Tabular Information Extraction

Master Thesis

Tobias Matysiak

18.12.2019

UNI FREIBURG

Albert-Ludwigs-Universität Freiburg

Knowledge Base Table Extractor

Table definition

SPARQL query

city | country | population

UNI FREIBURG

```
1. PREFIX wd: <http://www.wikidata.org/entity/>
2. PREFIX wdt: <http://www.wikidata.org/prop/direct/>
3. PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

4. SELECT DISTINCT ?city_name ?country_name ?population
   WHERE {
5.      ?city wdt:P31 wd:Q515 .
6.      ?city wdt:P17 ?country .
7.      ?city wdt:P1082 ?population .
8.      ?city rdfs:label ?city_name .
9.      ?country rdfs:label ?country_name .
10.     FILTER (lang(?city_name) = "en") .
11.     FILTER (lang(?country_name) = "en") .
12.}
```

Table Definition

- Column filters
  - city(="Berlin"@en) | country | population
  - city | country | population(>= 100000)
- Column orders
  - city | country | population [DESC]
  - city [ASC, 1] | country | population [DESC, 2]
- Exact IDs
  - Q515 | country | population
- Explicit linking
  - city | country | population -> 1

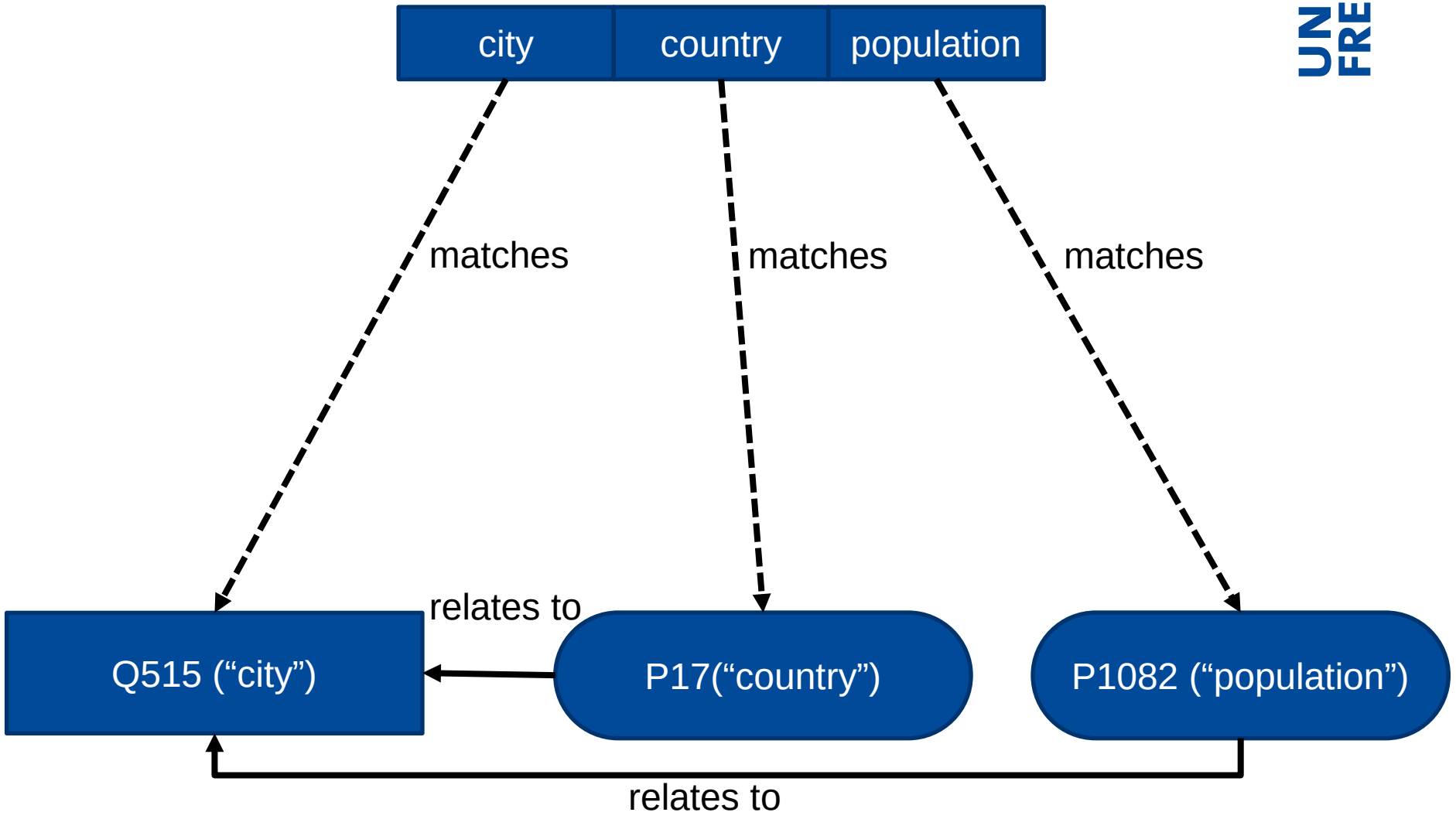
Table Matching - Problem Definition

UNI FREIBURG

- Input: table definition
- Output: SPARQL query

- Column matching
  - Find a class or property for each column
- Relations between columns
  - Each column must relate to another (except one “master column”)

Table Matching - Problem Definition



Approach

- Search index
  - find class or property for each column
- Column graph
  - find relations between two columns
  - SPARQL templates
- Pairwise matching of columns (Search index + Column graph)

Search Index

UNI FREIBURG

- Document-term matrix (tf-idf features)
  - Documents: labels of classes and properties
  - Terms: tri-grams of the labels
- Python library *scikit-learn*

CP Template (Class – Property)

UNI FREIBURG

```
1. SELECT ?o WHERE {
2.    ?s <is-a> [CLASS] .
3.    ?s [PROPERTY] ?o .
4. }
```

- `<is-a>` in Freebase: fb:type.object.type
- `<is-a>` in Wikidata: wdt:P31

- Example

CC Template (Class – Class)

UNI FREIBURG

```
1. SELECT ?p WHERE {
2.    ?s <is-a> [CLASS1] .
3.    ?o <is-a> [CLASS2] .
4.    ?s ?p ?o .
5. }
```

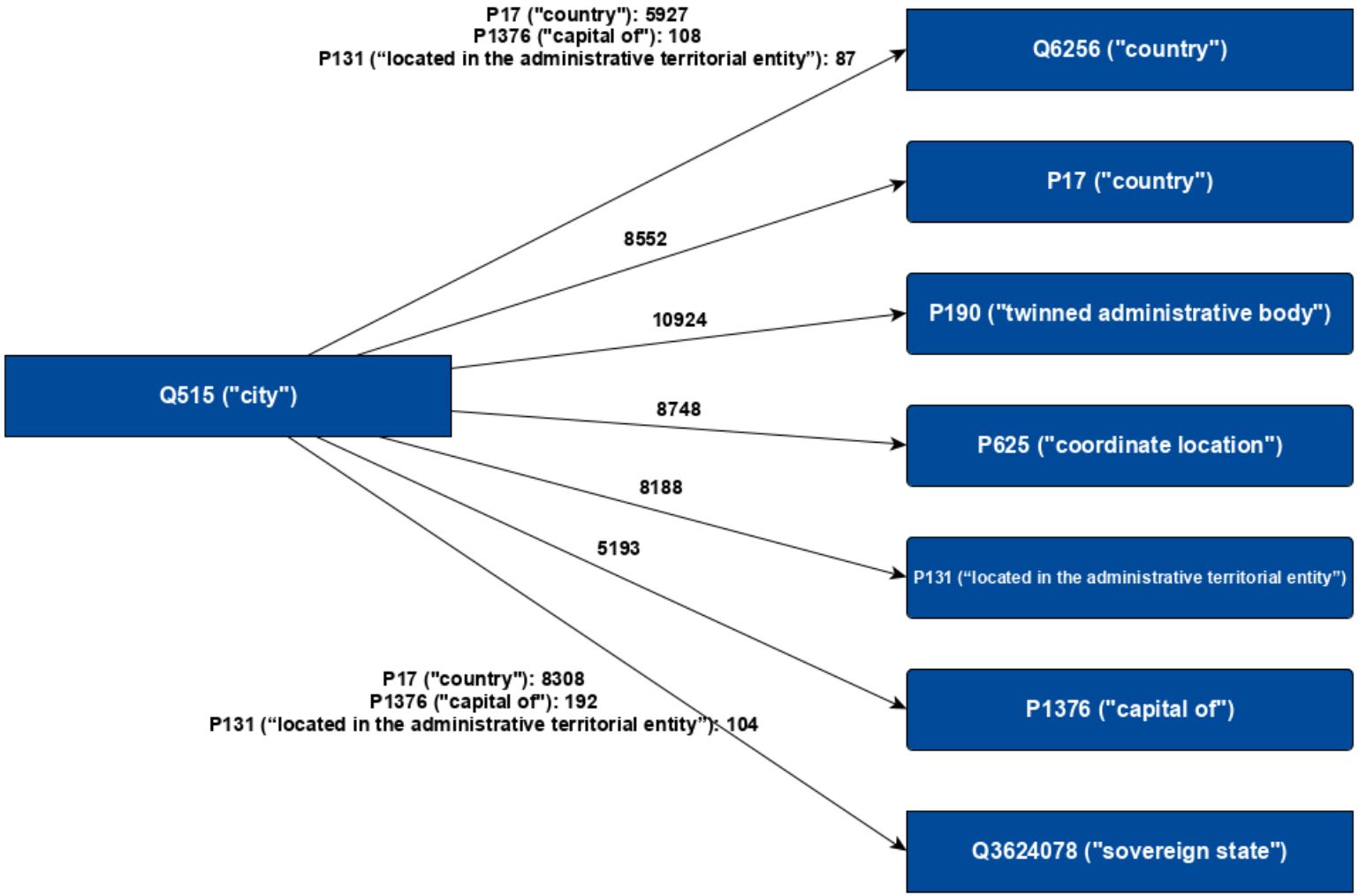
- Example

Column Graph

- A node for each class or property
- Edges between template matching nodes

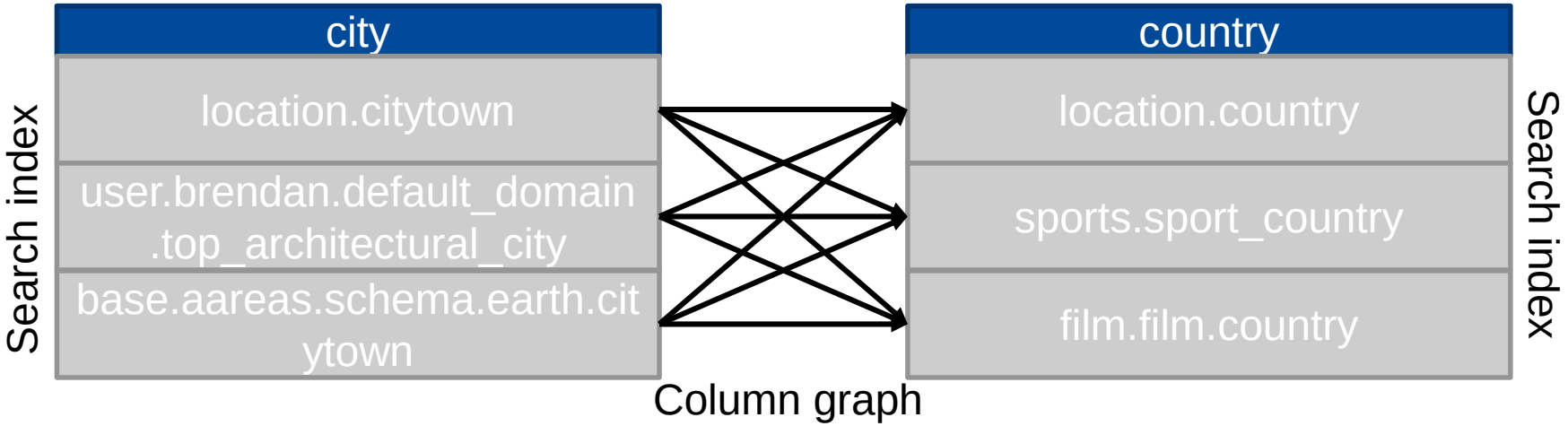
- Created from RDF dump
  - Entity dict
    - <Entity> <is-a> <Class> .
  - CC Template
    - <Entity1> <Property> <Entity2> .
  - CP Template
    - <Entity> <Property> <Value or other Non-entity> .

Column Graph



Column Pair Matching

- Input: a pair of column definitions



- Output: ranked list of best matching classes and properties
  - e.g. (location.citytown, location.country)
  - e.g. (location.citytown, film.film.country)

Baseline Table Matching Algorithm

1. For each column find best matching class or property using the search index
2. Pairwise matching of all best matches
3. Find best related column for each column

4. Master column: The column which is the best related column the most often

KBTE Table Matching Algorithm

1. Pairwise matching of all columns
2. Based on pairwise matchings
   - Rank columns → master column candidates
   - Create a ranking of the best classes for each candidate
3. Try each candidate and class
   - If each other column has a related column without contradiction → table matching found

Evaluation

- Evaluation on 5000 Wikipedia tables
- Evaluation on 15 handcrafted tables for Freebase and Wikidata

Wikipedia Tables – Dataset Creation

- Tables created from Wikipedia dump
  - 14,051,148 articles (18,665,935 pages)
    ⇩ Python library *WikiTextParser*
  - 2,372,431 tables
    ⇩ Filtering & Preprocessing
  - 1,066,292 tables

- Removing of references (`<ref></ref>`) and line breaks (`</br>`)
- Links (`[[target|text]]`) are replaced by their text
- Only tables with row span and col span 1
- Only tables where >50% of the columns are known to search index

Wikipedia Tables – Results

Outcome	Baseline FB	KBTE FB	Baseline WD	KBTE WD
kbte-error	3345 (85.4%)	2938 (75.0%)	3793 (96.9%)	3379 (86.3%)
sparql-error	46 (1.2%)	66 (1.7%)	2 (0.1%)	17 (0.4%)
sparql-empty	210 (5.4%)	436 (11.1%)	26 (0.7%)	247 (6.3%)
sparql-full	314 (8.0%)	475 (12.1%)	94 (2.4%)	272 (6.9%)
# tabledefs	3915			

- *kbte-error*: Error during table matching
- *sparql-error*: Error during execution of SPARQL query
- *sparql-empty*: Execution of SPARQL query yields no results
- *sparql-full*: Execution of SPARQL yields results

Handcrafted Tables – Wikidata (1)

Table number	Wikidata table definition & Ground truth
1 (Persons)	human | birth | death | spouse Q5 | P569 | P570 | P26
2 (Cities 1)	city | country | population | location Q515 | P17 | P1082 | P625
3 (Cities 2)	city | country | capital Q515 | P17 | P1376
4 (Mountains)	mountain | country | elevation | mountain range Q8502 | P17 | P2044 | P4552
5 (Movies 1)	film | publication date | genre | country | director Q11424 | P577 | P136 | P495 | P57
6 (Movies 2)	film | date | genre | country | director Q11424 | P577 | P136 | P495 | P57
7 (Books 1)	book | author | publication date Q571 | P50 | P577
8 (Books 2)	book | author | date Q571 | P50 | P577

Handcrafted Tables – Wikidata (2)

Table number	Wikidata table definition & Ground truth
9 (Space missions)	mission | human | occupation(="astronaut"@en) Q2133344 | Q5 | P106(=”astronaut”@en)
10 (Politicians)	human | occupation(="politician"@en) | gender | country | date of birth Q5 | P106(=”politician”@en) | P21 | P17 | P569
11 (Super Bowls)	super bowl | point in time | location | winner Q32096 | P585 | P276 | P1346
12 (Sports teams)	club | city | sport | league | venue Q847017 | Q515 | P641 | P118 | P115
13 (Buildings)	building | architect | country | height | floors Q41176 | P84 | P17 | P2048 | P1101
14 (Airports)	airport | city | country | patronage Q1248784 | Q515 | P17 | P3872
15 (Chemical elements)	element | number | symbol | mass Q11344 | P1086 | P246 | P2067

Handcrafted Tables – Freebase

Table number	Freebase table definition
1 (Persons)	person | birth | death | spouse
2 (Cities 1)	city | country | population | latitude | longitude
3 (Cities 2)	city | country | capital
4 (Mountains)	mountain | country | elevation | mountain range
5 (Movies 1)	film | release date | genre | country | director
6 (Movies 2)	film | date | genre | country | director
7 (Books 1)	book | author | publication date
8 (Books 2)	book | author | date
9 (Space missions)	mission | astronaut | start | end
10 (Politicians)	politician | gender | country | date of birth
11 (Super Bowls)	super bowl | date | location | champion | runner-up | result
12 (Sports teams)	sports teams | location | sport | league | venue
13 (Buildings)	building | architect | country | height | floors
14 (Airports)	airport | city | country | passengers
15 (Chemical elements)	element | number | symbol | mass

Handcrafted Tables - Results

UNI FREIBURG

Table number	Baseline FB	KBTE FB	Baseline WD	KBTE WD
1 (Persons)	3 / 4 (75%)	4 / 4 (100%)	4 / 4 (100%)	4 / 4 (100%)
2 (Cities 1)	4 / 5 (80%)	5 / 5 (100%)	3 / 4 (75%)	3 / 4 (75%)
3 (Cities 2)	2 / 3 (67%)	3 / 3 (100%)	1 / 3 (33%)	2 / 3 (67%)
4 (Mountains)	2 / 4 (50%)	4 / 4 (100%)	2 / 4 (50%)	4 / 4 (100%)
5 (Movies 1)	2 / 5 (40%)	5 / 5 (100%)	4 / 5 (80%)	5 / 5 (100%)
6 (Movies 2)	2 / 5 (40%)	5 / 5 (100%)	4 / 5 (80%)	5 / 5 (100%)
7 (Books 1)	1 / 3 (33%)	3 / 3 (100%)	2 / 3 (67%)	3 / 3 (100%)
8 (Books 2)	1 / 3 (33%)	3 / 3 (100%)	2 / 3 (67%)	3 / 3 (100%)
9 (Space missions)	kbte-error	4 / 4 (100%)	2 / 3 (67%)	2 / 3 (67%)
10 (Politicians)	2 / 4 (50%)	4 / 4 (100%)	3 / 5 (60%)	5 / 5 (100%)
11 (Super Bowls)	kbte-error	2 / 6 (33%)	kbte-error	4 / 4 (100%)
12 (Sports teams)	kbte-error	1 / 5 (20%)	4 / 5 (80%)	4 / 5 (80%)
13 (Buildings)	kbte-error	3 / 5 (60%)	3 / 5 (60%)	4 / 5 (80%)
14 (Airports)	kbte-error	4 / 4 (100%)	3 / 4 (75%)	3 / 4 (75%)
15 (Chemical elements)	kbte-error	3 / 4 (75%)	kbte-error	4 / 4 (100%)
overall	**19 / 64 (30%)**	**53 / 64 (83%)**	**37 / 61 (61%)**	**55 / 61 (90%)**

Conclusion

- KBTE algorithm is more robust and more accurate than baseline algorithm
- KBTE has difficulties with Wikipedia tables
  - Wikipedia tables often lack context
  - Table content may not be contained in FB & WD
- Possible improvements for KBTE (Wikidata)
  - Usage of alternative labels
  - Class hierarchy
  - Allow professions as columns

Summary

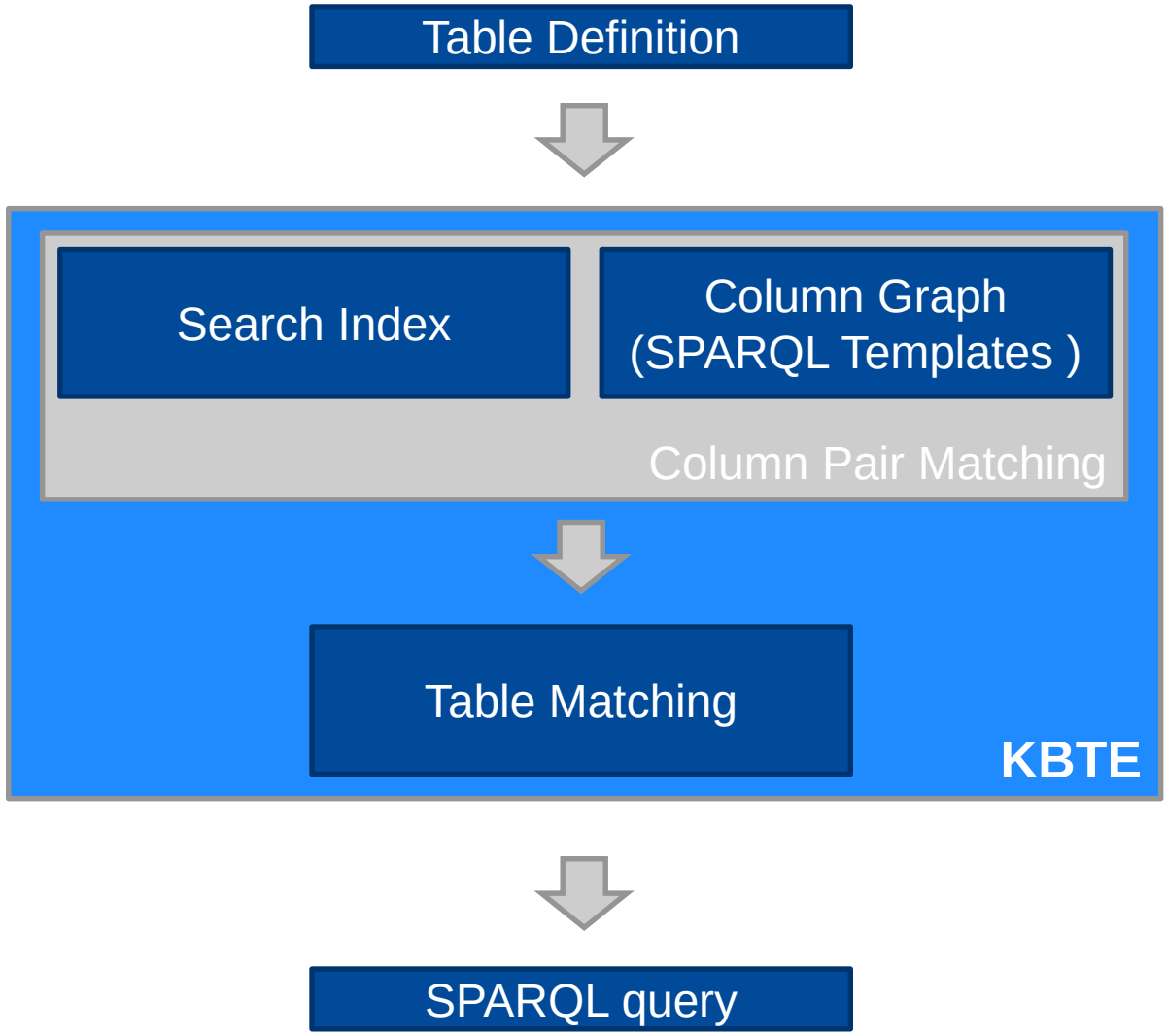


Table Definition
Search Index
Column Graph
(SPARQL Templates)
Column Pair Matching
Table Matching
KBTE
SPARQL query

UNI FREIBURG

Thank you for your attention!

Tabular Description Format

UNI FREIBURG

```
tabledef = columndef (separator columndef)*
columndef = string [filter] [order] [link]
filter = "(" comparator (string | number) ")"
order = "[" ("asc" | "desc") ("," number) "]"
link = "->" number
separator = "|"
string = CHAR+
number = DIGIT+
comparator = "!=" | "<=" | "<" | ">=" | ">" | "="
```

Search Index

Documents (classes and properties)	Terms (tri-grams)															
	␣pa	␣ra	ain	ang	ass	in␣	mou	n␣p	n␣r	nge	nta	oun	pas	ran	tai	unt
Q8502 ("mountain")			1				1				1	1			1	1
P4552 ("mountain range")		1	1	1		1	1		1	1	1	1		1	1	1
Q133056 ("mountain pass")	1		1		1	1	1	1			1	1	1		1	1

Figure 3.1: Term frequencies of three exemplary Wikidata documents

$$A = \begin{pmatrix} 0 & 0 & 0.41 & 0 & 0 & 0 & 0.41 & 0 & 0 & 0 & 0.41 & 0.41 & 0 & 0 & 0.41 & 0.41 \\ 0 & 0.36 & 0.21 & 0.36 & 0 & 0.28 & 0.21 & 0 & 0.36 & 0.36 & 0.21 & 0.21 & 0 & 0.36 & 0.21 & 0.21 \\ 0.39 & 0 & 0.23 & 0 & 0.39 & 0.29 & 0.23 & 0.39 & 0 & 0 & 0.23 & 0.23 & 0.39 & 0 & 0.23 & 0.23 \end{pmatrix}$$

$$q_{range}^{\top} = \begin{pmatrix} 0 & 0 & 0 & 0.58 & 0 & 0 & 0 & 0 & 0 & 0.58 & 0 & 0 & 0 & 0.58 & 0 & 0 \end{pmatrix}$$

$$q_{range} \cdot A^{\top} = \begin{pmatrix} 0 & 0.63 & 0 \end{pmatrix}$$

Figure 3.2: Example calculation of the relevance scores for the input "range"