



SQulD: Semantic Similarity-Aware Query Intent Discovery



Anna Fariha

afariha@cs.umass.edu

University of Massachusetts Amherst

Sheikh Muhammad Sarwar

smsarwar@cs.umass.edu

University of Massachusetts Amherst

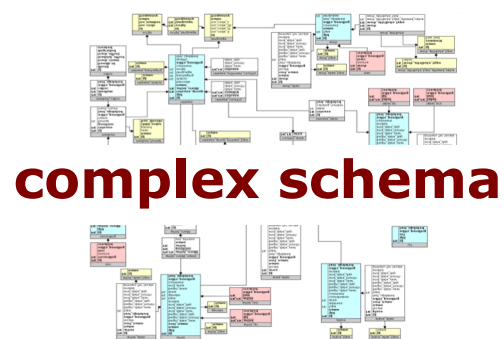
Alexandra Meliou

ameli@cs.umass.edu

University of Massachusetts Amherst

Traditional data retrieval is challenging

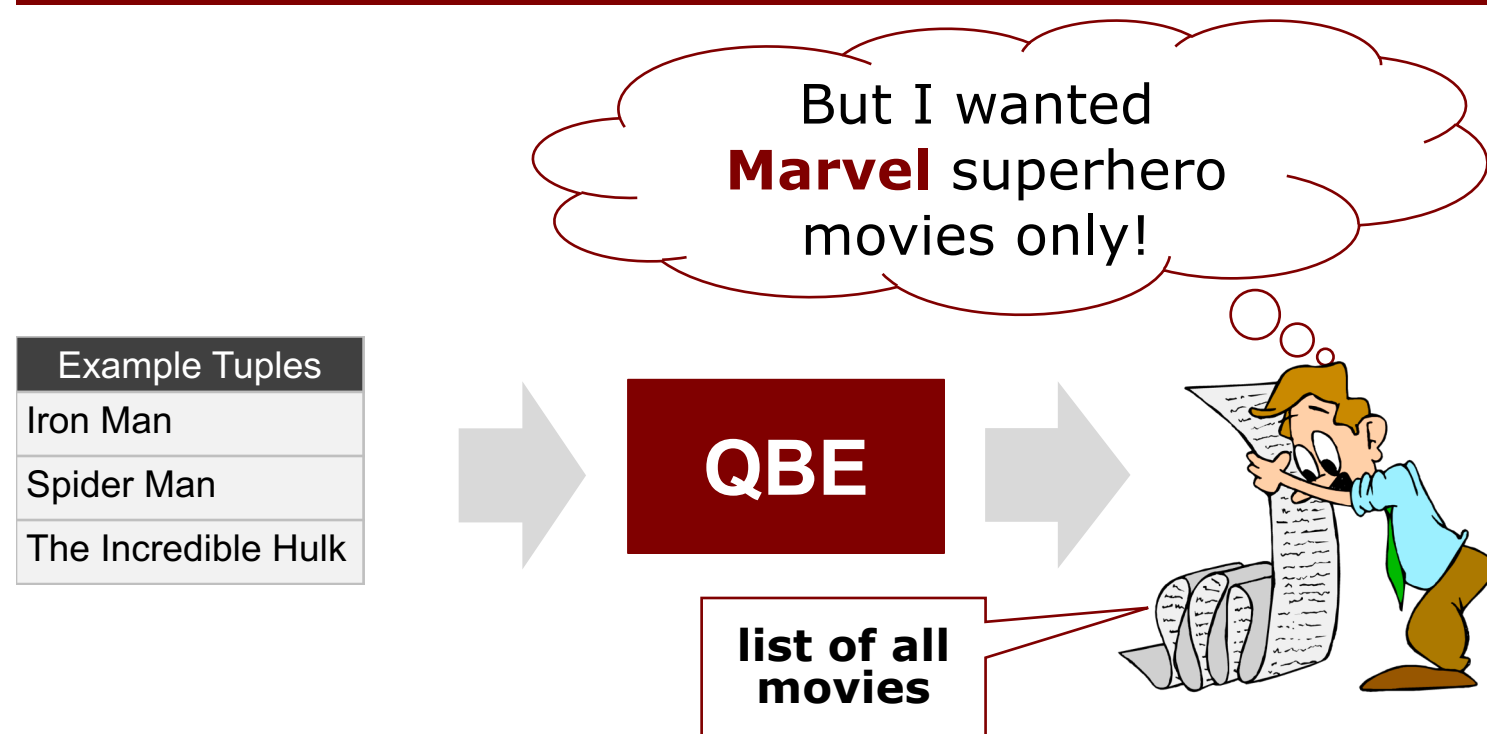
- ❑ Non expert users struggle to formulate complex SQL query



```
SELECT person.name
FROM person, castinfo, movietoggenre, genre
WHERE person.id = castinfo.person_id
AND genre.name = 'Comedy'
GROUP BY person.id
HAVING count(*) >= 40
```

no SQL expertise

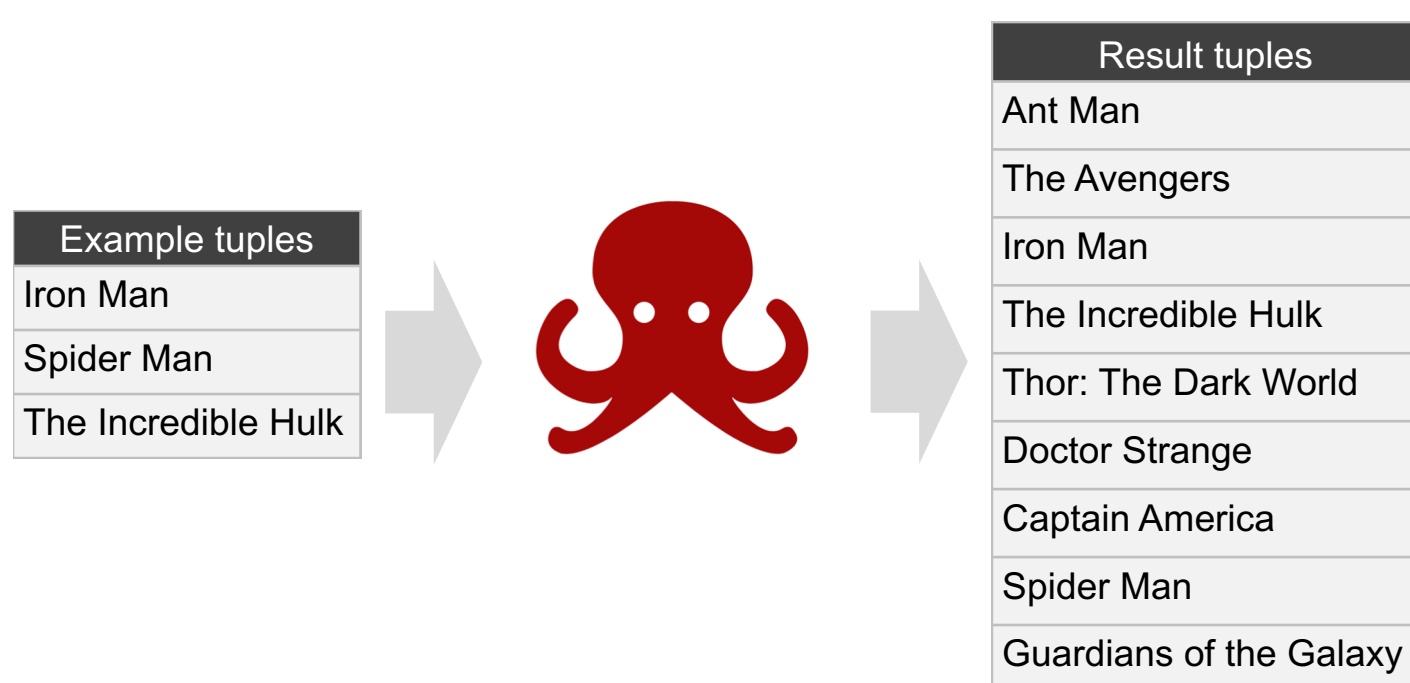
Query by Example (QBE) tries but fails



- ❑ Existing QBE approaches fail to capture semantic context (e.g., Marvel movies)

SQulD: a query intent discovery framework

- ✓ SQulD is aware of semantic similarity



Not all semantic similarities are intended

What are the intended semantic similarities?

- ✗ All are Male
 - but, so are 2 million other people
- ✗ All are Hollywood actors
 - but, so are 1.5 million other people
- ✓ All are **Funny Actors**
 - **very few** actors are funny

- ✓ SQulD rejects irrelevant semantic context

Semantic context can be complex

How to express **funny**?

- ❑ **Implicit** context
- ❑ Humans can tell easily!

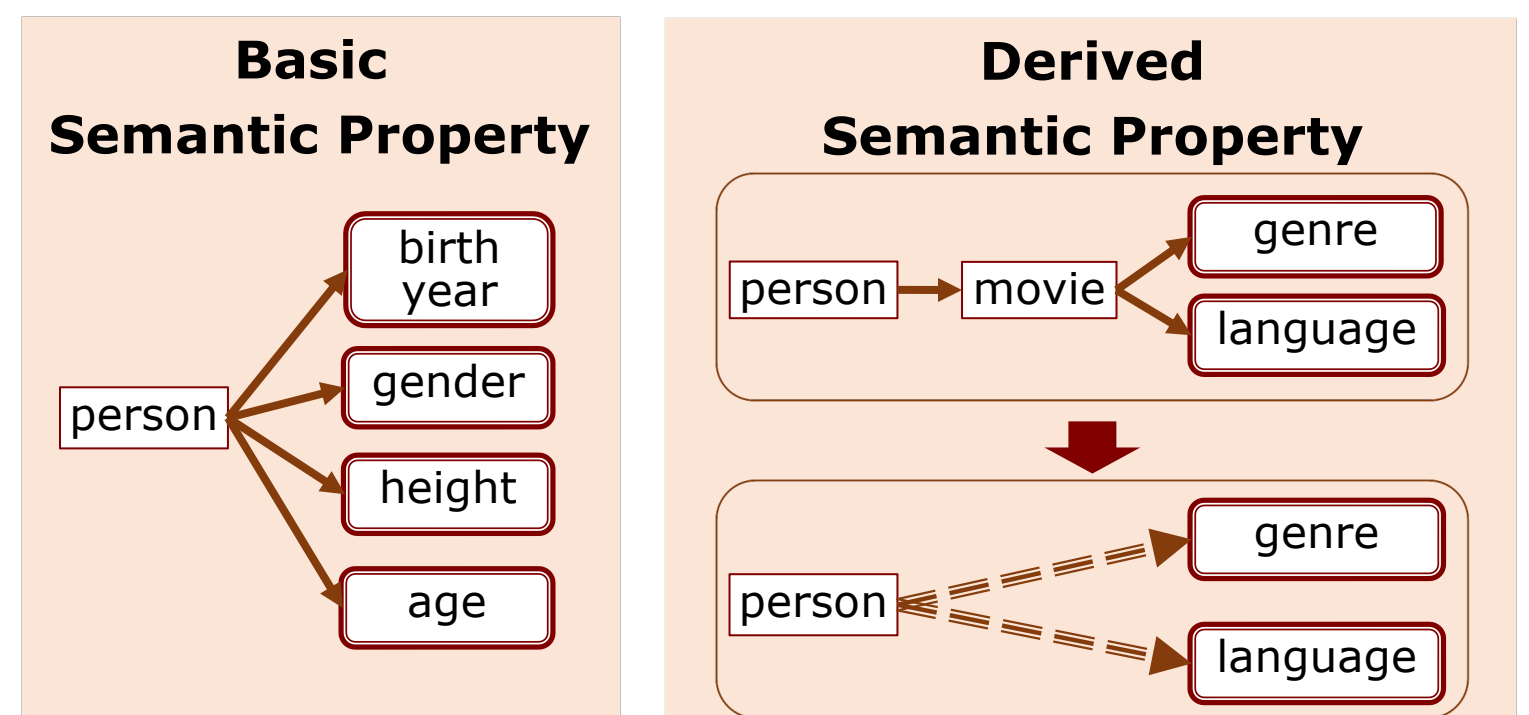
- ✓ SQulD captures implicit semantic context

Semantic context: **funny actors**

- ❑ No explicit attribute, but **hidden** in the data
- ❑ Appearing in **more than 40 comedy movies**

SQulD in a nutshell

- ❑ **Real-time performance** through precomputing basic and derived semantic properties



- ❑ **Filters** encode semantic properties and constitute selection predicates

```
SELECT name
FROM person, person_to_genre pg, genre
WHERE person.id = pg.person_id AND
pg.genre_id = genre.id AND
genre.name = "Comedy" AND pg.count >= 40
```

Architecture

- ❑ **Offline: Precomputes** derived semantic properties and related statistics for real-time performance
- ❑ **Online:** Discovers semantic context, captures **most likely query intent**, and constructs SQL query

