Goal: Useful, Useable, and Scalable FSM

- **Useful**: Develop general FSM algorithms that can be tailored to diverse applications, such as natural language processing, information extraction, web usage mining, market-basket analysis, and computational biology
- **Useable**: Flexible subsequence constraints allow applications to specify patterns of interest intuitively
- **Scalable**: Ability to deal with large input, search space, and output

**DESQ [1, 2]: Usefulness and Usability**

A unified FSM framework to specify flexible subsequence constraints in an intuitive, declarative way

1. **Hierarchies**
   - Allow for discovering non-trivial patterns
   - Example: DSLR → Tripod → Flash
2. **Pattern Expressions**
   - Specify subsequence constraints
   - Example: \((Book)\{0,2\}(Book)\{1,4\}
3. **FSTs**
   - Computational framework: translate input sequence to candidate subsequences
   - Example:

**Example**

<table>
<thead>
<tr>
<th>Sequence database: T</th>
<th>Item hierarchy:</th>
<th>Pattern expression:</th>
<th>Minimum support:</th>
<th>Output:</th>
</tr>
</thead>
<tbody>
<tr>
<td>T1: a,b,c</td>
<td>a,b</td>
<td>(?A{1}, ?B{2}.*)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>T2: abc</td>
<td>a,b,c</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T3: a,b,c</td>
<td>a,b,c</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T4: a,b,c</td>
<td>a,b,c</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>T5: a,b,c</td>
<td>a,b,c</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Our Contribution: Scalability**

1. A general framework for **distributed FSM** with flexible subsequence constraints
2. **Two algorithms** within this framework:
   - (a) D-CAND: Communicate compressed candidate subsequences
   - (b) D-SEQ: Communicate rewritten input sequences
3. **Large-scale** experimental study

**1. General Framework**

- We generalize existing item-based partitioning approaches (MG-FSM, LASH) to a general framework that supports flexible subsequence constraints
- Key questions: how to distribute computation and what to communicate

**2a. Communicate Candidates**

- Communicate candidate subsequences as compressed non-deterministic finite automata (NFA)
- Beneficial for selective constraints

**2b. Communicate Inputs**

- Communicate task-relevant items of input sequences
- Robust across a wide range of mining tasks

**3. Experimental Study**

- **Flexible Patterns**: Both algorithms outperformed naive approaches by up to 50x

\[ T1(0) \quad T2(10) \quad T3(100) \quad T4(1k) \quad T5(1k) \]

- **Traditional Patterns**: Both algorithms exhibited acceptable generalization overhead over existing, specialized methods

**References and Resources**


Code is open source and available at [https://github.com/rgemulla/desq/tree/distributed](https://github.com/rgemulla/desq/tree/distributed)