System Evolution Analytics based on Data Science

Dr. Animesh Chaturvedi (Assistant Professor: IIIT Dharwad)

Young Researcher: Pingala Interaction in Computing (PIC) and Heidelberg Laureate Forum (HLF)

Postdoc: KCL & Alan Turing Institute

PhD: IIT Indore MTech: IIITDM Jabalpur

Pingala Interactions In Computing

HEIDELBERG LAUREATE FORUM

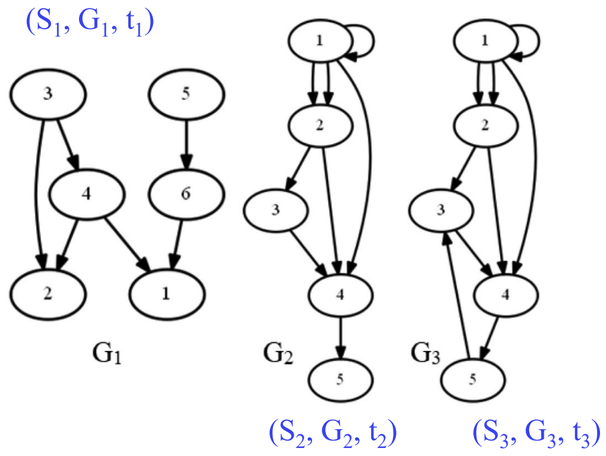
PDPM Indian Institute of Information Technology, Design and Manufacturing, Jabalpur

IIIT Dharwad ज्ञानेन विकासः — INDIAN INSTITUTE OF INFORMATION TECHNOLOGY

KING'S College LONDON — The Alan Turing Institute

Indian Institute of Technology Indore भारतीय प्रौद्योगिकी संस्थान इंदौर

State Series (SS)



Example of State series:
- software version series
- document versions

Challenges
- pre-processing requires a system domain expert
- processing multiple states separately with one semantic

We define (S_i, G_i, t_i) for an evolving system to make a **State Series (SS)** = $\{S_1, S_2 \dots S_N\}$, such that each state is represented as an Evolving Graph (network) $\{G_1, G_2 \dots G_N\}$, at various time points $\{t_1, t_2 \dots t_N\}$.

System Evolution Analytics

Domains	Evolving Systems	Networks (Graph)	System Evolution Analytics
Evolving Software System	Hadoop HDFS	Call graph	Software Evolution Analytics
Evolving Natural-language systems	Bible Translation	Words networks	Natural-language Evolution Analytics
	Multi-sport Events		
Evolving Retail Market System	Market Basket	Purchase network	Market Evolution Analytics
Evolving IMDb movie genre systems	Positive sentiment	Sentiment networks	Movie Evolution Analytics
	Negative sentiment		

System Network Evolution Analytics

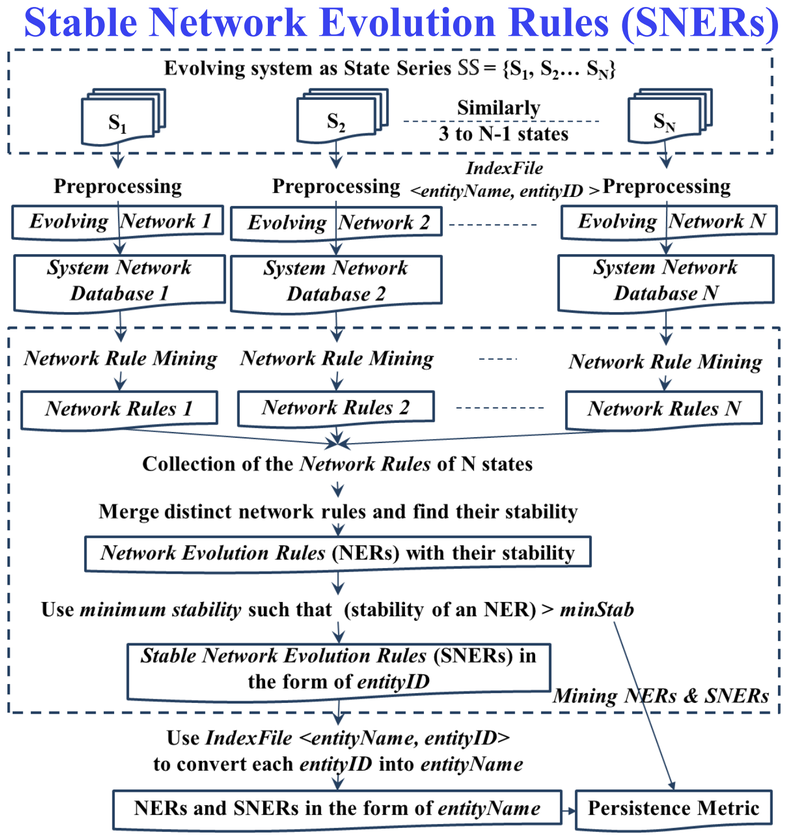
Stable Rules using threshold:
- minimum Stability (minStab)

Network Evolution Subgraph Mining
- Network Evolution Graphlets and Network Evolution Motifs

System Evolution Metrics
- Changeability and Stability of an Evolving System,
- System State Complexity of one state,
- Evolving System Complexity of a state series.

System Structure Learning
- to make System Neural Network (SysNN)
- to train machine about System structure and evolution
- to do Network Reconstruction
- to do System Evolution Recommendation about new versions

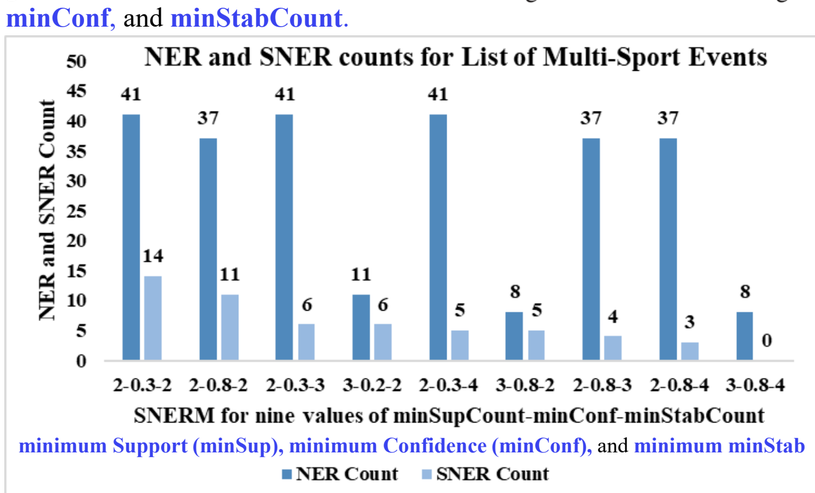
Stable Network Evolution Rules (SNERs)



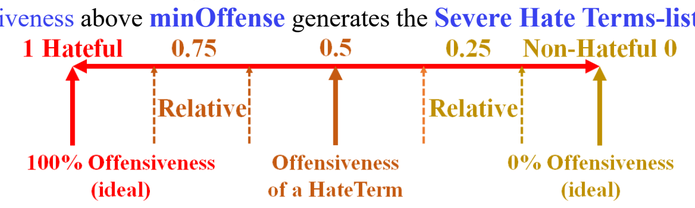
Search space of **Network Rule Mining (NRM)** is **System Network Database (SysNetDb)** of a state that contains connection pairs (CP). Interesting SNERs are generated with **minSupCount, minConf,** and **minStabCount.**

Hadoop-HDFS[1]

create ==> convert
readAll ==> readFully
checkAccess ==> getDelegationToken
close ==> getRemoteAddressString



Offensiveness: Percentage contribution of Hate Terms towards Hate Speech. Offensiveness above **minOffense** generates the **Severe Hate Terms-lists**.

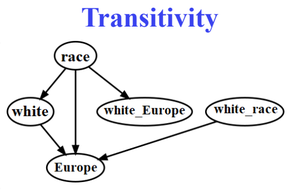


Stable Hate Rule mining to discover co-occurring Hate Terms

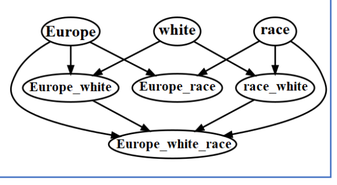
Concept, Rule

Europe race white 5
white → Europe
race → white Europe
white race → Europe
race → white
race → Europe

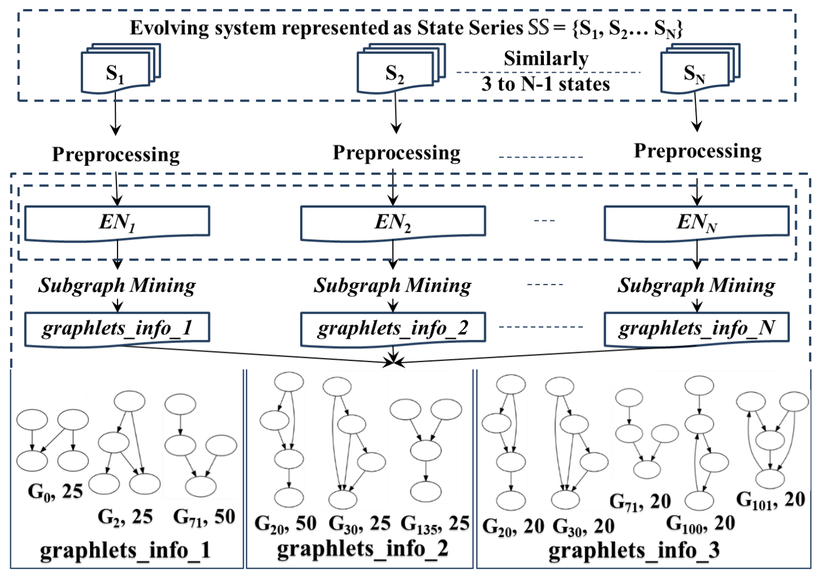
Transitivity



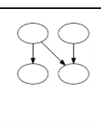
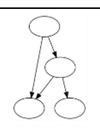
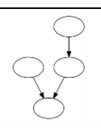
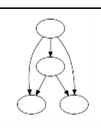
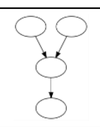
Lattice



Network Evolution Subgraphs and Complexity



Network Evolution Subgraphs and their complexities

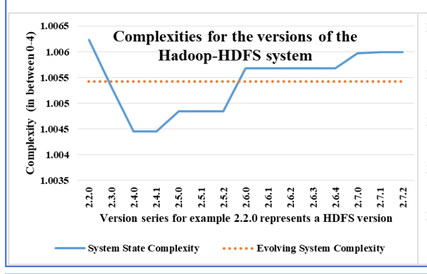
$M_0, C_0 = 1$	$M_2, C_2 = 2$	$M_{71}, C_{71} = 1$	$M_{185}, C_{185}=3$	$M_{135}, C_{135} = 1$
				

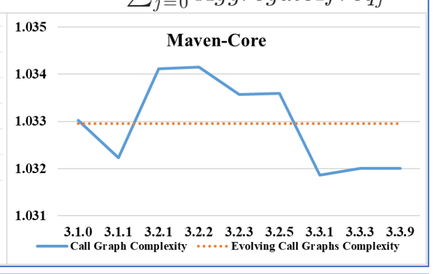
For each state, **System State Complexity**

For a state series, **Evolving System Complexity**

$$\text{SSC of } S_i\ (\text{SSC}_i) = \frac{\sum_{j=0}^{m} (freq_{ji} \times C_j)}{\sum_{j=0}^{m} freq_{ji}}$$

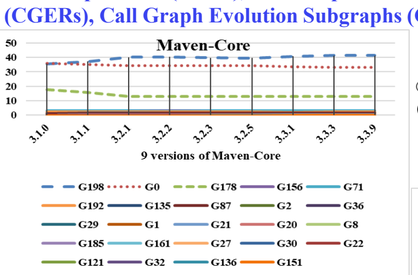
$$\text{ESC} = \frac{\sum_{j=0}^{m'} (Aggregate\_freq_j \times C_j)}{\sum_{j=0}^{m'} Aggregate\_freq_j}$$

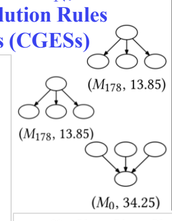




Evolving Call Graphs $\{CG_1, CG_2, \dots CG_N\}$ as a Software **Version Series** = $\{V_1, V_2, \dots V_N\}$

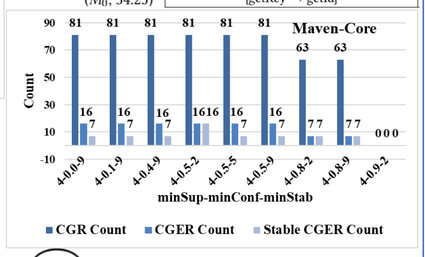
Call Graph Rules (CGRs), Call Graph Evolution Rules (CGERs), Call Graph Evolution Subgraphs (CGESs)

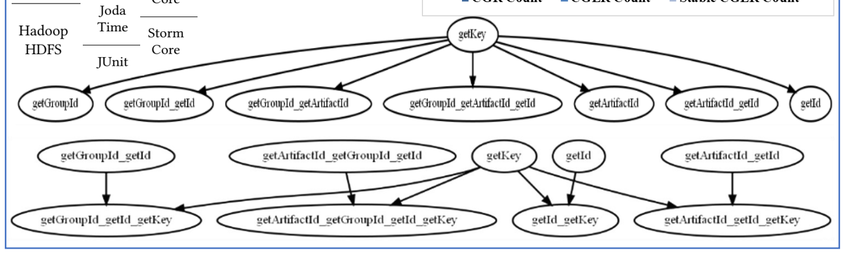


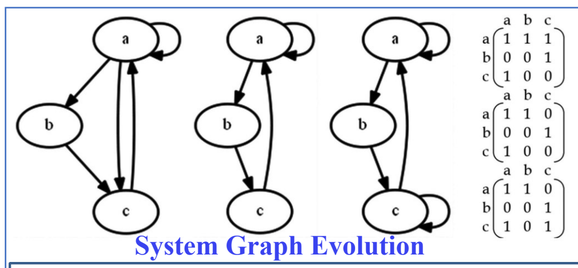


Stable CGERs
{getGroupId → getArtifactId}
{getGroupId → getArtifactId, getId}
{getArtifactId → getGroupId}
{getArtifactId → getGroupId, getId}
{getGroupId → getId}
{getId → getGroupId}
{getId → getGroupId, getArtifactId}
{getKey → getGroupId}
{getKey → getGroupId, getId}
{getKey → getGroupId, getArtifactId}
{getKey → getGroupId, getArtifactId, getId}
{getArtifactId → getId}
{getId → getArtifactId}
{getKey → getArtifactId}
{getKey → getArtifactId, getId}
{getKey → getId}

Evolving software		CGRs	CGERs	Stable CGERs
Commons Codec	HTTP Client	Log4J		
Guava	JMeter Core	Maven Core		
Hadoop HDFS	Joda Time	Storm Core		
	JUnit			







System Graph Evolution

System Structure Learning

to make

System Neural Network (SysNN)

$$\text{accuracy} = \frac{\left(\#\ \text{correctly recommended connections} + \text{correctly recommended no connections}\right)}{\#\text{all possible entity connections i.e. matrix size}}$$

$$\text{precision} = \frac{\#\text{correctly recommended connections}}{\#\text{all connections recommended by the tool}}$$

$$\text{recall} = \frac{\#\text{correctly recommended connections}}{\left(\#\ \text{correctly recomended connections + incorrectly recommended connections as no connections}\right)}$$

System Structure Learning (SSL) that uses **Deep Evolution Learner** (DEL)

Evolution Representor

Evolving Design Structure Matrix (EDSM), where $vector_i$ represents i^{th} state

$vector_1$ $[\{a_{11}.. a_{jk} \dots a_{1m}\}, \{a_{21}.. a_{jk} \dots a_{2m}\} \dots \{a_{m1}.. a_{jk}\dots a_{mm}\}]$

$vector_2$ $[\{b_{11}.. b_{jk} \dots b_{1m}\}, (b_{21}.. b_{jk} \dots b_{2m}) \dots (b_{m1},.. b_{jk}\dots b_{mm})]$

Evolving Design Structure Matrix (EDSM) or Evolving Connection Matrix (ECM) **DEL**

$vector_N$ $[(x_{11}.. x_{jk} \dots x_{1m}), (x_{21}.. x_{jk} \dots x_{2m}) \dots (x_{m1}.. x_{jk}\dots x_{mm})]$

0 0 .... 0 — Conversion — ***zero_vector*** (m × m)

Deep System Neural Network (Deep SysNN)

y_{11} y_{12} .... y_{mm} — ***output_vector*** (m × m)

Conversion

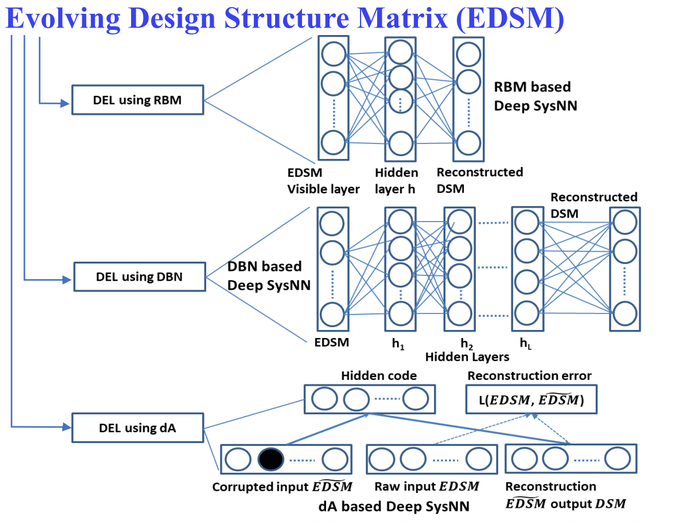
Symbol	Description
SS	$(vector_1, vector_2 \dots vector_N)$
ER	EDSM = f(SS)
a_{jk} b_{jk} x_{jk}	Elements of EDSM
input	*zero_vector*
g(ER)	$H_1, H_2, \dots H_L$
Y	*output_vector* (m × m)
M_{NO}	binary output matrix (m × m)

Zero Matrix of size (m × m)

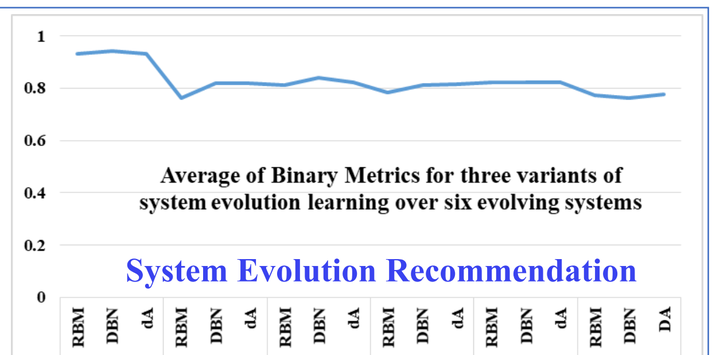
Output Matrix M_O (m × m)

Normalization

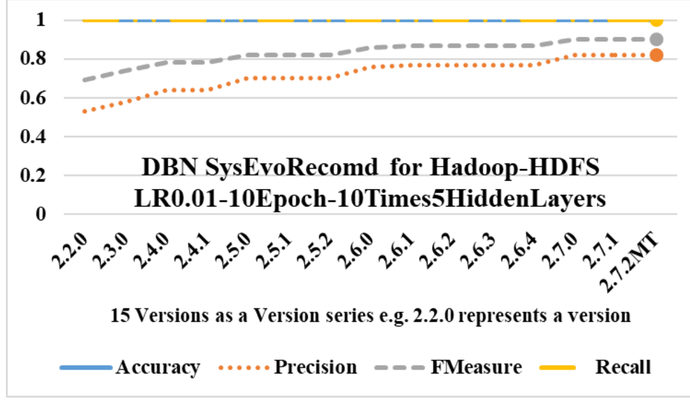
Evolving Design Structure Matrix (EDSM)



System Neural Network (SysNN)



System Evolution Recommendation



1. **Animesh Chaturvedi**, Aruna Tiwari, and Nicolas Spyratos. "minStab: Stable Network Evolution Rule Mining for System Changeability Analysis". *IEEE Transactions on Emerging Topics in Computational Intelligence*, Vol 5.2: 274-283 (April 2021). **DOI:** 10.1109/TETCI.2019.2892734.
2. **Animesh Chaturvedi**, Aruna Tiwari, and Nicolas Spyratos. "System Network Analytics: Evolution and Stable Rules of a State Series". *9th IEEE International Conference on Data Science and Advanced Analytics* (DSAA), 13-16 Oct, 2022, Shenzhen, China. IEEE, 2022. **DOI:** 10.1109/DSAA54385.2022.10032382. (Core A)
3. **Animesh Chaturvedi**, and Aruna Tiwari. "System network complexity: Network evolution subgraphs of system state series." *IEEE Transactions on Emerging Topics in Computational Intelligence,* Vol 4.2 (April 2020): 130-139. ISSN: 2471-285X. **DOI:** 10.1109/TETCI.2018.2848293.
4. **Animesh Chaturvedi**, and Aruna Tiwari. "System Evolution Analytics: Evolution and Change Pattern Mining of Inter-Connected Entities". 48th *IEEE International Conference on Systems, Man, and Cybernetics* (SMC), Miyazaki Japan, 7-10 October 2018, pp. 3877-3882. IEEE SMC Society **DOI:** 10.1109/SMC.2018.00750.
5. **Animesh Chaturvedi**, Aruna Tiwari, and Shubhangi Chaturvedi. "SysEvoRecomd: Network Reconstruction by Graph Evolution and Change Learning", *IEEE Systems Journal*, Vol. 14.3, pp. 4007 - 4014, Sept. 2020. ISSN: 1937-9234. **DOI:** 10.1109/JSYST.2020.2988037.
6. **Animesh Chaturvedi**, and Aruna Tiwari. "SysEvoRecomd: Graph Evolution and Change Learning based System Evolution Recommender". 18th *IEEE International Conference on Data Mining Workshops* (ICDMW). Singapore, 17-20 Nov 2018, pp. 1499-1500. IEEE Computer Society **DOI:** 10.1109/ICDMW.2018.00217.
7. **Animesh Chaturvedi**, and Aruna Tiwari. "System Evolution Analytics: Deep Evolution and Change Learning of Inter-Connected Entities". 48th *IEEE International Conference on Systems, Man, and Cybernetics* (SMC), Miyazaki Japan, 7-10 October 2018, pp. 3075-3080. IEEE SMC Society **DOI:** 10.1109/SMC.2018.00657.
8. **Animesh Chaturvedi**, Aruna Tiwari, Shubhangi Chaturvedi, Pietro Lio'. "System Neural Network: Evolution and Change based Structure Learning". *IEEE Transactions on Artificial Intelligence*, Vol. 3.3, pp. 426 - 435, June 2022. ISSN: 2691-4581. **DOI:** 10.1109/TAI.2022.3143778.
9. **Animesh Chaturvedi**, and Rajesh Sharma. "minOffense: Inter-Agreement Hate Terms for Stable Rules, Concepts, Transitivities, and Lattices". *9th IEEE International Conference on Data Science and Advanced Analytics* (DSAA), 13-16 Oct 2022, Shenzhen China. **DOI:** 10.1109/DSAA54385.2022.10032389. (Core A)
10. **Animesh Chaturvedi**. "Call Graph Evolution Analytics over a Version Series of an Evolving Software System". *37th IEEE/ACM International Conference on Automated Software Engineering* (ASE '22), October 10-14, 2022, Rochester, MI, USA. **DOI:** 10.1145/3551349.3559573. (Core A*)