterings, but as reusable vocabularies;
- different downstream tasks exploit different regime artifacts: prototypes, labels, conditioned subseries, transitions, and regime means;
- downstream utility is empirically predictable from two label-free structural indicators: the *prototype discrimination ratio* Δ and *within-regime variance*.

Sections 5 and 6 show that all three claims hold empirically across five downstream tasks and six datasets, and that the two structural indicators predict downstream success before any task is run.

The remainder of the paper is organised as follows. Section 2 reviews related work and identifies the open gap this paper addresses. Section 3 defines the regime vocabulary and describes the construction pipeline. Section 4 describes the evaluation design. Section 5 presents downstream utility results for all five tasks. Section 6 synthesises the cross-application findings and identifies the two label-free structural predictors. Section 7 discusses limitations, and Section 8 concludes.

2 Related Work

Segmentation, regime detection, and pattern discovery. Offline change-point detection [1] and temporal clustering methods such as TICC [3], k-Shape [4], adap-

tive FastDTW-based clustering [18], and random-kernel or kernel-ensemble clustering [16, 19] recover coherent segments or regime partitions from unlabelled time series; HMM-style models [8, 9] additionally capture regime-transition probabilities. Pattern discovery methods — motif search [12], the matrix profile [5, 6, 17], shapelets [11], and DTW matching [2] — find or compare recurring subsequences in isolation. All evaluate their output at a single endpoint (cluster quality, motif distance, or supervised accuracy) without asking what the discovered structure enables for subsequent analysis.

Anomaly detection, classification, and forecasting. Anomaly detection spans statistical, distance-based, and reconstruction methods [10], motivating our multi-signal APP3 design. Supervised classification benchmarks, including the UCR archive [15], and multivariate forecasting models such as DPFN [13] require task-specific supervision unavailable in our setting.

Open gap. None of the above treats the complete collection of regime artifacts — prototypes, labels, transition matrices, and quality scores — as a unified, reusable vocabulary, and none asks which structural properties of that vocabulary predict downstream task success before deployment. Our contribution is not a new segmentation or forecasting method, but a pattern-mining perspective on detected regimes as reusable vocabularies whose utility varies systematically across downstream tasks.

3 What Is a Regime Vocabulary?

3.1 Time Series, Segments, Regimes, and Patterns

A *time series* is an ordered sequence of observations $x_{1:T} = (x_1, \ldots, x_T)$ from one sensor or aggregated signal. A *segment* is a contiguous interval $s_i = x_{a_i:b_i}$ whose observations are assumed to be locally coherent. Change-point detection partitions the full signal into such intervals. A *regime* is a recurring class of segments with similar shape, level, temporal context, or statistical behaviour. A regime becomes a *mined pattern* when its artifacts — the medoid prototype, transition probabilities, duration statistics, and recurring motifs — are extracted and made available for downstream reuse; the full collection of such artifacts across all detected regimes constitutes the *regime vocabulary*.

Figure 1 gives the visual intuition: boundaries split the signal into segments, while colours show that the same regime can recur in non-contiguous intervals. Regime detection therefore means finding both the segment boundaries and the recurring regimes those segments belong to.

Figure 2 shows why utility varies by dataset. The S3c hierarchical dataset has day-type structure, the Power dataset [14] has repeated weekday peaks and lower baselines, and WRM is sparse and event-driven. These signal differences later explain why the same vocabulary helps some tasks more than others.

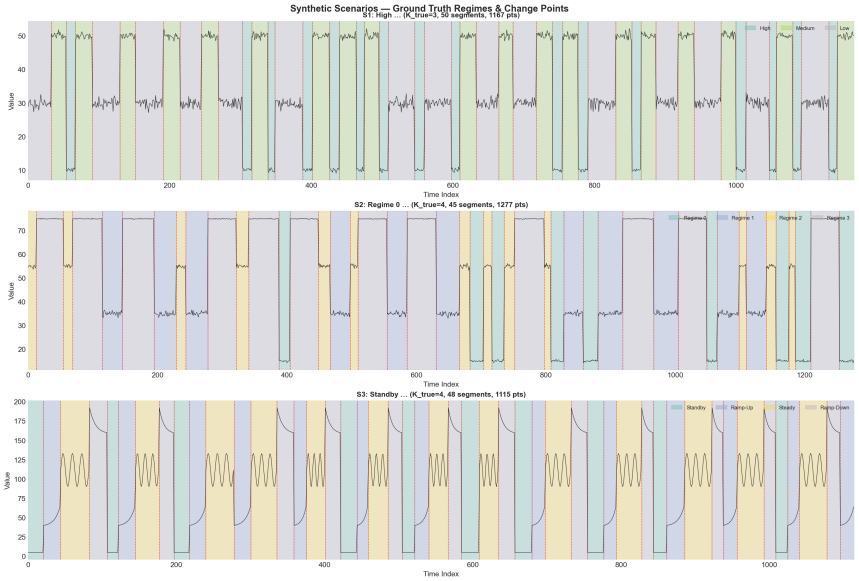
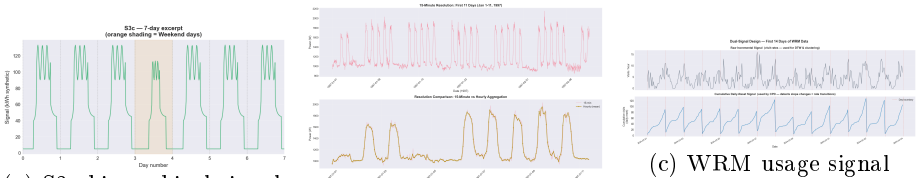
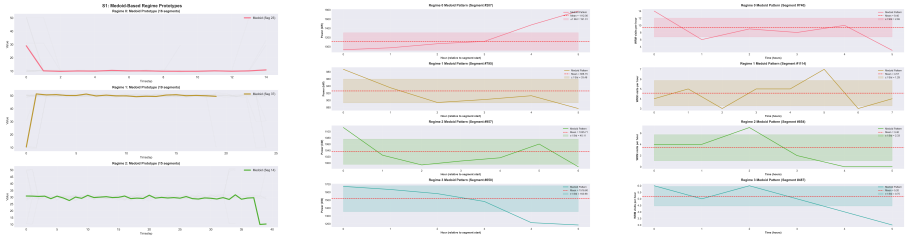


Fig. 1. Time series, segments, and regimes. Dashed lines mark segment boundaries; colours indicate recurring regimes. S1/S2 are statistical regimes, while S3 contains industrial operating phases.



(a) S3c hierarchical signal (b) Power excerpt (c) WRM usage signal

Fig. 2. Dataset gallery. S3c has hierarchical day-type structure; Power is level-driven and periodic; WRM is event-driven and stochastic.



(a) Synthetic S1 prototypes (b) Power prototypes (c) WRM prototypes

Fig. 3. Detected regime shapes. Synthetic regimes are clean, Power regimes are level-driven, and WRM regimes are noisy and event-driven.

We start from detected regimes and ask whether their artifacts are useful. Figure 3 shows medoid prototypes: observed segments, not averages, that can be inspected, matched, and reused.

Table 1 summarizes the detected vocabularies and their expected downstream behaviour.

Table 1. Detected-regime atlas used by the downstream tasks (K^* = selected number of regimes per dataset).

Dataset	K^*	Regime meaning	Expected downstream behaviour
S1	3	clean statistical regimes	high annotation, anomaly, reconstruction
S2	4	overlapping statistical regimes	high utility; forecasting sample-size limited
S3	4	standby/ramp/steady phases	strong motifs and transitions
S3c	4	day-types plus intra-day phases	hierarchy useful; persistence strong
Power	4	weekend/rest to peak demand	strong reconstruction and forecasting
WRM	4	near-idle to high usage	motifs/anomaly useful; annotation weaker

3.2 High-Level Construction Process

The detection pipeline is not the contribution of this paper; Fig. 4 shows how raw sensor data is converted into the regime vocabulary through segmentation, pairwise segment comparison, clustering, and artifact extraction. The five downstream tasks are APP1 annotation, APP2 motif discovery, APP3 anomaly detection, APP4 forecasting, and APP5 reconstruction.

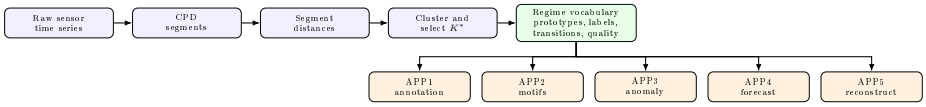


Fig. 4. Construction and reuse of a regime vocabulary for five downstream tasks (APP1–APP5).

4 Evaluation Design

The evaluation uses the six datasets in Table 1 and the five tasks in Table 2. Anomaly detection and forecasting are validated against explicit baselines; annotation, motif discovery, and reconstruction assess the structural content of the regime vocabulary directly.

Structural predictors. Two label-free quantities serve as pre-deployment screening indicators throughout § 5–6. The *prototype discrimination ratio* Δ compares inter-prototype separability with within-regime scatter: $\Delta = \bar{d}(\mathcal{P})/\bar{\sigma}_{\text{intra}}$,

Table 2. Input–operation–output view of the five downstream tasks.

Task	Input from vocabulary	Operation	Output / metric
APP1 annotation	prototypes, segment distances	nearest-prototype / ensemble / classifier	segment labels, accuracy, Δ
APP2 motifs	regime-conditioned subseries	matrix profile search; DTW/prototype matching	motif pairs, motif distance
APP3 anomalies	transition matrix, silhouette, distances	low-probability transition and multi-signal scoring	AUC, consensus anomaly flags
APP4 forecasting	regime sequence, transition matrix	Markov1/Markov2 prediction	next-regime accuracy, horizon decay
APP5 reconstruction	labels, prototypes, regime means	prototype or regime-mean substitution	R^2, RMSE reduction

where $\bar{d}(\mathcal{P}) = \frac{2}{K^*(K^*-1)} \sum_{j<k} \mathrm{DTW}(p_j, p_k)$ is the mean pairwise DTW distance between prototypes and $\bar{\sigma}_{\text{intra}} = \frac{1}{K^*} \sum_k \mathrm{mean}_{s_i \in C_k} \mathrm{DTW}(s_i, p_k)$ is the mean within-regime scatter; $\Delta > 1$ indicates prototypes are better separated than within-regime noise, $\Delta < 1$ warns of overlap. The *within-regime variance* $\bar{\sigma}^2 = \frac{1}{K^*} \sum_k \mathrm{var}_{s_i \in C_k} \mathrm{DTW}(s_i, p_k)$ measures internal regime heterogeneity. Both are empirical indicators, not theoretical guarantees.

5 Downstream Utility Results

5.1 Prototype-Based Annotation

Prototype-based annotation treats the medoids as a reference library. A new segment is assigned to the nearest prototype under DTW distance:

$$\hat{r}_i = \arg\min_k \mathrm{DTW}(s_i, p_k), \tag{1}$$

where s_i is the i-th test segment, p_k the prototype of regime k ($k = 1, \ldots, K^*$), and $\hat{r}_i$ the assigned regime label. We ask whether the discovered prototypes are discriminative enough to label new segments without any supervised training.

The results show a clear separability effect. Synthetic datasets achieve 100% annotation accuracy, with discrimination ratios Δ between 1.947 and 4.534. Power reaches 80.7% accuracy with $\Delta = 1.759$. WRM is much harder: its best method, a logistic-regression variant using DTW and temporal features, reaches only 49.1%, although this improves over single-medoid DTW (28.7%) and medoid ensembles (39.6%). The reason is structural: WRM has $\Delta = 0.691 < 1$, meaning within-regime scatter exceeds inter-prototype separation.

Takeaway. Annotation is viable when prototypes are more separated than the typical within-regime scatter; the discrimination ratio Δ is a label-free warning signal before deployment.

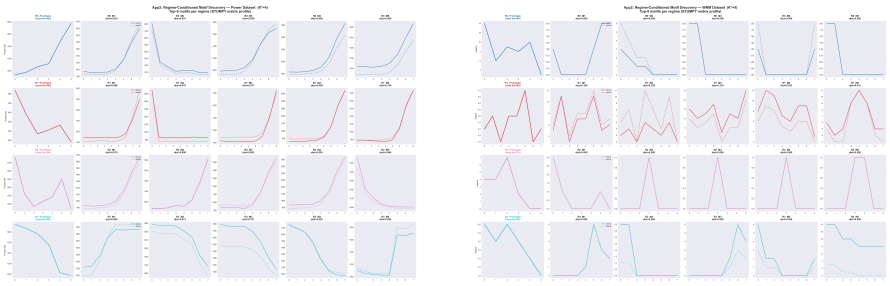
5.2 Regime-Conditioned Motif Discovery

Motif discovery asks whether recurring subsequences are present in a signal. Applied globally to a heterogeneous sensor stream, motif search can mix motifs from different operating conditions. Regime conditioning restricts the search to subseries belonging to the same regime, making motifs easier to interpret. We use matrix profile search [5, 6] within each regime subseries.

APP2 applies matrix profile within each regime-conditioned subseries for efficient fixed-length motif discovery; APP1 and the prototype similarity comparisons use DTW instead, since DTW handles variable-length segments and is robust to local time distortion.

The top-1 motif distances are low across nearly all regimes. Power regimes have distances between 0.035 and 0.079, indicating high-fidelity repeated shapes. WRM has near-zero distances for R0, R2, and R3, reflecting repeated baseline or near-idle usage patterns, while the high-usage regime R1 is more variable (0.306). S3 also shows coherent motifs, with distances below 0.04 for the non-constant phases and near zero for the standby phase. S1 and S2 are omitted from Table 3 because all their regimes yield near-zero top-1 distances, making a worst-case value uninformative; S3c is omitted because hierarchical motif distances were not separately tabulated at the flat-regime level.

Takeaway. Motif discovery succeeds because regime conditioning separates heterogeneous operating contexts before search; even in noisy signals such as WRM, baseline and near-idle regimes yield near-perfect repeated motifs.



(a) Power motifs (b) WRM motifs

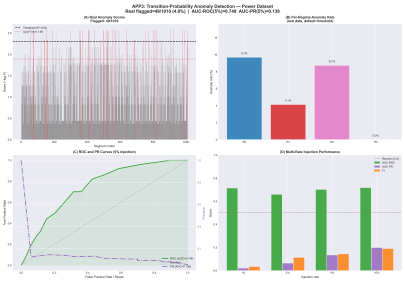
Fig. 5. APP2 regime-conditioned motif discovery. Matrix profile is applied inside each regime subseries. Power shows consistently low motif distances; WRM baseline and near-idle regimes contain almost exact repeated motifs.

5.3 Transition-Probability Anomaly Detection

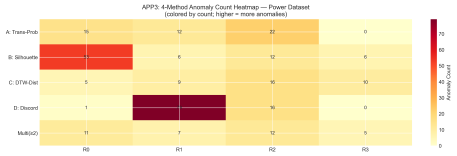
Anomaly detection uses the transition matrix and per-segment quality scores. A low-probability transition under the transition matrix $\mathbf{T}$ is anomalous because it violates the learned regime succession structure. We complement this transition signal with three additional indicators — silhouette score, prototype distance, and matrix-profile discord — and flag segments where at least two independent signals agree as high-confidence anomaly candidates.

Injected-label experiments show moderate to strong anomaly ranking. On synthetic datasets, the best transition-based area-under-the-ROC-curve (AUC) reaches 1.000. On real data, the best AUC is 0.748 for Power and 0.707 for WRM. On the original uninjected series, the multi-signal consensus produces 35 Power flags (3.5% of segments) and 8 WRM flags (0.7%), useful as compact review lists for operational inspection.

Takeaway. Structured transitions enable label-free anomaly scoring, while diffuse geometry, as in WRM, warns that silhouette-based thresholds must be recalibrated.



(a) Power anomaly flags



(b) Multi-method comparison

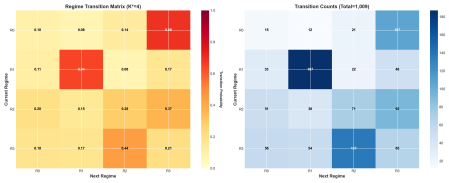
Fig. 6. APP3 anomaly detection on Power. Transition, silhouette, DTW-distance, and discord signals flag complementary unusual segments; multi-signal agreement gives a compact review list.

5.4 Markov Regime Forecasting

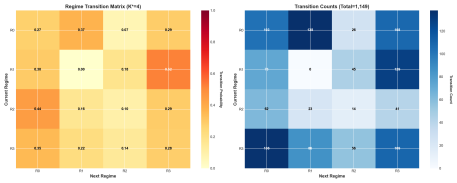
Forecasting evaluates whether the regime sequence contains temporal predictive structure. Given the current regime label, the target is to predict the next regime: we use first- and second-order Markov chains and compare their accuracy against majority-class, persistence, ARIMA, and MLP baselines. Markov2 conditions on the two most recent regime labels, capturing short sequential dependencies; the forecast target is always the immediate next regime ($h = 1$ step ahead), and improvement over the majority-class baseline is the primary metric because regime class distributions are typically unbalanced.

The transition matrix is the most direct representation of regime-sequence structure: each row corresponds to the current regime and each column to the next. Concentrated rows indicate predictable transitions; nearly uniform rows indicate weak temporal memory. The same matrix therefore supports two tasks: low-probability transitions are anomaly candidates, and high-probability transitions are forecasts.

Figure 7 shows transition matrices for the two real datasets. Power contains several concentrated rows, which is consistent with its stronger Markov forecasting gain. WRM is more diffuse and event-driven, which explains why its forecasting gain is smaller even though the same regime vocabulary remains useful for motif discovery and anomaly review.



(a) Power transition matrix



(b) WRM transition matrix

Fig. 7. Regime transition matrices for the two real datasets. Unlikely transitions support anomaly review, while high-probability transitions drive Markov forecasts. Power is more structured; WRM is more diffuse.

The best gains occur when the regime process has clear transition structure. S3 improves by 55.6 percentage points over majority-class prediction. Power improves by 24.4 points, with Markov2 reaching 57.2% accuracy and slightly exceeding the MLP baseline (56.3%). WRM improves by only 9.6 points because its transitions are more stochastic. S3c is an instructive edge case: persistence outperforms Markov variants because the sequence is near-cyclic, not simply first-order Markov.

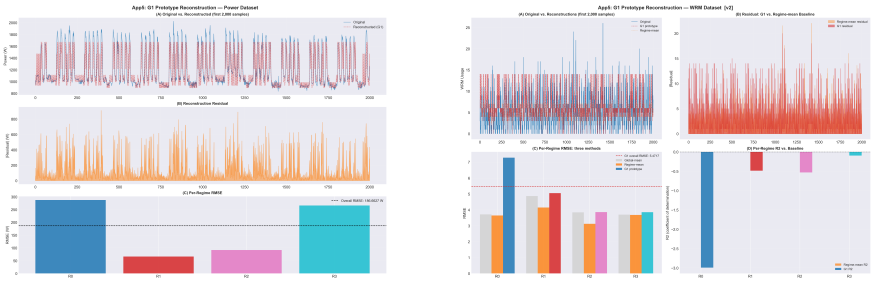
Takeaway. Forecasting benefits when transitions are directional; nearly memoryless or cyclic regime sequences require different baselines such as persistence or seasonal models.

5.5 Prototype and Regime-Mean Reconstruction

Reconstruction asks how much of the raw signal variance is explained by regime membership. Two strategies are compared: prototype-waveform substitution replaces each segment with its regime medoid, while regime-mean substitution sets each observation to the scalar mean of its assigned regime. The latter is simpler and better suited when regimes encode signal level rather than detailed waveform shape.

On synthetic datasets, prototype and regime-mean reconstruction both work well, with R^2 values around 0.80–0.92. On the S3c cyclic dataset, reconstruction is almost perfect. The real datasets separate sharply. Power has poor prototype reconstruction but strong regime-mean reconstruction: $R^2 = 0.718$ (coefficient of determination, where 1.0 = perfect fit), corresponding to a 46.9% RMSE (root mean squared error) reduction. WRM improves over the failed prototype reconstruction but remains modest, with $R^2 = 0.134$ and 6.95% RMSE reduction.

Takeaway. Reconstruction succeeds when regime means capture dominant signal variance; it is limited when within-regime stochasticity dominates, as in WRM.



(a) Power reconstruction (b) WRM reconstruction

Fig. 8. APP5 reconstruction contrast. Power is level-driven, so regime means explain substantial variance ($R^2 = 0.718$). WRM is event-driven, so regime means explain less raw signal variance ($R^2 = 0.134$).

6 Cross-Application Analysis

Table 3 synthesizes results across all five tasks and six datasets, each drawing from the same regime vocabulary. Three cross-cutting conclusions emerge, each corresponding to one of the paper's claims: regime artifacts are reusable across tasks rather than single-purpose clusterings; different tasks exploit different artifact types; and the two label-free structural indicators — the prototype discrimination ratio Δ and within-regime variance — jointly predict which dataset–task combinations benefit.

Table 3. Cross-application summary. *Acc.*: annotation accuracy (%); *APP2 motif dist.*: top-1 motif distance for the most variable regime (lower = stronger motif coherence; — = all regimes near-zero or not separately evaluated at flat-regime level for that dataset, see § 5.2); *AUC*: area under ROC curve; *Gain*: next-regime accuracy improvement over majority-class baseline (pp = percentage points); *APP5B* R^2: coefficient of determination for regime-mean reconstruction; Δ: prototype discrimination ratio ($\Delta > 1$ = good prototype separation).

Dataset	APP1 Acc.	APP2 motif dist.	APP3 AUC	APP4 Gain	APP5B R^2	Δ	Dominant reason
S1	100.0%	—	1.000	+22.2 pp	0.812	4.534	clean separation
S2	100.0%	—	1.000	0.0 pp	0.916	4.342	small forecast test set
S3	100.0%	<0.04	1.000	+55.6 pp	0.827	1.947	structured phase order
S3c	92.2%	—	0.775	0.0 pp	≈1.000	–	cyclic day types
Power	80.7%	0.079	0.748	+24.4 pp	0.718	1.759	level-driven regimes
WRM	49.1%	0.306	0.707	+9.6 pp	0.134	0.691	stochastic usage

6.1 When Regimes Help

Regimes help when vocabulary structure matches the downstream task. As formalised in Section 4, $\Delta > 1$ predicts annotation success; within-regime variance screens reconstruction; and regime-transition concentration screens both forecasting and anomaly detection — concentrated rows support Markov prediction and transition-probability anomaly scoring, while diffuse rows require multi-signal consensus.

6.2 When Regimes Do Not Help

Failures are informative. WRM annotation underperforms because prototypes overlap; WRM reconstruction is limited by stochastic within-regime usage; S3c forecasting is limited because persistence fits the cyclic sequence better than Markov transitions. The vocabulary therefore explains both success and limited utility.

This is the paper's central contribution: regime detection should not be evaluated only as a clustering endpoint, but as a pattern vocabulary whose downstream utility depends on measurable structural properties of the discovered regimes.

7 Limitations

The study has four main limitations. First, all downstream tasks are evaluated offline; no live operational deployment or user study is included. Second, real-world validation covers only two sensor domains (Power and WRM), so generalisation to other domains remains open. Third, the predictors are empirical: $\Delta > 1$ is a consistently observed threshold but has not been formally proved. Fourth, baselines are intentionally simple to isolate the value of the regime vocabulary, not to compete with specialised forecasting or anomaly-detection models. These limitations clarify scope without undermining the main finding: regime artifacts predict their own utility.

8 Conclusion

This paper studied downstream utility of unsupervised regime vocabularies in sensor time series, and confirmed three claims across five downstream tasks and six datasets. First, regime artifacts are reusable vocabularies: prototypes, labels, transitions, and quality scores each supported distinct downstream tasks without re-running the detector. Second, different tasks exploit different artifact types: motif discovery drew on regime-conditioned subseries, anomaly detection on transition probability, forecasting on regime-sequence structure, and reconstruction on regime-level means. Third, downstream utility is empirically predictable from two label-free structural indicators: the prototype discrimination ratio Δ reliably screens annotation success ($\Delta > 1$ across all high-accuracy datasets), and within-regime variance explains the sharp Power–WRM reconstruction contrast.

The broader message is that unsupervised regimes should be judged not only by whether they form stable clusters, but by whether they become useful patterns. A good regime vocabulary is therefore both a compact description of the past and a reusable tool for later analysis. This framing is especially relevant for complex sensor streams, where labels are scarce but repeated operational structure is abundant.

References

1. Truong, C., Oudre, L., Vayatis, N.: Selective review of offline change point detection methods. Signal Processing **167**, 107299 (2020)
2. Sakoe, H., Chiba, S.: Dynamic programming algorithm optimization for spoken word recognition. IEEE Transactions on Acoustics, Speech, and Signal Processing **26**(1), 43–49 (1978)
3. Hallac, D., Nystrup, P., Boyd, S.: Toeplitz inverse covariance-based clustering of multivariate time series data. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 215–223 (2017)
4. Paparrizos, J., Gravano, L.: k-Shape: Efficient and accurate clustering of time series. In: Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, pp. 1855–1870 (2015)

5. Yeh, C.-C.M., Zhu, Y., Ulanova, L., Begum, N., Ding, Y., Dau, H.A., Silva, D.F., Mueen, A., Keogh, E.: Matrix profile I: All pairs similarity joins for time series. In: 2016 IEEE 16th International Conference on Data Mining, pp. 1317–1322 (2016)
6. Law, S.M.: STUMPY: A powerful and scalable Python library for time series data mining. Journal of Open Source Software **4**(39), 1504 (2019)
7. Rousseeuw, P.J.: Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics **20**, 53–65 (1987)
8. Rabiner, L.R.: A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE **77**(2), 257–286 (1989)
9. Hamilton, J.D.: A new approach to the economic analysis of nonstationary time series and the business cycle. Econometrica **57**(2), 357–384 (1989)
10. Chandola, V., Banerjee, A., Kumar, V.: Anomaly detection: A survey. ACM Computing Surveys **41**(3), 1–58 (2009)
11. Ye, L., Keogh, E.: Time series shapelets: A new primitive for data mining. In: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 947–956 (2009)
12. Mueen, A., Keogh, E., Zhu, Q., Cash, S., Westover, B.: Exact discovery of time series motifs. In: SIAM International Conference on Data Mining, pp. 473–484 (2009)
13. Cai, Q., Hu, M., Lian, Z.: Decomposed prototype forecasting network for multivariate time series forecasting. In: ICEAAI, pp. 1227–1232 (2026). doi:10.1109/ICEAAI68945.2026.11442350
14. Keogh, E., Lin, J., Fu, A.: HOT SAX: Efficiently finding the most unusual time series subsequence. In: Proceedings of the 5th IEEE International Conference on Data Mining (ICDM 2005), pp. 226–233 (2005). Power demand dataset: `https://www.cs.ucr.edu/~eamonn/discords/power_data.txt`
15. Dau, H.A., Keogh, E., Kamgar, K., Yeh, C.-C.M., Zhu, Y., Gharghabi, S., Ratanamahatana, C.A., Chen, Y., Hu, B., Begum, N., Bagnall, A., Mueen, A., Batista, G.: The UCR time series classification archive (2019). `https://www.cs.ucr.edu/~eamonn/time_series_data_2018/`
16. Jorge, M.B., Rubén, C.: Time series clustering with random convolutional kernels. Data Mining and Knowledge Discovery **38**(4), 1862–1888 (2024)
17. Middlehurst, M., Ismail-Fawaz, A., Guillaume, A., et al.: aeon: A Python toolkit for learning from time series. Journal of Machine Learning Research **25**(289), 1–10 (2024)
18. Tao, J., Wu, Q., Zhao, L., Liang, C.: Adaptive clustering framework for time-series data using enhanced FastDTW. In: CCDC, pp. 5408–5412 (2025). doi:10.1109/CCDC65474.2025.11090860
19. Zhang, H., Li, J., Yao, Q.: RACER: Fast and accurate time series clustering with random convolutional kernels and ensemble methods. IEEE Internet Things J. **13**(9), 19430–19442 (2026). doi:10.1109/JIOT.2026.3662758