Scalable Frequent Sequence Mining With Flexible Subsequence Constraints

Alexander Renz-Wieland[1], Matthias Bertsch[2], Rainer Gemulla[2]

[1]Technische Universität Berlin, [2]Universität Mannheim

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

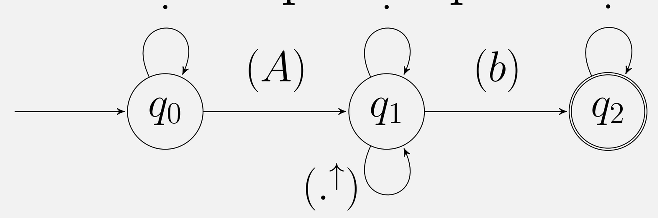
- Allow for discovering non-trivial patterns
- Example: *DSLR* → *Tripod* → *Flash*

(2) Pattern Expressions

- Specify subsequence constraints
- Example: $(Book)[.\{0,2\}(Book)]\{1,4\}$

(3) FSTs

- Computational framework: translate input sequence to candidate subsequences
  - Example:

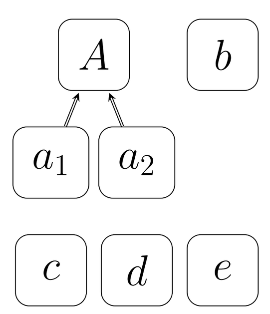


Example

Sequence database:

T_1: a_1cdcb
T_2: eea_1ea_1eb
T_3: $cdcb$
T_4: a_2db
T_5: a_1a_1b

Item hierarchy:



Pattern expression:
$.^*(A)[(.\uparrow).^*]^*(b).^*$

Minimum support:
$\sigma = 2$

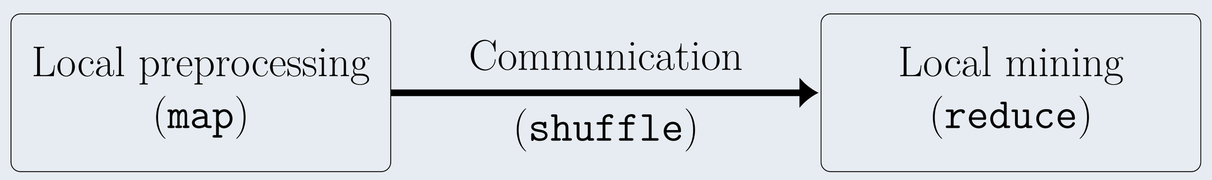
Output:
a_1b (3), a_1a_1b (2), a_1Ab (2)

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

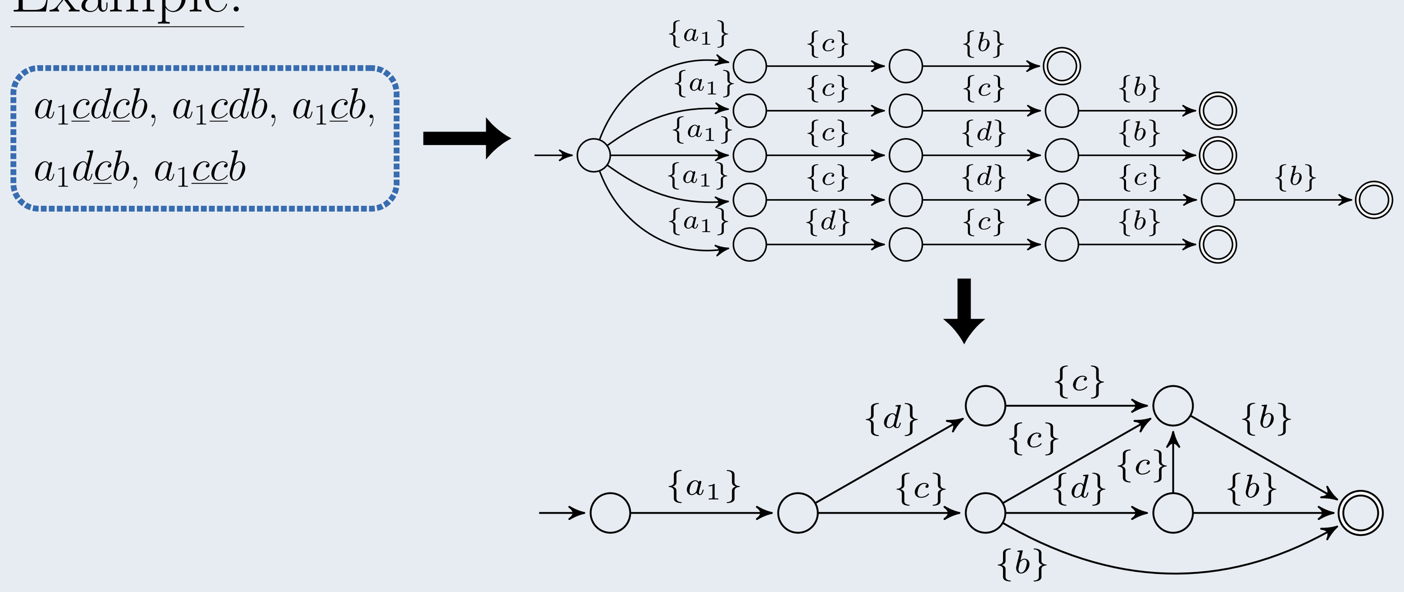
1. General Framework

- 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 nondeterministic finite automata (NFA)
- Beneficial for selective constraints
- Example:



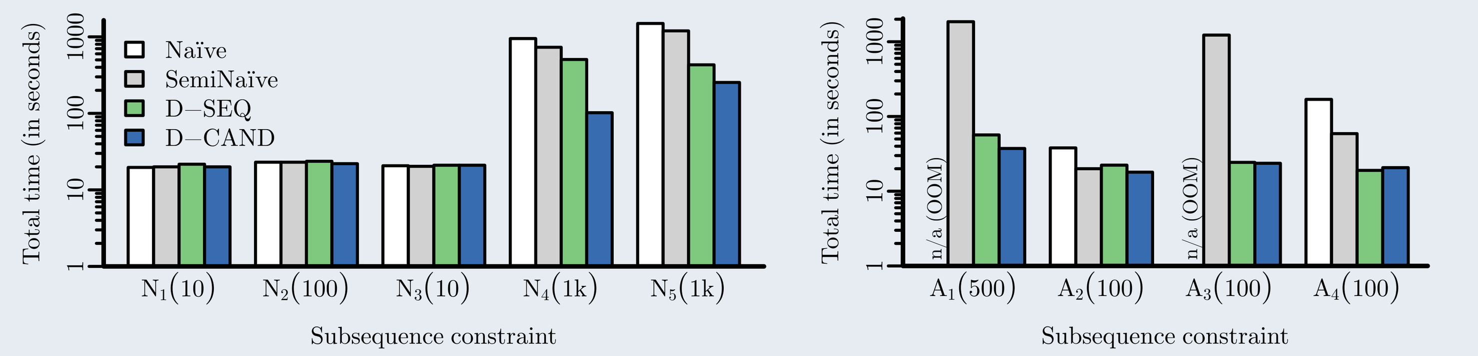
2b. Communicate Inputs

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

$ee\,a_1ea_1eb\,eeee$

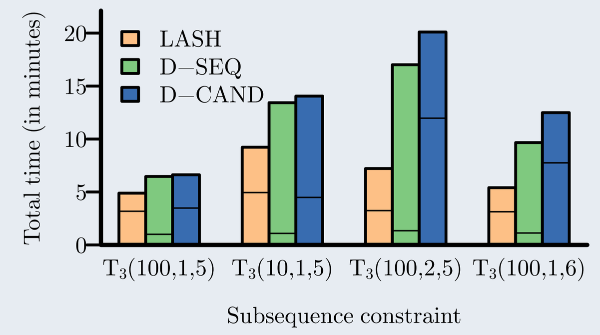
3. Experimental Study

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



(a) New York Times data (b) Amazon Review data

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



References and Resources

[1] K. Beedkar and R. Gemulla.
DESQ: Frequent sequence mining with subsequence constraints.
IEEE International Conference on Data Mining '16.

[2] K. Beedkar, R. Gemulla, and W. Martens.
A unified framework for frequent sequence mining with subsequence constraints.
To appear in ACM Transactions on Database Systems '19.

Code is **open source** and available at
`https://github.com/rgemulla/desq/tree/distributed`.