Extracting Descriptions of Location Relations from Implicit Textual Networks

Andreas Spitz, Gloria Feher, Michael Gertz

Heidelberg University, Institute of Computer Science
Database Systems Research Group

{spitz,gertz}@informatik.uni-heidelberg.de
{feher}@stud.uni-heidelberg.de

11th GIR Workshop
Heidelberg, November 30, 2017

What are the relations between

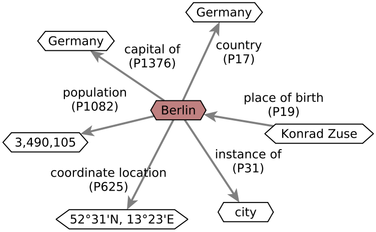
Berlin and Vienna?

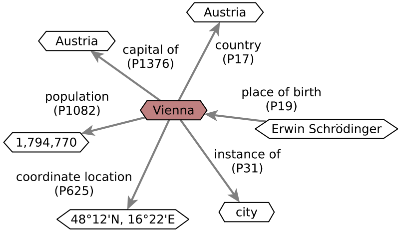
source: cdn.getyourguide.com

source: www.wien.info

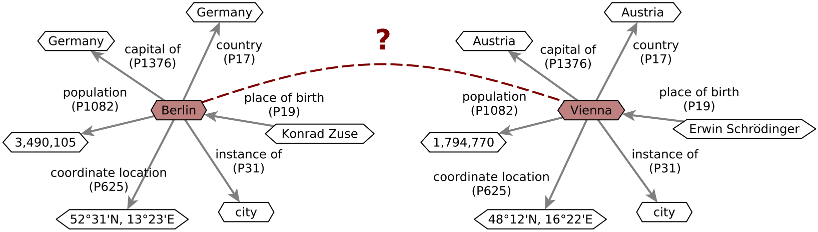
Relations between Berlin and Vienna

both are capitals
spoken language is German
located in Europe
population $> 1{,}000{,}000$





source: www.wikidata.org



source: www.wikidata.org

How can we extract other non-trivial connections from texts?

Outline

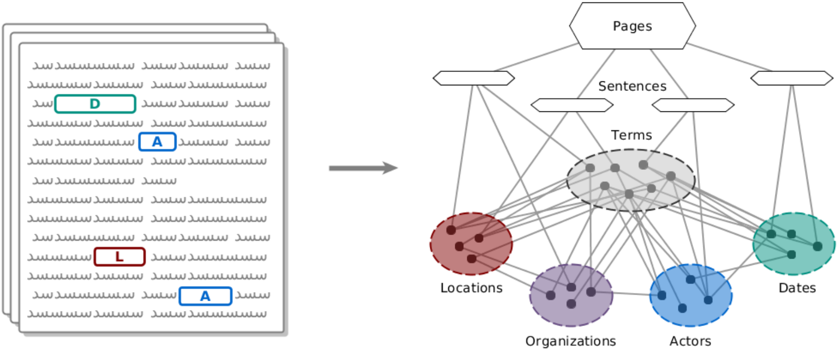
(1) The what and why of implicit textual networks

(2) Identifying related locations and geo-entities

(3) Extracting descriptive sentences

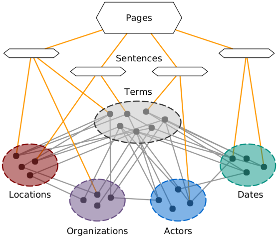
(4) Exploratory results and discussion

What is an Implicit Network?



Spitz and Gertz, *Terms over LOAD* (2016)

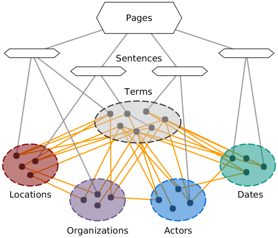
Implicit Network Edge Weights

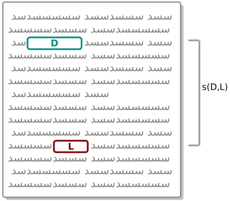


For edges (x, y) in which y is a page or sentence, count only (co-) occurrences:

$$\omega(x, y) = \begin{cases} 1 & \text{if } y \text{ contains } x \\ 0 & \text{otherwise} \end{cases}$$

Implicit Network Edge Weights





For edges (x, y) in which y is a page or sentence, count only (co-) occurrences:

$$\omega(x,y) = \begin{cases} 1 & \text{if } y \text{ contains } x \\ 0 & \text{otherwise} \end{cases}$$

For edges (x, y) between entity types and terms, aggregate co-occurrence instances I: sum over similarities derived from sentence distances s.

$$\omega(x,y) := \sum_{i \in I} \exp(-s(x,y,i))$$

Why Use Implicit Networks?

Existing approaches

- Knowledge Extraction
  $\Rightarrow$ Limited by identifiable patterns or predicates

Why Use Implicit Networks?

Existing approaches

- Knowledge Extraction
  $\Rightarrow$ Limited by identifiable patterns or predicates
- Summarization
  $\Rightarrow$ Severe scaling limitations for large input collections

Why Use Implicit Networks?

Existing approaches

- Knowledge Extraction
  ⇒ Limited by identifiable patterns or predicates
- Summarization
  ⇒ Severe scaling limitations for large input collections
- Vector embeddings
  ⇒ Encode *similarity* of contexts, not *relatedness* of entities

Why Use Implicit Networks?

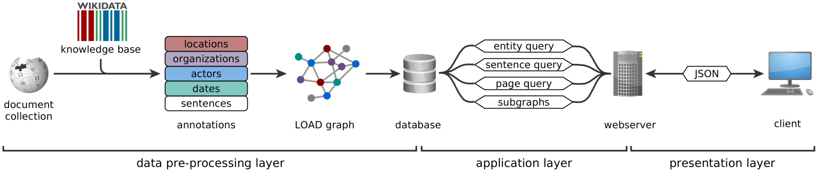
Existing approaches

- Knowledge Extraction
  $\Rightarrow$ Limited by identifiable patterns or predicates
- Summarization
  $\Rightarrow$ Severe scaling limitations for large input collections
- Vector embeddings
  $\Rightarrow$ Encode *similarity* of contexts, not *relatedness* of entities

Implicit networks

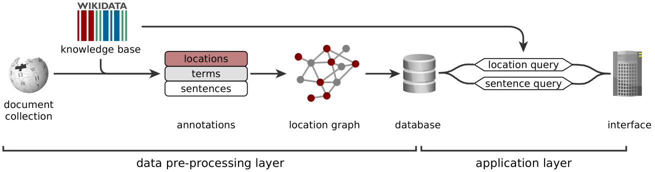
- Scale well to large document collections
- Collocation-based weights encode relatedness of entities
- Work well with dynamic text data

Implicit Network Exploration Pipeline



Spitz, Almasian, Gertz, *EVELIN* (2017)

Implicit Network Exploration Pipeline



Overview: Location Relation Extraction

Extracting descriptive sentences for pairs of locations

(1) Find closely related pairs of locations
(2) Filter relations that exist in knowledge bases
(3) Identify descriptive sentences for the remaining pairs

Identifying Closely Related Locations

Obtain a location ranking from the network by

(1) Creating weights for directed edges between nodes $x \in X$ and $y \in Y$ in entity sets X and Y in the implicit network

$$\vec{\omega}(x|y) = \omega(x,y) \log \frac{|Y|}{|N(x) \cap Y|}$$

(2) For a given query location $q \in L$, ranking all $l \in L$ by $\vec{\omega}(l|q)$

Rousseau and Vazirgiannis, *Graph-of-word* (2013)
Spitz and Gertz, *Terms over LOAD* (2016)

Location Ranking Example

Berlin (Q64)

location	wikiID	score
Germany	Q183	1.00
West Berlin	Q56036	0.42
East Germany	Q16957	0.32
Hamburg	Q1055	0.31
Munich	Q1726	0.29
Brandenburg	Q1208	0.29
Paris	Q90	0.27

Vienna (Q1741)

location	wikiID	score
Austria	Q40	1.00
Berlin	Q64	0.25
Prague	Q1085	0.23
Paris	Q90	0.19
Munich	Q1726	0.16
Austria-Hungary	Q28513	0.15
Graz	Q13298	0.14

Coverage Estimation Data

Input location data (Wikipedia):

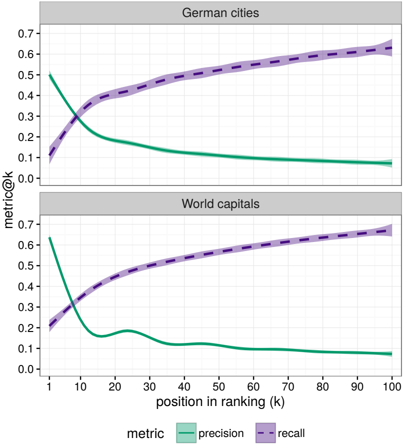
- List of largest German cities (79 locations)
- List of international capitals (250 locations)

Knowledge Base:

- Wikidata

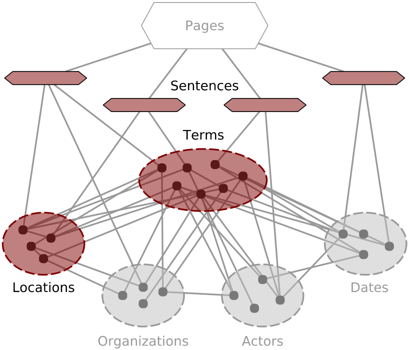
$\Rightarrow$ Inverse evaluation:
How "poorly" does the ranking reflect Wikidata properties?

Coverage of Location Relations



- Precision
  Fraction of location pairs in ranking that are connected by a property in Wikidata
- Recall
  Fraction of Wikidata properties that are in the ranked list of location relations

Sentence Extraction: Intuition



Basic Sentence Ranking Methods

Rank a sentence s by a set of query entities Q (here: locations), based on its neighbourhood $N(s)$ and a number n of relevant terms $T_n(Q)$.

Basic Sentence Ranking Methods

Rank a sentence s by a set of query entities Q (here: locations), based on its neighbourhood $N(s)$ and a number n of relevant terms $T_n(Q)$.

M1 Entity count (baseline)

$$r_1(s, Q) := |N(s) \cap Q|$$

- Rank by adjacent query entities

Basic Sentence Ranking Methods

Rank a sentence s by a set of query entities Q (here: locations), based on its neighbourhood $N(s)$ and a number n of relevant terms $T_n(Q)$.

M1 Entity count (baseline)

$$r_1(s, Q) := |N(s) \cap Q|$$

- Rank by adjacent query entities

M2 Term influence

$$r_2(s, Q, n) := |N(s) \cap Q| + \frac{|N(s) \cap T_n(Q)|}{|T_n(Q)| + 1}$$

- Rank first by entity count
- Then rank by number of contained relevant terms

Normalized Sentence Ranking Methods

Rank a sentence s by a set of query entities Q (here: locations), based on its neighbourhood $N(s)$ and a number n of relevant terms $T_n(Q)$.

Normalized Sentence Ranking Methods

Rank a sentence s by a set of query entities Q (here: locations), based on its neighbourhood $N(s)$ and a number n of relevant terms $T_n(Q)$.

M3 Normalization by length

$$r_3(s, Q, n) := \frac{1}{\log \operatorname{len}(s)} \left[|N(s) \cap Q| + \frac{|N(s) \cap T_n(Q)|}{|T_n(Q)| + 1} \right]$$

- Penalize term influence logarithmically with sentence length

Normalized Sentence Ranking Methods

Rank a sentence s by a set of query entities Q (here: locations), based on its neighbourhood $N(s)$ and a number n of relevant terms $T_n(Q)$.

M3 Normalization by length

$$r_3(s, Q, n) := \frac{1}{\log \operatorname{len}(s)} \left[|N(s) \cap Q| + \frac{|N(s) \cap T_n(Q)|}{|T_n(Q)| + 1} \right]$$

- Penalize term influence logarithmically with sentence length

M4 Normalization by count

$$r_4(s, Q, n) := \frac{|N(s) \cap Q|}{|N(s) \cap \mathcal{E}|} + \frac{|N(s) \cap T_n(Q)|}{|T_n(Q)| \cdot (|N(s) \cap \mathcal{T}| + 1)}$$

- Normalize contained query entities by total entity count
- Normalize relevant terms by total term count

Evaluation Data

Wikipedia glossary pages on

- astronomy (18)
- biology (167)
- chemistry (177)
- geology (225)

Example glossary entries (Geology)

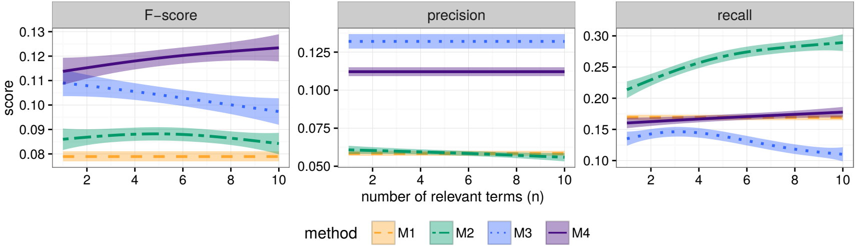
entity	wikidata	description
archipelago	Q33837	a chain or cluster of islands
tectonics	Q193343	large-scale processes affecting the structure of the earth's crust

Evaluation Results (1)

set	M1			M2		
	p	r	F1	p	r	F1
astronomy	0.069	0.207	0.099	0.064	**0.248**	0.096
biology	0.086	0.181	0.105	0.075	**0.302**	0.106
chemistry	0.039	0.180	0.062	0.044	**0.316**	0.074
geology	0.053	0.144	0.072	0.061	**0.215**	0.090
all	0.059	0.167	0.079	0.060	**0.271**	0.090

set	M3			M4		
	p	r	F1	p	r	F1
astronomy	0.078	0.184	0.097	**0.084**	0.199	**0.109**
biology	**0.212**	0.133	0.127	0.160	0.179	**0.151**
chemistry	0.082	0.149	0.093	**0.084**	0.187	**0.107**
geology	**0.114**	0.129	0.100	0.105	0.150	**0.111**
all	**0.131**	0.138	0.105	0.113	0.171	**0.121**

Evaluation Results (2)



Performance of sentence extraction methods for varying numbers of relevant terms.

Example: Athens and Sparta

Athens (Q1524) – Sparta (Q5690)

(1) Although Thebes had traditionally been antagonistic to whichever state led the Greek world, siding with the Persians when they invaded against the **Athenian-Spartan alliance**, **siding with Sparta when Athens seemed omnipotent**, and famously derailing the Spartan invasion of Persia by Agesilaus.

(2) The Greek historian Thucydides wrote in his History of the Peloponnesian War of how, in 416 BC, Athens attacked Milos for refusing to submit tribute and refusing to join **Athens' alliance against Sparta**.

(3) In the wake of this battle, Athens, Thebes, Corinth, and Argos joined together to form an **anti-Spartan alliance**, with its forces commanded by a council at Corinth.

Example: Rome and Milan

Rome (Q220) – Milan (Q490)

(1) It was set up in 1958 in Rome and now is settled in Milan and represents all the highest cultural values of **Italian Fashion**.

(2) **Italian fashion is dominated by Milan, Rome**, and to a lesser extent, Florence, with the former two being included in the top 30 **fashion capitals of the world**.

(3) Alberico Archinto (born November 8, 1698, Milan, died September 30, 1758, Rome) was an Italian cardinal and papal diplomat.

Issues and Challenges

- Interactions between entity types in different domains
- Extension to other entity types
- Extension to data from the news domain

Berlin and Vienna

Berlin Q64 – Vienna Q1741

(1) In the same way that Vienna was the center of Austrian operetta, Berlin was the center of German operetta.

Vienna's Operetta Theater, www.theater-wien.at

Implicit network exploration online

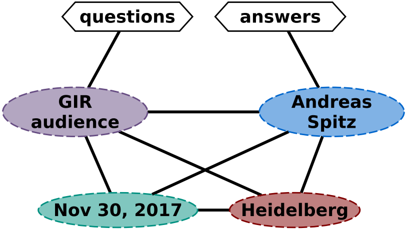
- Uses Wikipedia implicit entity network
- Location ranking
- Descriptive sentence extraction
- Subgraph exploration

http://evelin.ifi.uni-heidelberg.de

Implicit network exploration online

- Uses Wikipedia implicit entity network
- Location ranking
- Descriptive sentence extraction
- Subgraph exploration

`http://evelin.ifi.uni-heidelberg.de`



Bibliography I

François Rousseau and Michalis Vazirgiannis.
Graph-of-word and TW-IDF: New Approach to Ad Hoc IR.
In *CIKM*, 2013.

Andreas Spitz, Satya Almasian, and Michael Gertz.
EVELIN: Exploration of Event and Entity Links in Implicit Networks.
In *WWW*, 2017.

Andreas Spitz and Michael Gertz.
Terms over LOAD: Leveraging Named Entities for Cross-Document Extraction and Summarization of Events.
In *SIGIR*, 2016.