Condensed Representations for Contrast Sequential Pattern Mining in ASP

Gioacchino STERLICCHIO[1,2,4] **and Francesca A. LISI**[2,3]
[1]*DMMM* Polytechnic University of Bari
[2]*DIB* University of Bari "Aldo Moro"
[3]*CILA* University of Bari "Aldo Moro"
[4]*SnT*, University of Luxembourg

CILC 2024, Rome, June 26-28, 2024

Outline

- Preliminaries
  - Pattern Mining
  - Declarative Pattern Mining
  - Contrast Sequential Pattern Mining
- Answer Set Programming
  - Data encoding
  - Problem encoding
- Evaluation
  - Goal
  - Datasets
  - Results
- Final remarks

Pattern Mining in brief

- Class of data mining tasks aimed at the *discovery of interesting regularities* in dataset
  - e.g., itemset mining, sequential mining, graph mining
- The interestingness measure of a pattern is, in most of the algorithms, the number of its occurrences (*frequency/support*) in the dataset
- Frequent pattern mining: given a *threshold k*, interesting patterns are those that *occur at least in k data instances*
- Our work: *contrast sequential pattern mining*

What is DPM

Declarative Pattern Mining (DPM) is a stream of research aims at encoding pattern tasks in a declarative framework:

- (Guyet et al., 2014) explore a first attempt to solve sequential pattern mining in ASP
- (Gebser et al., 2016) use ASP for extracting condensed representation of sequential patterns
- (Samet et al., 2017) show a method for mining meaningful rare sequential patterns with ASP
- (Guyet et al., 2018) analyse ASP efficiency for sequential pattern mining and compare with constraint programming (CP)
- (Paramonov et al., 2019) combine dedicated pattern mining algorithms and ASP
- (Besnard and Guyet, 2020) address the task of mining negative sequential patterns in ASP
- (Lisi and Sterlicchio, 2022a 2022b) adapt the Guyet's ASP encodings for SPM to address the requirements of an application in the digital forensics domain
- (Lisi and Sterlicchio, 2022) show MASS-CP, an ASP-based approach to contrast pattern mining
- (Lisi and Sterlicchio, 2023) show MASS-CSP, an ASP-based approach to contrast sequential pattern mining

Sequential and Contrast Pattern Mining

- *Sequential pattern Mining*: finding statistically relevant patterns between data examples where the values are delivered in a sequence (Mooney and Roddick, 2013)
  - Values are discrete within a time series

- *Contrast Pattern Mining*: detecting statistically significant differences (contrast) between two or more disjoint sets of transactions (Dong and Bailey, 2012)
  - Class labels are introduced to partition the dataset
  - Halfway between characterization and discrimination

Contrast Sequential Pattern Mining (CSPM)

A contrast sequential pattern is defined as a sequential pattern that occurs frequently in one sequences dataset but not in the others (Chen et al., 2022)

Given two sequences dataset D_1 labelled with class C_1 and D_2 with class C_2:

- the *growth rate* from D_1 to D_2 of a sequential pattern s is $GR_{c_1}(s) = \frac{supp(s,D_1)/|D_1|}{supp(s,D_2)/|D_2|}$
- the *growth rate* from D_2 to D_1 of a sequential pattern s is $GR_{c_2}(s) = \frac{supp(s,D_2)/|D_2|}{supp(s,D_1)/|D_1|}$
- the *contrast rate* of s is $CR(s) = \max\{GR_{c_1}, GR_{c_2}\}$

A sequence s is said to be a *contrast sequential pattern* if $CR(s) \geq mincr$

- $mincr$ is the minimum contrast rate threshold

CSPM > example

ID	Sequence	Class
1	<*a b* a *c* d>	C_1
2	<*a b c*>	C_1
3	<c *a b c*>	C_1
4	<c>	C_1
5	<b c a a>	C_2
6	<c b a>	C_2
7	<c b a>	C_2
8	<*a b* a *c* b a>	C_2

given $s =< a\ b\ c >$ and $minsup = 2$

- $cover(s) = \{1, 2, 3, 8\}$
- $supp(s) = 4$
- *frequent sequential pattern*

given $mincr = 2$

- $supp(s, C_1) = 3$
- $supp(s, C_2) = 1$
- $GR_{c_1}(s) = 3$
- $GR_{c_2}(s) = 0.33$
- $CR(s) = 3$
- *contrast sequential pattern* for C_1

Condensed Representations of CSPM

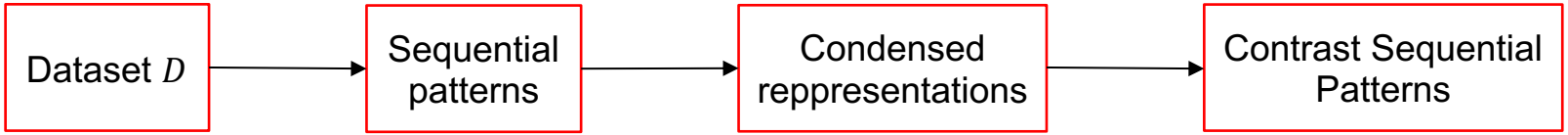
- Problem: large number of patterns
  - redundant information
  - overlapping information
  - difficulty in interpretation
  - inefficiency in storing and processing

- *Condensed representations* summarize patterns w.r.t a dataset D into a compact form without losing important insights
  - s is *maximal* if there are no other patterns t s.t. $s \subseteq t \; and \; supp(s, D) \geq minsupp$
  - s is *closed* if no other pattern t exist s.t. $s \subseteq t \; and \; supp(s, D) = supp(t, D)$

Condensed Representations of CSPM

- Problem: large number of patterns
  - redundant information
  - overlapping information
  - difficulty in interpretation
  - inefficiency in storing and processing

- *Condensed representations* summarize patterns w.r.t a dataset D into a compact form without losing important insights
  - s is *maximal* if there are no other patterns t s.t. $s \subseteq t \; and \; supp(s, D) \geq minsupp$
  - s is *closed* if no other pattern t exist s.t. $s \subseteq t \; and \; supp(s, D) = supp(t, D)$



ASP in a nutshell

- Logic programming paradigm under *answer set* (or *"stable model"*) semantics (Brewka et al., 2011)
- Highly declarative and expressive programming language, oriented towards difficult search problems
- Powerful tool for solving problems in various domains such as planning, scheduling, reasoning about actions, and knowledge representation
- In ASP, search problems are reduced to computing answer sets, and an ASP solver (i.e., a program for generating stable models) is used to find solutions

ASP data encoding

ID	Sequence	Class
1	<a b a c d>	C_1
2	<a b c>	C_1
3	<c a b c>	C_1
4	<c>	C_1
5	<b c a a>	C_2
6	<c b a>	C_2
7	<c b a>	C_2
8	<a b a c b a>	C_2

```
cl(1,c1). seq(1,1,a). seq(1,2,b). seq(1,3,a). seq(1,4,c). seq(1,4,d).
cl(2,c1). seq(2,1,a). seq(2,2,b). seq(2,3,c).
cl(3,c1). seq(3,1,c). seq(3,2,a). seq(3,3,b). seq(3,4,c).
cl(4,c1). seq(4,1,c).

cl(5,c2). seq(5,1,b). seq(5,2,c). seq(5,3,a). seq(5,4,a).
cl(6,c2). seq(6,1,c). seq(6,2,b). seq(6,3,a).
cl(7,c2). seq(7,1,c). seq(7,2,b). seq(7,3,a).
cl(8,c2). seq(8,1,a). seq(8,2,b). seq(8,3,a). seq(8,4,c). seq(8,5,b).
seq(8,6,a).
```

ASP problem encoding

From sequences to sequential patterns (Guyet et al., 2018)

```
item(I) :- seq(_, _, I).

patpos(1).
{ patpos(X+1) } :- patpos(X), X < maxlen.
patlen(L) :- patpos(L), not patpos(L+1).
1 {pat(X, I): item(I)} 1 :- patpos(X).

:- patlen(L), L < minlen

occ(T, 1, P) :- seq(T, P, I), pat(1, I).
occ(T, L, P) :- occ(T, L,   P-1), seq(T, P, _).
occ(T, L, P) :- occ(T, L-1, P-1), seq(T, P, C), pat(L, C).

seqlen(T, L) :- seq(T, L, _), not seq(T, L+1, _).
support(T) :- occ(T, L, LS), patlen(L), seqlen(T, LS).
:- { support(T) } < th.
```

sequential pattern generation

minlen constraint

pattern embedding

minsup constraint

ASP problem encoding

Keep only closed or maximal sequential patterns

```
% maximal
:- item(I), X = 1..maxlen+1, {ins(T,X,I) : support(T)} >= th.
```

```
% closed
:- item(I), X = 1..maxlen+1, {ins(T,X,I)} >= th, ins(T,X,I) : support(T).
```

ASP problem encoding

From condensed sequential patterns to condensed contrast sequential patterns

```
card(Card, c1) :- Card = #count{T : cl(T, c1)}.
card(Card, c2) :- Card = #count{T : cl(T, c2)}.
```

dataset cardinality

```
sup(Sup, c1) :- Sup = #count{T : support(T), seq(T, _, _), cl(T, c1)}.
sup(Sup, c2) :- Sup = #count{T : support(T), seq(T, _, _), cl(T, c2)}.
```

class support

```
gr(inf, c1) :- sup(Sup1, c1), Sup1 != 0, sup(0, c2).
gr(inf, c2) :- sup(Sup2, c2), Sup2 != 0, sup(0, c1).
gr(@gr(Sup1, Card1, Sup2, Card2), c1) :- sup(Sup1, c1), card(Card1, c1),
            sup(Sup2, c2), card(Card2, c2), Sup1 > 0, Sup2 > 0.
gr(@gr(Sup2, Card2, Sup1, Card1), c2) :- sup(Sup1, c1), card(Card1, c1),
            sup(Sup2, c2), card(Card2, c2), Sup1 > 0, Sup2 > 0.
```

growth rate

```
contr_pat(yes, Class) :- growth_rate(inf, Class).
contr_pat(@csp(Cr, mincr), Class) :- growth_rate(Cr, Class), Cr != inf.
:- contr_pat(no, c1), contr_pat(no, c2).
```

contrast rate

Evaluation: goal

- Scalability tests to assess time and memory of condensed representations
- Comparison against MASS-CSP (Lisi and Sterlicchio, 2023)

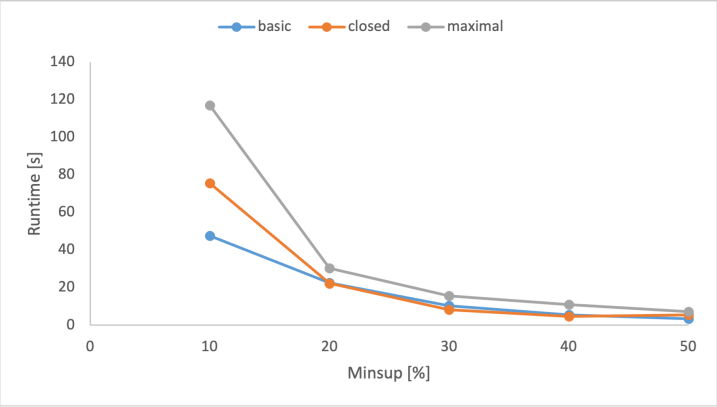
Evaluation: datasets

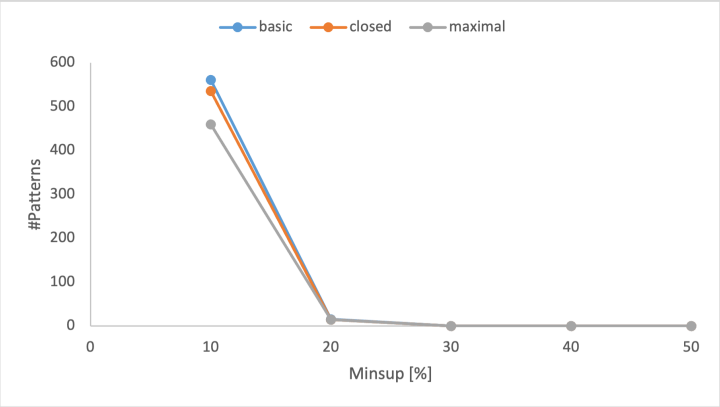
Dataset	$\|\Sigma\|$	$\|D\|$	$\|\|D\|\|$	max$\|T\|$	avg$\|T\|$	density
iPRG	21	8628	111,743	12	11.95	0.62
iPRG_25_25	20	50	657	12	11.88	0.64
iPRG_100_100	20	200	2591	12	11.83	0.64
iPRG_500_500	21	1000	12,933	12	11.92	0.62
iPRG_1000_1000	21	2000	25,841	12	11.91	0.61
UNIX	2672	9099	165,748	1256	18.22	0.01
UNIX_25_25	70	50	365	55	7.3	0.10
UNIX_100_100	178	200	2281	175	11.41	0.06
UNIX_500_500	420	1000	13,289	187	13.29	0.03
UNIX_755_755	540	1510	20,234	214	13.4	0.02

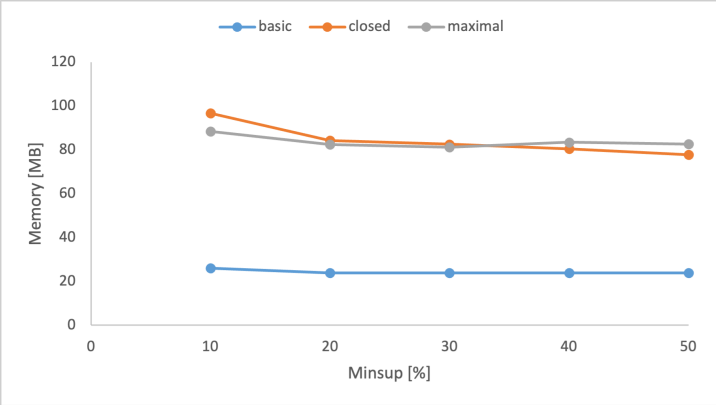
$$density = \frac{||D||}{|\Sigma||D|}$$

Evaluation: results

- Clingo as ASP platform
- Ubuntu 20.04.4
- AMD Ryzen 5 3500U @ 2.10 GHz, 8GB RAM







Final Remarks

- CSPM explores sequential data and discovers meaningful patterns by highlighting differences between groups of sequences
- Domains of application: market analysis, fraud detection, …
- The versatility of ASP:
  - addition/deletion of constraints allows the modelling of problem variants
  - development effort is lower
- Combinatorial explosion is involved in pattern mining and condensed representations can be a solutions
  - decreasing of number of patterns
  - increasing of computational resources
- What's next?
  - How can a hybrid approach improve performance?
  - How improving performance by optimizing embedding lookups?

References

- Guyet, T., Moinard, Y., Quiniou, R.: Using answer set programming for pattern mining. arXiv preprint arXiv:1409.7777 (2014)
- Gebser, M., Guyet, T., Quiniou, R., Romero, J., Schaub, T.: Knowledge-based sequence mining with asp. In: 25th International Joint Conference on Artificial Intelligence, IJCAI 2016, p. 8. AAAI (2016)
- Samet, A., Guyet, T., Negrevergne, B.: Mining rare sequential patterns with ASP. In: ILP 2017–27th International Conference on Inductive Logic Programming (2017)
- Besnard, P., Guyet, T.: Declarative mining of negative sequential patterns. In: DPSW 2020–1st Declarative Problem Solving Workshop, pp. 1–8 (2020)
- Lisi, F.A., Sterlicchio, G.: Declarative pattern mining in digital forensics: preliminary results. In: Calegari, R., Ciatto, G., Omicini, A. (eds.) Proceedings of the 37th Italian Conference on Computational Logic, Bologna, Italy, 29 June–1 July 2022. CEURWorkshop Proceedings, vol. 3204, pp. 232–246. CEUR-WS.org (2022a). http://ceur-ws.org/Vol-3204/paper 23.pdf
- Lisi, F.A., Sterlicchio, G.: Mining sequences in phone recordings with answer set programming. In: Bruno, P., Calimeri, F., Cauteruccio, F., Maratea, M., Terracina, G., Vallati, M. (eds.) Joint Proceedings of the 1st International Workshop on HYbrid Models for Coupling Deductive and Inductive ReAsoning (HYDRA 2022) and the 29th RCRA Workshop on Experimental Evaluation of Algorithms for Solving Problems with Combinatorial Explosion (RCRA 2022) co-located with the 16th International Conference on Logic Programming and Non-monotonic Reasoning (LPNMR 2022), Genova Nervi, Italy, 5 September 2022. CEUR Workshop Proceedings, vol. 3281, pp. 34–50. CEUR-WS.org (2022b). http://ceur-ws.org/Vol-3281/paper4.pdf
- Lisi, F.A., Sterlicchio, G.: A declarative approach to contrast pattern mining. In: Dovier, A., Montanari, A., Orlandini, A. (eds.) AIxIA 2022. LNCS, vol. 13796, pp. 17–30. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-27181-6 2
- F. A. Lisi, G. Sterlicchio, Mining contrast sequential patterns with ASP, in: R. Basili, D. Lembo, C. Limongelli, A. Orlandini (Eds.), AIxIA 2023 - Advances in Artificial Intelligence - XXIInd International Conference of the Italian Association for Artificial Intelligence, AIxIA 2023, Rome, Italy, November 6-9, 2023, Proceedings, volume 14318 of Lecture Notes in Computer Science, Springer, 2023, pp. 44–57. URL: https://doi.org/10.1007/978-3-031-47546-7_4. doi:10.1007/978-3-031-47546-7_4.

References

- Paramonov, S., Stepanova, D., Miettinen, P.: Hybrid ASP-based approach to pattern mining. Theory Pract. Log. Program. 19(4), 505–535 (2019). https://doi.org/10.1017/S1471068418000467
- Guyet, T., Moinard, Y., Quiniou, R., & Schaub, T. (2018). Efficiency analysis of ASP encodings for sequential pattern mining tasks. Advances in Knowledge Discovery and Management: Volume 7, 41-81.
- Han, J., Pei, J., Mortazavi-Asl, B., Pinto, H., Chen, Q., Dayal, U., Hsu, M.: PrefixSpan: Mining sequential patterns efficiently by prefix-projected pattern growth. In: proceedings of the 17th international conference on data engineering. pp. 215–224. IEEE (2001)
- Brewka, G., Eiter, T., and Truszczynski, M. (2011). Answer set programming at a glance. Communications of the ACM, 54(12):92-103.
- Mooney, C., Roddick, J.F.: Sequential pattern mining - approaches and algorithms. ACM Comput. Surv. 45(2), 19:1–19:39 (2013).
- Dong, G. and Bailey, J. (2012). Contrast data mining: concepts, algorithms, and applications. CRC Press.
- Chen, Y., Gan, W., Wu, Y., Yu, P.S.: Contrast pattern mining: A survey. arXiv preprint arXiv:2209.13556 (2022)
- Martin Gebser, Roland Kaminski, Benjamin Kaufmann and Torsten Schaub, Multi-shot ASP solving with clingo, TPLP, 19(1), 27–82, 2019

This work was partially supported by the project FAIR - Future AI Research (PE00000013), under the NRRP MUR program funded by the NextGenerationEU.

Thanks!

Do you have any questions?

g.sterlicchio@phd.poliba.it