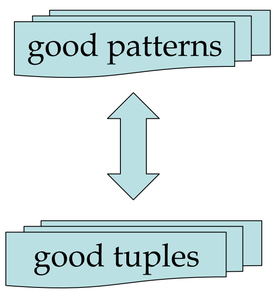
Searching Patterns for Relation Extraction over the Web: Rediscovering the Pattern-Relation Duality

Yuan Fang[1,2] **Kevin Chen-Chuan Chang**[1,2]

[1] University of Illinois at Urbana-Champaign, United States
[2] Advanced Digital Sciences Center, Singapore

Key Finding: *Pattern-Relation Duality (PR Duality)*

Original Intuition [Brin'98]



Given a good set of patterns, we can build a good set of tuples. Given a good set of tuples, we can build a good set of patterns.

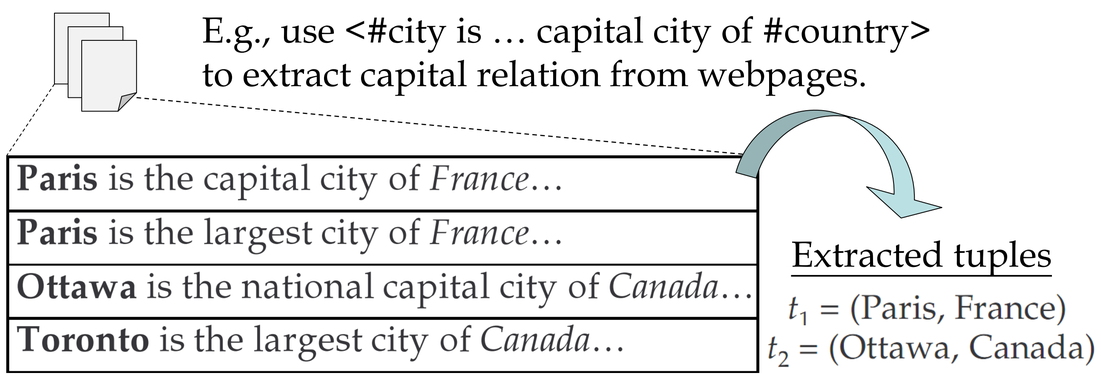
But…
What are considered "good?"
How does the "goodness" mutually reinforce?

Formal Principle

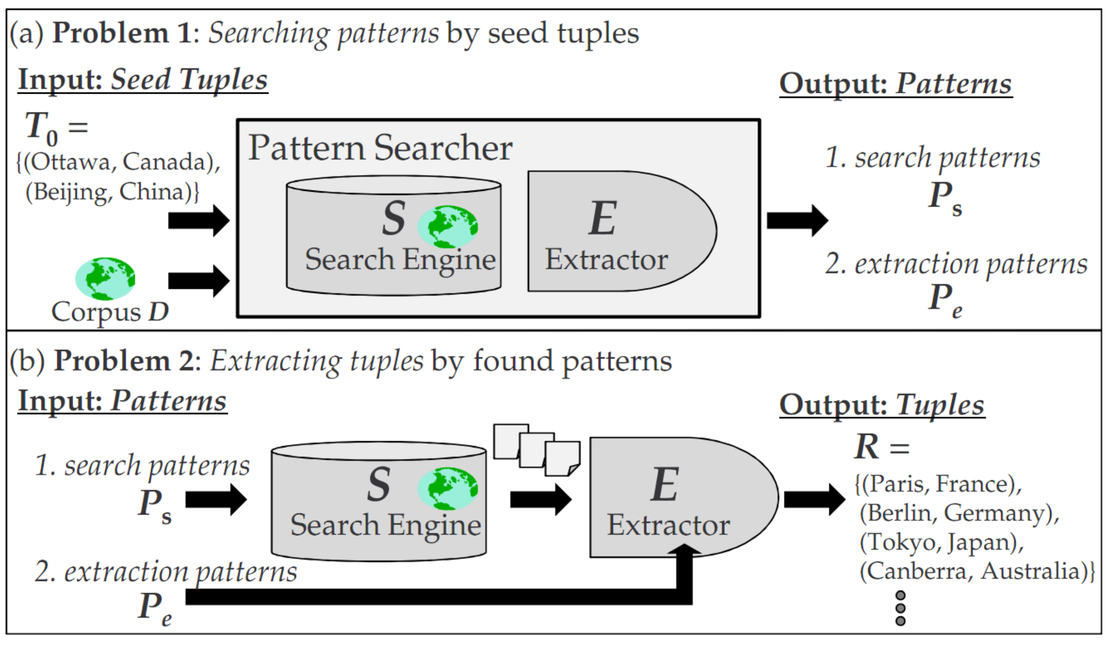
Tuples and patterns for a desired relation R can be qualified by the metrics of **precision** and **recall**, both of which are propagated between matching patterns and tuples–essentially such propagations correspond to random walks on a graph of interrelated patterns and tuples: recall is a **forward walk** from R to tuples and patterns, and precision is a **backward walk** from tuples and patterns to R.

Problem: *Pattern Search*

- **Motivation:** Use syntactic *patterns* to extract *tuples*.



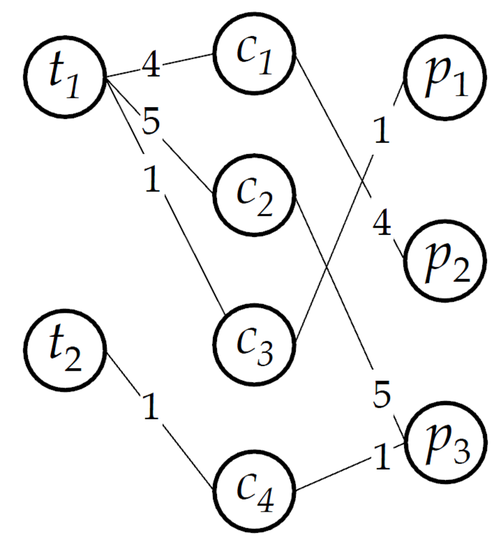
- Dual problems:
  - **How to rank patterns?**
  - **How to rank tuples using the patterns?**



How to Interrelate Patterns and Tuples?

- Need to interrelate them for mutual reinforcement
- Patterns and tuples co-occur in text fragments
- Co-occurring *p* and *t* form a context *c* = (*t*, *p*)
- A context is a particular "interpretation"
  - Whether the pair (*t*, *p*) is relevant or irrelevant
- Contexts thus interrelate tuples and patterns into a Context Graph
  - Which is an affinity graph of semantic relationships

Example Context Graph $G = (T, C, P)$



t_1 = (Beijing, China)
t_2 = (Shanghai, China)

p_1 = #city is … capital of #country
p_2 = #city is an ancient city in #country
p_3 = #city, a big city in #country

t_1 and p_2 co-occur as c_1 in text fragment *"Beijing is an ancient city in China, …"* for 4 times.

t_1 and p_3 co-occur as c_2 in text fragment *"Beijing, a big city in China, …"* for 5 times.

How to Qualify Patterns and Tuples?

- Let C_R be the set of relevant contexts
- View patterns as retrieving a set of contexts I_p
- Deterministic precision and recall:
  $\mathcal{P}(p) = |C_R \cap I_p|/|I_p| \qquad \mathcal{R}(p) = |C_R \cap I_p|/|C_R|$
- Probabilistic precision and recall:
  $\mathcal{P}(p) = \Pr(c \in C_R | c \in I_p) \quad \mathcal{R}(p) = \Pr(c \in I_p | c \in C_R)$
- Can be similarly defined on tuples

How to Propagate the Metrics?

- Probabilistic Inferences between tuples and patterns
- Through contexts (acting as bridges)

1) QuestP: ***Quest Backward for Precision Inference***

$$\textbf{\textit{P1}}: \mathcal{P}(p) = \sum_{t_i \in \tau(p)} \mathcal{P}(t_i) \cdot \frac{|I_{t_i p}|}{|I_p|}$$

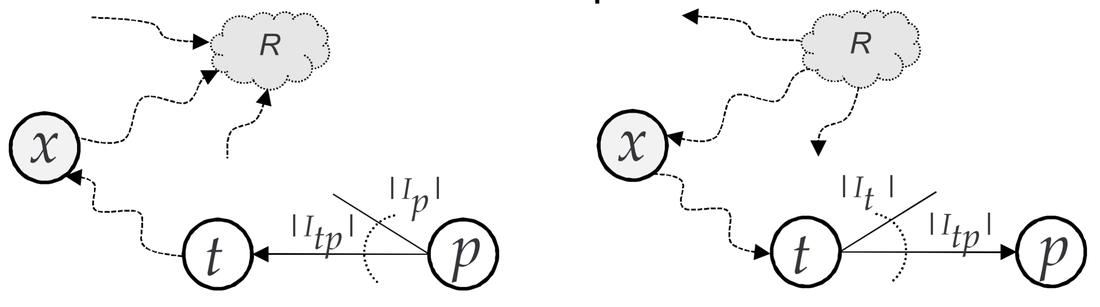
$$\textbf{\textit{P2}}: \mathcal{P}(t) = \begin{cases} \mathcal{P}_0(t) & \text{if } t \in T_0; \\ \mathcal{P}(t) = \sum_{p_i \in \pi(t)} \mathcal{P}(p_i) \cdot \frac{|I_{t p_i}|}{|I_t|} & \text{otherwise.} \end{cases}$$

2) QuestR: ***Quest Forward for Recall Inference***

$$\textbf{\textit{R1}}: \mathcal{R}(p) = \sum_{t_i \in \tau(p)} \frac{|I_{t_i p}|}{|I_{t_i}|} \mathcal{R}(t_i)$$

$$\textbf{\textit{R2}}: \mathcal{R}(t) = \sum_{p_i \in \pi(t)} \frac{|I_{t p_i}|}{|I_{p_i}|} \mathcal{R}(p_i)$$

- The inferences can be interpreted as random walks



(a) *Precision* by backward random walk. (b) *Recall* by forward random walk.

- "Rediscovery" of PR Duality (see our ***Key Finding***)

Experiment: *Our Results*

- Extracting three target relations on the Web
- Baselines: QXtract and Snowball (Q&S)
- Three different schemes of PRDualRank:
  - Dual-Ext (scoring tuples with extraction patterns only)
  - Dual-Sch (scoring tuples with search patterns only)
  - Dual-Combine (average of the above two)