The 9th IEEE International Conference on Data Science and Advanced Analytics

System Network Analytics: Evolution and Stable Rules of a State Series

Dr. Animesh Chaturvedi
Indian Institute of Information Technology Dharwad, Dharwad, Karnataka, India
animesh.chaturvedi88@gmail.com

Prof. Aruna Tiwari
Indian Institute of Technology Indore, Indore, India
artiwari@iiti.ac.in

Emeritus Prof. Nicolas Spyratos
University of Paris-Saclay, Paris, France
spyratos@lri.fr

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY Dharwad ज्ञानेन विकासः

Indian Institute of Technology Indore भारतीय प्रौद्योगिकी संस्थान इंदौर

université PARIS-SACLAY

Overview

- **Motivation:** Given a state series $SS = \{S_1, S_2 \ldots S_N\}$, a rule might be interesting in one state, but not interesting in another state.
- **Contributions:**

  (a) introduced Evolution rules and Stable rules;

  (b) proposed System Network Databases, Network Evolution Rules (NERs), Stable NERs (SNERs), Persistence metric, and SNP metric;

  (c) System Network Analytics (SysNet-Analytics): algorithm
- Experimentations with results
- Literature surveys on network evolution rules

IEEE | Cosponsers ACM/ASA/CCF

DSAA 2022

http://dsaa2022.dsaa.co

Definitions and Concepts

Contributions:

(a) introduced Evolution rules and Stable rules;

(b) proposed System Network Databases, Network Evolution Rules (NERs), Stable NERs (SNERs), Persistence metric, and SNP metric;

(c) System Network Analytics (SysNet-Analytics): Algorithm

Definitions and Concepts

- ***System Evolution Analytics*** on a system that evolves is a challenge because it makes a ***State Series*** **SS = {S_1, S_2… S_N}** (i.e., a set of states ordered by time) with several inter-connected entities changing over time.
- We defined and presented
  - ***Stability*** characteristics of interesting ***evolution rules*** occurring in multiple states.
  - ***Evolution rule*** with its ***stability*** as the fraction of states in which the rule is interesting.
  - ***Stable rule*** is the evolution rule having stability that exceeds a given threshold ***minimum stability*** (**minStab**).
  - ***Persistence metric***, a quantitative measure of persistent entity-connections.

Evolution Rule and Stable Rule

- Suppose a state series SS makes a **Database Series DS = {D_1, D_2… D_N}**,
  - assume a rule (X→Y) in database D_i for state S_i.
  - The rule is *interesting* in D_i, if its support and confidence exceeds given thresholds.
- A distinct rule occurring in multiple states is said to be an ***Evolution Rule*** that has some ***Stability*** in the state series, where stability is the fraction of states in which the rule is interesting.
- An evolution rule is said to be a ***Stable Rule*** if its stability exceeds a given threshold named **minimum stability** (**minStab**).

System Network Database

- A **connection pair** (**CP**) is defined as **(L, R)**, where the symbols **L** and **R** as mnemonics for **Left** and **Right**, respectively.
- A **connection pair** in S_i is an ordered pair of two subsets: **source entities L** followed by their **target entities R**.
- A ***System Network Database*** (**SysNetDb**) is made-up of a set of *connection pairs*
  - $\{CP_1, CP_2 \ldots CP_M\}$, where **M** is the total number of connection pairs.
- The **SysNetDb** of state S_i is denoted as ***SysNetDb_i***.
- Dynamic databases for a state series are
  - ***SysNetDbs*** = {**SysNetDb_1, SysNetDb_2… SysNetDb_N**}.

Stable Network Evolution Rule (Stable NER)

Let SysNetDbs contains a set of entities $X \cup Y$ to form a rule $X \rightarrow Y$ of a state series SS

- The ***support*** of $X \rightarrow Y$ in S_i
  - $sup(X \rightarrow Y, S_i) = minSupCount \div M$,
  - CPs in which $X \cup Y$ occurs divided by total number of CPs in a SysNetDb.
- The ***confidence*** of $X \rightarrow Y$ in S_i
  - $conf(X \rightarrow Y, S_i) = sup(X \cup Y, S_i) \div sup(X, S_i)$
- A ***network rule*** ($X \rightarrow Y$) is called ***interesting***,
  - if support and confidence are greater than thresholds minSupCount and minConf.
- The $X \rightarrow Y$ is a **NER**, a distinct and identical network rule present in multiple states.
- The ***stability*** of NER ($X \rightarrow Y$) in state series (SS)
  - $stab(X \rightarrow Y, SS) = stabilityCount \div N$,
  - number of states in which $X \rightarrow Y$ is interesting divided by the number of states in SS.
- A NER $X \rightarrow Y$ is ***Stable NER*** in SS
  - if its stability count is greater than ($\geq$) a threshold $minStab = minStabCount \div N$.

Persistence metric

- *Number of persistent connections is directly proportional to the minStab*

$$\text{Persistent Connections} \propto minStab = \frac{\text{minStabCount}}{\text{N}} \quad \ldots (1).$$

- *Number of persistent connections is directly proportional to the stable rule count and inversely proportional to the distinct evolution rule count*

$$\text{Persistent Connections} \propto \frac{\text{SR\_Count}}{\text{ER\_Count}} \quad \ldots (2).$$

- Given a state series with its evolution rules and stable rules

$$\text{Persistence metric} = \text{minStab} \times \frac{\text{SR\_Count}}{\text{ER\_Count}} \times 100 \quad \ldots (3).$$

Physical significance of Persistence metric

- Low values of the *persistence metric* means
  - low values of minStab and SR_count
  - few stable rules occur in a smaller number of states
  - few persistent entity-connections
  - lower bound is 0 for a volatile system (no common rule between any two states).
- High value of *persistence metric* means
  - high values of minStab and SR_count produces
  - many stable rules occur in many states,
  - many persistent entity-connections,
  - upper bound is 100 for a constant system (not change with time & has identical states).
- Both lower and upper bound are ideal conditions; rarely occur with a system.

Physical significance of Persistence metric

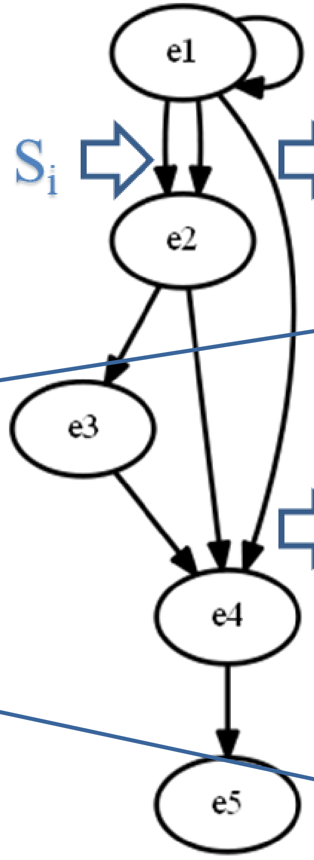
- Persistence metric value varies due to three reasons:
  - the system domain,
  - the fraction of (minStabCount ÷ N), and
  - the fraction of (SR_count ÷ ER_count)
- Use the *persistence metric* to know about the stability of a new state.
  - Suppose the *persistence metric* is PM_N for N states.
  - Add a new state to the system, then the new *persistence metric* is PM_{N+1} for N+1 states.
  - Three conditions are possible
    1. $PM_{N+1} > PM_N$ or
    2. $PM_{N+1} < PM_N$ or
    3. $PM_{N+1} = PM_N$.

An Illustrative Example

An Illustrative Example

- Assume
  - $E = \{e_1, e_2, .. e_K\}$ be a set of entities
  - *SysNetDb_i_ID* $= \{CP_1, CP_2 \ldots CP_M\}$ is a set of connection pairs
- Figure, assume a connection relationship provides
  - 3 connection pairs with
  - 5 entities in the network of state S_i
  - K=5 and M=3.

S_i



Connection Pairs with (L, R) where L is set of source entities and R is set of target entities
Input of Network Rule Mining

CP ID	*SysNetDb_i_ID*
CP_1	{1, 2, 3} {2, 3, 4}
CP_2	{1, 2} {2, 4}
CP_3	{4, 1, 1} {5, 1, 4}

Index

Entity ID	Entity
1	e_1
2	e_2
3	e_3
4	e_4
5	e_5

minSup-minConf Network Rule Mining

Network Rules net_Rules_i_ID

Input of NRM	
CP ID	*SysNetDb_1_ID*
CP_1	{1, 2, 3} {2, 3, 4}
CP_2	{1, 2} {2, 4}
CP_3	{4, 1, 1} {5, 1, 4}

Input of NRM	
CP ID	*SysNetDb_2_ID*
CP_1	{1, 2, 3} {2, 3, 4}
CP_2	{1, 2} {2, 4}
CP_3	{4, 1, 2} {5, 1, 4}

Input of NRM	
CP ID	*SysNetDb_3_ID*
CP_1	{1, 2, 3} {2, 3, 4}
CP_2	{1, 3} {2, 4}
CP_3	{4, 1, 2} {5, 1, 4}

Output of NRM for SysNetDb_1_ID		
Network Rules	Sup Count	Conf
1 → 4	3	1.0
1, 2 → 4	2	1.0
2 → 4	2	1.0
1 → 2	2	0.6
1 → 2, 4	2	0.6

Output of NRM for SysNetDb_2_ID		
Network Rules	Sup Count	Conf
1 → 4	3	1.0
1, 2 → 4	3	1.0
2 → 4	3	1.0
1 → 2	2	0.6
1 → 2, 4	2	0.6

Output of NRM for SysNetDb_3_ID		
Network Rules	Sup Count	Conf
1 → 4	3	1.0
1, 3 → 2	2	1.0
1, 3 → 2, 4	2	1.0
1, 3 → 4	2	1.0
3 → 2	2	1.0
3 → 2, 4	2	1.0
3 → 4	2	1.0
1, 2 → 4	2	0.6
2 → 4	2	0.6
1 → 2	2	0.6
1 → 2, 4	2	0.6

Collections of Network Rules		
1 → 4	1 → 4	1 → 4
--	--	1, 3 → 2
--	--	1, 3 → 2, 4
--	--	1, 3 → 4
--	--	3 → 2
--	--	3 → 2, 4
--	--	3 → 4
1, 2 → 4	1, 2 → 4	1, 2 → 4
2 → 4	2 → 4	2 → 4
1 → 2	1 → 2	1 → 2
1 → 2, 4	1 → 2, 4	1 → 2, 4

Network Evolution Rules (NERs)	Stability Count
1 ==> 4	3
1, 2 ==> 4	3
2 ==> 4	3
1 ==> 2	3
1 ==> 2, 4	3
1, 3 ==> 2	1
1, 3 ==> 2, 4	1
1, 3 ==> 4	1
3 ==> 2	1
3 ==> 2, 4	1
3 ==> 4	1

Stable Network Evolution Rules (SNERs)
1 → 4
1, 2 → 4
2 → 4
1 → 2
1 → 2, 4

Search space of NRM is SysNetDb of a state that contains connection pairs.

the interesting network rules are generated according to $minSupCount = 2$ and $minConf = 0.5$.

$minStabCount = 2$, $N = 3$, $SNERs = 5$, $NERs = 11$

System Network Persistence metric SNP

$$= minStab \times \frac{SNER\_Count}{NER\_Count} \times 100$$

$$= \{(2 \div 3) \times (5 \div 11)\} \times 100$$

System Network Analytics: Algorithm

Contributions:

(a) introduced Evolution rules and Stable rules;

(b) proposed System Network Databases, Network Evolution Rules (NERs), Stable NERs (SNERs), Persistence metric, and SNP metric;

(c) System Network Analytics (SysNet-Analytics): Algorithm

System Network Analytics (SysNet-Analytics)

- uses minStab to retrieve
  - **Network Evolution Rules** (**NERs**) and
  - **Stable NERs** (**SNERs**).
- Calculate a proposed **System Network Persistence** (**SNP**) metric.
- Automated as a **SysNet-Analytics Tool**

Algorithm **SysNet-Analytics(***repository***)**

Retrieve N system states and store them in a *repository*.

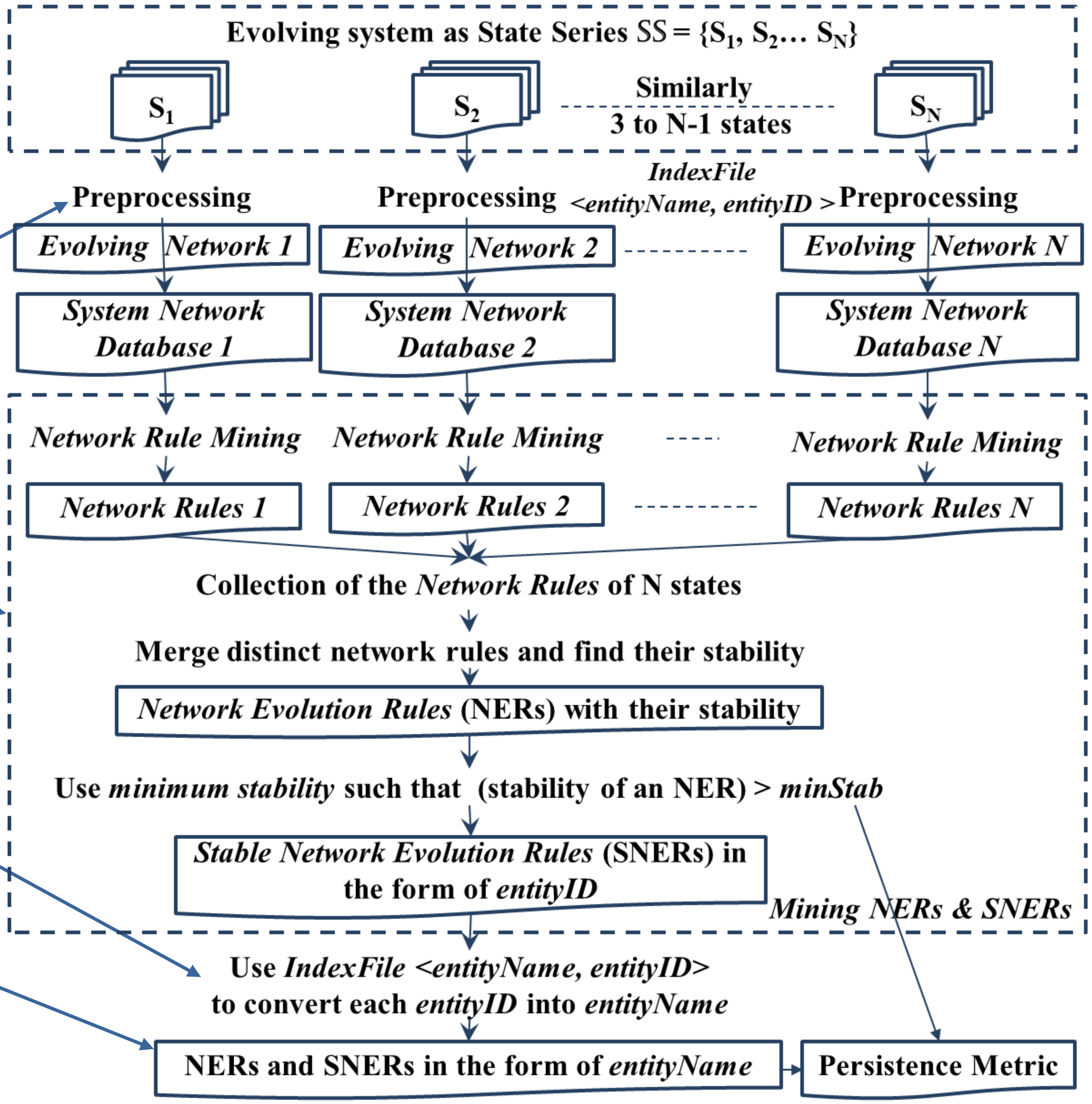
1. *SysNetDbs* & *IndexFile* = **Pre-process(***repository***)**
2. *NERs_ID* & *SNERs_ID* = **Mining_NERs_SNERs (***SysNetDbs*, *minSupCount*, *minConf*, *minStabCount***)**
3. *NERs_Name* & *SNERs_Name* = **Indexing(***NERs_ID*, *SNERs_ID*, *IndexFile***)**
4. *SNP_metric* = **SNP_Metric(***minStab*, *SNER_Count*, *NER_Count***)**

SysNet-Analytics

Algorithm **SysNet-Analytics(***repository***)**

Retrieve N system states and store them in a *repository*.

1. *SysNetDbs* & *IndexFile* = **Pre-process(***repository***)**
2. *NERs_ID* & *SNERs_ID* = **Mining_NERs_SNERs (***SysNetDbs*, *minSupCount*, *minConf*, *minStabCount***)**
3. *NERs_Name* & *SNERs_Name* = **Indexing(***NERs_ID*, *SNERs_ID*, *IndexFile***)**
4. *SNP_metric* = **SNP_Metric(***minStab*, *SNER_Count*, *NER_Count***)**



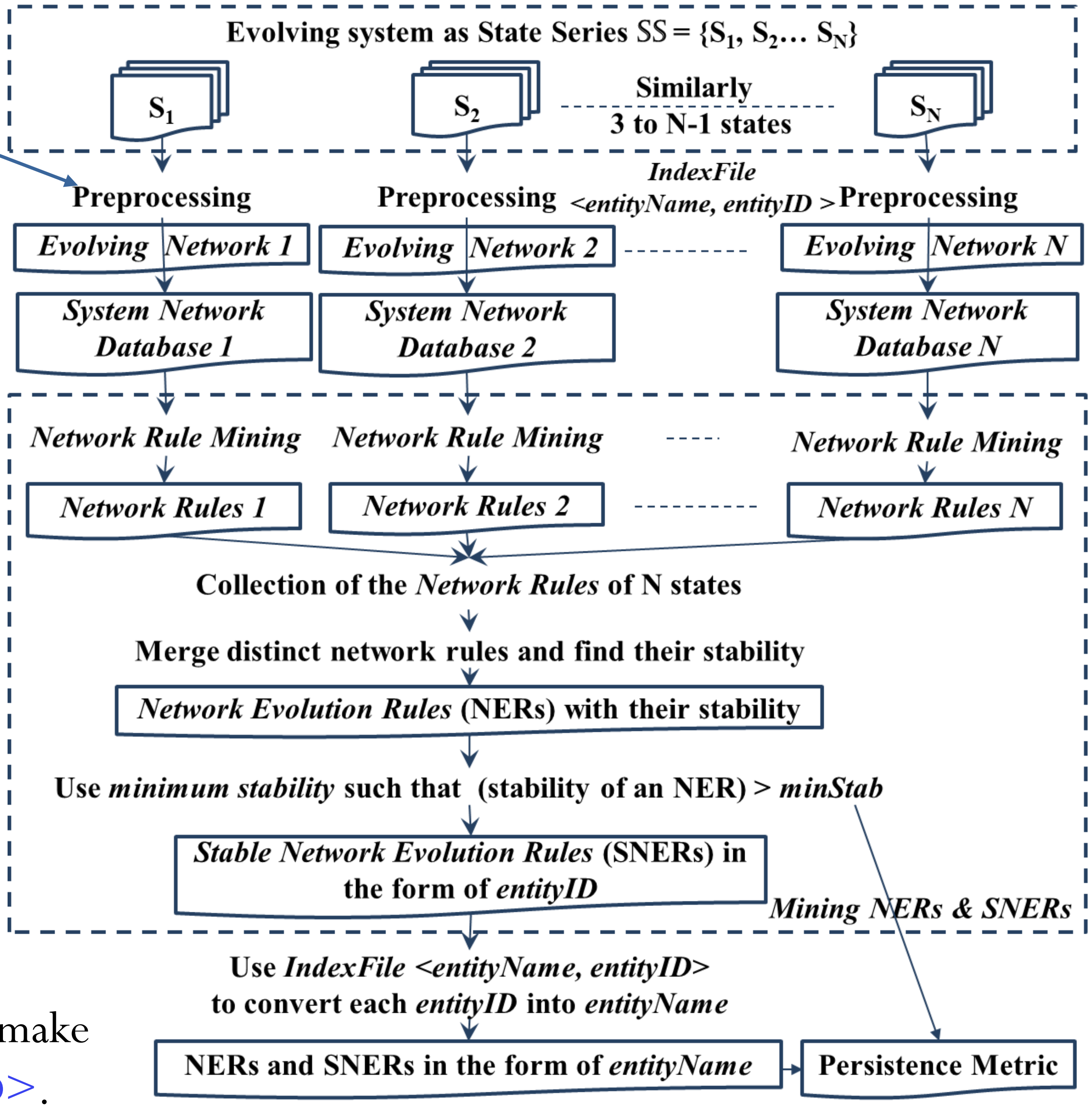
Algorithm 1 **Pre-process(*repository*)**

Initialize HashMap *Index*< *entityName*, *entityID* >
Initialize integer *counter* = 1
Initialize String Buffer *buffer*

For each state S_i where i ∈ integer and i varies from 1 to N
  Depending on type of repository, extract relationship between set of inter-connected entities to create a *SysNetDb_i* for a state S_i
  Read *SysNetDb_i* and store it in *buffer* until **end of file**
  For each line of *buffer*, **scan** *entityName*
    If an *entityName* is in the *Index*
      In *buffer*, replace the *entityName* with its *entityID*
    Else
      entityID = *counter*
      Add the new tuple < *entityName*, *entityID* > in the *Index*
      In *buffer*, replace the *entityName* with its *entityID*
      Increment the *counter* by 1 i.e. *counter* = *counter* + 1
  End of Scan when end of *buffer* is reached
Write the *buffer* in *SysNetDb_i_ID*
Store the file *SysNetDb_i_ID* in directory *SysNetDbs*
End For when all the states are pre-processed

Make an *IndexFile* and store *Index*<*entityName*, *entityID*>.

Return SysNetDbs & IndexFile



Pre-process N states stored in a repository to make N *SysNetDbs* and an *Index* <*entityName*, *entityID*>.

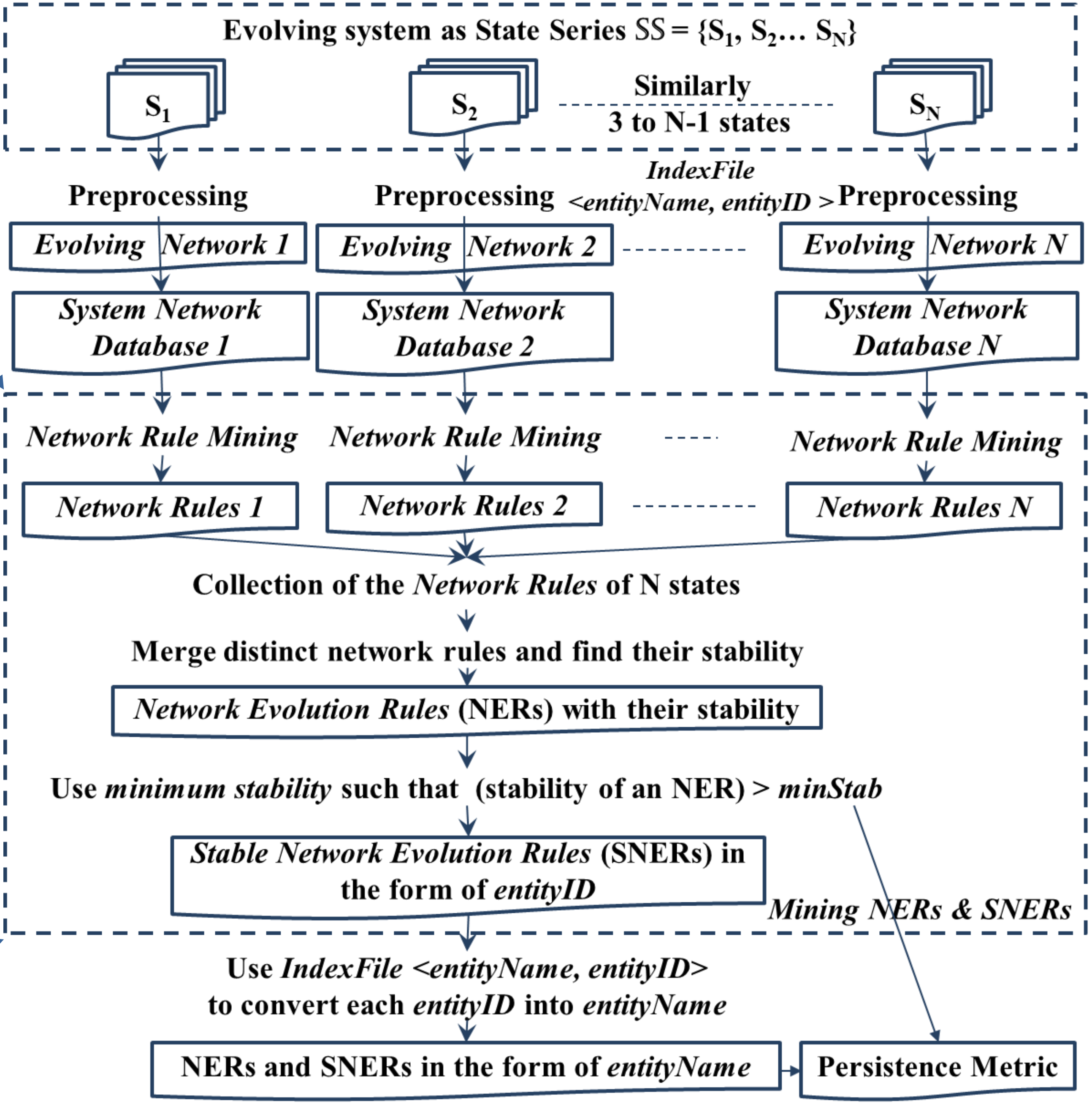
```
Algorithm 2 Mining_NERs_SNERs(SysNetDbs, minSupCount,
                               minConf, minStabCount)

Initialize File net_Rules_i_ID NERs_ID, SNERs_ID
Initialize Array Collect_NRs_ID,
Initialize HashMap NERs_HM < NER_ID, stability >
Initialize i ∈ integer
For each SysNetDb_i_ID in SysNetDbs, where i varies from 1 to N
   net_Rules_i_ID= NRM(SysNetDb_i_ID, minSupCount, minConf)
   Store net_Rules_i_ID file in directory netRules
End For
For each state net_Rules_i_ID in netRules, where i varies 1 to N
   Collect_NRs_ID = Merge(Collect_NRs_ID, net_Rules_i_ID)
End For
For each distinct rule (as NER_ID) in Collect_NRs_ID
   Initialize int stabilityCount = 0
   For each rule_x in Collect_NRs_ID
      if(NER_ID equal to rule_x)
         then stabilityCount++
      end if
   End for
   Initialize float stability = stabilityCount ÷ N
   Initialize float minStab = minStabCount ÷ N
   if(NER_ID is not in NERs_HM)
      then Add(<NER_ID, stability> to NERs_HM)
   end if
   if(stability > minStab)
      if(NER_ID is not in SNERs_ID)
         then Add(NER_ID to SNERs_ID)
      end if
   end if
End For
NERs_ID = NERs_HM
Return NERs_ID and SNERs_ID
```

Four inputs

1. SysNetDbs,
2. minSupCount,
3. minConf,
4. minStabCount

Retrieves

1. NERs_ID and
2. SNERs_ID



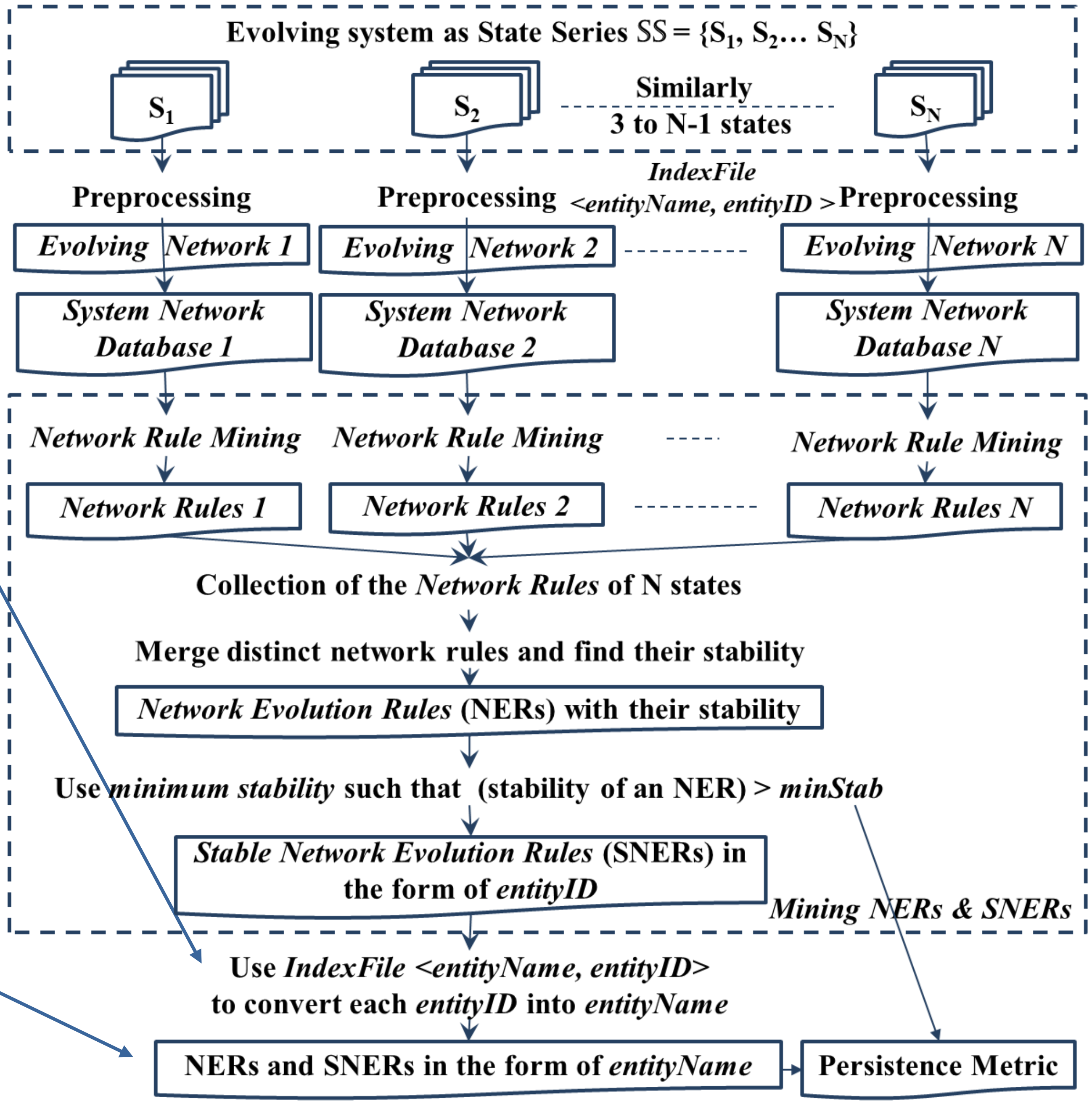
Algorithm 3 **Indexing(***NERs_ID, SNERs_ID, IndexFile***)**

Initialize HashMap *Index<entityName, entityID>*
Initialize integer *counter* = 1
Initialize String Buffer *buffer₁*, *buffer₂*
Make two files *NERs_Name*, *SNERs_Name*
For each line of *IndexFile*
  Scan line and **Store** *<entityName, entityID>* in *Index*
End For when *IndexFile* is completely scanned
For each line of file *NERs_ID*,
  Scan and **store** the line in *buffer*
  In *buffer*, **replace** each *entityID* with its *entityName* in *Index*
  Store the buffer in *NERs_Name*
End For when *NERs_ID* is completely scanned
For each line of file *SNERs_ID*,
  Scan and **store** the line in *buffer*
  In *buffer*, **replace** each *entityID* with its *entityName* in *Index*
  Store the buffer in *SNERs_Name*
End For when *SNERs_ID* is completely scanned
Return *NERs_Name*, *SNERs_Name*

{1, 2} ➔ {3}, where X = {1, 2} and Y = {3}.

entityName format rules like

{butter, jam} ➔ {milk} where 1, 2, and 3 stands for butter, jam, and milk, respectively.



System Network Persistence metric (Proposed)

- Keep note of the minStab used to retrieve SNERs.
- Calculate SNP metric given in Equation (4) using the minStab, number of SNERs retrieved (SNER_Count), and number of NERs retrieved (NER_Count).

System Network Persistence metric **(SNP metric)**

$$\text{SNP metric} = minStab \times \frac{\text{SNER\_Count}}{\text{NER\_Count}} \times 100 \quad \ldots (4)$$

where,

  - SNER_Count stands for the SNER count and
  - NER_Count stands for the NER count

- It measures the persistence quantitatively for entity-connections.

SysNet-Analytics Tool

- Based on Algorithm SysNet-Analytics, we developed an automated SysNet-Analytics Tool using Java technology (JRE and JDK).
- As input, the tool takes N *System Network Databases* (SysNetDbs) with thresholds.
- As output, the tool retrieves network rules, collection of NERs, and SNERs.
- The tool discovers these system evolution rules using three components based on the three algorithms:
  - 'Pre-processing',
  - 'Mining_NERs_SNERs'
  - 'Indexing'.

System Evolution Analytics Experiments

- ***Domain Information***
- *Experiments of SysNet-Analytics on four domains*
- *System Network Persistence (SNP) Experimental Results*

Application on Real-world systems

1. Software system,
2. Natural-language system,
3. Retail market system, and
4. IMDb system.

- Quantified **Stability** and **Persistence** of entity-connections in a system state series.
- The **evolution information** helps in **System Evolution Analytics**.

DOMAIN INFORMATION OF 6 EVOLVING SYSTEMS AND THEIR EVOLVING NETWORKS TO PERFORM SYSTEM EVOLUTION ANALYTICS

Domains of System Evolution Analytics	Evolving Systems	N	'Source' and 'Target' Entities	Type of Relationship	Type of network	Number of entities	Average number of neighbours	Number of aggregated connections
(A) Software Evolution Analytics	Hadoop HDFS[1]	15	'Caller' and 'Callee' Procedures	Procedural calls	Call graph	3129	2.166	15×3129×2.166 = 101661.21
(B) Natural-language Evolution Analytics	List of Bible Translation[2]	13	Words in 'Source biblical language' and 'English variant'	Translations	Word Network	246	1.456	13×246×1.456 = 4656.288
	List of Multi-sport Events[3]	13	Words in 'Titles (name)' and Scopes (region) of events	Regional names	Word network	141	1.786	13×141×1.786 = 3273.73
(C) Market Evolution Analytics	Retail Market[4]	13	'Products description' and 'Customer IDs'	Purchases	Purchase network	1872	7.204	13×1872×7.20 = 175219.2
(D) Movie Evolution Analytics on Evolving IMDb System[5]	Positive sentiment[6] of movie genres[5]	16	'Positive words in names' and 'genres' of movies	Sentiments	Positive sentiment network	284	2.661	16×284×2.661 = 12091.58
	Negative sentiment[6] of movie genres[5]	16	'Negative words in names' and 'genres' of movies	Sentiments	Negative sentiment network	510	3.303	16×510×3.303 = 26952.48

1. https://mvnrepository.com/artifact/org.apache.hadoop/hadoop-hdfs
2. https://en.wikipedia.org/wiki/List_of_English_Bible_translations
3. https://en.wikipedia.org/wiki/List_of_multi-sport_events
4. https://archive.ics.uci.edu/ml/datasets/Online+Retail
5. http://www.imdb.com/interfaces/
6. https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html

System Evolution Analytics Experiments

- *Domain Information*
- ***Experiments of SysNet-Analytics on four domains***
- *System Network Persistence (SNP) Experimental Results*

Experiments of SysNet-Analytics on four domains

- To find the optimum number of SNERs, we *explored* (minSupCount and minConf) and *exploited* (minStab) for the best possible high values of thresholds.
- Four Domains and 6 evolving systems

A. Software Evolution Analytics for Call graph

B. Natural-language Evolution Analytics for two lists

1. List of Bible Translations
2. List of Multi-sport Events

a) Market Evolution Analytics for Retail-market

A. Movie Evolution Analytics for IMDb

1. Positive sentiment of movie genres
2. Negative sentiment of movie genres

A. Software Evolution Analytics for Call graph

- preprocessed 15 jars of Hadoop-HDFS to make 15 *evolving call graphs* (networks) that further makes 15 SysNetDbs for 15 versions (states).

SNER (create → convert)
'create' is a caller procedure and 'convert' is a callee procedure.

The rule suggests that the procedure 'create' frequently calls procedure 'convert'.

This reveals that the changes in procedure 'convert' may also affect the procedure 'create'.

	minSupCount-minConf-minStabCount	N	minStab	SNER Count	Total NER Count	NERs fraction	SNP metric
Hadoop-HDFS as an Evolving Software System [16]							
1	4-0.6-5	15	0.33	2	2	1	33.33
2	4-0.6-4	15	0.27	2	2	1	26.67
3	4-0.4-5	15	0.33	3	5	0.6	20
4	4-0.6-3	15	0.2	2	2	1	20
5	4-0.2-7	15	0.47	2	5	0.4	18.67
6	4-0.8-5	15	0.33	1	2	0.5	16.67
7	4-0.4-3	15	0.2	4	5	0.8	16
8	4-0.4-4	15	0.27	3	5	0.6	16
9	4-0.8-6	15	0.4	0	2	0	0

B.1 Natural-language Evolution Analytics for List of Bible Translations

- extracted 13 evolving word networks that were converted to 13 SysNetDbs for 13 centuries.

SNERs (English, Modern → Greek) and (English → Vulgate) suggest that bible in the Modern English language is likely to be translated from Vulgate or Greek.

Our tool automatically retrieved many other such evolution rules (NERs and SNERs).

minSupCount-minConf-minStabCount	N	minStab	SNER Count	Total NER Count	NERs fraction	SNP metric
List of Bible translations as an Evolving Natural-language System						
3-0.2-2	13	0.15	25	3715	0.007	0.1
2-0.3-2	13	0.15	28	5397	0.005	0.08
3-0.3-2	13	0.15	16	3715	0.004	0.07
2-0.2-4	13	0.31	12	5644	0.002	0.07
3-0.3-3	13	0.23	6	3715	0.002	0.04
2-0.3-3	13	0.23	6	5397	0.001	0.03
2-0.3-4	13	0.31	3	5397	0.001	0.02
3-0.3-4	13	0.31	3	3715	0.001	0.02
2-0.2-5	13	0.38	0	5644	0	0

B.2 Natural-language Evolution Analytics for List of Multi-sport Events

- 13 evolving word networks that were converted to 13 SysNetDbs for 13 decades

SNERs (Games, Asian → Regional) and (Games, World → International) suggest that the 'Asian' 'Games' are of type 'Regional' sports. Similarly, we can deduce 'World' 'Games' are of type 'International' sports.

Our tool automatically retrieved many other such NERs and SNERs.

	minSupCount-minConf-minStabCount	N	minStab	SNER Count	Total NER Count	NERs fraction	SNP metric
List of Multi-sport events as an Evolving Natural-language System							
1	3-0.8-2	13	0.15	5	8	0.625	9.62
2	3-0.2-2	13	0.15	6	11	0.545	8.39
3	2-0.3-2	13	0.15	14	41	0.341	5.25
4	2-0.8-2	13	0.15	11	37	0.297	4.57
5	2-0.3-4	13	0.31	5	41	0.122	3.75
6	2-0.3-3	13	0.23	6	41	0.146	3.38
7	2-0.8-3	13	0.23	4	37	0.108	2.49
8	2-0.8-4	13	0.31	3	37	0.081	2.49
9	3-0.8-4	13	0.31	0	8	0	0

C. Market Evolution Analytics for Retail-market

- 13 evolving purchase networks, which are further converted to 13 SysNetDbs for 13 months

SNER
"SUKI SHOULDER BAG → 17841" suggests that the product with description (source words) 'SUKI SHOULDER BAG' is frequently purchased by the customer (target ID) '17841'.

For low threshold values, there are several SNERs, which are useful to do target marketing.

minSupCount-minConf-minStabCount	N	minStab	SNER Count	Total NER Count	NERs fraction	SNP metric
Evolving Retail Market System						
4-0.6-2	13	0.15	25	131	0.191	2.94
5-0.6-2	13	0.15	12	71	0.169	2.6
4-0.8-2	13	0.15	14	86	0.163	2.5
4-0.6-3	13	0.23	12	131	0.092	2.11
4-0.6-4	13	0.31	7	131	0.053	1.64
5-0.6-3	13	0.23	5	71	0.07	1.63
4-0.8-3	13	0.23	6	86	0.07	1.61
5-0.6-4	13	0.31	3	71	0.042	1.3
4-0.8-4	13	0.31	3	86	0.035	1.07

D.1 Movie Evolution Analytics for IMDb Positive sentiment of movie genres

- Evolving networks has connections between positive words in 'movie names' as source entities and their 'genres' as target entities.

SNER "premier → Short" suggests that the movie name containing 'premier' appears mostly in 'Short' genre movies.

While naming a movie for a genre, SNERs are useful to find positive words, which are suitable for a genre and to target a positive audience.

	minSupCount-minConf-minStabCount	N	minStab	SNER Count	Total NER Count	NERs fraction	SNP metric
Positive sentiment of movie genres in Evolving IMDb System							
1	2-0.3-3	16	0.19	21	146	0.144	2.70
2	2-0.3-4	16	0.25	5	146	0.034	0.86
3	2-0.3-5	16	0.31	4	146	0.027	0.86
4	3-0.3-4	16	0.25	4	74	0.054	1.35
5	4-0.3-3	16	0.19	4	49	0.082	1.53
6	3-0.4-4	16	0.25	3	60	0.050	1.25
7	3-0.5-4	16	0.25	2	48	0.042	1.04
8	4-0.3-4	16	0.25	1	49	0.020	0.51
9	4-0.3-5	16	0.31	0	49	0.000	0.00

D.2 Movie Evolution Analytics for IMDb Negative sentiment of movie genres

- Made evolving networks using sentiment-list of negative words

SNER "sin → Drama" suggests that the movie name containing 'sin' appears mostly in the 'Drama' genre

SNERs are useful to find negative words that are suitable for a genre and to target a negative audience.

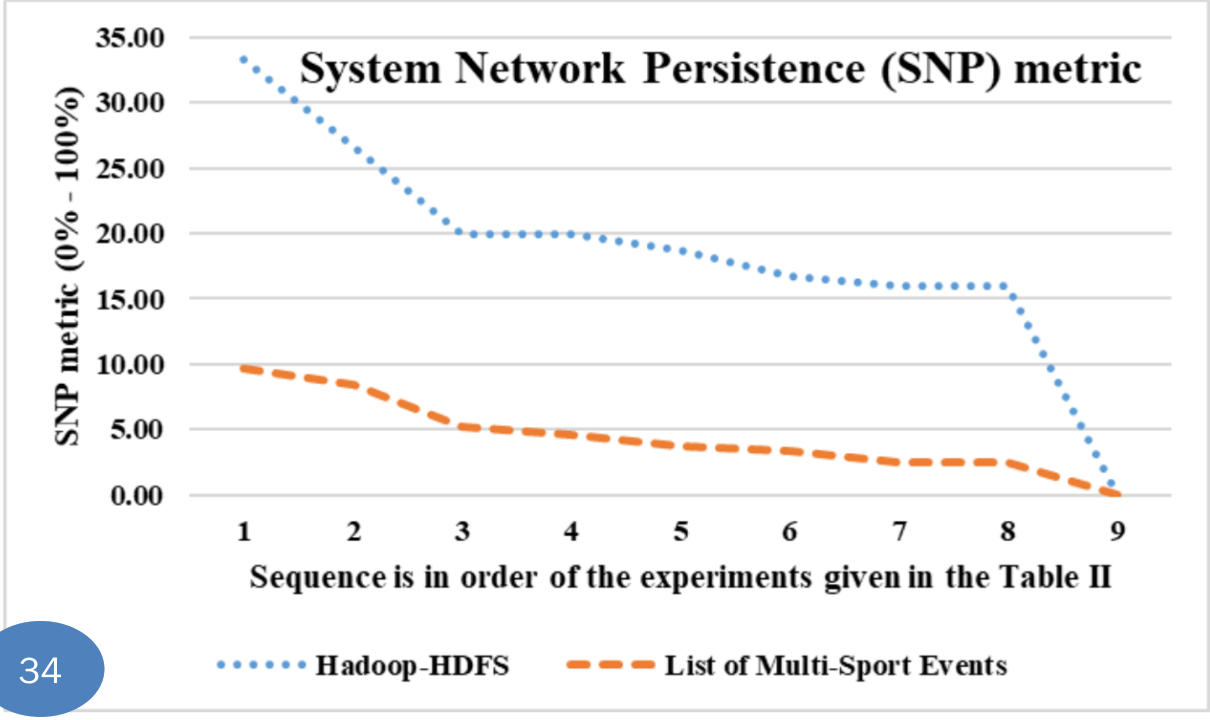
minSupCount-minConf-minStabCount	N	minStab	SNER Count	Total NER Count	NERs fraction	SNP metric
Negative sentiment of movie genres in Evolving IMDb System						
2-0.3-3	16	0.19	31	258	0.12	2.25
4-0.3-3	16	0.19	7	79	0.089	1.66
2-0.3-4	16	0.25	10	258	0.039	0.97
4-0.3-4	16	0.25	3	79	0.038	0.95
3-0.3-4	16	0.25	4	116	0.034	0.86
3-0.4-4	16	0.25	2	84	0.024	0.6
3-0.5-4	16	0.25	1	63	0.016	0.4
4-0.3-5	16	0.31	1	79	0.013	0.4
2-0.3-5	16	0.31	3	258	0.012	0.36

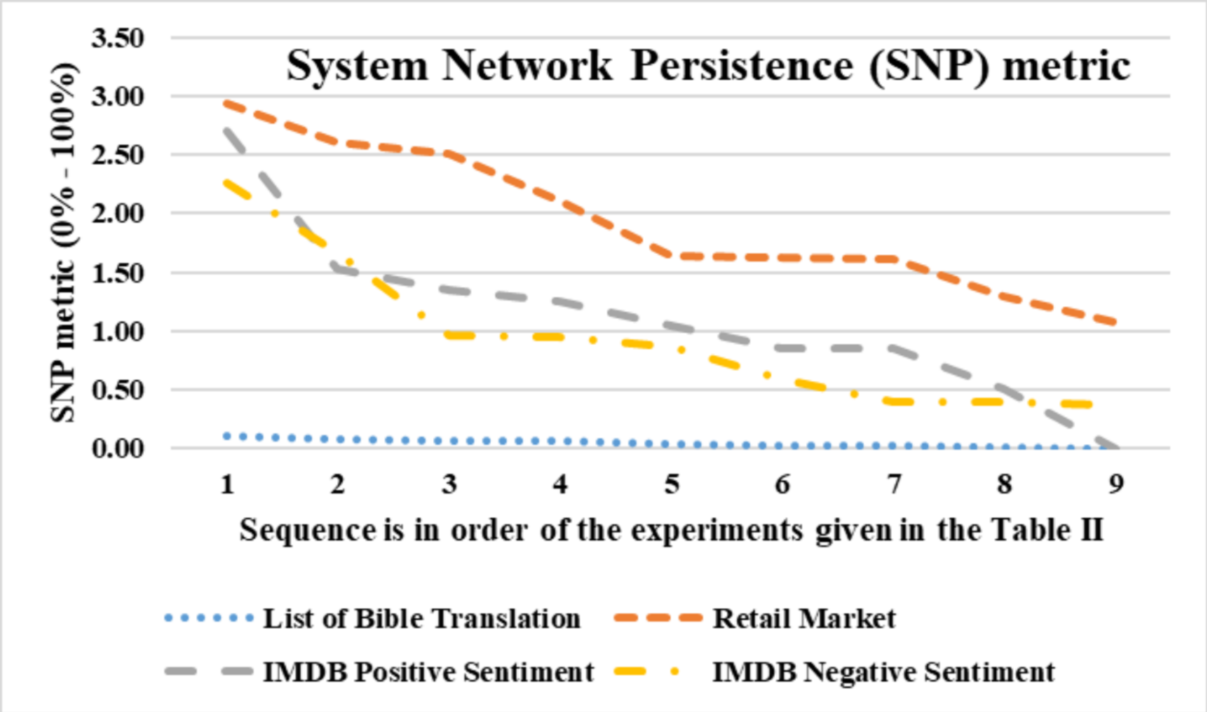
System Evolution Analytics Experiments

- *Domain Information*
- *Experiments of SysNet-Analytics on four domains*
- ***System Network Persistence (SNP) Experimental Results***

System Network Persistence (SNP) Experiments

- SNP metrics using the nine experiments for all six evolving systems.
- Six series to demonstrate persistence metric values for all six evolving systems.
- Each series has nine coordinates that represent nine SNP metric values for nine experiments. The X-axis has the same sequence as mentioned in Tables.





Related Works & Discussions

Existing State-of-the-Arts

- Intense research activities on mining networks that change over time:
  - Graph Evolution Rules (GERs) by Berlingerio et al. [18];
  - Link Formation Rules (LFRs) by Leung et al. [19];
  - Graph-Pattern Association Rules (GPARs) by Fan et al. [20];
  - EvoMine tool by Scharwächter et al. [21];
  - Graph Temporal Association Rules (GTARs) by Namaki et al. [22];
  - Attribute Evolution Rules (AERs) by Fournier-Viger et al. [23]; and
  - TACOs (TemporAl event prediCtiOn rules) by Wenfei et al. [24].

[18] M. Berlingerio, et al. "Mining Graph Evolution Rules". *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer Berlin Heidelberg, 2009.

[19] C. W. Leung, et al. "Mining Interesting Link Formation Rules in Social Networks". *19th ACM IEEE International Conference on Information and Knowledge Management*. ACM, 2010.

[20] W. Fan, et al. "Association Rules with Graph patterns". *VLDB Endowment* 8.12 (2015): 1502-1513.

[21] E. Scharwächter, et al. "Detecting Change Processes in Dynamic Networks by Frequent Graph Evolution Rule Mining"., *IEEE 16th International Conference on Data Mining (ICDM)*. IEEE, 2016.

[22] M. H. Namaki, et al. "Discovering Graph Temporal Association Rules". *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. ACM, 2017.

[23] P. Fournier-Viger, et al. "Mining Attribute Evolution Rules in Dynamic Attributed Graphs". *International Conference on Big Data Analytics and Knowledge Discovery*. Springer, Cham, 2020.

[24] W. Fan, et al. "Towards Event Prediction in Temporal Graphs". *Proceedings of the VLDB Endowment* 15.9 (2022): 1861-1874.

Existing State-of-the-Arts

Graph Evolution Rules (GERs); Link Formation Rules (LFRs); Graph-Pattern Association Rules (GPARs); EvoMine tool; Graph Temporal Association Rules (GTARs); Attribute Evolution Rules (AERs); and TACOs (TemporAl event prediCtiOn rules)

Rules	**Contribution**	**Future Work**
GERs (2009)	Describes the local changes and applied to four real-world networks.	Investigate rule confidence to predict graph evolution.
LFRs (2010)	Edge labels but unlabeled nodes are applied to two real-world datasets. LFR contains link patterns as dyadic and/or triadic structures in social networks	Study of rules at a certain time point and multigraphs.
GPARs (2015)	Discover regularities between entities on social media graphs with a parallel (scalable) algorithm. Help in marketing by identifying and influencing customers.	To support graph patterns as consequent and provide other matching semantics as graph simulation.
EvoMine (2016)	Mines graphs - edge insertions and deletions - and - node and edge relabelling. Provide results for comparing EvoMine with the GERM and LFR miner.	Time and space analysis for quality and confidence of rules.
GTARs (2017)	Class of temporal association rules and used for activity prediction.	Exploring other quality metrics of rules and rules over graphs.
AERs (2020)	Demonstrated changes of attributed values of multiple vertices on dynamic graphs.	To discover concise AERs with specific temporal constraints.
TACOs (2022)	Demonstrated a system TASTE to discover TACOs on temporal graphs.	To make TASTE for finance and real-time prediction.
SNERs (2019 [12] and this paper)	We defined stability of NERs over states to retrieve Stable NERs. We defined persistence metrics. Shown System Evolution Analytics on six state series.	Other mining techniques with network evolution to make new hybrid mining. Application on Big graphs of other systems.

[12] A. Chaturvedi, A. Tiwari, and N. Spyratos "minStab: Stable Network Evolution Rule Mining for System Changeability Analysis". *IEEE Trans. on Emerging Topics in Computational Intelligence*, 2019.

Comparison with Existing State-of-the-Arts

- NERs is a concept equivalent to rules given in the Table GERs, LFRs, GPARs.
  - Introduced a novel method to retrieve NERs.
  - Optimized the number of SNERs over multiple states
  - Compare the reduction in the number of rules experiment Tables i.e., SNER Count and Total NER Count.
  - Smaller number of SNERs are more interesting and stable than NERs.
- Usually, network and rule mining are independent of time. Evolution (temporal) mining is independent of inter-connected entities.
  - Our approach is better in the sense that it intelligently combines network, rule, and evolution mining.

Conclusions

Conclusions

- Evolution rules, Stable rules, minStab, NERs, SNERs, Persistence metric, SNP metric
- SysNet-Analytics approach is applicable to a set of evolving networks.
- Analyses a pre-evolved system using a threshold minStab to generate SNERs.
- Quantifies the stability and persistence of entity-connections in an evolving system.
- Real-world applications for 4 domains based on the 6 State-Series of 6 evolving systems
  - SysNet-Analytics Tool automatically retrieves NERs and SNERs,
  - calculated 9 SNP metric values for 9 experiments on each evolving system,
  - to measure persistence of entity-connections for an evolving system,
  - to deduce system evolution information, which is interesting and non-obvious.
- In future,
  - other mining techniques with network evolution analytics; our approach can be applicable to large networks or big graphs; our experimentation with other datasets.

Related Publications

Citation:

Animesh Chaturvedi, Aruna Tiwari, and Nicolas Spyratos.

"**System Network Analytics: Evolution and Stable Rules of a State Series**."

IEEE 9th International Conference on Data Science and Advanced Analytics (DSAA). IEEE, 2022.

Evolution Rule and Stable Rule Mining

- Previous publication of this work
  - A. Chaturvedi, A. Tiwari, and N. Spyratos "**minStab: Stable Network Evolution Rule Mining** for System Changeability Analysis". *IEEE Trans. on Emerging Topics in Computational Intelligence*, 2019.

- Applied the concept of minStab to retrieve significant **Stable Hate Rules**
  - A. Chaturvedi, and R. Sharma. **"minOffense: Inter-Agreement Hate Terms for Stable Rules, Concepts, Transitivities, and Lattices"**. *IEEE 9th International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, 2022.

- System Evolution Analytics
  - A. Chaturvedi and A. Tiwari. "**System Evolution Analytics**: Evolution and Change Pattern Mining of Inter-Connected Entities". *IEEE International Conference on Systems, Man, and Cybernetics (SMC)* 2018.
  - A. Chaturvedi and A. Tiwari. "**System Evolution Analytics**: Deep Evolution and Change Learning of Inter-Connected Entities". *IEEE International Conference on Systems, Man, and Cybernetics (SMC)* 2018.

Other Related Paper

- Subgraph evolution mining to calculate **System Network Complexity**.
  - A. Chaturvedi and A. Tiwari. "**System Network Complexity**: Network Evolution Subgraphs of System State series". *IEEE Transactions on Emerging Topics in Computational Intelligence* 4.2 (2018): 130-139.
- Deep Evolution Learning led to the construction of System Evolution Recommender tool, which further led to the invention of **System Neural Network**
  - A. Chaturvedi and A. Tiwari. "**SysEvoRecomd**: Graph Evolution and Change Learning based System Evolution Recommender". *IEEE International Conference on Data Mining Workshops (ICDMW)*. IEEE, 2018.
  - A. Chaturvedi, A. Tiwari, and S. Chaturvedi. "**SysEvoRecomd**: Network Reconstruction by Graph Evolution and Change Learning". *IEEE Systems Journal* 14.3 (2020): 4007-4014.
  - A. Chaturvedi, et al. "**System Neural Network**: Evolution and Change based Structure Learning". *IEEE Transactions on Artificial Intelligence* 3.3 (2022): 426-435.

ขอบคุณ

Thai

Grazie

Italian

תודה רבה

Hebrew

धन्यवादः

Sanskrit

ಧನ್ಯವಾದಗಳು

Kannada

Ευχαριστώ

Greek

Thank You

English

Gracias

Spanish

Спасибо

Russian

Obrigado

Portuguese

شكراً

Arabic

https://sites.google.com/site/animeshchaturvedi07

Merci

French

多謝

Traditional Chinese

धन्यवाद

Hindi

Danke

German

多谢

Simplified Chinese

நன்றி

Tamil

Tamil

ありがとうございました

Japanese

감사합니다

Korean