Ontology-based Subgraph Querying

Yinghui Wu, Shengqi Yang, Xifeng Yan
{yinghui, sqyang, xyan}@cs.ucsb.edu
Dept of Computer Science, University of California, Santa Barbara

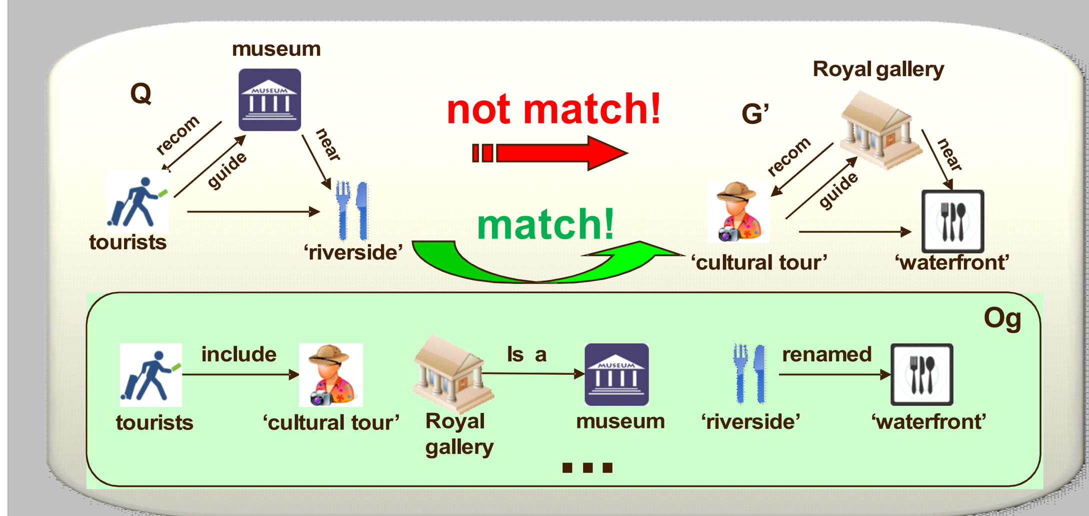
Introduction

- Traditional subgraph querying requires identical label matching, which is too restrictive to capture **semantically related matches** for query graphs. Moreover, it is nontrivial to capture the semantic similarity using the query graph and data graph alone.
- We introduce ontology-based subgraph matching, a revision of the traditional subgraph queries by enabling ontology graph-based searching.
- We introduce a metric to measure the similarity of the matches. We propose a querying framework to find top K best matches in terms of the metric. This framework is also flexible enough to incorporate prior knowledge for edge-typed subgraph matching, as well as approximate structural matching. It can also be efficiently maintained to deal with dynamic data graphs.

Ontology-based Subgraph Matching

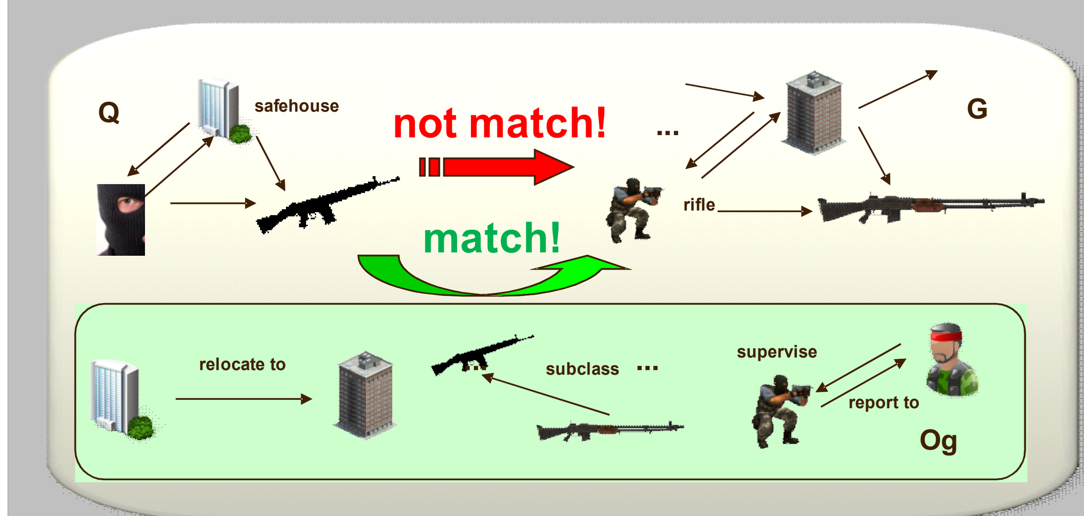
- **Traditional subgraph querying vs. subgraph querying using ontology**

Q: "*find tourists who recommend a museum with guide service, and favor restaurant 'riverside' close to the museum*."



A: *We found 'cultural tour' group who recommend royal gallery with guide service. They like a nearby restaurant 'waterfront' which used to be the 'riverside'.*

Q: "*find suspect A using class I gun in safehouse A*"



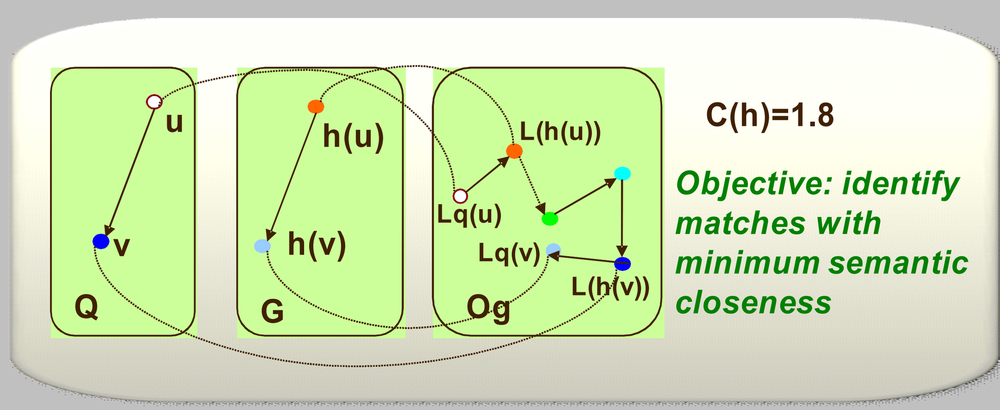
A:"we identify a suspect [who report to A] using class II guns [a type of class I guns] in a [relocated] safehouse B"

- **Ontology-based subgraph querying**
  - An ontology graph represents ontologies and their relationships
  - A similarity function **sim**(.) calculates ontology similarities
  - Ontology-based subgraph querying is to find subgraphs isomorphic to query graphs, where nodes are matched w.r.t ontology similarity sim(.)

Quality measurement

- Semantic closeness C(h) for a mapping function h from Q to G

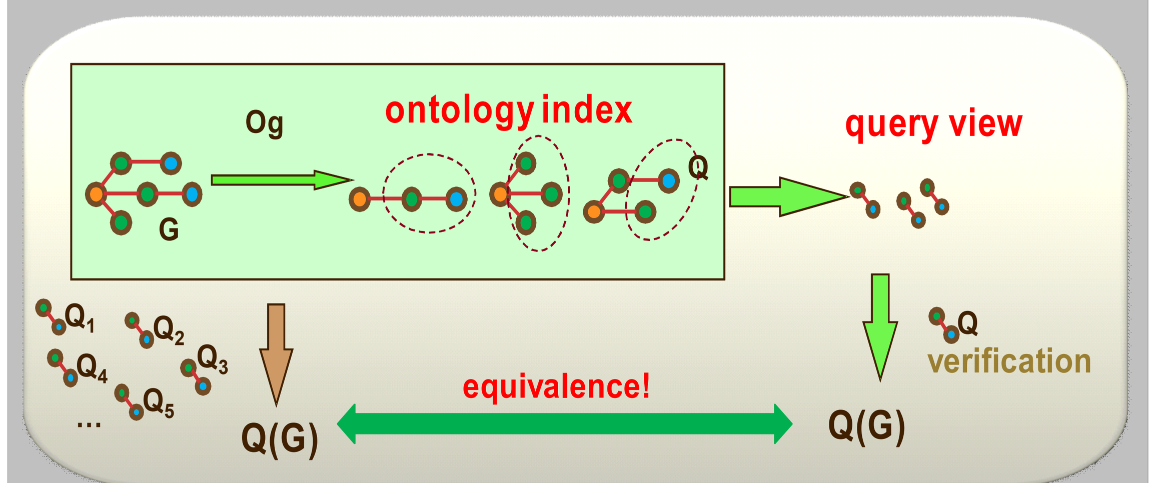
$$C(h)=\Sigma\ sim(L_q(u), L(h(u))),\ u \in V_q$$



- Top-K ontology-based subgraph query: identify k matches specified by h that maximizes C(h)

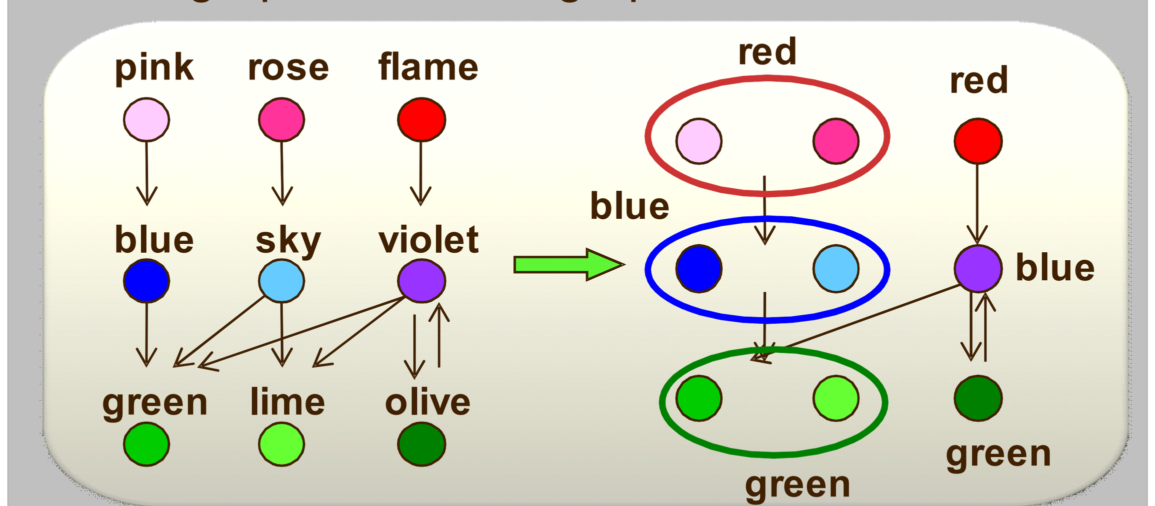
Querying framework

- construct ontology index as a set of concept graphs of G, by summarizing G using the ontology graph Og
- use concept graphs to evaluate Q via filtering-and-verification process

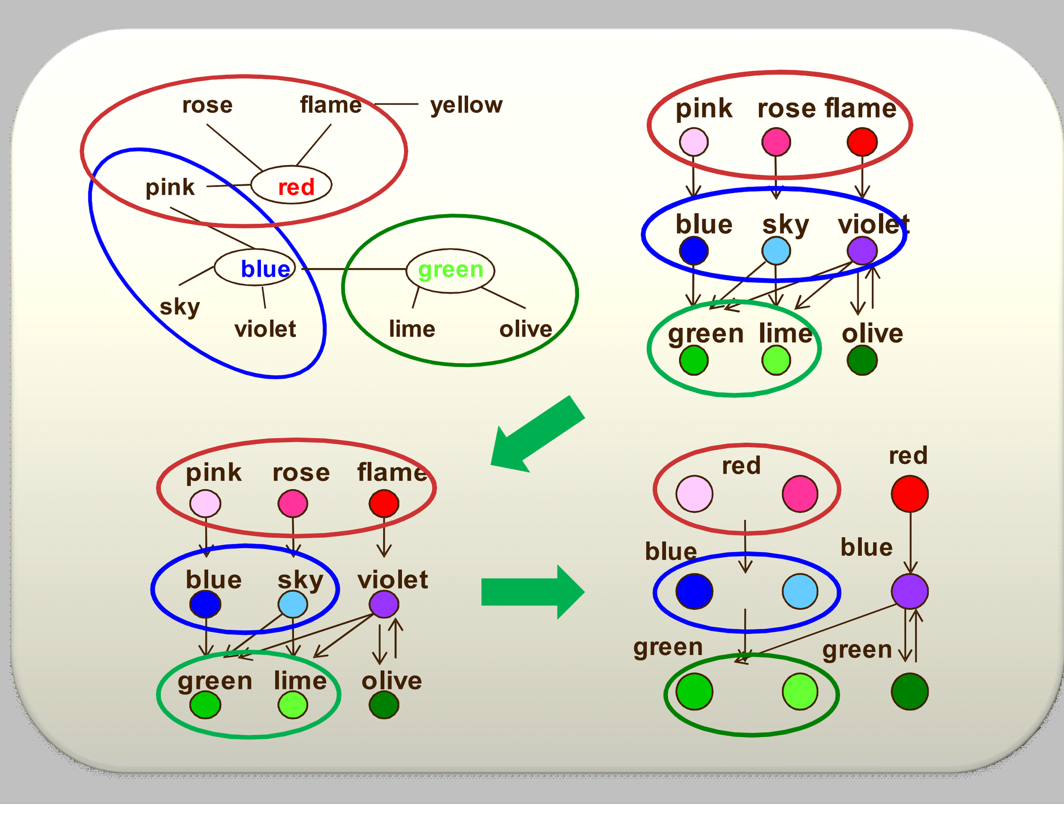


Ontology index

- a concept graph is an abstraction of data graph G, where each node represents a group of nodes having similar label to the same label in ontology graph, and each edge preserves node connection relation.
- An ontology index is a set of concept graphs of a data graph G

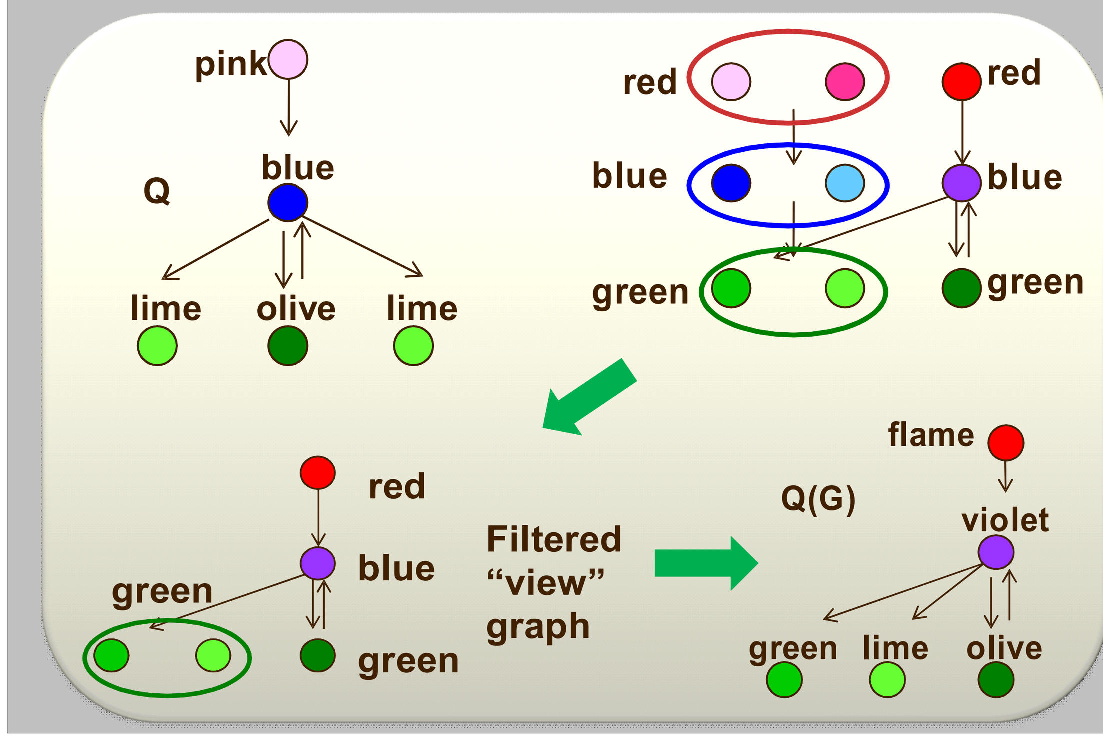


- An algorithm to construct ontology index
  - Partition the ontology graph O
  - Select a concept label in each cluster; repeats until all labels in O are "covered"
  - Construct a partition of G by grouping nodes with labels similar to concept labels
  - Iteratively refine the partitions



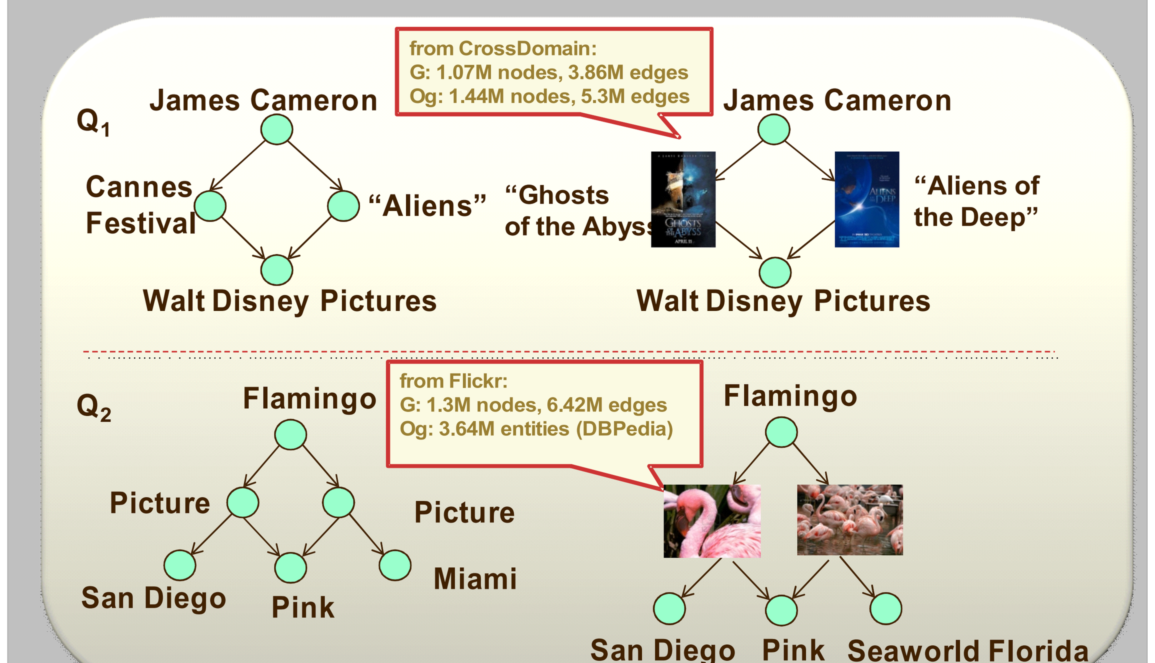
Filtering and verification

- Matching: select candidates for each query node in Q (using a lazy strategy); compute a matching relation M from Q to each concept graph Gc;
- Subgraph extraction: compute intersection of the matches M from Q to each Gc; return the induced subgraph Gv
- Verification: extract top-K matches from Gv

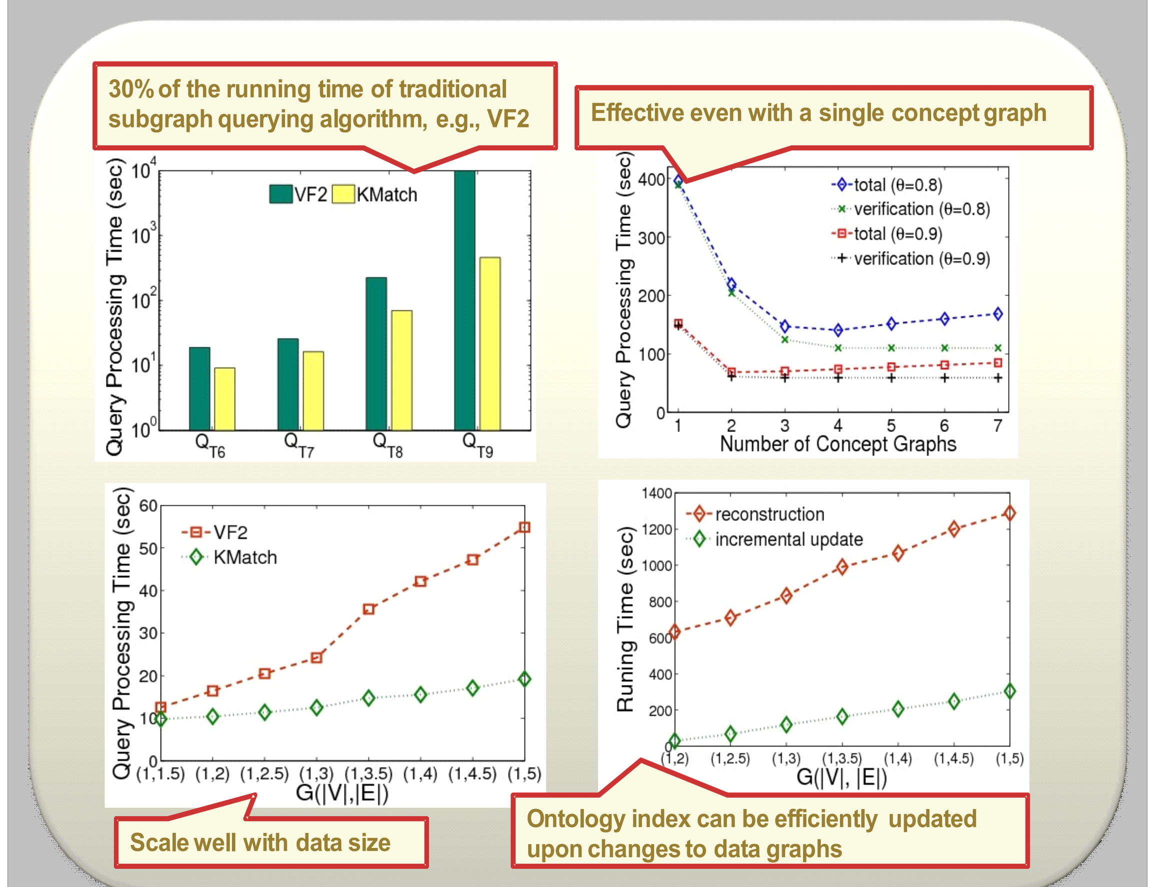


Experimental Study

- Effectiveness: ontology-based querying over real-life graphs



- Efficiency: improving traditional subgraph isomorphism algorithms by 70%, and can be efficiently maintained



Conclusion

- Traditional graph matching is too restrictive to identify "hidden matches". Ontologies enables subgraph querying to identify semantically related matches effectively.
- Our ontology-based querying framework is efficient, scale well with the graph size and query size, and can be efficiently maintained.
- A good source of future work includes: (1) extending the techniques for various queries and similarity measurements; and (2) ontology-based query suggestion and interpretation.