Match-Based Candidate Network Generation for Keyword Queries over Relational Database

Pericles de Oliveira[1,2] Altigran da Silva[1] Edleno de Moura[1] Rosiane Rodrigues [1]

NOKIA Solutions and Networks, Rio de Janeiro, Brazil[2]

Introduction

- R-KwS systems: take queries as a set of keywords and return JNTs – joint networks of tuples – that fulfill the user needs
- Candidate Networks (CNs): Relational Join Expressions automatically generated from input keywords that when evaluated by a RDBMS produce relevant JNTs
- Current CN generation approaches require an exhaustive exploration of a large number of combinations of keyword occurrences in the DB
- For certain queries, they can take too long to produce answers and may even fail to return results (e.g., by exhausting memory)
- We propose MatCNGen, a novel approach for generating CNs that drastically reduces the time required to generate CNs
- Besides improving CN generation, MatCNGen also has a positive impact on the evaluation of CNs: run faster and produces better results.

MatCNGen

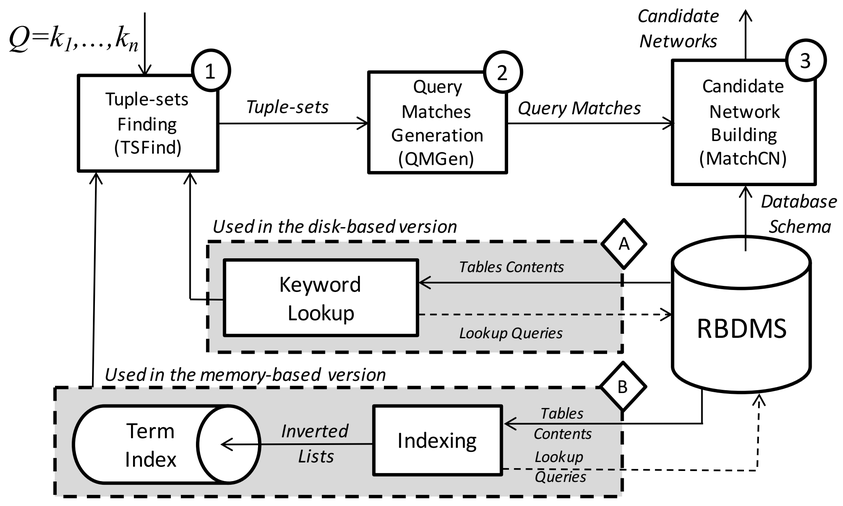


Figure: MatCNGen main steps.

Step 1 – Finding Tuple Sets

What are the relevant pieces of information in the DB?

CHAR

Id	Char_name	....
CH1	George Washington	...
CH2	Mr. Washington	...
CH3	Gangster	...
CH4	Agent Doug Carlin	....

ROLE

Id	Role	....
R1	Actor	...
R2	Actress	...
R3	Writer	...
R4	Producer	....

CAST

Id	MOV_id	CHAR_id	ROLE_id	PER_id	Note
C1	M3	CH3	R1	P3	... gangster ...
C2	011	004	001	001	... gangster ...
C3	...	...	...	...	... Denzel ... Washington ... gangster
C4	...	...	...	...	.... Washington ....

MOV

Id	Title	year	....
M1	George Washington	1923	...
M2	Deja Vu	2006	...
M3	American Gangster	2007	...

PER

Id	Person	....
P1	Denzel Philip	...
P2	Steven Washington	...
P3	Denzel Washington	...

Database Instance

Tuple-Sets

CHAR{gangster}

Id	...	Note
C1	...	... gangster ...
C2	...	... gangster ...

CAST{denzel,washington,gangster}

Id	...	Note
C3	...	... Denzel ... Washington ... gangster

CAST{washington}

Id	...	Note
C4	...	.... Washington ....

PER{denzel}

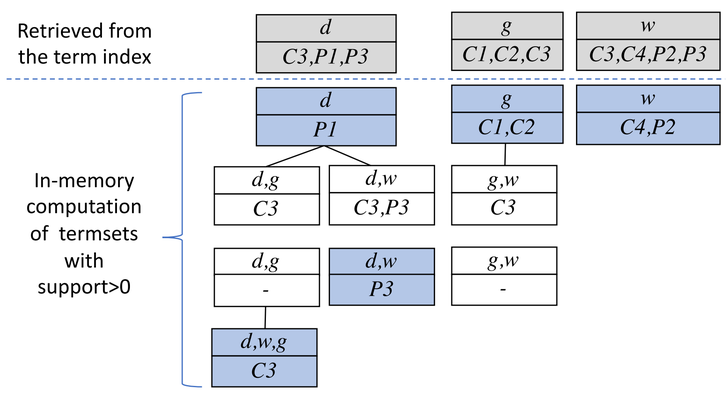
Id	Person	....
P1	Denzel Philip	...

PER{washington}

Id	Person	....
P2	Steven Washington	...

PER{denzel,Washington}

Id	Person	....
P3	Denzel Washington	...



Step 2 – Query Matches Generation

Which pieces fit together to produce meaningful answers?

All Tuple-sets

MOV{w} MOV{w,g} MOV{d,w} MOV{g}

CAST{d,w} CAST{d} CAST{w}

CHAR{w} CHAR{d}

PER{d,w}

Query Matches
Keywords form covers of {d,w,g}

MOV{g} PER{d,w}

CHAR{g} CAST{d,w}

MOV{w,g} CAST{d,w}

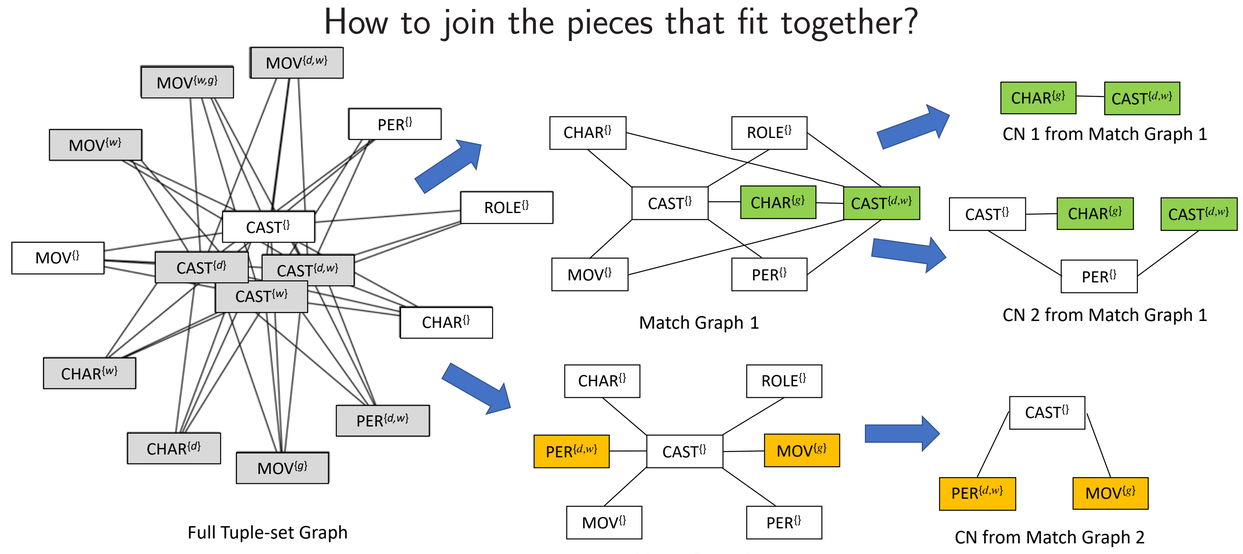
MOV{g} CAST{w} Non-total

CAST{d,w} CHAR{w} MOV{g} Non-minimal

- For a query Q, there is no Query Match with more then $|Q|$ tuple-sets!!

Step 3 – Candidate Network Building

How to join the pieces that fit together?



$$\sigma_{\text{title}\supseteq\{gangster\}}\,\text{MOV} \bowtie_{\text{id=cid}} \text{CAST} \bowtie_{\text{cid=pid}} \sigma_{\text{name}\supseteq\{denzel,washington\}}\,\text{PER}.$$

Experimental Datasets

Dataset	Size (MB)	Relations	Tuples	RIC
Mondial	9	28	17,115	104
IMDb	516	5	1,673,074	4
Wikipedia	550	6	206,318	5
DBLP	40	6	878,065	6
TPC-H	876	8	2,389,071	11

Table: Characteristics of the datasets used.

Results

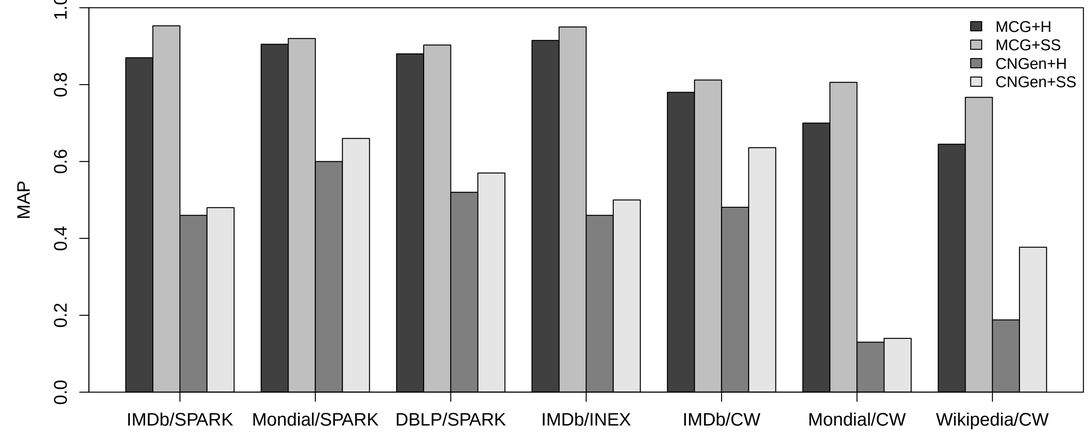


Figure: MAP results for all query sets.

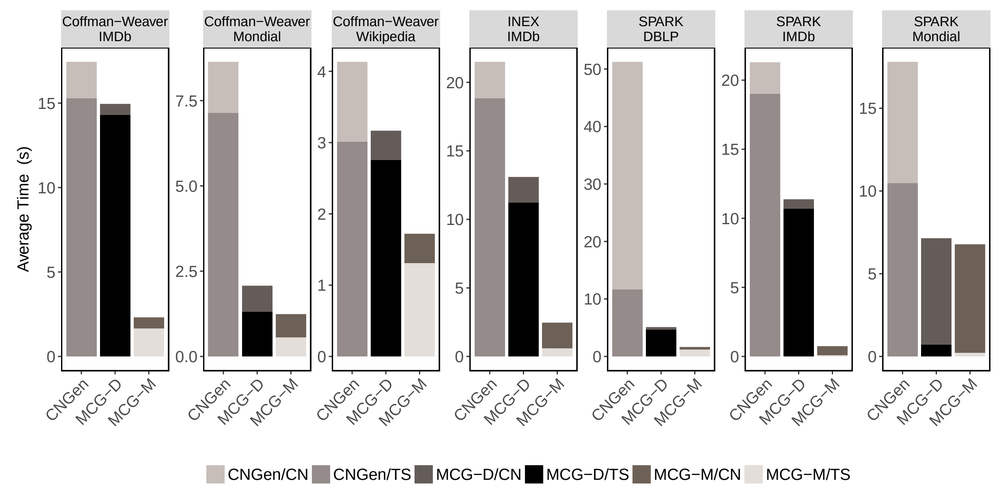
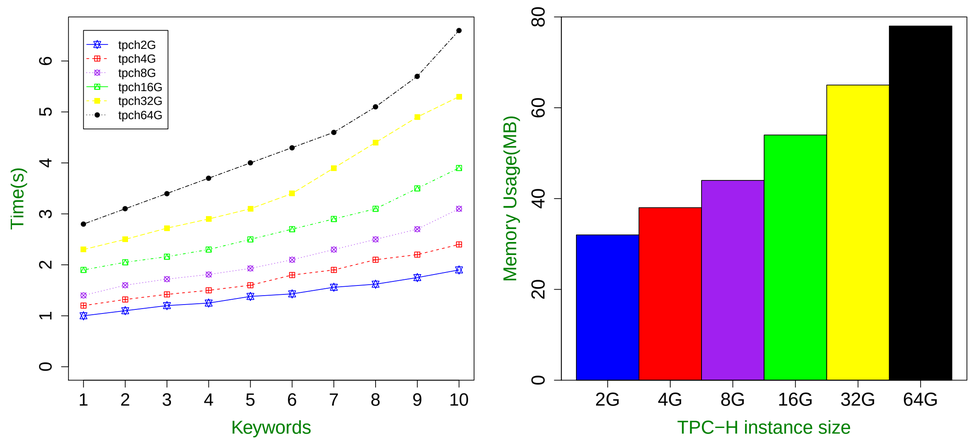


Figure: Time to generate CNs using CNGen and MatCNGen.



Response Time (a) Memory Usage (b)

Figure: Scalability with instances of varying sizes.

{pericles,alti,edleno}@icomp.ufam.edu.br