

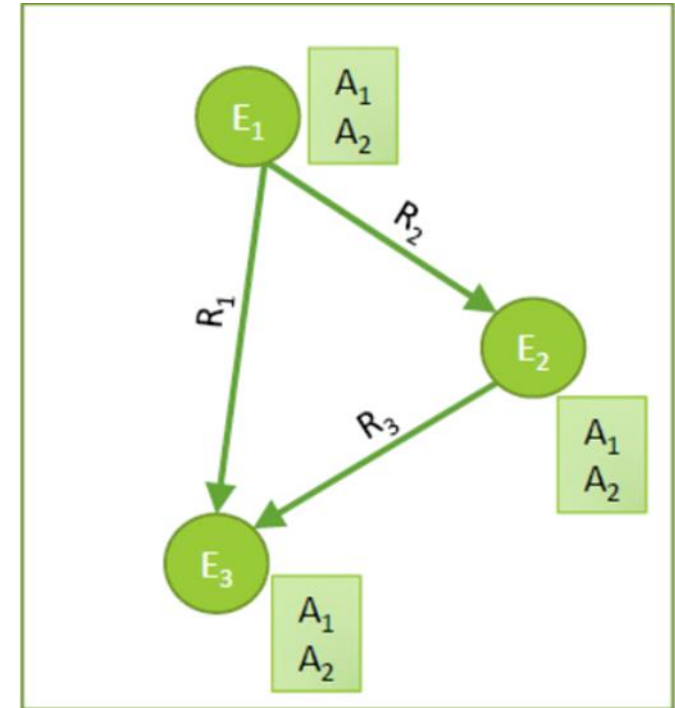
Knowledge Graph

Lecture Outline

- Probabilistic models for Knowledge Graph construction
 - Graphical model based on Probabilistic Soft Logic
 - Random walk model for KG construction

Main Problems

- Who are the entities (nodes) in the graph?
- What are their attributes and types (labels)?
- How are they related (edges)?



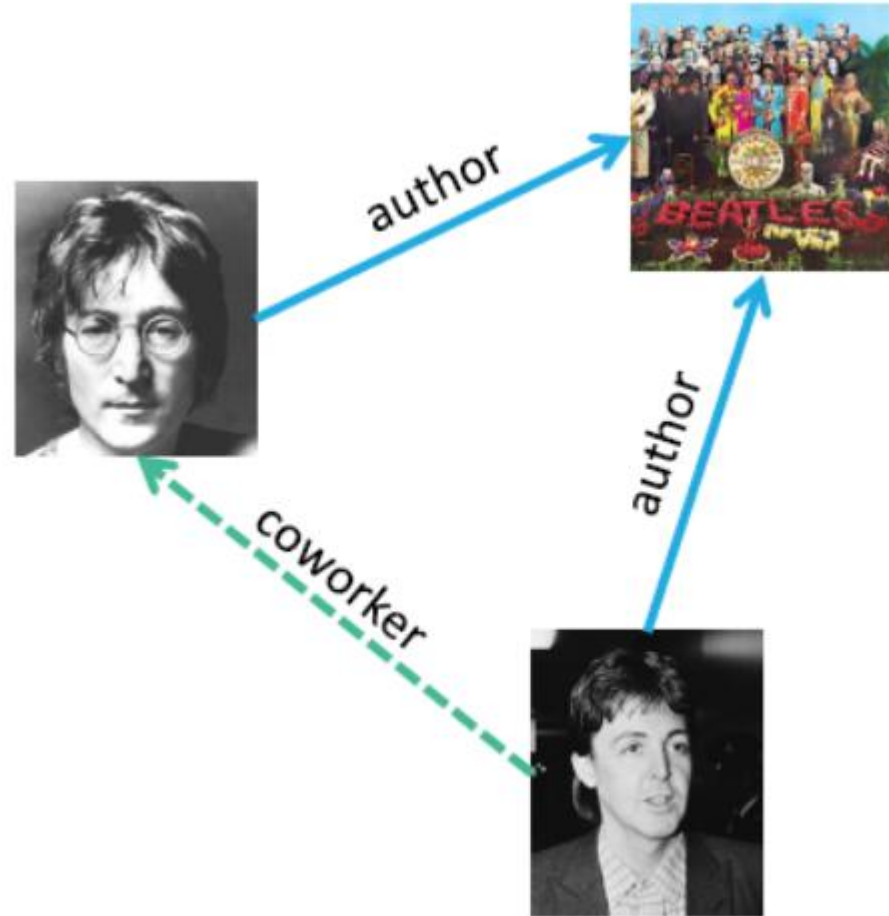
KG Construction: Issues

- Ambiguity
 - Example: citizenOf, livedIn, bornIn
 - Example: Beetles, beetles, Beatles



KG Construction: Issues

- Incomplete facts
 - missing relationships
 - missing labels
 - missing entities



KG Construction: Issues

- Inconsistency of extracted knowledge

- Ex: Cynthia Lennon, Yoko Ono
- Ex: exclusive labels (alive, dead)
- Ex: domain-range constraints



spouse

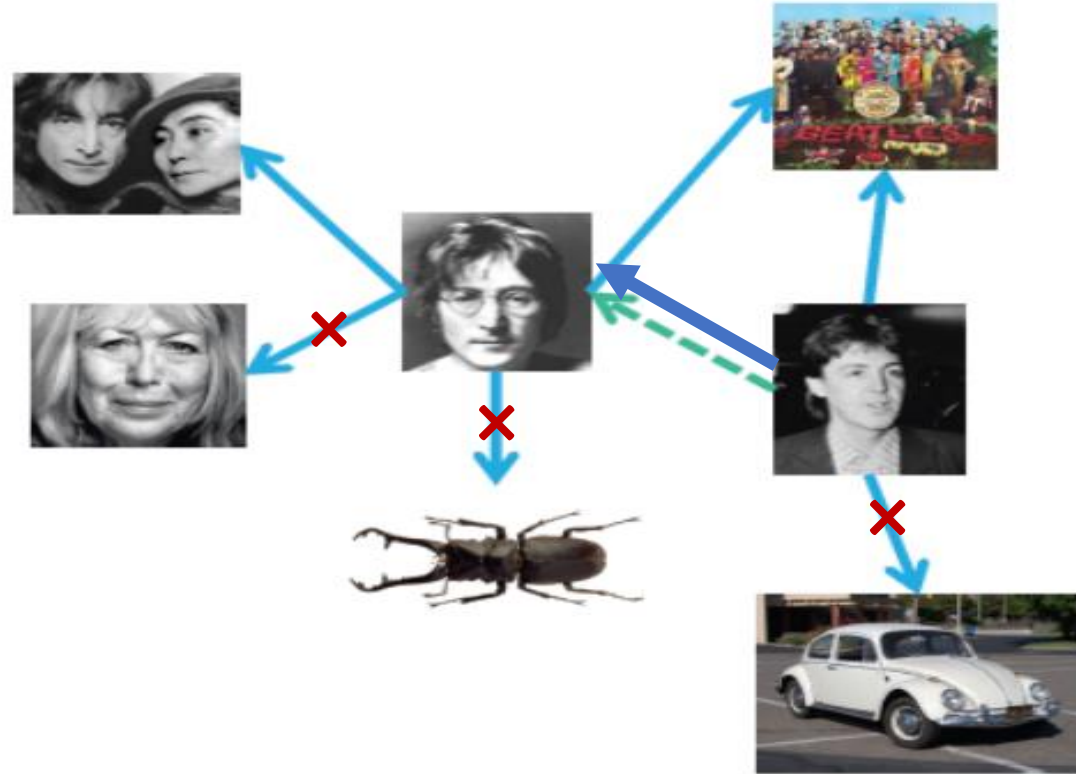


spouse



KG Issues

- Need to minimize
 - ambiguity, incompleteness, inconsistency



Objectives of KG Construction Methods

- Graph construction cleans and completes extraction graph
- Incorporate ontological constraints and relational patterns
 - $R(P, \text{father}, B) \rightarrow \neg R(Q, \text{father}, B)$
- Discover statistical relationships within knowledge graph

Knowledge Graph Identification

- Objective:
 - Given the extracted knowledge, find the following things simultaneously
- **Who** are the entities (nodes) in the graph?
- **What** are their attributes and types (labels)?
- **How** are they related (edges)?

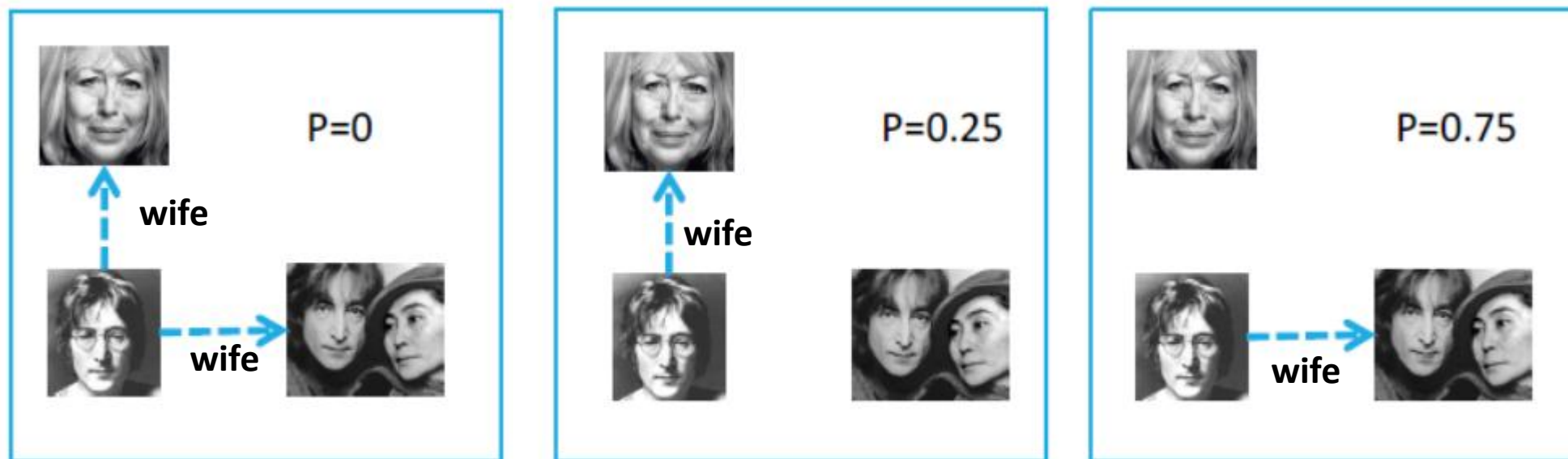
$P(\text{Who, What, How} \mid \text{Extracted knowledge})$

Why Probabilistic Models?

- Limits of Pure Reasoning
 - Classical AI approach to knowledge: reasoning
 - $\text{Lbl}(\text{Socrates}, \text{Man}) \ \& \ \text{Sub}(\text{Man}, \text{Mortal}) \rightarrow \text{Lbl}(\text{Socrates}, \text{Mortal})$
 - Reasoning difficult when extracted knowledge has errors
 - Solution to pure reasoning: Probabilistic models
 - $P(\text{Lbl}(\text{Socrates}, \text{Mortal}) \mid \text{Lbl}(\text{Socrates}, \text{Man})=0.9)$

Probabilistic Models: Intuitions

- Use dependencies between facts in KG
- Probability defined *jointly* over facts



How to get probability values?

- Statistical signals from text extractors and classifiers
 - $P(R(\text{John}, \text{Spouse}, \text{Yoko})) = 0.75$; $P(R(\text{John}, \text{Spouse}, \text{Cynthia})) = 0.25$
 - $\text{LevenshteinSimilarity}(\text{Beatles}, \text{Beetles}) = 0.9$
- Ontological knowledge about domain
 - $\text{Functional}(\text{Spouse}) \ \& \ R(A, \text{Spouse}, B) \rightarrow \neg R(A, \text{Spouse}, C)$
 - $\text{Range}(\text{Spouse}, \text{Person}) \ \& \ R(A, \text{Spouse}, B) \rightarrow \text{Type}(B, \text{Person})$
- Rules and patterns mined from data
 - $R(A, \text{Spouse}, B) \ \& \ R(A, \text{Lives}, L) \rightarrow R(B, \text{Lives}, L)$
 - $R(A, \text{Spouse}, B) \ \& \ R(A, \text{Child}, C) \rightarrow R(B, \text{Child}, C)$

Illustration of KG Identification

Uncertain Extractions:

- .5: Lbl(Fab Four, novel)
- .7: Lbl(Fab Four, musician)
- .9: Lbl(Beatles, musician)
- .8: Rel(Beatles, AlbumArtist, Abbey Road)

(Annotated) Extraction Graph

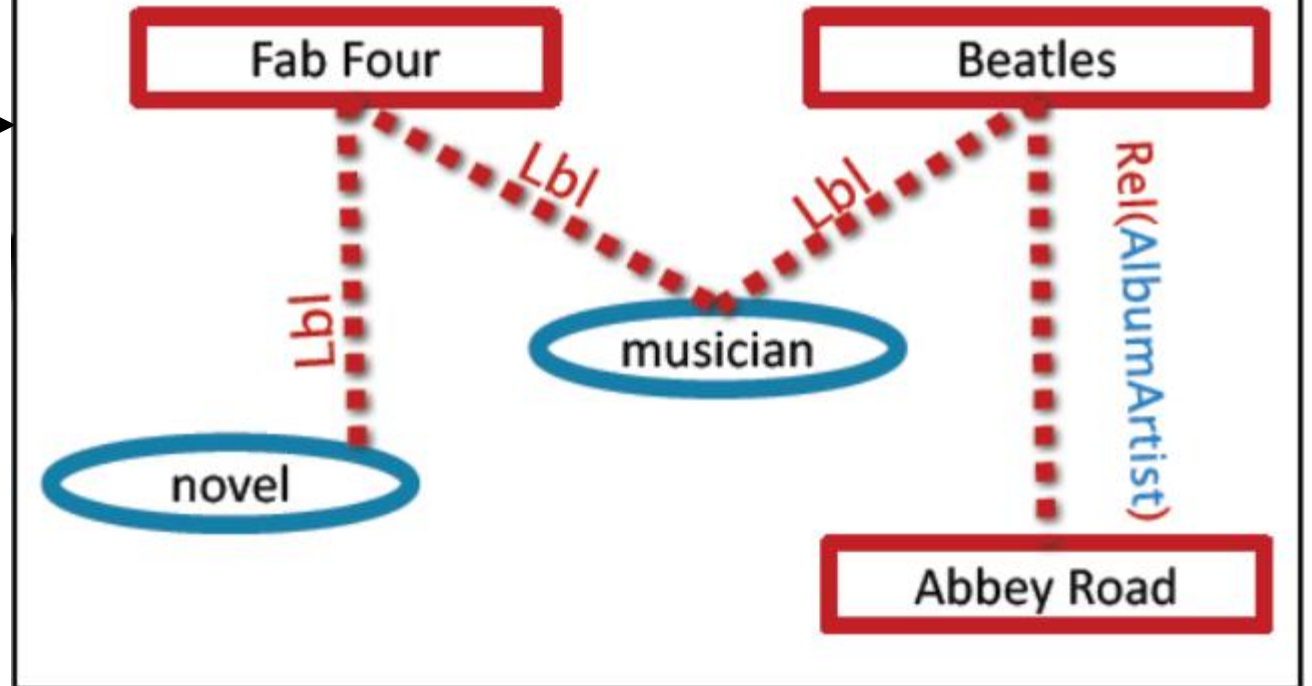


Illustration of KG Identification

Uncertain Extractions:

- .5: Lbl(Fab Four, novel)
- .7: Lbl(Fab Four, musician)
- .9: Lbl(Beatles, musician)
- .8: Rel(Beatles, AlbumArtist, Abbey Road)

Ontologies:

- Dom(albumArtist, musician)
- Mut(novel, musician)

(Annotated) Extraction Graph

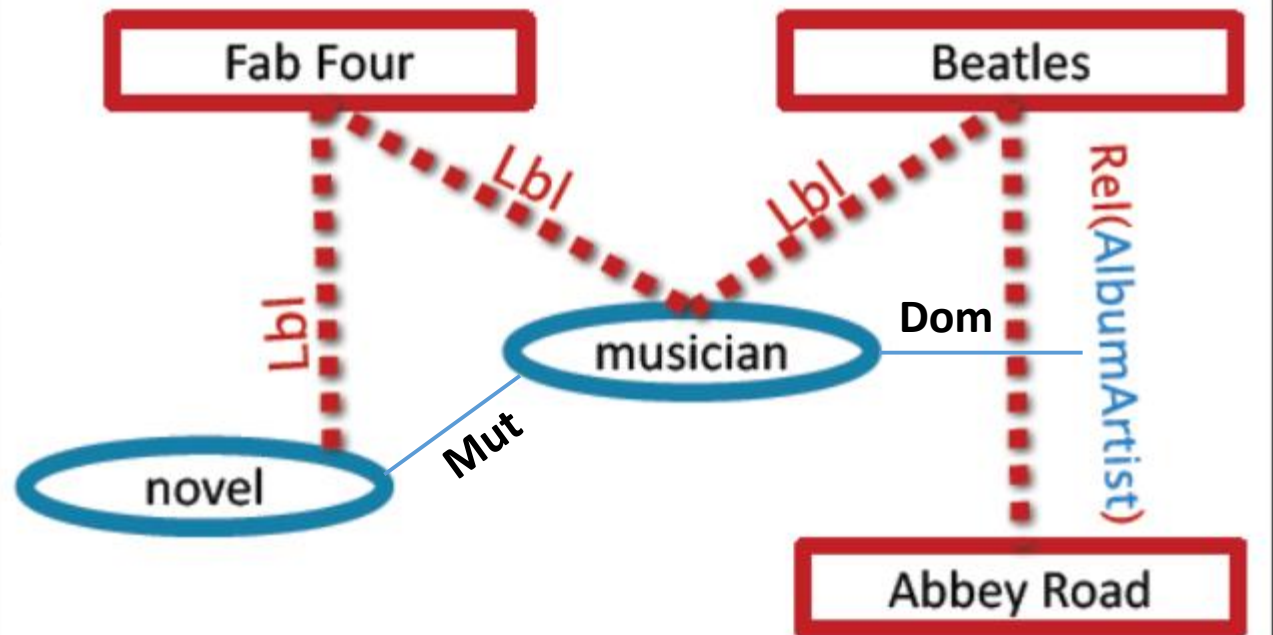


Illustration of KG Identification

Uncertain Extractions:

0.5: Lbl(Fab Four, novel)

0.7: Lbl(Fab Four, musician)

0.9: Lbl(Beatles, musician)

0.8: Rel(Beatles, AlbumArtist, Abbey Road)

Ontologies:

Dom(albumArtist, musician)

Mut(novel, musician)

Entity Resolution:

SameEnt(Fab Four, Beatles)

(Annotated) Extraction Graph

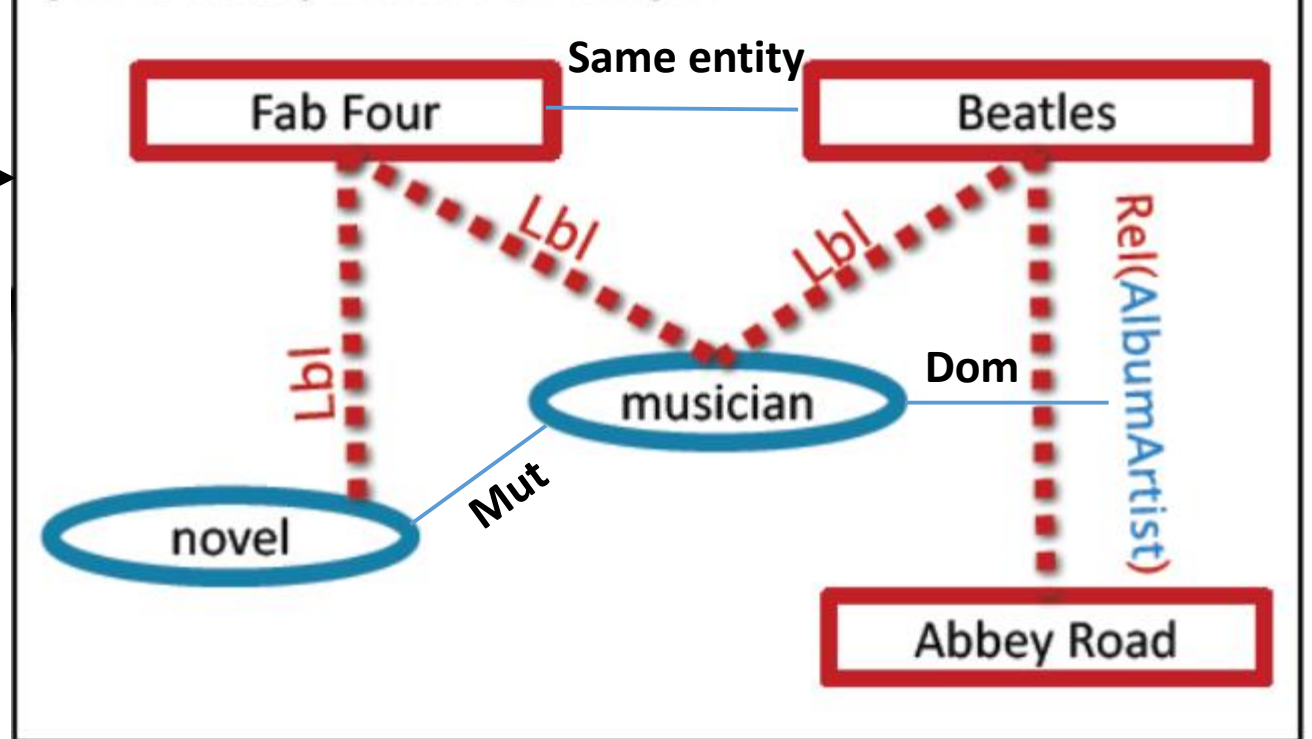


Illustration of KG Identification

Uncertain Extractions:

0.5: Lbl(Fab Four, novel)

0.7: Lbl(Fab Four, musician)

0.9: Lbl(Beatles, musician)

0.8: Rel(Beatles, AlbumArtist, Abbey Road)

Ontologies:

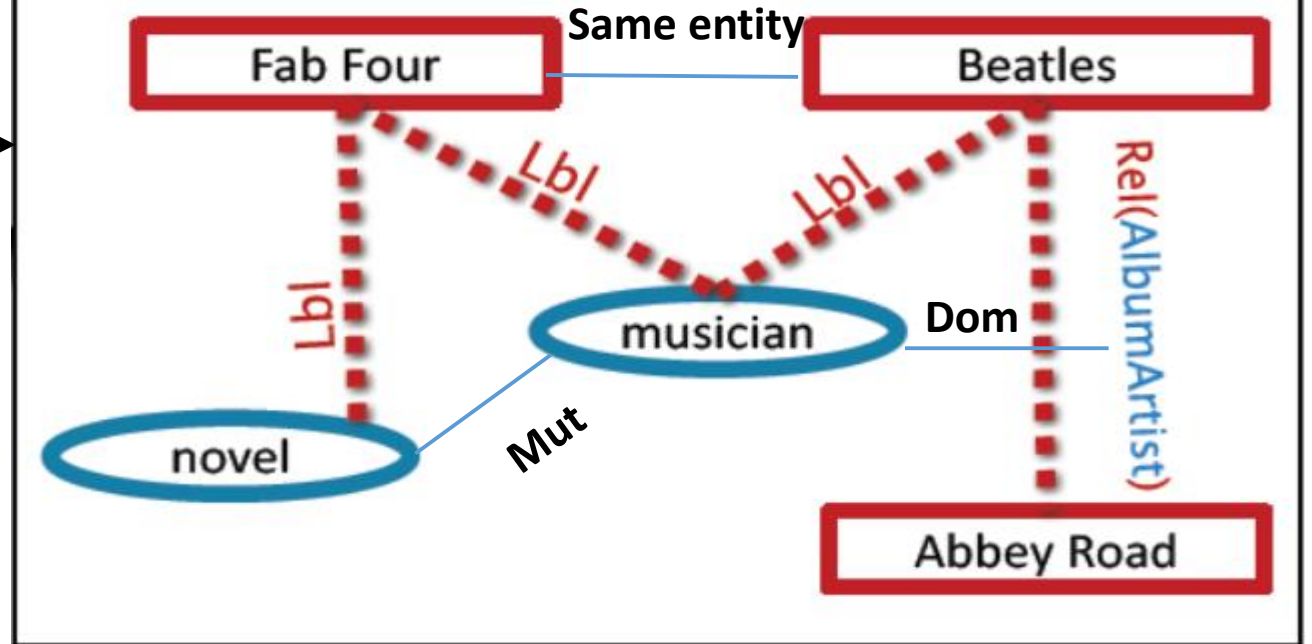
Dom(albumArtist, musician)

Mut(novel, musician)

Entity Resolution:

SameEnt(Fab Four, Beatles)

(Annotated) Extraction Graph



After Knowledge Graph Identification



Graphical Model for KG: Overview

- Uses set of rules
- Uses Probabilistic soft logic (PSL)
- Goal is to infer the truth value of each fact in extraction graph via joint probability distribution

Rules for KG Model

```
100: Subsumes(L1,L2) & Label(E,L1) -> Label(E,L2)
100: Exclusive(L1,L2) & Label(E,L1) -> !Label(E,L2)

100: Inverse(R1,R2) & Relation(R1,E,O) -> Relation(R2,O,E)
100: Subsumes(R1,R2) & Relation(R1,E,O) -> Relation(R2,E,O)
100: Exclusive(R1,R2) & Relation(R1,E,O) -> !Relation(R2,E,O)

100: Domain(R,L) & Relation(R,E,O) -> Label(E,L)
100: Range(R,L) & Relation(R,E,O) -> Label(O,L)

10: SameEntity(E1,E2) & Label(E1,L) -> Label(E2,L)
10: SameEntity(E1,E2) & Relation(R,E1,O) -> Relation(R,E2,O)

1: Label_OBIE(E,L) -> Label(E,L)
1: Label_OpenIE(E,L) -> Label(E,L)
1: Relation_Pattern(R,E,O) -> Relation(R,E,O)
1: -> !Relation(R,E,O)
1: -> !Label(E,L)
```

Probabilistic soft logic (PSL): Overview

- PSL model is composed of a set of weighted, first-order logic rules

- Example: $0.3 : \text{friend}(B, A) \wedge \text{votesFor}(A, P) \rightarrow \text{votesFor}(B, P)$
 $0.8 : \text{spouse}(B, A) \wedge \text{votesFor}(A, P) \rightarrow \text{votesFor}(B, P).$

the weight of the rule

- A, B, P are the universally quantified variables
- *Friend*, *votesFor*, *spouse* are predicates

Probabilistic soft logic (PSL)

- Grounding of a rule:

- Substitute variables by constants/literals
- In a sense, it is an instance of the rule

- Ex: $P(A, B) \tilde{\wedge} Q(B, C) \stackrel{w}{\Rightarrow} R(A, B, C) \Rightarrow P(a, b) \tilde{\wedge} Q(b, c) \Rightarrow R(a, b, c)$

- Ground atoms:

- $P(a, b)$
- $Q(b, c)$
- $R(a, b, c)$

- Each ground atom has a soft value in the range [0-1].

Probabilistic soft logic (PSL)

- Interpretations:

- A mapping (I) from a set of atoms to soft values in $[0-1]$

$$I: A \rightarrow [0,1]^n$$

$A \rightarrow \text{set of atoms}$

- Example:

$$I = \{spouse(b, a) \rightarrow 1, votesFor(a, p) \rightarrow 0.9, votesFor(b, p) \rightarrow 0.3\}$$

Probabilistic soft logic (PSL)

- Distance to satisfaction

- PSL associates a numeric score to each ground rule under an interpretation
- It measures the degree to which this condition is violated

- Computing distance to satisfaction

- $\phi_r(I) = \max(0, I(r_{body}) - I(r_{head}))$

- Example:

$I = \{\text{spouse}(b, a) \rightarrow 1, \text{votesFor}(a, p) \rightarrow 0.9, \text{votesFor}(b, p) \rightarrow 0.3\}$

$r = 0.8 : \text{spouse}(B, A) \wedge \text{votesFor}(A, P) \rightarrow \text{votesFor}(B, P)$

$I(r_{body}) = \max\{0, 1 + 0.9 - 1\} = 0.9$ (using **Lukasiewicz t-norm**)

$\phi_r(I) = \max\{0, 0.9 - 0.3\} = 0.6,$

Lukasiewicz t-norm

$$p \tilde{\wedge} q = \max(0, p + q - 1)$$

$$p \tilde{\vee} q = \min(1, p + q)$$

$$\tilde{\neg} p = 1 - p$$

Probabilistic soft logic (PSL)

- **Computing distributions:**

- PSL defines a probability distribution over interpretations
- Those satisfying more ground rule instances more probable

$$f(I) = \frac{1}{Z} \exp \left[- \sum_{r \in R} w_r \phi_r(I)^p \right]$$

Normalization factor

rule

Weight of
the rule

Distance to satisfaction

$$\phi_r(I) = 1 - T_r(I),$$

$T_r(I)$ = soft-truth value from the
Lukasiewicz t-norm

PSL for KG

- Use the set of rules
- Find an interpretation that maximizes the joint probability
- An interpretation in this case is the set of facts in the extraction graph along with their confidence score

Probability Distribution over KG

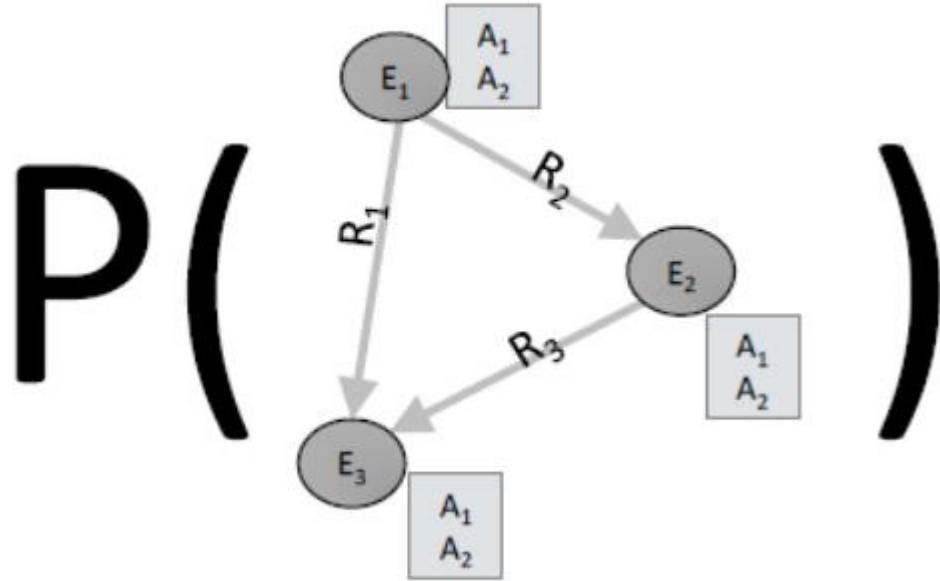
$$P(G | E) = \frac{1}{Z} \exp\left[- \sum_{r \in R} w_r \phi_r(G)\right]$$

Example Rule set

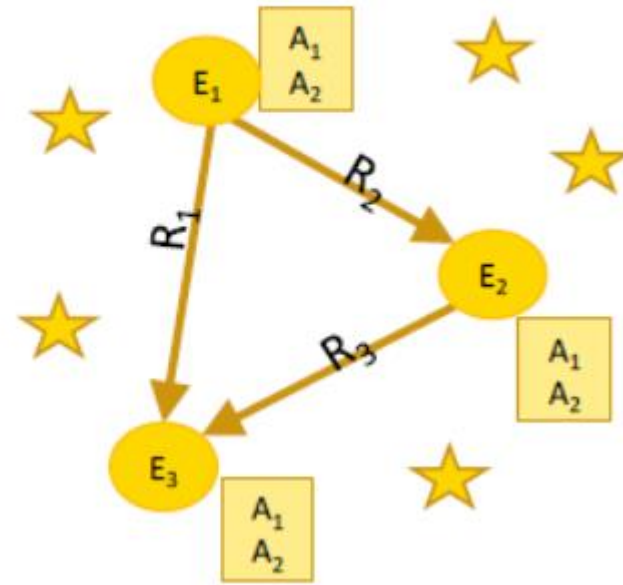
<code>CANDLBL_T(FabFour, novel)</code>	\Rightarrow <code>LBL(FabFour, novel)</code>
<code>MUT(novel, musician)</code>	\wedge <code>LBL(Beatles, novel)</code> \Rightarrow \neg <code>LBL(Beatles, musician)</code>
<code>SAMEENT(Beatles, FabFour)</code>	\wedge <code>LBL(Beatles, musician)</code> \Rightarrow <code>LBL(FabFour, musician)</code>

Getting Final Knowledge Graph

Have: $P(\text{KG})$ for all KGs



Need: best KG

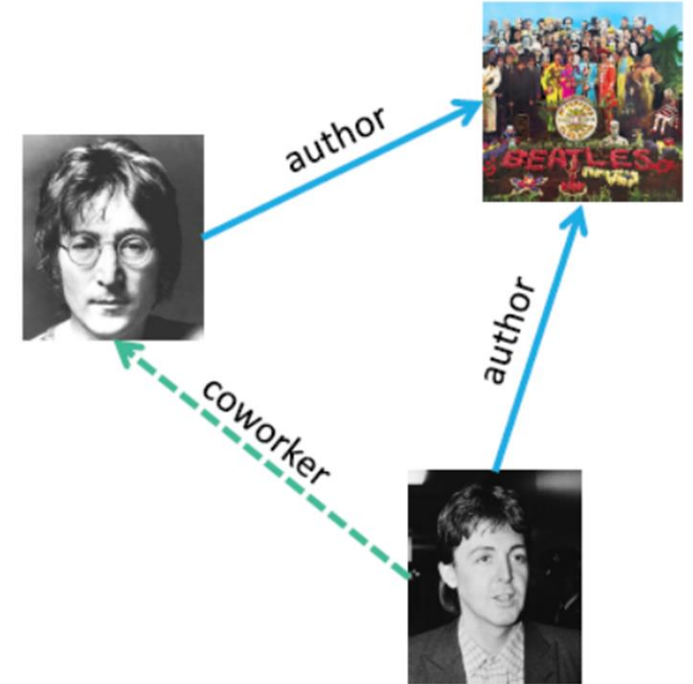


MAP inference: optimizing over distribution to find the best knowledge graph

Relational Learning using Random Walk

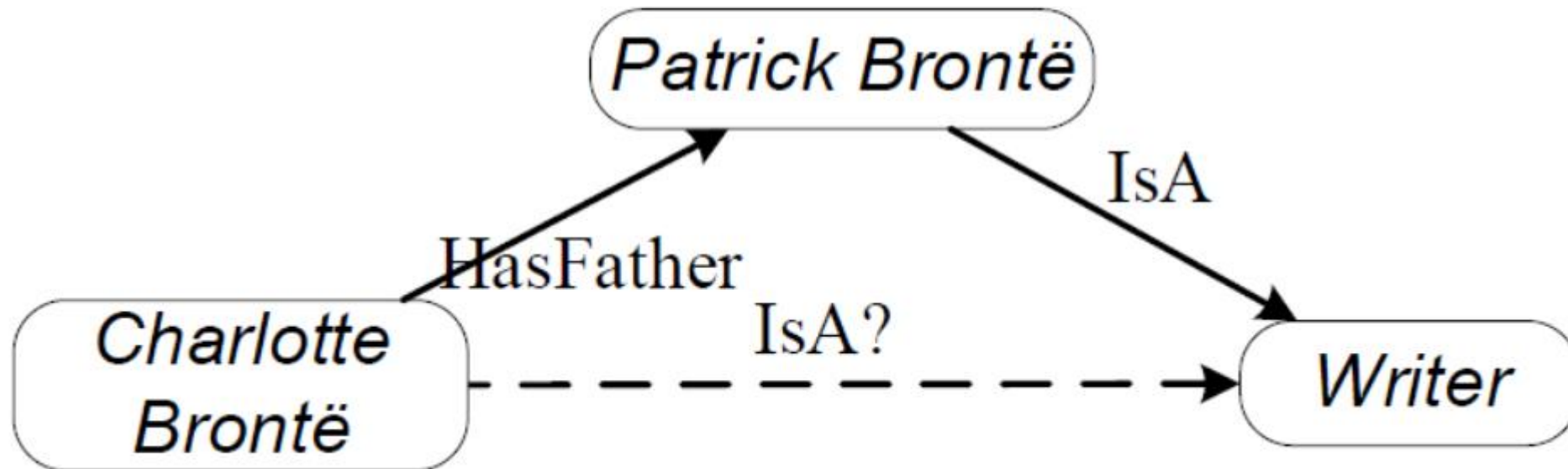
Relational Learning

- Given
 - a directed heterogeneous graph G
 - a starting node s
 - edge type R
- Find
 - nodes t which should have edge R with s



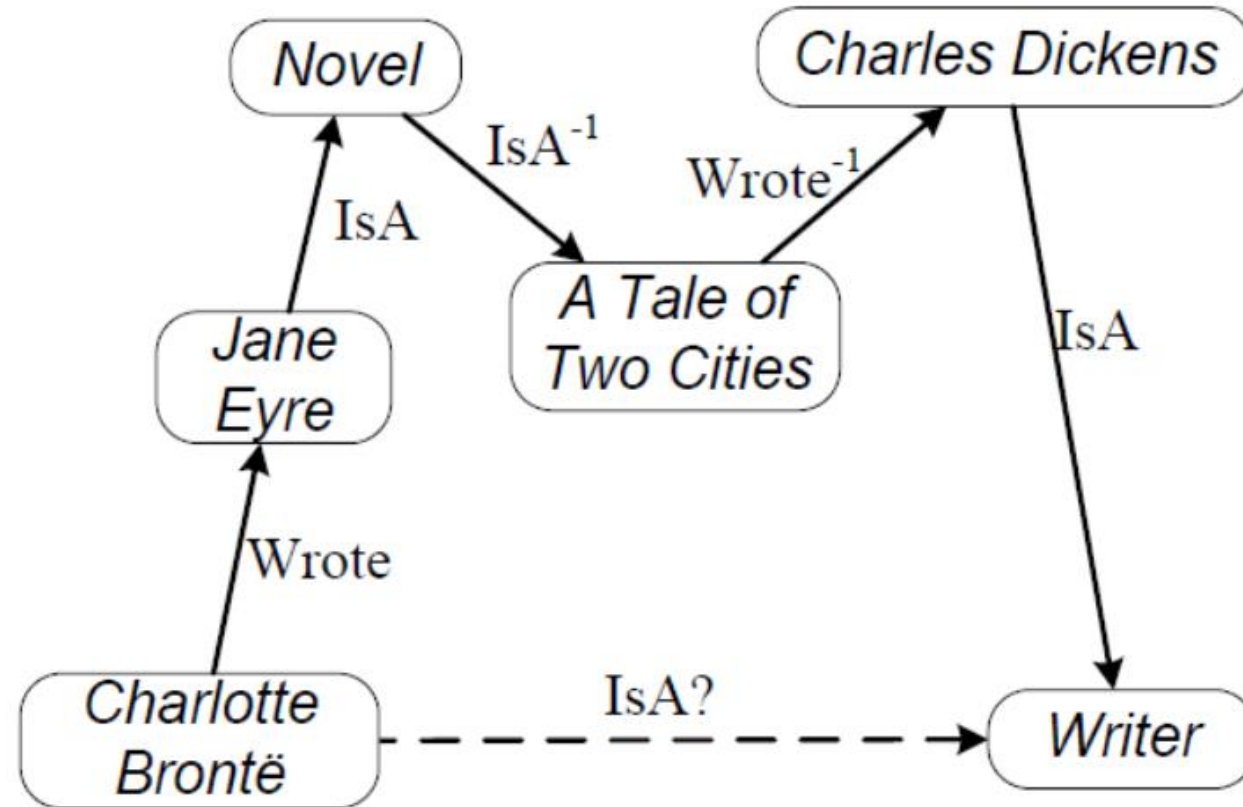
Relational Learning

- Consider friends/family



Relational Learning

Consider people's behavior

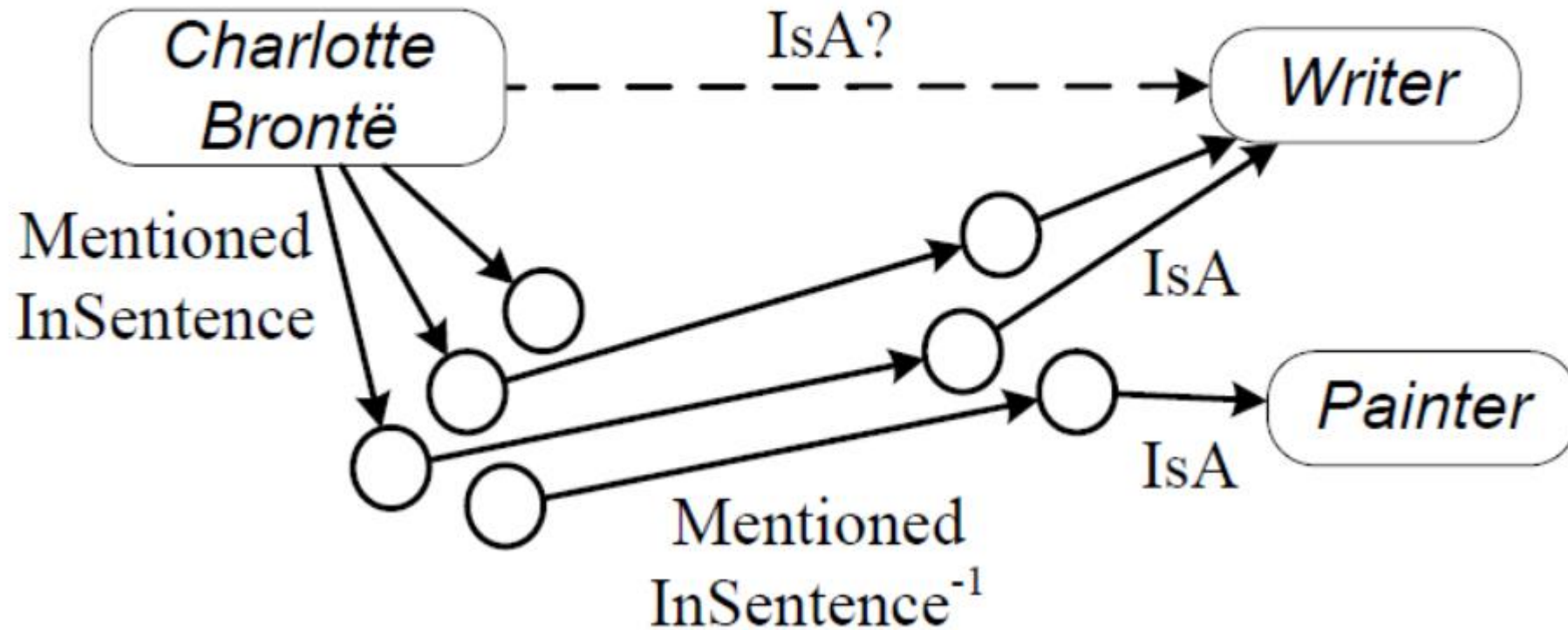


IsA^{-1} is the reverse of IsA relation

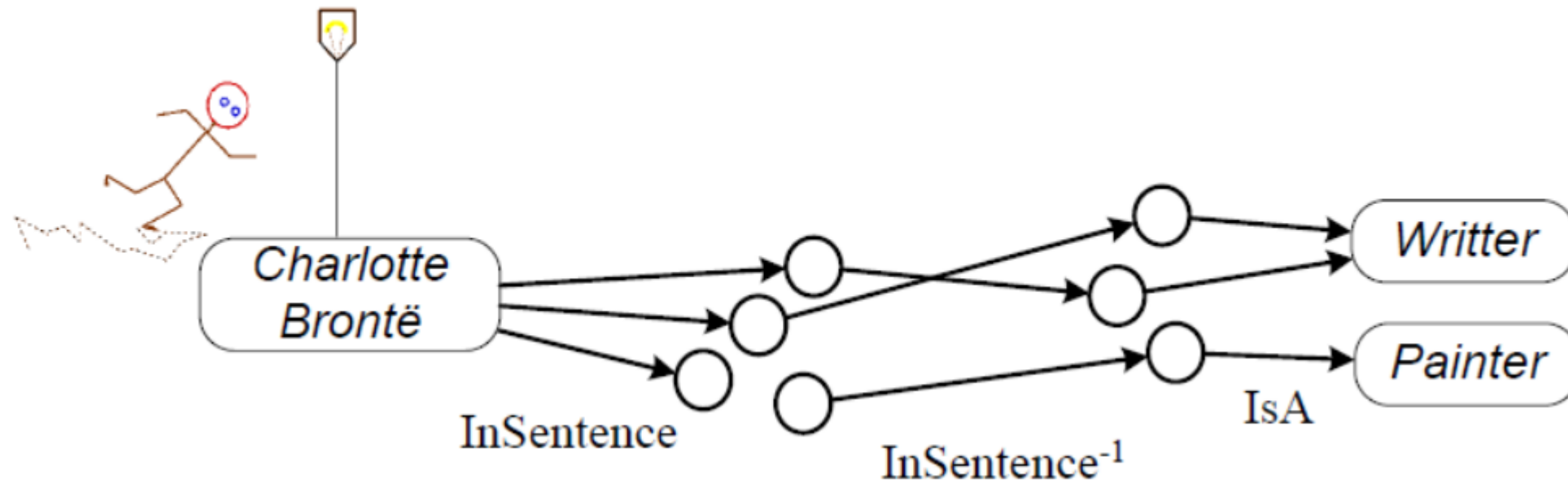
Wrote^{-1} is the reverse of Wrote relation

Relational Learning

Consider literature/publication



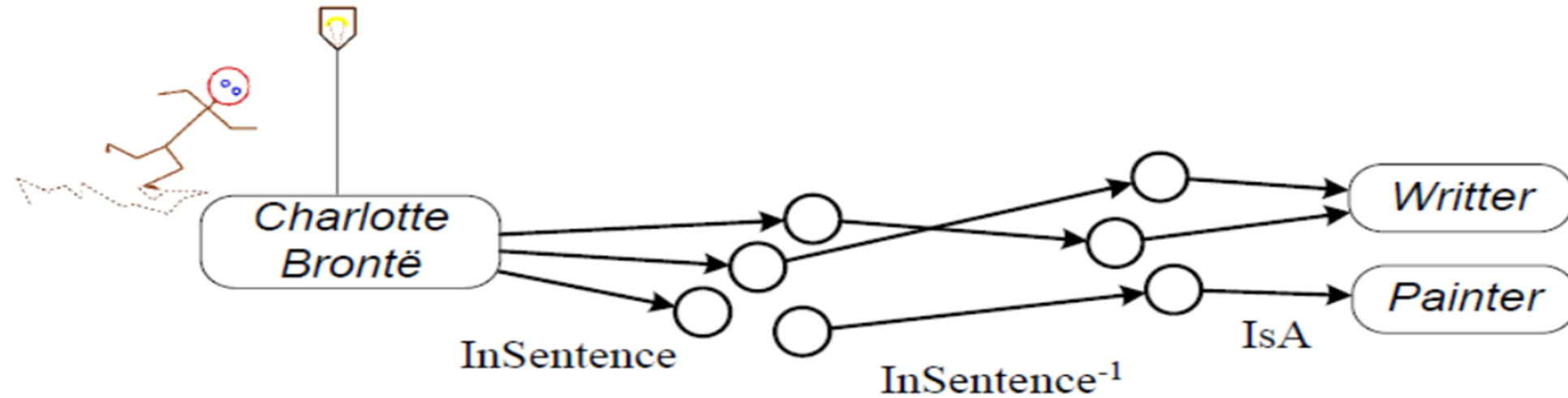
Random Walk Inference



$$\text{Prob}(\text{Charlotte} \rightarrow \text{Writer} \mid \text{InSentence}, \text{InSentence}^{-1}, \text{IsA})$$

Random Walk Inference

Combining features via multiple path



$\text{Prob}(\text{Charlotte} \rightarrow \text{Writer} \mid \text{HasFather, isa})$

$\text{Prob}(\text{Charlotte} \rightarrow \text{Writer} \mid \text{Write, isa, isa}^{-1}, \text{Write, isa})$

$\text{Prob}(\text{Charlotte} \rightarrow \text{Writer} \mid \text{InSentence, InSentence}^{-1}, \text{isa})$

Path Ranking Algorithm

- A relation path $P = (R_1, R_2, \dots, R_n)$ is a sequence of relations
- A PRA model scores a source-target pair by a linear function of their path features

$$score(s, t) = \sum_{P \in \mathcal{P}} \text{Prob}(s \rightarrow t; P) \theta_P$$


- \mathcal{P} is the set of all relation paths within a given length
- E.g. IsA(Charlotte, ???)

Parameter for path importance.
To be trained from data

Prob(Charlotte -> Writer | HasFather, isa)

Prob(Charlotte -> Writer | Write, isa, isa-1, Write, isa)

Prob(Charlotte -> Writer | InSentence, InSentence-1, isa)

Training

- For a relation R and a set of node pairs $\{(s_i, t_i)\}$, construct a training dataset $D = \{(x_i, y_i)\}$
 - x_i is a vector of all the path features for (s_i, t_i)
 - y_i indicates whether $R(s_i, t_i)$ is true or not
 - e.g. $s_i \rightarrow$ Charlotte, $t_i \rightarrow$ painter/writer
- θ is estimated using a classifier

Thank you!