CIKM 2016
Indianapolis
October 24-28, 2016

Approximating Graph Pattern Queries Using Views

Jia Li[1], Yang Cao[2], Xudong Liu[1]

[1]Beihang University, China

[2]University of Edinburgh, UK

北京航空航天大學
BEIHANG UNIVERSITY

University of Edinburgh

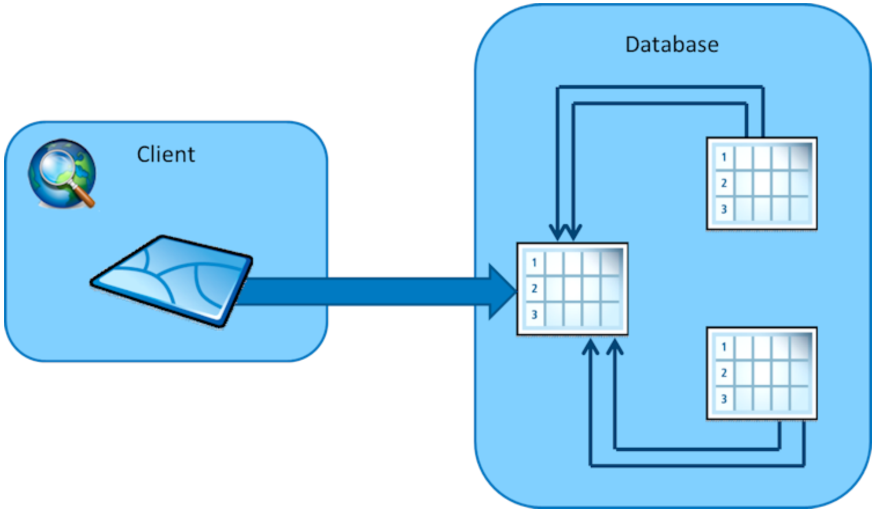
Background

Answering queries using materialized views

- materialize views over the database previously
- speed up query processing

Application

- relational queries
- graph pattern queries
  - ✓ SPARQL
  - ✓ graph simulation



Challenges

- Answering simulation queries using views

Input: A simulation pattern query Q, a set $\mathcal{V}$ of pattern views $V_1, V_2, \ldots, V_n$, data graph G, the materialized view answers $V_1(G), V_2(G), \ldots, V_n(G)$ in G.

Question: Can Q be answered using views $\mathcal{V}$, i.e. the matches Q(G) to Q in G can be computed using nodes and edges in the view answers only.

- Good news: If so, desirable performance
- Bad news: the physical storage is limited, such that the cached views are limited

In many cases, queries cannot be exactly answered using $\mathcal{V}$

Q can be upper and lower "bounded" via *approximations* which can be answered using $\mathcal{V}$

- Approximating answering simulation queries using views

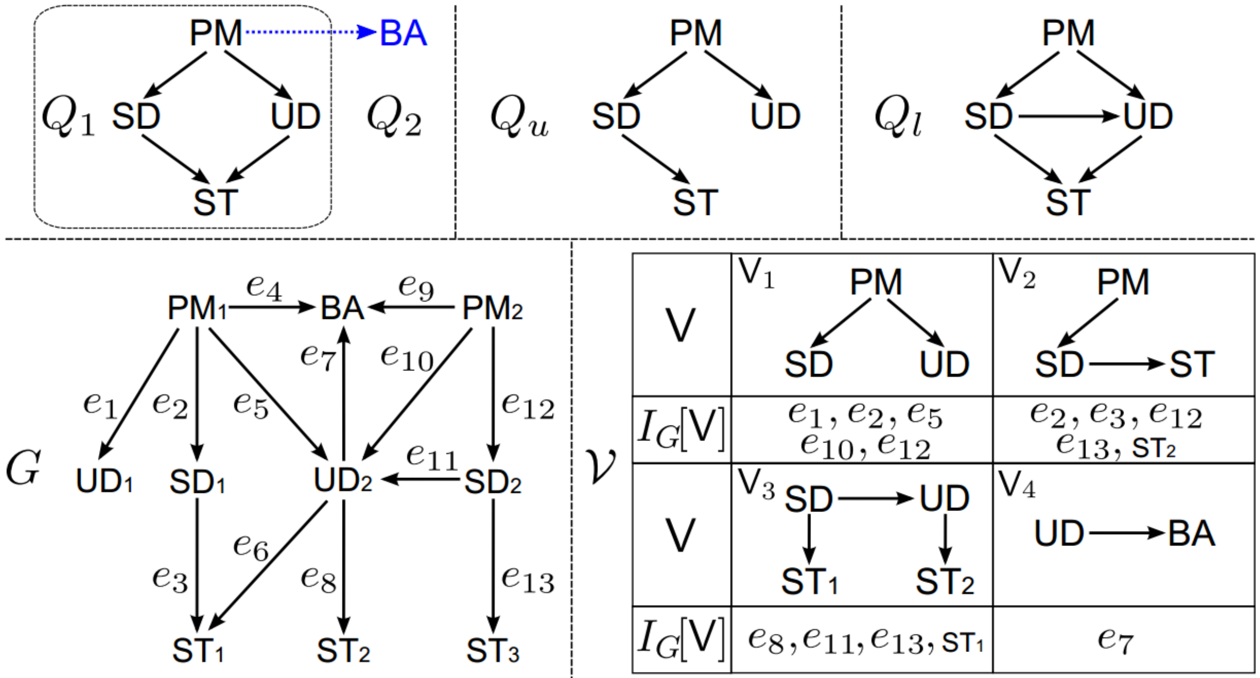
Question: Are there exist two pattern graphs Q_u and Q_l, such that

  - both Q_u and Q_l can be answered using $\mathcal{V}$, and
  - for *all* data graphs G, $Q_l(G) \subseteq Q(G)$ and $Q(G) \subseteq Q_u(G)$.

We call Q_u and Q_l *upper* and *lower* approximations of Q w.r.t. $\mathcal{V}$

An example: graph pattern queries (graph simulation)

Querying a recommendation network:

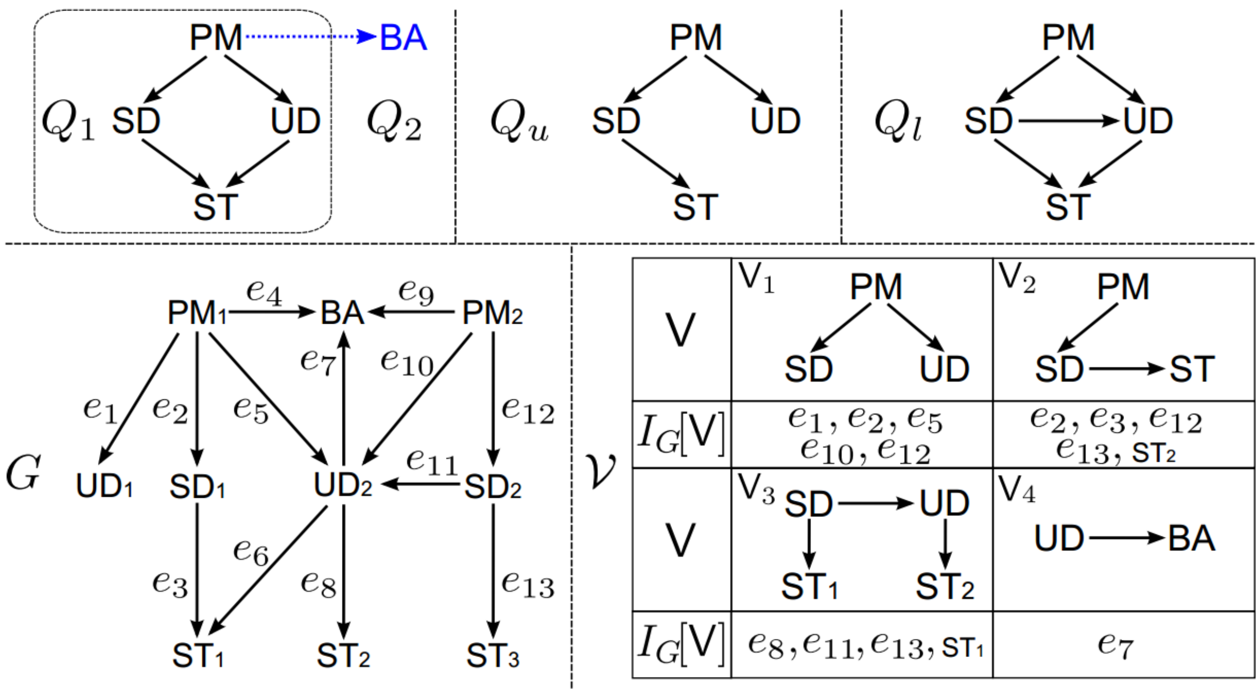


- (1) Q_1 cannot be answered using view answers.
- (2) There exist Q_u and Q_l can be answered using $I_G(V_1)$, $I_G(V_2)$ and $I_G(V_1)$, $I_G(V_3)$.
- (3) $Q_l(G) \subseteq Q_1(G) \subseteq Q_u(G)$.

Q_1 is *"completely" upper* and *lower* approximated by Q_u and Q_l w.r.t. $\mathcal{V}$

An example: graph pattern queries (graph simulation)

Querying a recommendation network:



- (1) Q_2 cannot be answered using view answers.
- (2) There exist no Q_u and Q_l that can "completely" upper and lower bound Q_2
- (3) The answers to subgraphs of Q_2 can be bounded, i.e. $Q_l(G) \subseteq Q_1(G) \subseteq Q_u(G)$.

> Q_2 is *"partially" upper* and *lower* approximated by Q_u and Q_l w.r.t. $\mathcal{V}$

Overview

- Formalization of upper and lower approximation of pattern queries
  - Simulation queries (graph simulation)
  - Subgraph queries (subgraph isomorphism)
- Investigating fundamental problems for simulation queries
  - Existence of (complete) upper approximation: EUA, EUA^c
  - Existence of (complete) lower approximation: ELA, ELA^c
  - Closest (complete) upper approximation: CUA, CUA^c
  - Closest (complete) lower approximation: CLA, CLA^c
- Developing algorithms with provable guarantees for the problems
- Extending the study to subgraph queries
  - Fundamental problems
  - Answering subgraph queries using views

Formalization of upper and lower approximation of pattern queries

Upper and lower approximation

- Query answering using views: A query Q is answerable using views in $\mathcal{V}$, for any data graph G, if Q(G) can be identified by accessing the answers of views in G only.

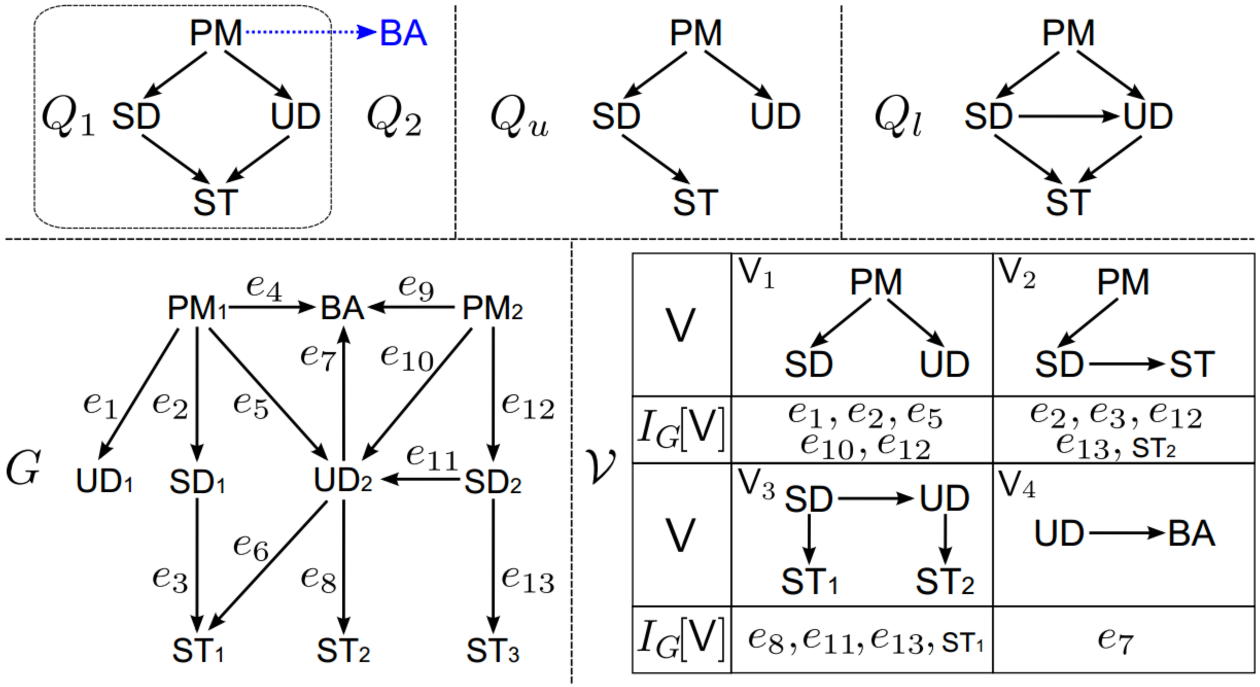
- Partial and complete query containment:
  - ✓ $Q \sqsubseteq_{\mathcal{U}} Q_u$: if there exists an induced subgraph Q_s of Q such that $Q_s(G) \subseteq Q_u(G)$
  - ✓ $Q_l \sqsubseteq_{\mathcal{L}} Q$: if there exists an induced subgraph Q_s of Q such that $Q_l(G) \subseteq Q_s(G)$
  - ✓ $Q \sqsubseteq^{c}_{\mathcal{U}} Q_u$: when Q_s above is Q
  - ✓ $Q_l \sqsubseteq^{c}_{\mathcal{L}} Q$: when Q_s above is Q

  We call Q is *partially upper contained* in pattern Q_u, *partially lower contains* Q_l, is *completely upper contained* in Q_u, and *completely lower contains* Q_l.

- Upper and lower approximation:
  - ✓ Q_u is an *upper* approximation of Q : if (1) $Q \sqsubseteq_{\mathcal{U}} Q_u$; (2) Q_u is answerable using $\mathcal{V}$
  - ✓ Q_u is a *complete upper* approximation of Q : if (1) $Q \sqsubseteq^{c}_{\mathcal{U}} Q_u$; (2) Q_u is answerable
  - ✓ Q_l is a *lower* approximation of Q : if (1) $Q_l \sqsubseteq_{\mathcal{L}} Q$; (2) Q_l is answerable using $\mathcal{V}$
  - ✓ Q_l is a *complete lower* approximation of Q : if (1) $Q_l \sqsubseteq^{c}_{\mathcal{L}} Q$; (2) Q_l is answerable

An example: graph pattern queries (graph simulation)

Querying a recommendation network:



- (1) $Q_1 \sqsubseteq^c_{\mathcal{U}} Q_u$, Q_u is answerable using $\mathcal{V}$, Q_u is a complete upper approximation of Q_1,
- (2) $Q_2 \sqsubseteq_{\mathcal{U}} Q_u$, Q_u is answerable using $\mathcal{V}$, Q_u is an upper approximation of Q_2,
- (3) $Q_l \sqsubseteq^c_{\mathcal{L}} Q_1$, Q_l is answerable using $\mathcal{V}$, Q_l is a complete lower approximation of Q_1,
- (4) $Q_l \sqsubseteq_{\mathcal{L}} Q_2$, Q_l is answerable using $\mathcal{V}$, Q_l is a lower approximation of Q_2.

Fundamental problems for simulation queries

Existence of approximation

- Existence of upper approximation (EUA) : Given a simulation query Q and $\mathcal{V}$, whether there exists an upper approximation Q_u for Q w.r.t. $\mathcal{V}$.
- Existence of complete upper approximation (EUA^c)

For a simulation query Q and a set $\mathcal{V}$ of views,

- ✓ there exists an upper approximation for Q using $\mathcal{V}$ iff there exists V ∈ $\mathcal{V}$ such that the match result V(Q) ≠ ∅
- ✓ there exists a complete upper approximation for Q using $\mathcal{V}$ iff $V_Q = \bigcup_{V \in \mathcal{V}} V_{I_Q[V]}$
- ✓ EUA and EUA^c are quadratic time in |Q| and $|\mathcal{V}|$

- Existence of lower approximation (ELA)
- Existence of complete lower approximation (ELA^c)

For a simulation query Q and a set $\mathcal{V}$ of views,

- ✓ there exists a complete lower approximation for Q using $\mathcal{V}$ iff $E_Q \subseteq \bigcup_{V \in \mathcal{V}} E_{I_{\hat{Q}}[V]}$
- ✓ ELA^c is in $O(|\mathcal{V}||Q|^2)$ time
- ✓ ELA is NP-complete

Here $\hat{Q}$ is the complete graph of Q.

Closest approximation

- Closeness: the closeness clo(Q',Q) of Q' and Q, is the number of edges in Q' and Q that are not in the edge-induced maximum common subgraph of Q' and Q.

- Closest upper approximation (CUA): Given a simulation query Q and $\mathcal{V}$, find the upper approximation Q_u that is closest to Q, i.e. for any other Q'_u, $clo(Q_u, Q) \leq clo(Q'_u, Q)$.
- Complete closest upper approximation (CUA^c)
- Closest lower approximation (CLA)
- Complete closest lower approximation (CLA^c)

For a simulation query Q and a set $\mathcal{V}$ of views,

- ✓ CUA and CUA^c are quadratic time in |Q| and $|\mathcal{V}|$.
- ✓ DCLA and $DCLA^c$ are NP-complete.
- ✓ OCLA and $OCLA^c$ are not in APX.

Algorithms for problem EUA, EUA^c, ELA, ELA^c, CUA, CUA^c, CLA, CLA^c

Computing upper and lower approximation

Algorithms for upper approximation

- Algorithm CUASim^c and EUASim^c is in $O(|Q||\mathcal{V}|+|V_Q|^2)$ time
- Algorithm CUASim and EUASim is in $O(|Q||\mathcal{V}|)$ time

Algorithms for lower approximation

- Algorithm ELASim^c is in $O(|\mathcal{V}||Q|^2)$ time
- Algorithm CLASim^c is a $\max_{e \in E_{\hat{Q}} \setminus E_Q} \mathsf{occ}(e) \cdot \ln(\max_{\mathsf{V} \in \mathcal{V}} |E_{I_{\hat{Q}}[\mathsf{V}]} \cap E_Q|)$ - approximation algorithm that always returns a complete lower approximation of Q w.r.t. $\mathcal{V}$ in $O(|\mathcal{V}||Q|^2)$ time when there exists one
- Algorithm CLASim and ELASim is a heuristic algorithm runs in $O(|\mathcal{V}||Q|^2)$ time

Extending to subgraph queries

Approximation for subgraph queries

- Existence of approximation

For a subgraph query Q and a set $\mathcal{V}$ of views

- ✓ there exists an upper approximation for Q using $\mathcal{V}$ iff there exists V ∈ $\mathcal{V}$ such that the match result V(Q) ≠ ∅
- ✓ there exists a complete upper approximation for Q using $\mathcal{V}$ iff $V_Q = \bigcup_{\mathsf{V}\in\mathcal{V}} V_{I_Q[\mathsf{V}]}$
- ✓ there exists a complete lower approximation of Q using $\mathcal{V}$ iff $E_Q \subseteq \bigcup_{\mathsf{V}\in\mathcal{V}} E_{I_{\widehat{Q}}[\mathsf{V}]}$
- ✓ problems EUA, EUA^c, ELA, ELA^c are all NP-complete

- Closest approximation

For a subgraph query Q and a set $\mathcal{V}$ of views

- ✓ pattern graph $Q_u(\bigcup_{\mathsf{V}\in\mathcal{V}} V_{I_Q[\mathsf{V}]}, \bigcup_{\mathsf{V}\in\mathcal{V}} E_{I_Q[\mathsf{V}]})$ is the closest upper approximation of Q
- ✓ if $\bigcup_{\mathsf{V}\in\mathcal{V}} V_{I_Q[\mathsf{V}]} = V_Q$, $Q_u^c(\bigcup_{\mathsf{V}\in\mathcal{V}} V_{I_Q[\mathsf{V}]}, \bigcup_{\mathsf{V}\in\mathcal{V}} E_{I_Q[\mathsf{V}]})$ is the complete closest upper approximation of Q
- ✓ problems CUA, CUA^c, CLA, CLA^c are all NP-complete

- Subgraph query answering using views

- ✓ Q can be answered using $\mathcal{V}$ iff $E_Q = \bigcup_{\mathsf{V}\in\mathcal{V}} E_{I_Q[\mathsf{V}]}$
- ✓ it is NP-complete to decide whether Q can be answered using $\mathcal{V}$

Experimental study

Experimental study

- Experimental setting
- ✓ Real-life graphs: (a) DBpedia: (4.43M, 8.43M) (b) YouTube: (2.03M, 12.22M)
- ✓ Views: designed of sizes (2,1), (3,2), (4,3), (4,4), and varied structure of same sizes view answers in total take 32.58% of DBpedia dataset, and 34.29% of YouTube.
- ✓ Algorithms: CUAsim^c, CUAsim, CLAsim^c, CLAsim, CUAiso^c, CUAiso, CLAiso^c, CLAiso;

  QAViso, QAVsim*; gSim*,VF2*;

- Experimental results
- ✓ Percentage of queries approximable using views:
  - 25% of views are used, 65% (53%) simulation (subgraph) queries on DBpedia
  - 75% of views are used, 88% (81%) simulation (subgraph) queries on YouTube
- ✓ Accuracy of approximation using views:
  - three measures: F-measure (F), strong F-meadure (F_s), weak F-measure (F_w)
  - achieve accuracy (F_w) above 0.79 and 0.86 in all cases.
- ✓ Speed up of approximation using views:
  - scale with million graphs within 0.24*s* and 2.7*s* for simulation and subgraph queries, when it takes 42*s* and 5382*s* to evaluate queries directly.

Summing up

Approximating Graph Pattern Queries Using Views

- A framework of query-driven approximation using views (simulation and subgraph)

Given a pattern query Q and views $\mathcal{V}$,

a) first check whether Q can be exactly answered using $\mathcal{V}$, using algorithm QAViso and QAVsim*;

b) if so, generate query plans that exactly answer Q using $\mathcal{V}$ only, by algorithm QAViso and QAVsim*;

c) otherwise, check whether there exist upper and lower approximations of Q, and find the closest approximations Q_u and Q_l if exist, via algorithm CUAsim^c, CUAsim, CLAsim^c, CLAsim, CUAiso^c, CUAiso, CLAiso^c and CLAiso.

d) generate query plans that answer Q_u and Q_l by using $\mathcal{V}$ only, via algorithm QAViso and QAVsim*.

- Conclusion

✓ A notion of upper and lower approximation of pattern queries w.r.t. a set of views

✓ The properties and characterizations

✓ Eight fundamental problems for approximating using views, complexity, approximation-hardness

✓ Efficient exact algorithms, approximation algorithms, effective heuristic algorithms for computing closest/existence of, complete/general, upper/lower approximations, for simulation/subgraph queries

✓ Experimental results have verified the effectiveness and efficiency of techniques and framework

Thanks!