INSID&S Lab
Interaction and Semantics
for Innovation with Data & Services

ADVIS Advances in Data, Visual & Information Science

UIC UNIVERSITY OF ILLINOIS AT CHICAGO

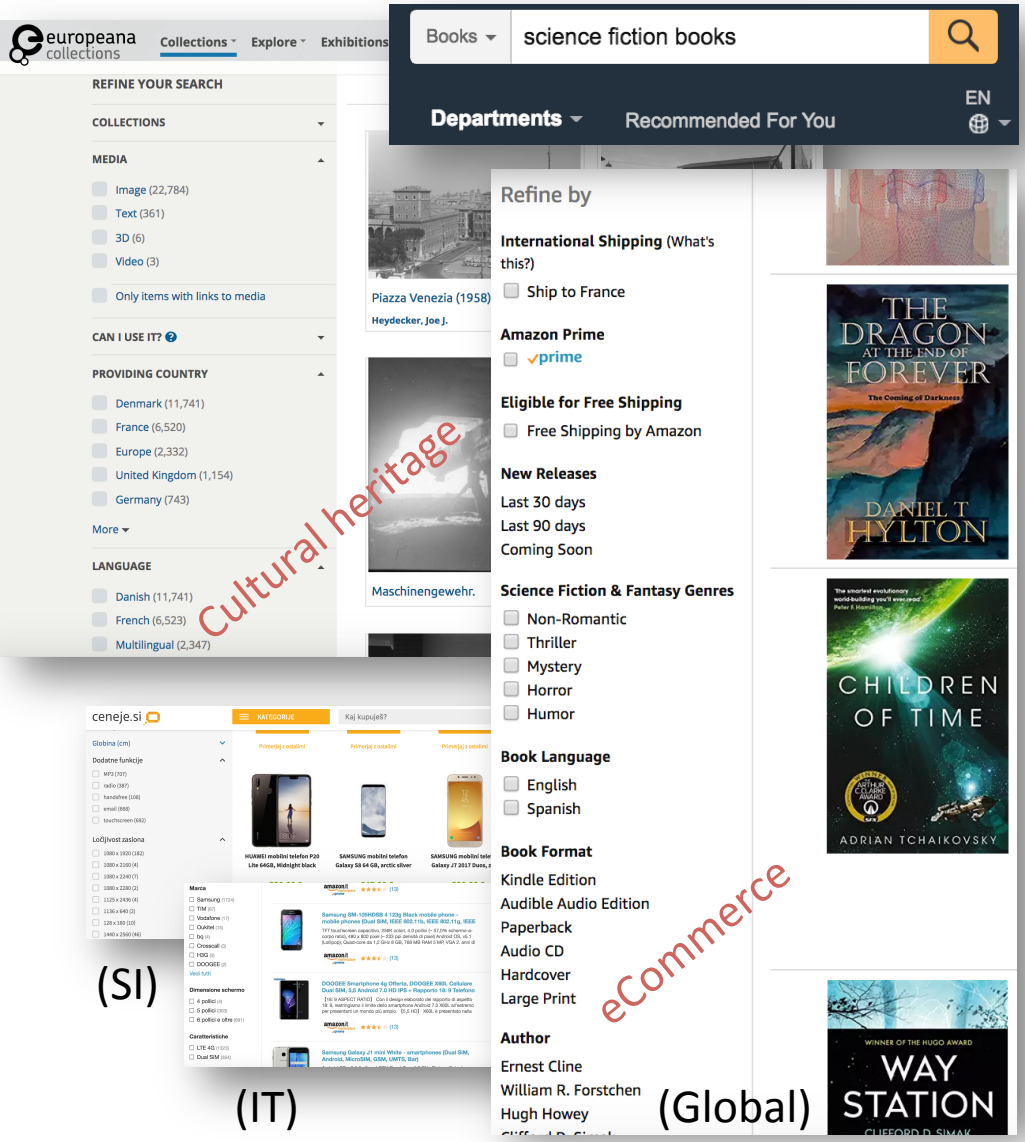
Facet Annotation Using Reference Knowledge Bases

Riccardo Porrini
Mia-platform
(7Pixel Srl)
(University of Milano-Bicocca)

Matteo Palmonari
Department of Information, Systems and Communication
University of Milano-Bicocca

Isabel F. Cruz
Department of Computer Science
University of Illinois at Chicago

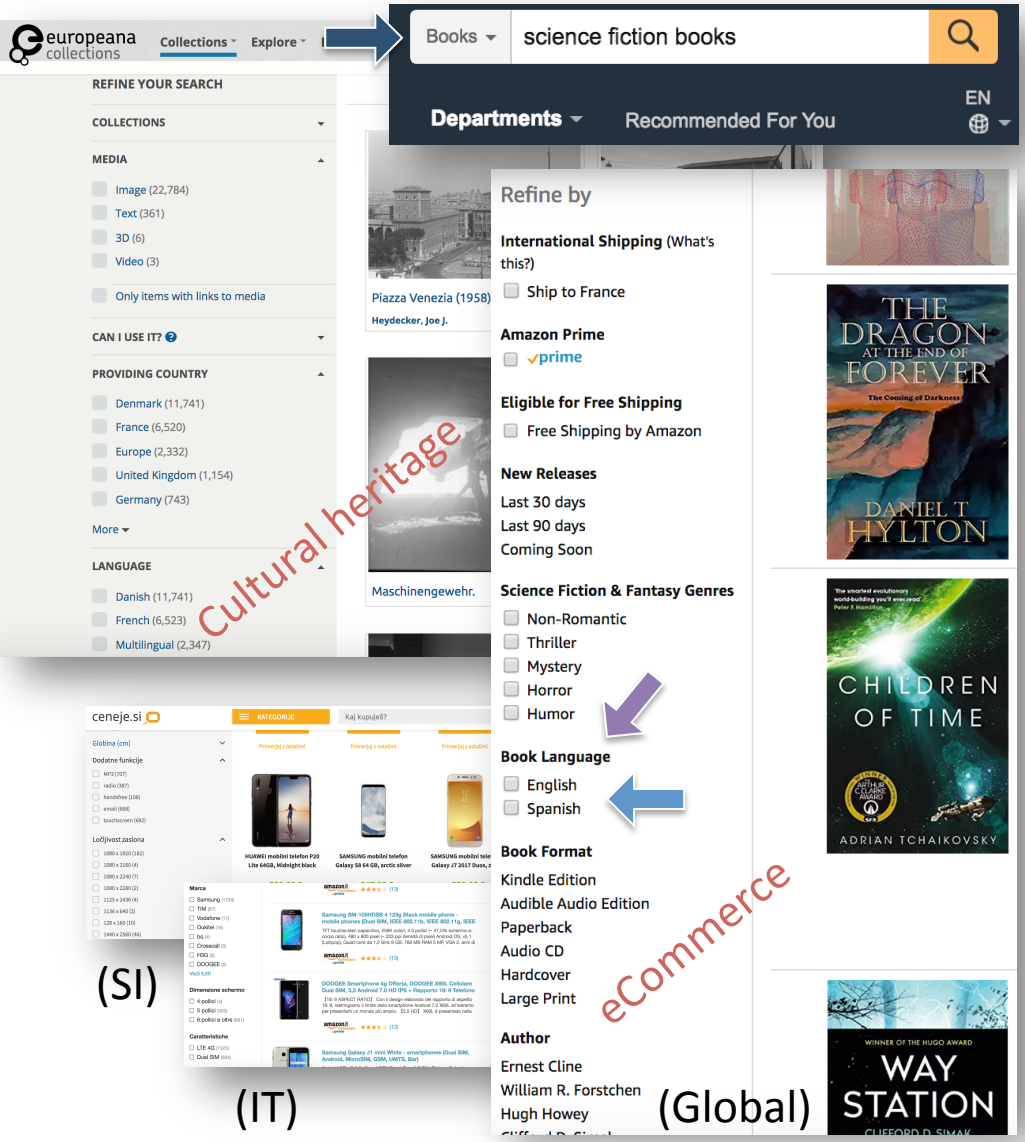
Using and Building Faceted Interfaces



Faceted interfaces

- Widespread in several domains
- Enhance user experience
  - Competitive advantage for web applications, e.g., in eCommerce

Using and Building Faceted Interfaces



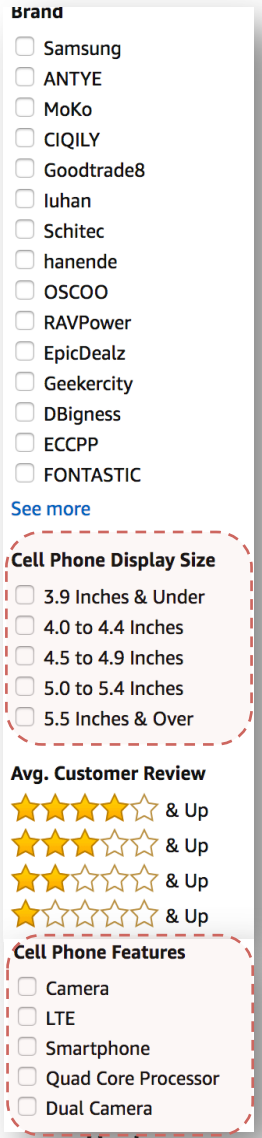
Faceted interfaces

- Widespread in several domains
- Enhance user experience
  - Competitive advantage for web applications, e.g., in eCommerce

A facet

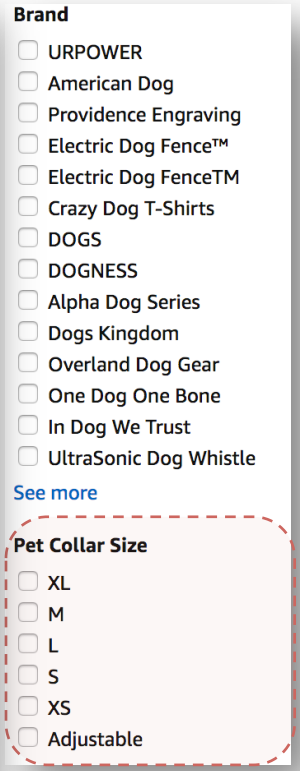
- Domain, property, values
  - <D,p,{v_1,…, v_n}>
  - <Books,book language, {English,Spanish}>

Using and Building Faceted Interfaces



cellphones

books



pet supplies

Faceted interfaces

- Widespread in several domains
- Enhance user experience
  - Competitive advantage for web applications, e.g., in eCommerce

A facet

- Domain, property, values
  - <D,p,{v_1,…, v_n}>
  - <Books,book language, {English,Spanish}>

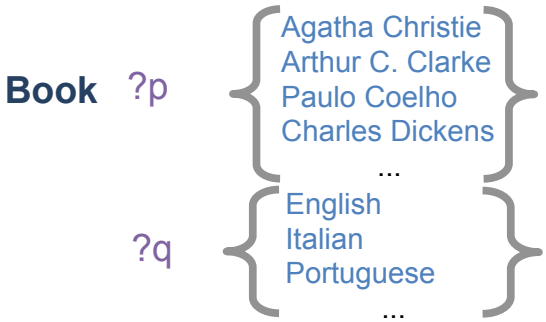
Hard to define well-crafted facets manually

- Many different types of entities require many domain-specific facets
  - E.g., 500+ product types

Facet Extraction & Annotation

Facet extraction

- Clusters of values for different characteristics
  - Offline vs. Online (on search results)



Facet annotation

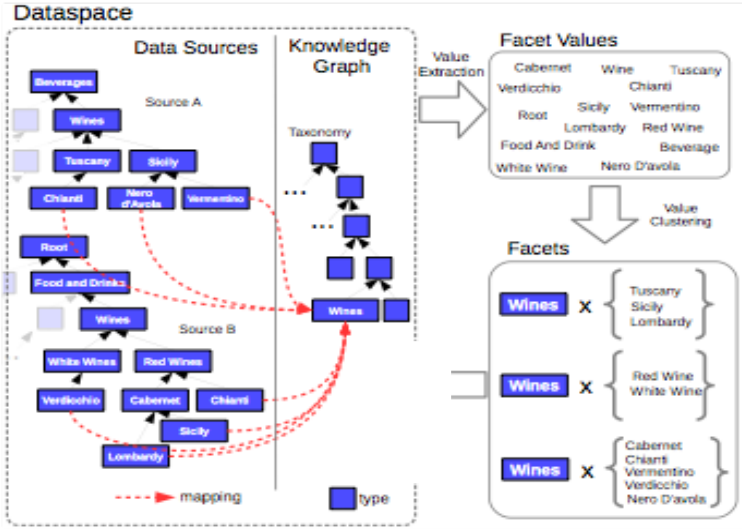
- *Assign a property to the automatically extracted set of values*
- Use a predicate of a KB to capture the semantics of the property
  - Human readable semantics
  - Machine-readable semantics → RDFa markup of faceted interfaces
  - Reduction of terminological entropy across domains

Main goals:

Build faceted interfaces on-the-fly for effective searching

Assist domain experts in creating and maintaining facets

Facet Extraction & Creation in Comparison Shopping Platforms (CSPs)

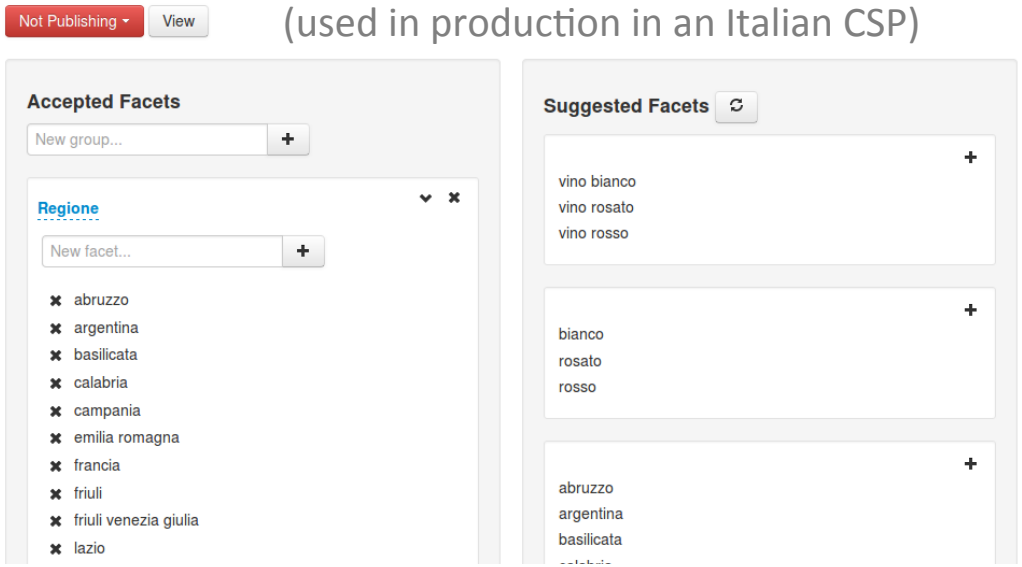


3K+ sources 500+ types

Automated clustering of facet values (Porrini&al.CAISE14)

- extracted from 3k+ data sources mapped to a CSP central schema with 500+ leaf types
- clustered by similarity

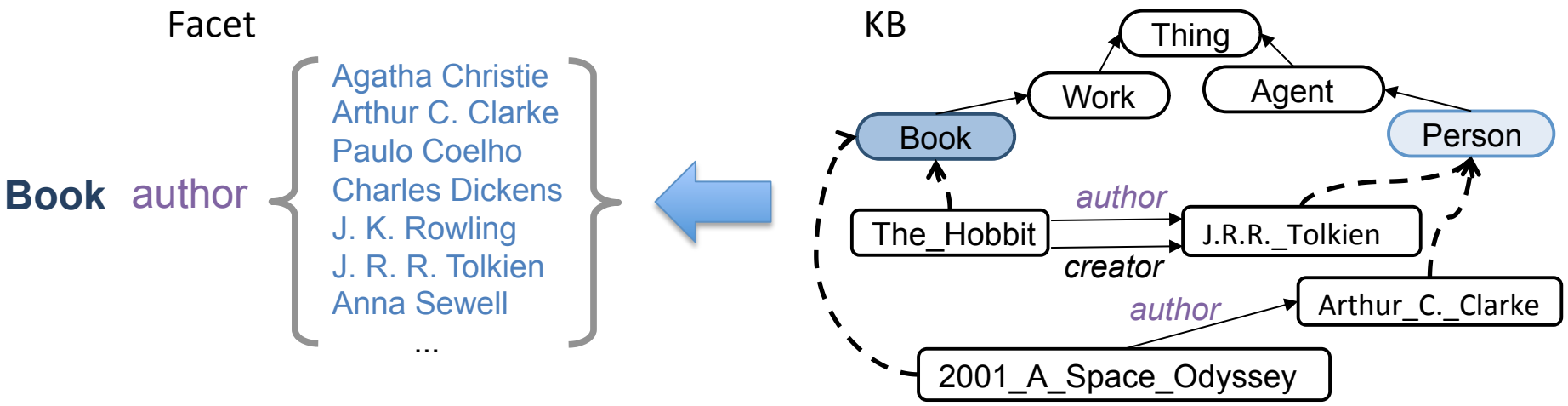
(used in production in an Italian CSP)



Experts refine the facet with a GUI

- Add/delete values
- Lexical specification of the facet property

Facet Annotation vs. Table Annotation



Facet annotation (unbalanced)

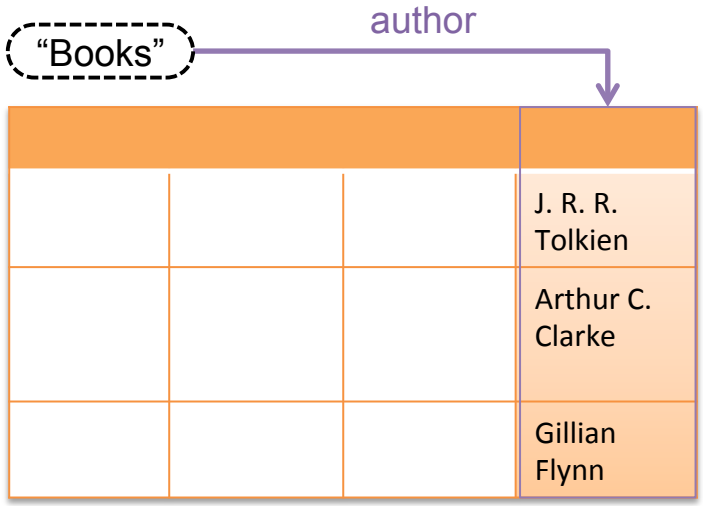
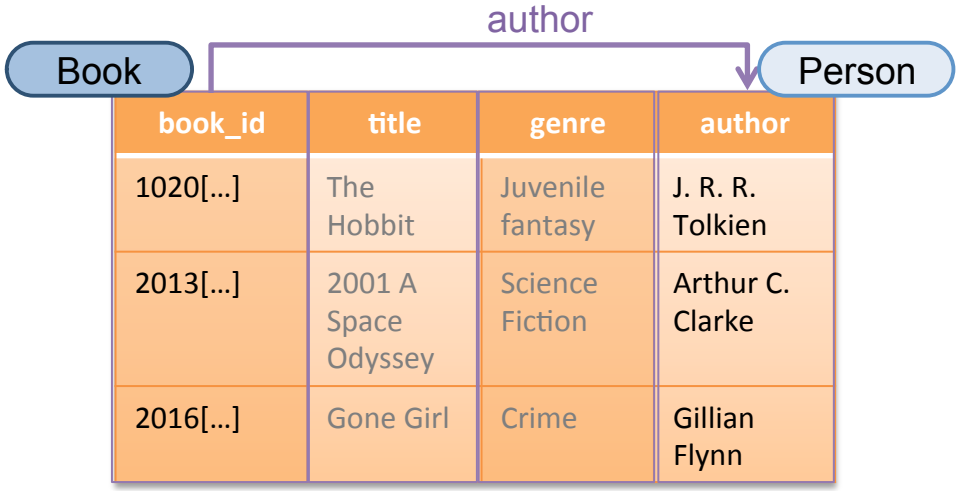
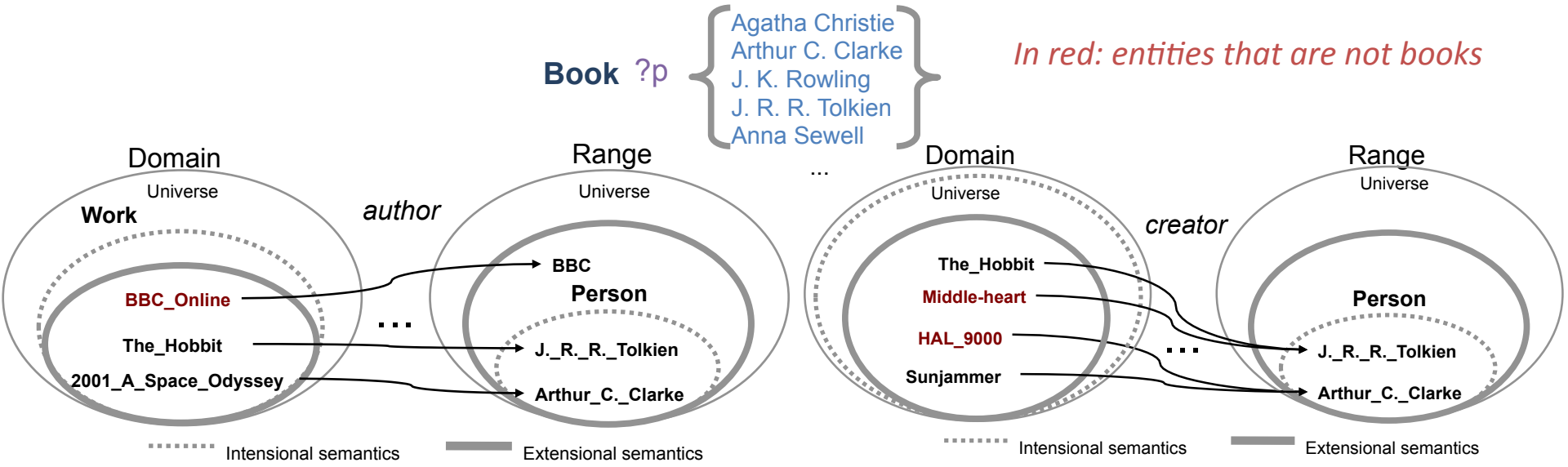


Table annotation (balanced)

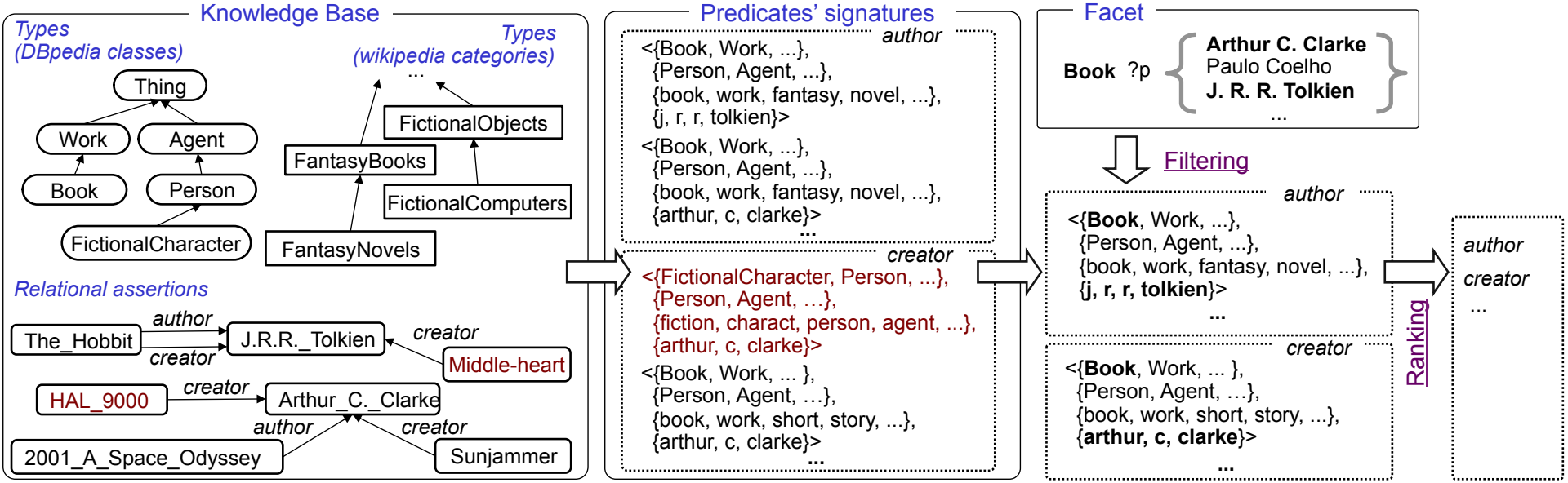


Facet Annotation: Challenges



- Challenges:
  - Many predicates, e.g., 53K+ DBpedia properties
  - Weak specification, e.g., no domain/range restrictions
  - Ambiguous values, i.e., same values in different predicates
  - Specificity of the predicate associated with a domain-specific facet

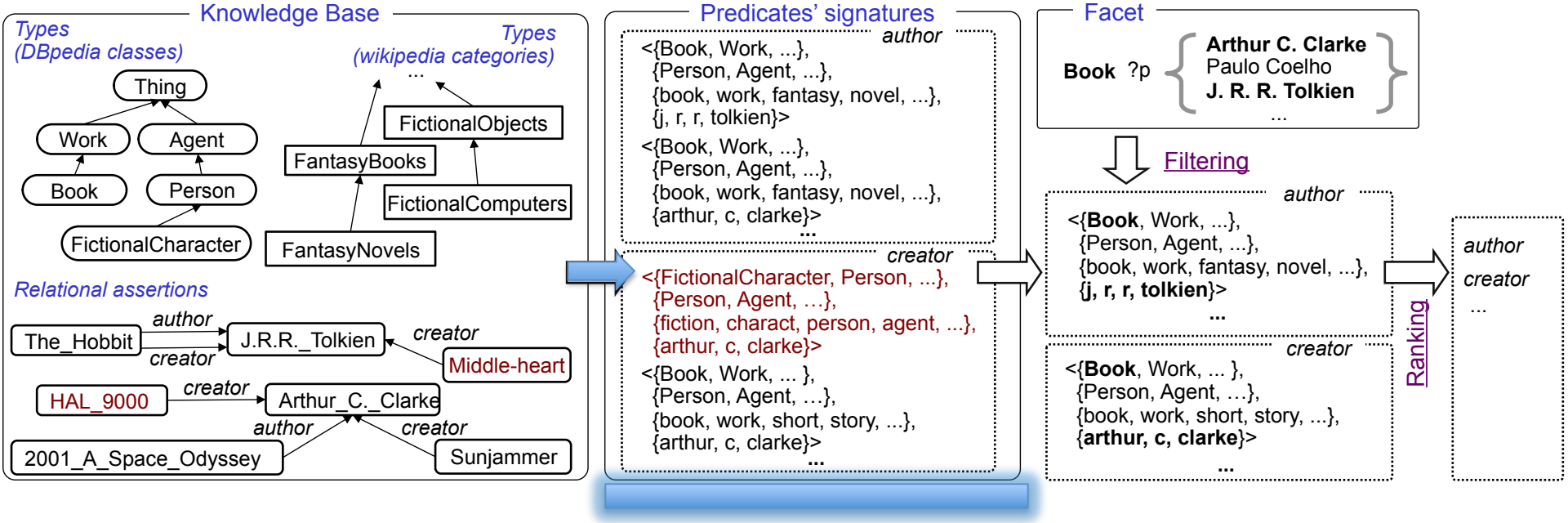
Approach: Overview



- Index the extensions of the KB predicates using **signatures**
- **Filter** predicates whose signatures match the facet domain and values
- **Rank** predicates by similarity (three criteria: frequency, coverage, specificity)

- Output: **ranked list of predicates**

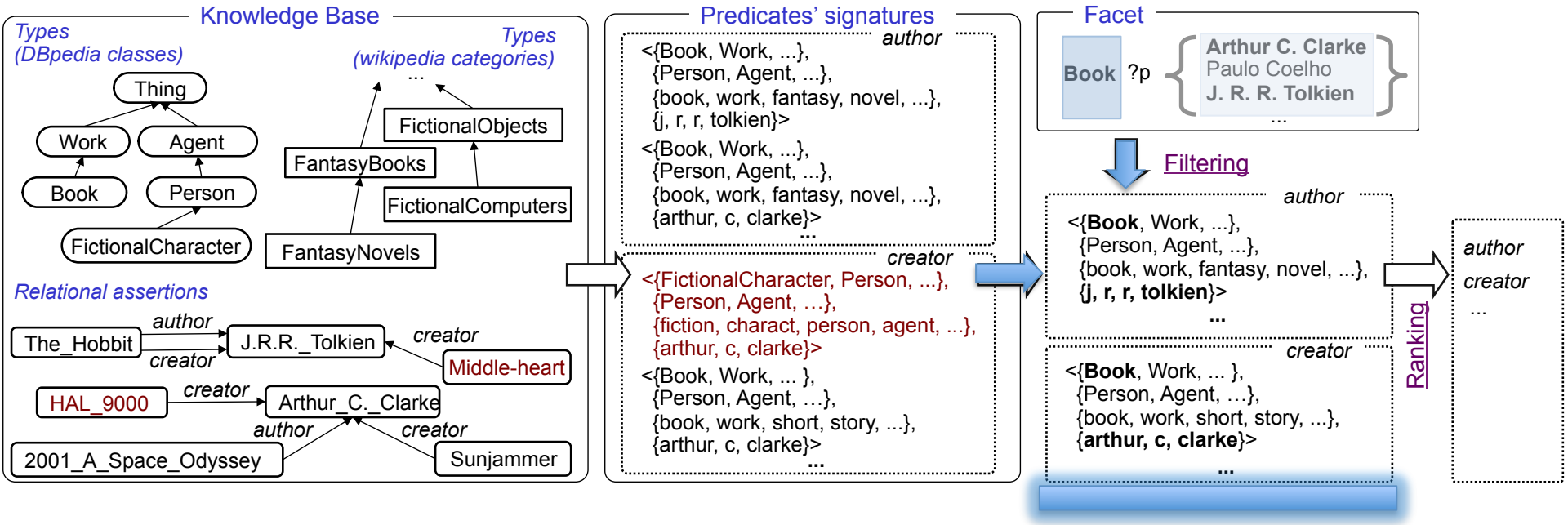
Approach: Signatures



Domain	Range
The_Hobbit	"J.R.R._Tolkien"
Middle-earth	"J.R.R._Tolkien"
HAL_9000	"Arthur_C._Clark"
Sunjammer	"Arthur_C._Clark"

Subject Types' Names	Subject Types' Lexicalizations	Object Types' Names	Objects' Lexicalizations
< {Book, Work, ... }	{*book*, *work*, *fantas*, *novel*, ...}	{Person, Agent, ...}	{j, r, r, *tolkien*} >
< {Thing, FictionalContinents,...}	{*thing*, *fiction*, *continent*,...,}	{Person, Agent, ...}	{*j*, *r*, *r*, *tolkien*} >
< {FictionalCharacter, Person, ...}	{*fiction*, *charact*, *person*, *agent*, ...}	{Person, Agent, ...}	{*arthur*, c, *clarke*} >
< {Book, Work, ... }	{*book*, *work*, *short*, *story*, ...}	{Person, Agent, ...}	{*arthur*, c, *clarke*} >

Approach: Filtering



D-match

- subject types vs. facet domain (lexicalizations)

R-match

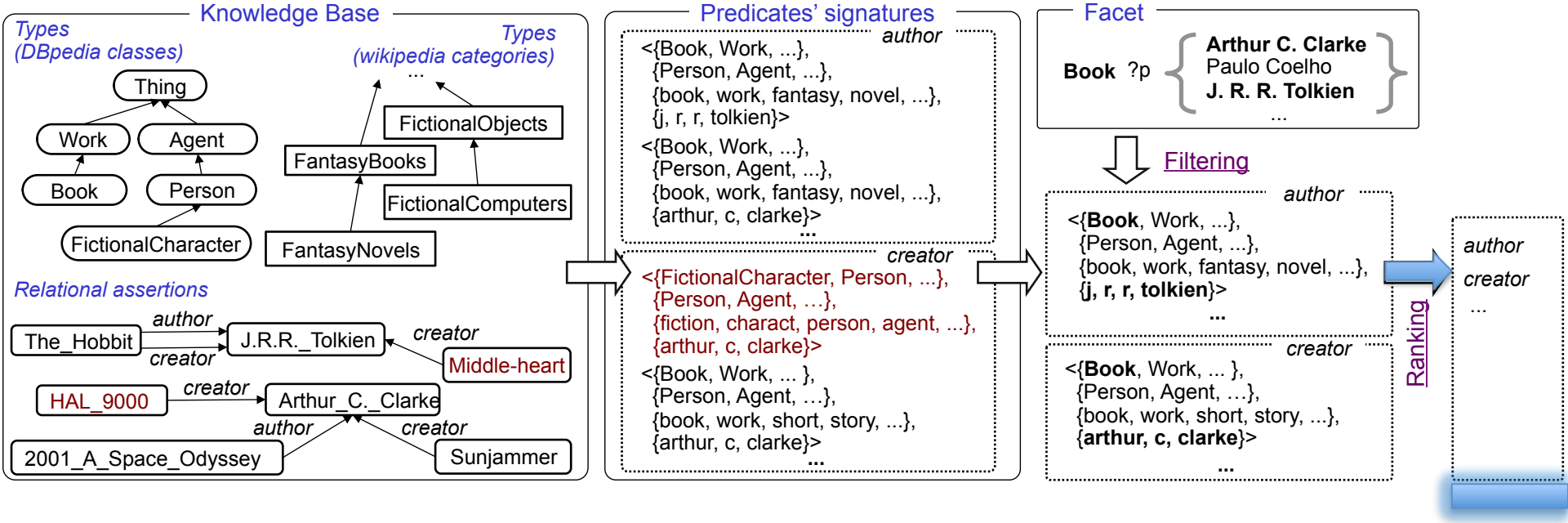
- object vs. at least 1 facet value (lexicalizations)

Signature matches if

- d-match & r-match

	Subject Types' Names	Subject Types' Lexicalizations	Object Types' Names	Objects' Lexicalizations
✓	< {Book, Work, ... }	{*book, work, fantas, novel, ...*}	{Person, Agent, ...}	{j, r, r, *tolkien*} >
✗	< {Thing, FictionalContinents,...}	{*thing, fiction, continent,...,*}	{Person, Agent, ...}	{*j, r, r, tolkien*} >
✗	< {FictionalCharacter, Person, ...}	{*fiction, charact, person, agent, ...*}	{Person, Agent, ...}	{*arthur*, c, *clarke*} >
✓	< {Book, Work, ... }	{*book, work, short, story, ...*}	{Person, Agent, ...}	{*arthur*, c, *clarke*} >

Approach: Ranking



Three criteria:

- Coverage
- Weighted Frequency
- Specificity

Combination:

- Smoothing for comparability
- Multiplication

	Subject Types' Names	Subject Types' Lexicalizations	Object Types' Names	Objects' Lexicalizations
✓	< {Book, Work, ... }	{*book*, *work*, *fantas*, *novel*, ...}	{Person, Agent, ...}	{j, r, r, *tolkien*} >
✗	< {Thing, FictionalContinents,...}	{*thing*, *fiction*, *continent*,...,}	{Person, Agent, ...}	{*j*, *r*, *r*, *tolkien*} >
✗	< {FictionalCharacter, Person, ...}	{*fiction*, *charact*, *person*, *agent*, ...}	{Person, Agent, ...}	{*arthur*, c, *clarke*} >
✓	< {Book, Work, ... }	{*book*, *work*, *short*, *story*, ...}	{Person, Agent, ...}	{*arthur*, c, *clarke*} >

Ranking: Coverage and Weighted Frequency

- Coverage of a predicate *q*
  - *#facet_values_matched_by_q / #facet_values*

- Weighted Frequency of a predicate *q*

Scaled down by the number of facet values V^f …

$$freq(f,q) = \frac{1}{|V^f|} \sum_{\delta_{(s,o)} \in \Delta_q^f} \frac{|L^T(s) \cap L(D^f)|}{|L^T(s) \cup L(D^f)|}$$

Sum of signatures $\delta_{(s,o)}$ matching the facet…

… weighted by the Jaccard similarity between the lexicalizations of the subject type $L^T(s)$ and of the facet domain $L(D^f)$

Ranking: Specificity

author

Subject Types' Names		Object Types' Names		
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✓
< {Work, ... }	{...}	{Agent, ... }	{...} >	✗
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✓
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✗
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✗

creator

Subject Types' Names		Object Types' Names		
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✓
< {Thing, FictionalContinents,...}	{...}	{Person, Agent, ...}	{...} >	✗
< {FictionalCharacter, Person, ...}	{...}	{Person, Agent, ...}	{...} >	✗
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✓
< {Book, Work, ... }	{...}	{Company, Agent, ...}	{...} >	✗

- Matching signatures vs. non matching signatures
  - "more similar" in *author* than in *creator*

Ranking: Specificity

author

Subject Types' Names			Object Types' Names		
< {Book, Work, ... }		{...}	{Person, Agent, ...}	{...} >	✓
< {Work, ... }	*	{...}	{Agent, ... }	{...} >	✗
< {Book, Work, ... }		{...}	{Person, Agent, ...}	{...} >	✓
< {Book, Work, ... }	*	{...}	{Person, Agent, ...}	{...} >	✗
< {Book, Work, ... }	*	{...}	{Person, Agent, ...}	{...} >	✗

creator

Subject Types' Names			Object Types' Names		
< {Book, Work, ... }		{...}	{Person, Agent, ...}	{...} >	✓
< {Thing, FictionalContinents,...}	*	{...}	{Person, Agent, ...}	{...} >	✗
< {FictionalCharacter, Person, ...}	*	{...}	{Person, Agent, ...}	{...} >	✗
< {Book, Work, ... }		{...}	{Person, Agent, ...}	{...} >	✓
< {Book, Work, ... }	*	{...}	{Company, Agent, ...}	{...} >	✗

- Matching signatures vs. non matching signatures
  - "more similar" in *author* than in *creator*
- Compare types in signatures rather than signatures themselves
  - Counting matching vs. non matching signatures is unreliable (noisy, matching signatures << non matching signatures)

Ranking: Specificity

author

Subject Types' Names			Object Types' Names		
< {Book, ~~Work~~, ... }		{...}	{Person, ~~Agent~~, ...}	{...} >	✓
< {Work, ... }	*	{...}	{Agent, ... }	{...} >	✗
< {Book, ~~Work~~, ... }		{...}	{Person, ~~Agent~~, ...}	{...} >	✓
< {Book, ~~Work~~, ... }	*	{...}	{Person, ~~Agent~~, ...}	{...} >	✗
< {Book, ~~Work~~, ... }	*	{...}	{Person, ~~Agent~~, ...}	{...} >	✗

creator

Subject Types' Names			Object Types' Names		
< {Book, ~~Work~~, ... }		{...}	{Person, ~~Agent~~, ...}	{...} >	✓
< {Thing, FictionalContinents,...}	*	{...}	{Person, ~~Agent~~, ...}	{...} >	✗
< {FictionalCharacter, ~~Person~~, ...}	*	{...}	{Person, ~~Agent~~, ...}	{...} >	✗
< {Book, ~~Work~~, ... }		{...}	{Person, ~~Agent~~, ...}	{...} >	✓
< {Book, ~~Work~~, ... }	*	{...}	{Company, ~~Agent~~, ...}	{...} >	✗

- Matching signatures vs. non matching signatures
  - "more similar" in *author* than in *creator*
- Compare types in signatures rather than signatures themselves
  - Counting matching vs. non matching signatures is unreliable (noisy, matching signatures << non matching signatures)
  - Consider most specific types

Ranking: Specificity

author

Subject Types' Names		Object Types' Names		
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✓
< {Work, ... }	{...}	{Agent, ... }	{...} >	✗
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✓
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✗
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✗

creator

Subject Types' Names		Object Types' Names		
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✓
< {Thing, FictionalContinents,...}	{...}	{Person, Agent, ...}	{...} >	✗
< {FictionalCharacter, Person, ...}	{...}	{Person, Agent, ...}	{...} >	✗
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✓
< {Book, Work, ... }	{...}	{Company, Agent, ...}	{...} >	✗

Two similarity measures:

- *d-spec (on the domain)*
- *r-spec (on the range)*

Most specific subject types in non matching signatures

Most specific subject types in matching signatures

$$d\text{-}spec(f,q) = \frac{(D[\Delta_q] \cap D[\Delta_q^f])}{(D[\Delta_q] \cup D[\Delta_q^f])}$$

Ranking: Specificity

author

Subject Types' Names		Object Types' Names		
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✓
< {Work, ... }	{...}	{Agent, ... }	{...} >	✗
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✓
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✗
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✗

creator

Subject Types' Names		Object Types' Names		
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✓
< {Thing, FictionalContinents,...}	{...}	{Person, Agent, ...}	{...} >	✗
< {FictionalCharacter, Person, ...}	{...}	{Person, Agent, ...}	{...} >	✗
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✓
< {Book, Work, ... }	{...}	{Company, Agent, ...}	{...} >	✗

Two similarity measures:

- *d-spec (on the domain)*
- *r-spec (on the range)*

Most specific subject types in non matching signatures

Most specific subject types in matching signatures

$$d\text{-}spec(f,q) = \frac{w\text{-}card(D[\Delta_q] \cap D[\Delta_q^f])}{w\text{-}card(D[\Delta_q] \cup D[\Delta_q^f])}$$

$$w\text{-}card(T) = \sum_{t \in T} \frac{depth(t)}{\max_{t' \in DES[t]} depth(t')}$$

Each type is weighted by its depth

Ranking: Specificity

author

Subject Types' Names		Object Types' Names		
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✓
< {Work, ... }	{...}	{Agent, ... }	{...} >	✗
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✓
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✗
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✗

creator

Subject Types' Names		Object Types' Names		
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✓
< {Thing, FictionalContinents,...}	{...}	{Person, Agent, ...}	{...} >	✗
< {FictionalCharacter, Person, ...}	{...}	{Person, Agent, ...}	{...} >	✗
< {Book, Work, ... }	{...}	{Person, Agent, ...}	{...} >	✓
< {Book, Work, ... }	{...}	{Company, Agent, ...}	{...} >	✗

Two similarity measures:

- *d-spec (on the domain)*
- *r-spec (on the range)*

Most specific subject types in non matching signatures

Most specific subject types in matching signatures

$$d\text{-}spec(f,q) = \frac{w\text{-}card(D[\Delta_q] \cap D[\Delta_q^f])}{w\text{-}card(D[\Delta_q] \cup D[\Delta_q^f])}$$

$$w\text{-}card(T) = \sum_{t \in T} \frac{depth(t)}{\max_{t' \in DES[t]} depth(t')}$$

$$r\text{-}spec(f,q) = \frac{w\text{-}card(R[\Delta_q] \cap R[\Delta_q^f])}{w\text{-}card(R[\Delta_q] \cup R[\Delta_q^f])}$$

Each type is weighted by its depth

Experiments: Set Up

Our approach
DRC
Domain and Range Comparison

Table 1: Gold Standards' statistics.

	Facets		*Predicates*	
	#	Domains	Accurate	Correct
dbpedia-numbers	8	7	~4	~19
dbpedia-entities	31	13	~7	~7
dbpedia	39	13	~3	~9
yago-explicit	83	17	1	-
yago-ambiguous	83	10	1	-

Gold standards (122 facets)

- DBpedia [new!]
  - + properties (≈ 53K+)
  - + numbers
  - medium hierarchy
  - Ground truth: graded judgments (incorrect < correct < accurate)
- YAGO [Limaye&al.PVLDB2010]
  - - properties (= 89)
  - deep hierarchy
  - Ground truth: one correct predicate per facet

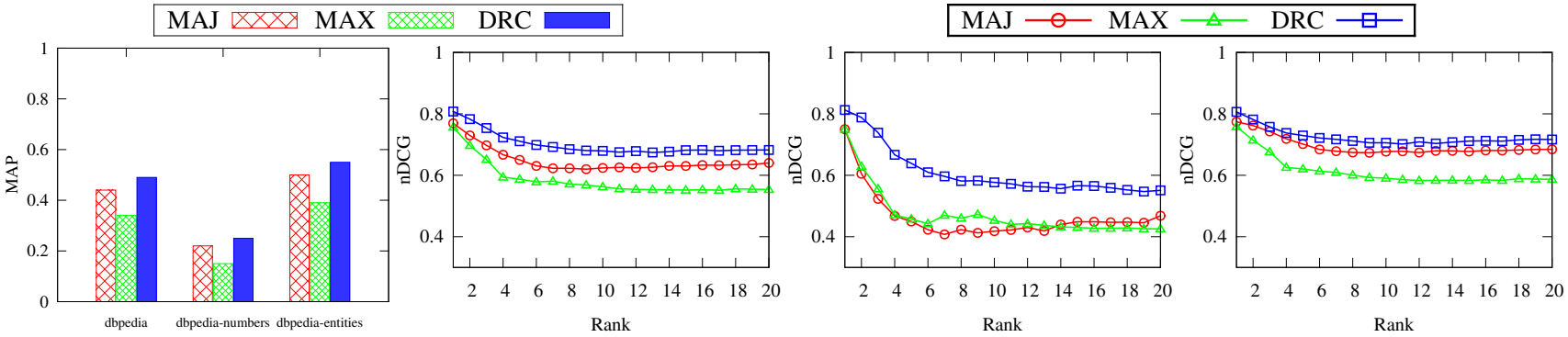
Baselines

- Majority voting (MAG)
- Maximum likelihood (MAX)
  - Adaptation of [Venetis&Al.VLDB2011]

Measures

- Mean Average Precision (MAP) – DBpedia
- Mean Reciprocal Rank (MRR) - YAGO
- Normalized Discounted Cumulative Gain (nDCG) – Dbpedia/YAGO

Experiments: DBpedia

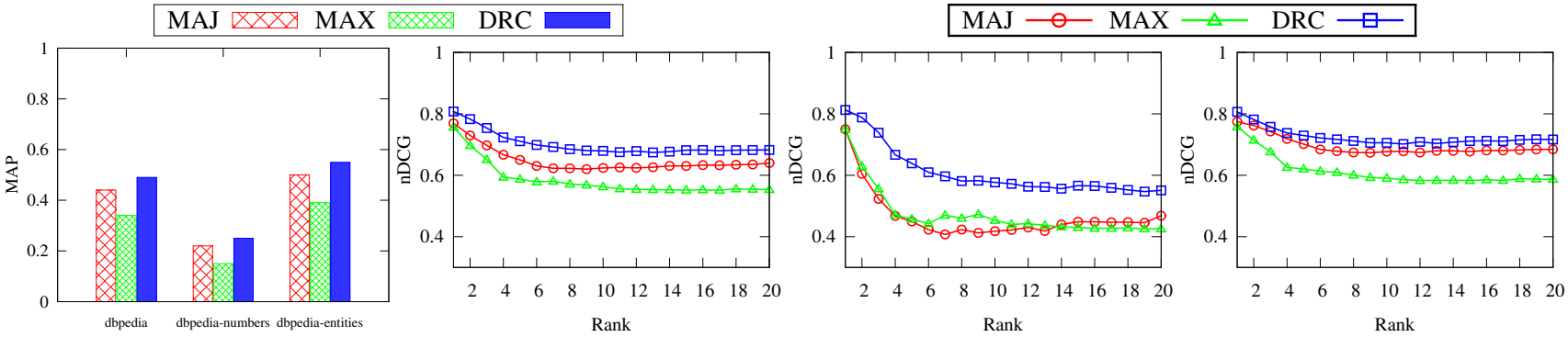


(a) Whole DBpedia dataset. (b) dbpedia. (c) dbpedia-numbers. (d) dbpedia-entities.

Highlights

- DRC always best results
- DRC particularly better with DBpedia numbers (values are more ambiguous)
- MAJ > MAX (adapted state-of-the-art approach)

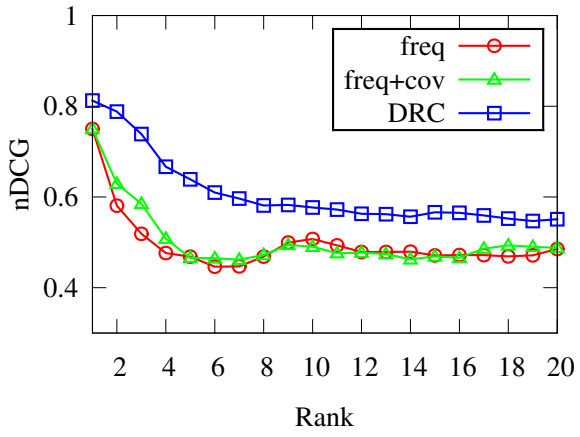
Experiments: DBpedia



(a) Whole DBpedia dataset. (b) dbpedia. (c) dbpedia-numbers. (d) dbpedia-entities.

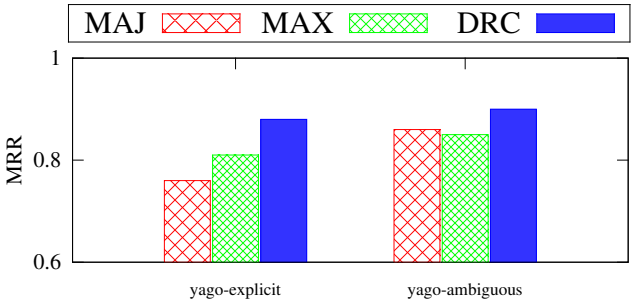
Highlights

- DRC always best results
- DRC particularly better with DBpedia numbers (values are more ambiguous)
- MAJ > MAX (adapted state-of-the-art approach)
- Specificity has impact on the quality of ranking

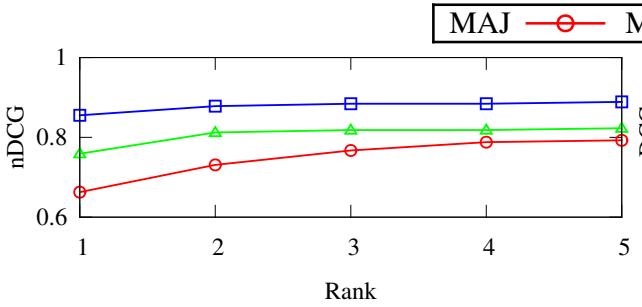


Effect of specificity on DRC

Experiments: YAGO

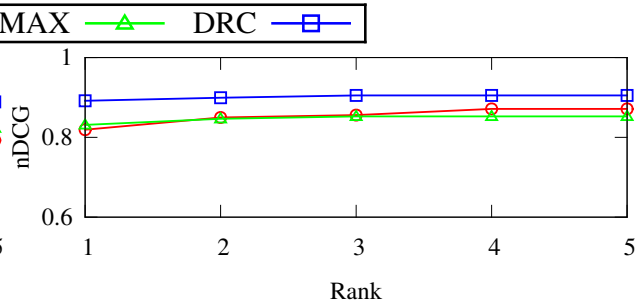


(a) Whole YAGO dataset.



(b) yago-explicit.

Facet domains:
type names in YAGO
(e.g., wikicategory-American-science-fiction-novels)



(c) yago-ambiguous.

Facet domains:
more general and intuitive than type names in YAGO
(e.g., novels)

Highlights

- DRC always best results
- DRC *and Specificity* are less effective than with DBpedia (in particular with **yago-ambigous)**
  - comparison of generic domains vs. very specific YAGO types
  - possible solution: take types of medium depth instead of most specific types

Conclusions & Future Work

- Facet annotation with reference KBs
  - Specificity-driven predicate ranking
  - End-to-end support domain experts in:
    - Creating new facets [Porrini&al.CAISE2014]
    - Labeling and lifting facets so as to transform lexical data into semantic-rich data

- Future work
  - Application to new domains (geospatial healthcare applications)
  - Improvement for KBs with very deep type hierarchy
  - Plug the approach into a table annotation framework

Contact: palmonari@disco.unimib.it

This research was partially supported by

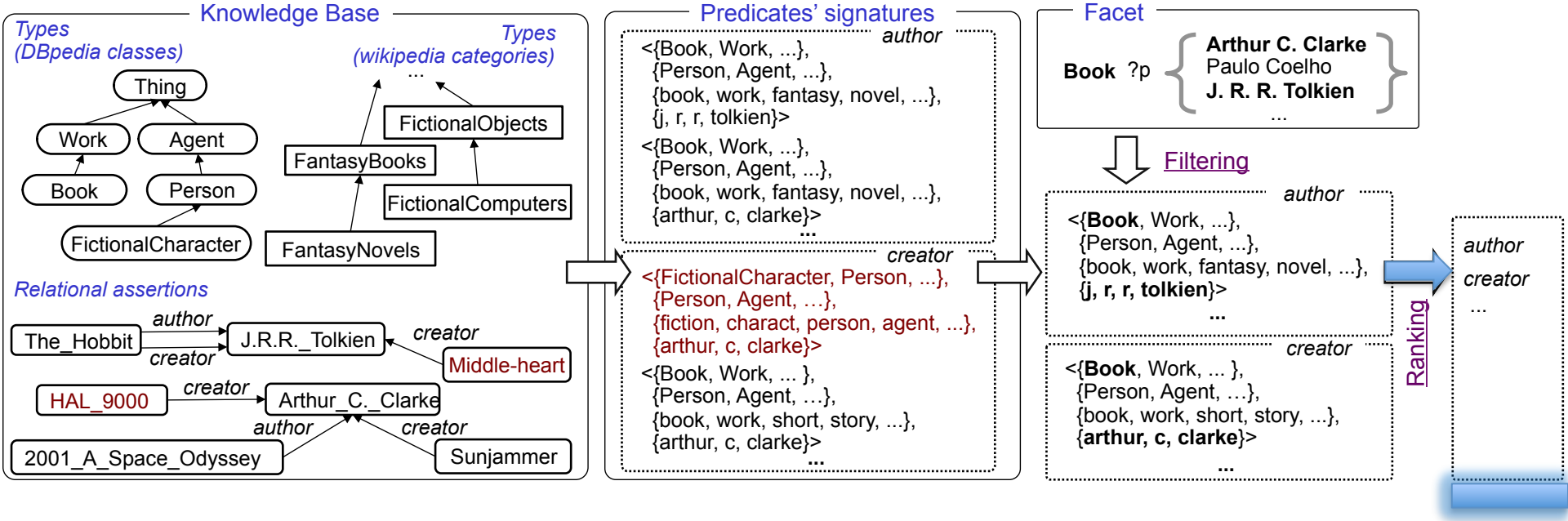
EWSHOPP
Horizon 2020
European Union Funding
for Research & Innovation

Supporting Event and Weather-based Data Analytics and Marketing along the Shopper Journey
www.ew-shopp.eu

NSF awards:
CNS-1646395, III-1618126, CCF-1331800
Bill & Melinda Gates Foundation
Grand Challenges Explorations

Additional Details

More Details: Combination of Ranking Criteria



Three criteria:

- Coverage, Weighted Frequency, Specificity

Combination:

- Smoothing for comparability
- Multiplication

$$drc(f,q) = \left(1 + \ln(d\text{-}spec(f,q) + 1)\right) \cdot \left(1 + \ln(r\text{-}spec(f,q) + 1)\right) \cdot \left(1 + \ln(cov(f,q) + 1)\right) \cdot \left(1 + \ln(freq(f,q) + 1)\right)$$