ITEM #220 - **Pattern Family Augmentation for Structural Indexing in Time-Series IR Systems – improving Recall While Preserving Structural Correctness**

Conversation：Time-Series IR Requirements

20260109

Authors: Sizhe Tan & GPT-Obot

DBM-COT ITEM #220

Pattern Family Augmentation for Structural Indexing in Time-Series IR Systems

Improving Recall While Preserving Structural Correctness

Abstract

In large-scale time-series intelligence systems, treating each known signal pattern as a single immutable template leads to fragile matching and poor recall. Real-world occurrences of the same structural pattern vary in speed, amplitude, phase, and noise characteristics. This ITEM introduces **Pattern Family Augmentation** as a mandatory component of **Structural Indexing (Step 5)** in the DBM Time-Series IR stack.

By generating a bounded set of **property-preserving IR variants** for each known pattern, structural indexing is upgraded from isolated template matching to robust **neighborhood search in metric space**. This approach significantly improves recall while maintaining interpretability, metric consistency, and runtime efficiency.

---

1. Motivation

Traditional time-series pattern matching systems suffer from a fundamental contradiction:

- Tight thresholds preserve precision but miss real signals.
- Loose thresholds increase recall but introduce false positives.

This contradiction arises because patterns are treated as **point objects** rather than **structural families**.

DBM resolves this by recognizing that:

A signal pattern is not a point in metric space, but a **local manifold of structurally equivalent IRs**.

Pattern Family Augmentation makes this manifold explicit and computable.

---

2. Pattern Family Definition

A **Pattern Family** is defined as a finite set of IR variants derived from a canonical pattern IR, subject to **property-preserving constraints**.

Formally:

```
PatternFamily(P) = {
    (IR_i, mutationSignature_i, mutationCost_i)
}
```

Where all `IR_i` preserve the essential structural properties of pattern `P`.

Preserved properties typically include:

- Event ordering and causality (Event → Event)
- Directional topology (Up / Down / Flat relations)
- Relative ladder / tier structure
- Core pattern constraints (pattern logic, not raw thresholds)

---

3. Property-Preserving Mutation Operators

Pattern Family Augmentation operates **at the IR level**, not at the raw time-series level. This ensures structural correctness and metric consistency.

Typical mutation operators include:

1. **Time-Warp Mutation**
   Varies event spacing while preserving order and topology.
2. **Amplitude Re-Quantization**
   Shifts events across adjacent ladder tiers without changing direction structure.
3. **Phase Jitter**
   Applies small temporal offsets to pattern boundaries.

4. **Noise Absorption / Decoration**
   Inserts or removes weak auxiliary events without breaking the main event chain.
5. **Substructure Optionality**
   Marks secondary substructures as optional, generating with/without variants.

Each mutation contributes a **mutation cost**, which can be incorporated into metric distance explanations.

---

4. Controlled Expansion and Budgeting

Unbounded mutation leads to combinatorial explosion. Therefore, Pattern Family generation must be governed by a **Mutation Policy**, enforcing:

- A maximum number of variants per pattern (e.g., K = 32–128)
- Coverage-oriented sampling in metric space
- Preservation of a minimal structural core
- Explicit mutation cost accounting

This ensures scalability and interpretability.

---

5. Integration with Structural Indexing

Pattern Family Augmentation is applied **before** building:

- Metric Differential Trees
- Two-Phases Search indices

Indexing strategy:

- **Index nodes store IR variants**
- **Leaf aggregation maps variants back to canonical pattern IDs**
- Query results are merged at the pattern-family level

This transforms indexing from pattern lookup into **family-aware neighborhood search**.

---

6. Runtime Benefits

Pattern Family Augmentation enables:

- Higher recall without relaxed thresholds
- More stable Differential Tree partitions
- Improved Two-Phases Search pruning

- Pattern-level confidence scoring based on multi-variant consistency
- Reduced false positives caused by single accidental matches

---

7. Impact on Online Decision and Learning Loop

With Pattern Families:

- Thresholding decisions are made over **family distance distributions**
- RHS statistics become more stable
- Pattern scores evolve based on family-level evidence
- Newly observed variants can be fed back to Offline pattern evolution

This turns pattern matching into a **living structural evidence system**.

---

8. Position in the DBM Time-Series Stack

Pattern Family Augmentation is a **mandatory enhancement** of:

Step 5: Structural Indexing

It bridges Offline structure discovery and Online robust recognition, ensuring that DBM Time-Series Intelligence remains both precise and resilient.

---

9. Summary Statement

Pattern Family Augmentation upgrades time-series pattern matching from fragile template comparison to robust structural neighborhood reasoning in metric space, preserving DBM's core principles of interpretability, consistency, and evolution.

DBM-COT ITEM #220（中文版）

时间序列 IR 结构索引中的模式族扩增

在保持结构正确性的前提下显著提升命中率

摘要

在大规模时间序列智能系统中，将每一个已知信号模式视为单一、不可变模板，会导致匹配脆弱、召回率低下。现实世界中，同一结构模式在速度、幅度、相位与噪声形态上天然存在变化。

本 ITEM 提出：**模式族扩增（Pattern Family Augmentation）** 应作为 DBM 时间序列 IR 应用栈中 **Step 5（Structural Indexing）** 的必要组成部分。

通过为每一个已知模式生成一组**保持结构性质不变的 IR 变体**，结构索引从“模板匹配”升级为**度量空间中的邻域搜索**，在不牺牲可解释性与一致性的前提下，大幅提升系统召回能力与稳定性。

---

1. 问题动机

传统时间序列系统面临一个根本矛盾：

- 阈值严格 → 精度高但漏信号
- 阈值放宽 → 召回高但假阳性泛滥

根源在于：
模式被错误地建模为“点”，而不是“结构族”。

DBM 的核心认知是：

一个信号模式不是度量空间中的单点，而是一个由结构等价 IR 构成的局部结构流形。

模式族扩增正是对这一结构事实的工程化表达。

---

2. 模式族定义

模式族（Pattern Family） 是由一个规范模式 IR 派生出的一组有限 IR 变体集合，所有变体均满足**性质保持约束**。

形式化表示为：

```
PatternFamily(P) = {
    (IR_i, mutationSignature_i, mutationCost_i)
}
```

所有 `IR_i` 必须保持模式 `P` 的核心结构性质。

---

3. 性质保持型 IR 变异操作

模式族扩增必须发生在 **IR 层**，而非原始时间序列层。

典型操作包括：

1. **时间扭曲（Time-Warp）**
   改变事件间距，不改变事件顺序与拓扑。
2. **幅度梯次重映射**
   在相邻 Ladder/Tier 之间移动事件。
3. **相位抖动**
   微调模式起止边界。
4. **噪声吸收 / 装饰事件**
   插入或移除弱事件而不破坏主事件链。
5. **子结构可选化**
   为次级结构生成有/无两类版本。

每一次变异都会产生**变异成本**，可直接参与度量距离解释。

---

4. 规模控制与预算机制

为避免组合爆炸，模式族生成必须受 **变异策略（Mutation Policy）** 控制：

- 每个模式限制生成 K 个变体（如 32–128）
- 优先覆盖度量空间邻域
- 保持最小结构核心
- 明确记录变异成本

---

5. 与结构索引的集成方式

模式族扩增在构建以下索引之前完成：

- 度量差分树（Differential Tree）
- 两阶段搜索引擎（Two-Phases Search）

工程原则：

- **索引节点存储 IR 变体**
- **叶子节点回归到 canonical pattern ID**
- **查询结果在模式族层面合并与解释**

---

6. 运行期收益

引入模式族后：

- 召回率显著提升而无需放宽阈值
- 差分树结构更稳定
- 两阶段搜索剪枝更有效
- 置信度来自“同族一致性”而非单点命中
- 假阳性明显减少

7. 对在线决策与学习闭环的影响

模式族使得：

- 阈值判断基于族级距离分布
- RHS 统计更稳定
- 模式评分随证据演化
- 新变体可回灌 Offline，推动模式进化

8. 在 DBM 时间序列体系中的定位

模式族扩增是 Step 5（Structural Indexing）的必要增强，而非可选优化。

它连接了 Offline 结构发现与 Online 稳健识别，是 DBM 时间序列智能具备工程可行性的关键环节。

9. 总结陈述

模式族扩增使时间序列识别从脆弱的模板匹配，跃迁为度量空间中的结构邻域推理，完整体现了 DBM 的结构智能哲学。