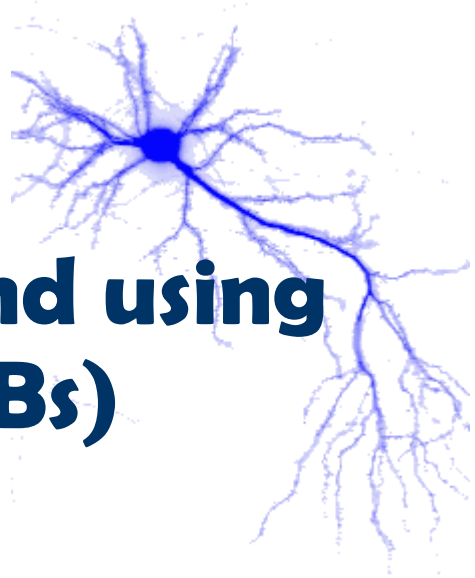


Building, maintaining and using Knowledge Bases(KBs)

Suravi saha roy
Department of computer science
University of Helsinki

Knowledge Bases(KBs)

- Concept taxonomy
- Instances
- Relationships

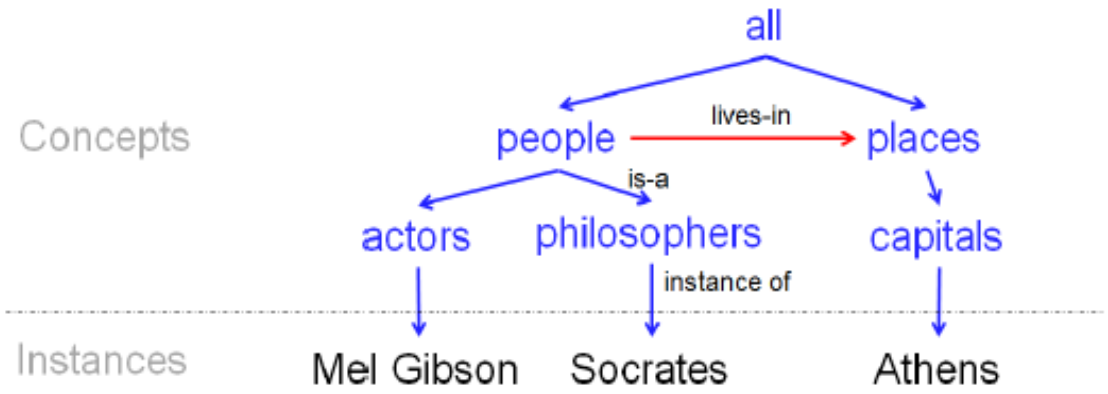
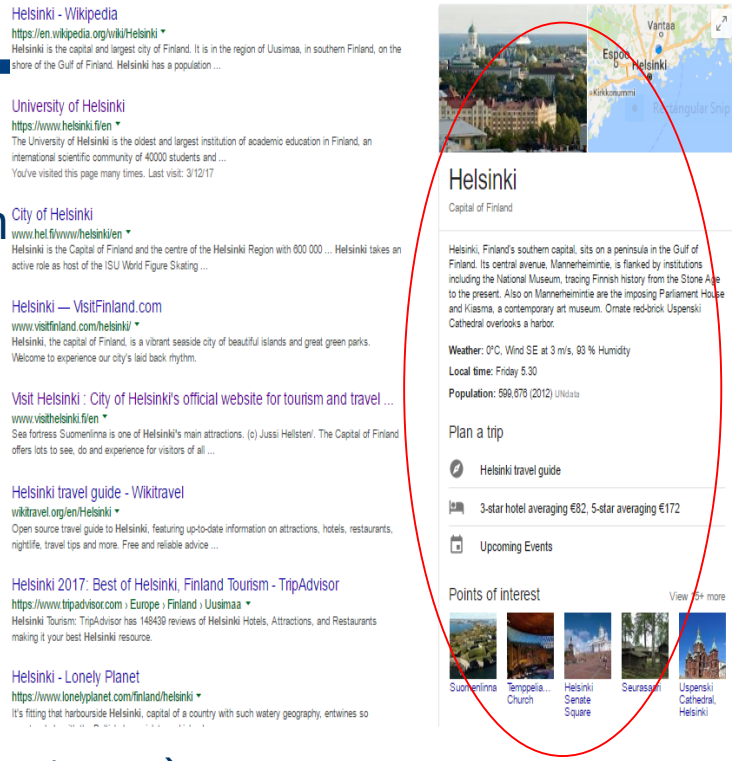


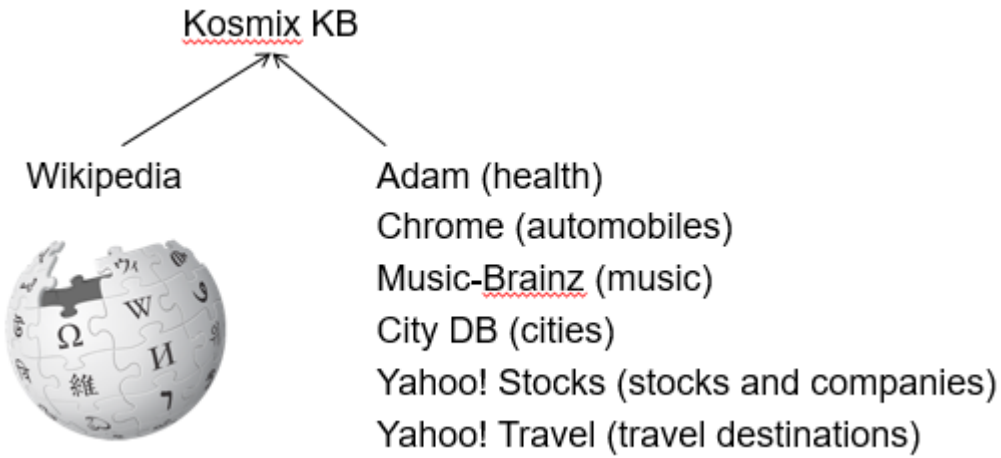
Figure 1: A tiny example of a KB

Increasingly Critical to a Wide Variety of Applications



- General search
  - Google search using Knowledge Graph
- Product search
  - Walmart.com, Amazon.com
- Question answering
  - IBM Watson, Apple Siri
- Advertising
- Information extraction
- Deep Web search
- Recommendation, playlisting, music(echonest.com)
- Biomedical expert finding (knode.com)
- Social commerce & media analysis (event discovery, event monitoring)

Example Knowledge Base: Kosmix KB



- 6.5M concepts, 6.7M concept instances, 165M relationship instances
- 23 verticals, 30G of disk space
- First built around 2005 at Kosmix
  - for Deep Web search, advertising, social media analysis
- Has been significantly expanded at WalmartLabs since 2011
  - for product search, social commerce, mining of social media, understanding Web data

State of the Art

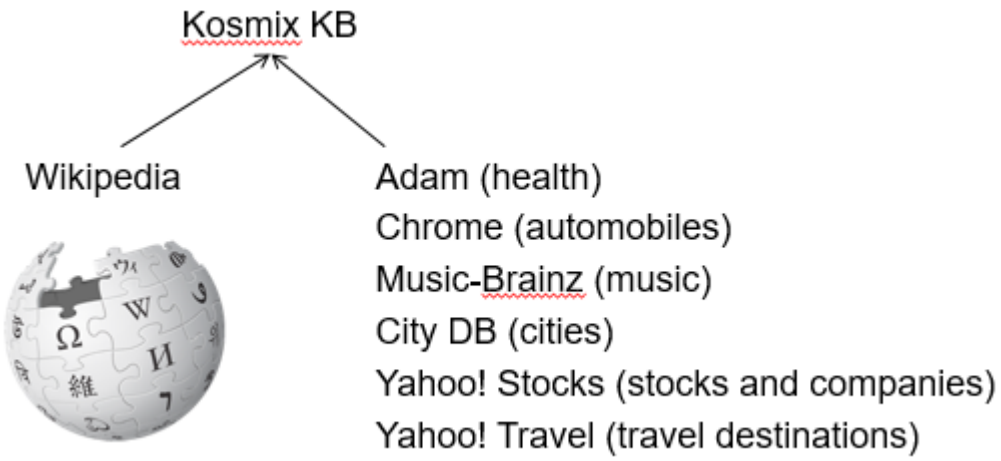
- Significant & growing interest in academia and industry for a wide variety of applications
- Important for Big Data
  - Big Data needs big semantics, which often come in form of large KBs
- But little has been published about building, maintaining, using KBs
- Current works have addressed only isolated aspects:
  - Initial construction, data representation, storage format, query APIs
- No work has addressed the end-to-end process
- This work: end-to-end process of building, maintaining, using Kosmix KB
  - How to maintain the KB over time?
  - How to handle human feedback?
  - How to integrate various data sources?
  - What kinds of applications is a not-so-accurate KB good for?
  - How big of a team is required to build such a KB? What should the team do?

Key Distinguishing Aspects of Kosmix KB

- Building the KB
  - started with Wikipedia, added many more data sources
  - extracting a KB from Wikipedia is non-trivial, use Web and social data / curation to guide the process
  - adding a lot of social / Web metadata to KB nodes
- Updating the KB
  - rerun from scratch instead of incremental updating
  - must reuse human curation
- Curating the KB
  - ongoing process, regularly evaluate the KB
  - add curations in form of commands which enable reusing of human curation can curate multiple errors all at once

Building the Kosmix KB

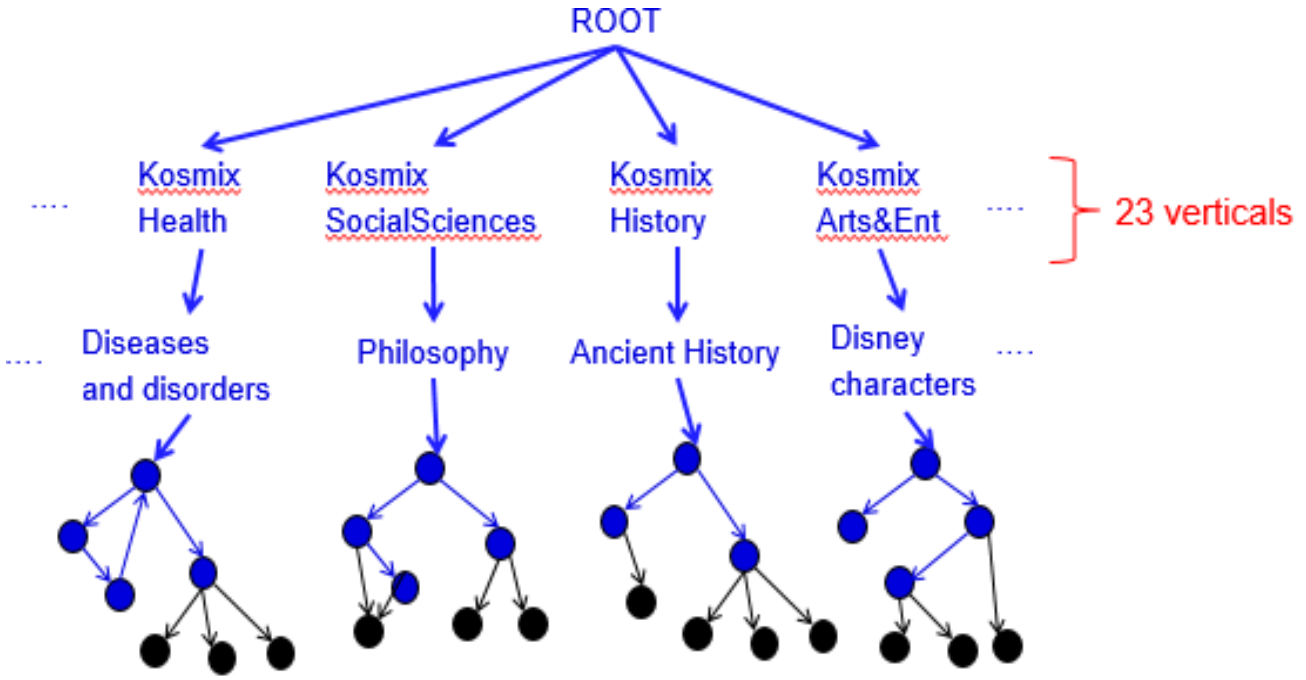
- **Convert Wikipedia into a KB, then add more data sources**



- **Why starting with Wikipedia?**
  - must process social media because it often mentions latest events/persons so we need them to be in our KB asap
  - Wikipedia is ideal for this
    - e.g. very soon after Malala Yousafzai became famous in 2012, Wikipedia had a homepage for her.

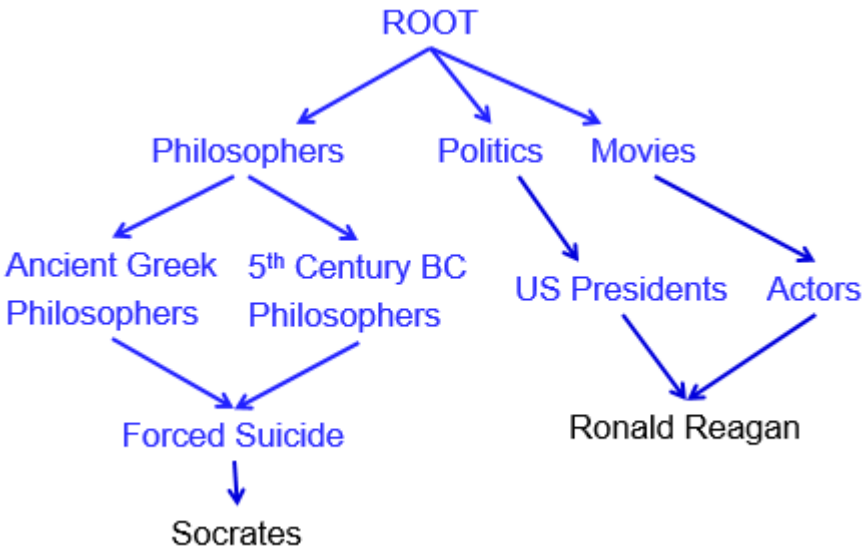
1. Convert Wikipedia into a Graph

- Crawl Wikipedia, parse & construct a graph
  - nodes = Wikipedia pages, edges = links among Wikipedia pages
- Remove irrelevant parts of graph
  - administration, help, discussion
- Add remaining parts into a new graph with a ROOT node



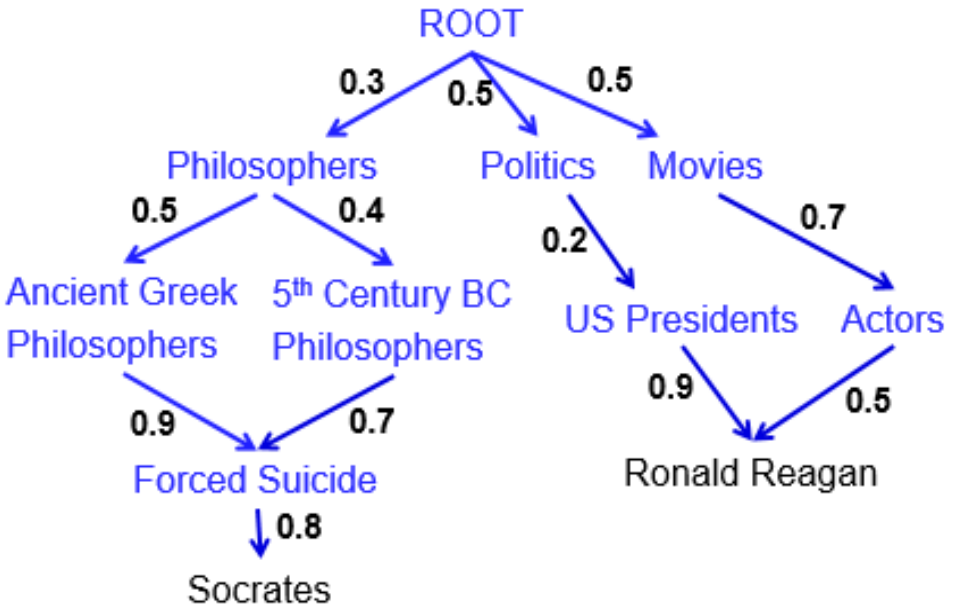
2. Extract Taxonomy of Concepts from Graph

- To obtain taxonomic tree
  - for each node, find a single path to ROOT
- But nodes can have multiple paths to ROOT
  - which one to pick?
- Picking wrong path causes many problems
  - e.g. ROOT ➔ Movies ➔ Actors ➔ Ronald Reagan
    "Reagan left a mixed legacy": will be classified incorrectly under "Movies"



2. Extract Taxonomy of Concepts from Graph (continued)

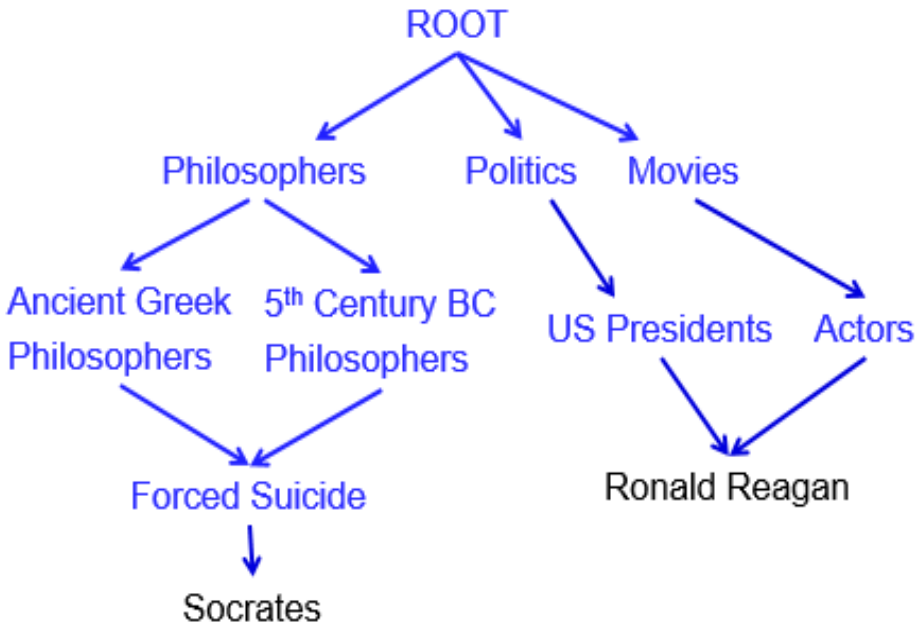
- Intuitively, pick most popular/important/relevant path
  - e.g. most people know Reagan as a president, not as an actor
- Solution:
  - assign to each edge A➔B a weight to capture its popularity/importance/relevance
  - run a spanning tree discovery algorithm using these weights
  - output a maximum spanning tree



2. Extract Taxonomy of Concepts from Graph (continued)

- How to assign weights to edge A➔B?
  - assign multiple weights, they form a weight vector
- Examples
  - Web signal: co-occurrence count of A and B on the Web
    - e.g. how many times “Ronald Reagan” and “President” co-occur in same Web page?
  - Social signal: same as Web signal, but measure co-occurrence in social media
  - List signal: how many times A and B co-occur in the same Wikipedia list?
  - Similarity in the names of the two nodes
    - e.g. “Actors” and “Actors by Nationality”
  - analyst can also assign weights to the edges

2. Extract Taxonomy of Concepts from Graph (continued)

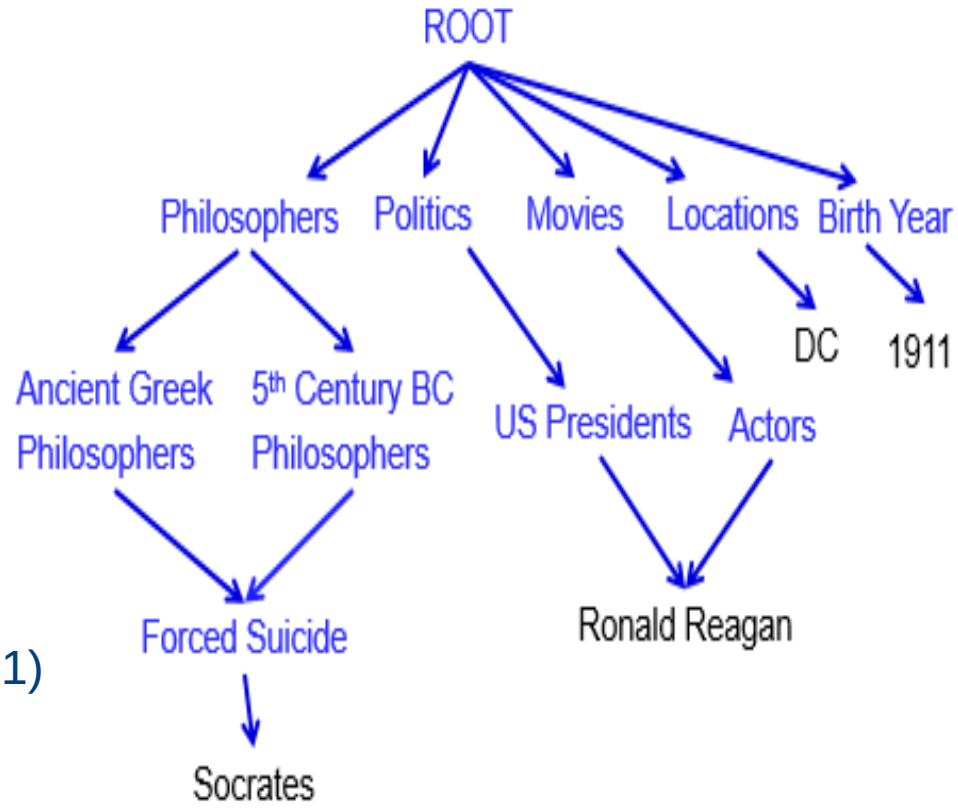


- It keeps all paths for the nodes
  - very useful for applications
- To keep all paths, must detect and break cycles
- End result: DAG of concepts + taxonomic tree imposed on the DAG

3. Extract Relations for the KB

Typical solution:

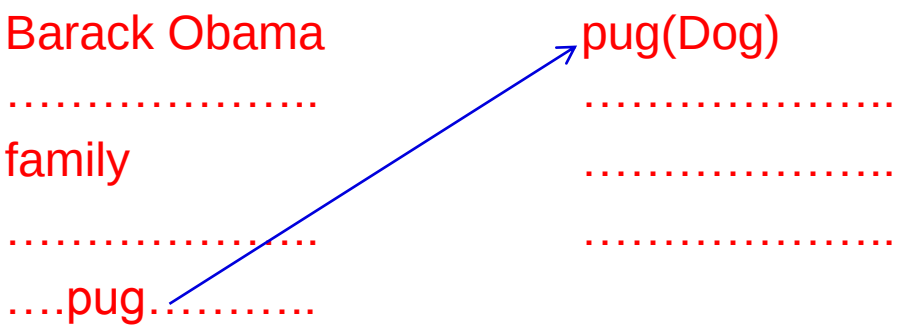
- **Define a set of relations**
  - livesIn, birthYear
- **Write extractors for them**
  - using rules
  - machine learning
- **Apply extractors**
  - livesIn(Reagan, DC),
  - birthYear(Reagan, 1911)



- **Problems:**
  - Wikipedia has 10,000+ interesting relations
  - can't manually define and extract all
  - difficult to obtain high accuracy

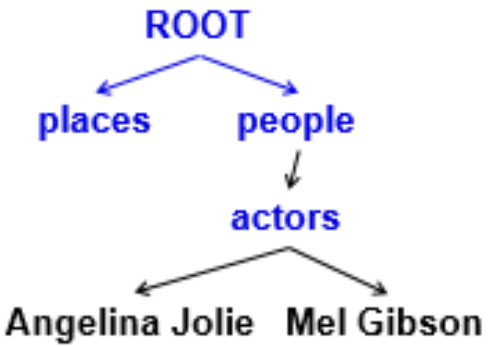
3. Extract Relations for the KB (continued)

- **Our solution: extract fuzzy relations**
- **Extract <Barack Obama, Bo (dog), Family> as a relation**
  - a relation exists between “Barack Obama” and “pug (dog)”, encoded by string “Family”
  - but we don’t know anything more precise



- **Yet this is already quite useful**
- **Example: Querying “Obama family” on a search engine**
  - search query and relations which contains “family”
  - can return “pug(dog)” as an answer
  - even though word “family” never appears in the page “pug (dog)”

4. Extract Metadata for KB Instances



Web URLs

- en.wikipedia.org/wiki/Mel_Gibson
- movies.yahoo.com/person/mel-gibson/
- imdb.com/name/nm0000154/

Twitter ID

- @melgibson

Wikipedia page visits (last day, last week,..)

- 7, 33, …

Web signature

- “actor”, “Hollywood”, “Oscar”, …

Social signature (last 3 hours)

- “car”, “crash”, “Maserati”, …

5.1 Add More Data Sources to the KB

- Challenges:
    1. Match source taxonomy to KB taxonomy
    2. Match source instances to KB instances

- Key innovations:
    1. Interleave taxonomy matching and instance matching
    2. Heavily use node metadata to match instances

5.2 Updating the KB

- Typical solution : Incremental updates
  - fast, relatively easy to preserve human curations
- But difficult in this case
  - They use “global” algorithms (e.g. spanning tree discovery) during KB construction
- Proposed solution
  - Run the pipeline from the scratch daily
  - Challenge: how to preserve human curation?

5.3 Human Curation

- Automatically constructed KB often contains errors
  - automatic version of Kosmix KB is about 70% accurate
  - need human curation
- A human analyst
  - evaluates the quality of the KB and writes curations
- Evaluate quality
  - samples paths and examines their accuracy
  - checks parent assignment for all nodes having at least 200 children
  - gets alerted by developers in case of quality issues
- Curate by writing commands
- Current KB contains several thousand commands (written over 3-4 years)
- Raises the accuracy of the KB to well above 90%

Observation and conclusion

- Possible to build relatively large KBs with modest hardware and team size
- Human curation is important
- An Imperfect relationships still quite useful
  - provide contexts for KB nodes, show how they relate to one another
- Capturing contexts is critical for processing social media
  - especially social contexts
- Important to have clear & proven methodologies to build & maintain KBs as multiple teams try to build their own KBs
- Reference: **Building, Maintaining, and Using Knowledge Bases: A Report from the Trenches** Omkar Deshpande1, Digvijay S. Lamba1, Michel Tourn2, Sanjib Das3, Sri Subramaniam1, Anand Rajaraman, Venky Harinarayan, AnHai Doan1,31@WalmartLabs, 2Google, 3University of Wisconsin-Madison

Thank You