W8. Lecture Notes — By Junyi

Summary
- Knowledge Graphs have a life-cycle
•Must evolve over a period of time
•Must address both social and technical concerns

- Techniques and algorithms
•Schema evolution
•View maintenance
•Truth maintenance

- Overview
  - change
    - in the real-wrold
    - in the business requirements
  - → revise
    - schema
    - ground facts
  - → approaches
    - technical challenges
    - social challenges
- examples requiring change
  - Changing world: Amazon Product Knowledge Graph
  - Changing requirements: Google Knowledge Graph
  - Changing sources (Google Knowledge Graph)
  - Changes affecting previous inferences:

- Consider the constraint that a movie theater only shows movies
  •Using this constraint a KG might have previously inferred that certain events are movies
- More recently the movie theaters are being used for operas, and social events
  •If we had previously derived such events to be movies, we must update them

▼ Changes requiring redesign

▼ change management techniques

▼ schema evolution

▼ For a relational database

▼ Adding/removing a column, renaming an attribute

▼ For a knowledge graph

Approach is to maintain invariants, and make system-specific decisions

▼ Adding/removing a class

▼ Adding/removing a superclass

▼ Adding/removing a property

▼ Adding/removing a constraint

▼ view maintenance

A mechanism from databases to name a query
•Query is defined with respect to one or more tables (known as base tables)
•If we store the results of the query, the stored data is called materialized view

▼ If the base data changes, the materialized view must be updated

▼ truth maintenance

A mechanism from rule-based systems
•Tracks how each conclusion was derived

▼ A popular implementation: Justification based system

•Each derived conclusion records the fact or rule that was used in derivation
•Any time that fact or rule updates, the conclusion must be revised

▼ summary