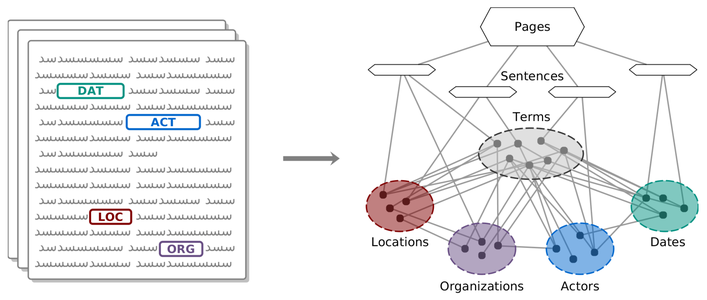
EVELIN:
Exploration of Event and Entity Links in Implicit Networks

Andreas Spitz, Satya Almasian and Michael Gertz

Database Systems Research Group, Heidelberg University, Im Neuenheimer Feld 205, 69120 Heidelberg, Germany

Representing Document Collections as Implicit Networks of Entities

Implicit entity networks are graph structures that can be constructed easily from large collections of text documents. Unlike knowledge bases, implicit networks do not encode explicit, discrete relations between entities. Instead, they represent the strength of entity relations as weights of the edges between entities. These weights support ad-hoc information retrieval on the underlying document collection. The recently introduced LOAD networks [1] include entities of the types **location**, **organization**, **actor**, and **date**, which frequently appear in the description of events. EVELIN is an interactive online tool for the exploration and analysis of such entity-centric networks.



How It Works:

For edges between entities $x \in X$ and $y \in Y$, we generate weights $\omega(x, y)$ from their textual distances δ (measured in the number of sentences) over all instances I in which x and y are mentioned together. We normalize by neighbourhood size N.

$$\omega(x, y) := \left(\log \frac{|Y|}{|N(x) \cap Y|}\right) \sum_{i \in I} e^{-\delta(x,y,i)}$$

EVELIN Online Query Interface

Demo Document Collection:

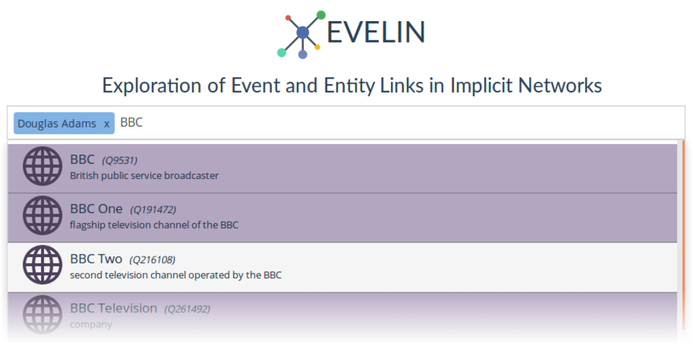
LOAD network created from the text of the English Wikipedia (excluding structured tables). All entity mentions are linked to Wikidata entities by following Wikipedia links. Temporal expressions are annotated with HeidelTime [2], entity types classified according to YAGO [3].

Online Demonstration:

The EVELIN online search interface for the demonstration data is available as a web application. The interface is mobile friendly and you can try it out for yourself at:

evelin.ifi.uni-heidelberg.de

Input Entity Selection



Composing Queries from Entities:

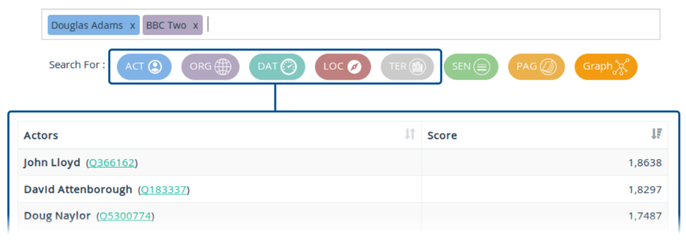
Enter a string and use the pop-up menu to select one of the suggested Wikidata entities from the database of stored entities that occur in the documents. Alternatively, press `enter` to input the string as an untyped search term. Continue adding more entities or launch a query.

Entity Queries and Term Queries

A query consists of a set of query entities Q and some target set X. They are launched by selecting the corresponding target set button. The result of a query is a ranking of all entities in X according to their relation strength to entities in the query set. An entity query results in a ranking of locations, organizations, actors or dates. Entities in the result include links to the Wikidata knowledge base. A term query returns a ranking of stemmed keywords for the query entities.

Query Extension:

Discovered entities can be added to grow the query with a double click on the entity.



How It Works:

We rank all entities x in target set X in relation to entities in the query set Q in a two component ranking scheme. It accounts for the number of common neighbours and the weights of edges between the target and query entities.

$$r(x|Q) := |N(x) \cap Q| - 1 + \frac{1}{s_{max}} \sum_{q \in Q} \omega(q, x)$$

Sentence Queries

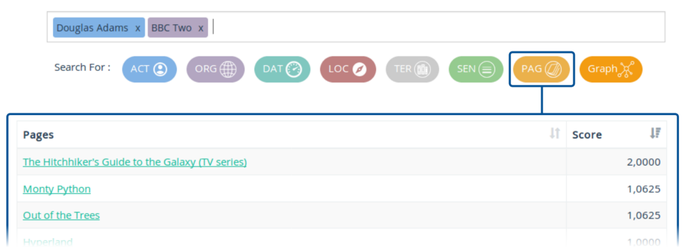
Similar to entities, sentences can be used as a query target. Here, the aim is to extract sentences that best describe an entity or a combination of entities.

How It Works:

A set of the most relevant terms T_Q for the query entities in Q is determined and used to refine the ranking of sentences s in a two component scheme.

$$r(s|Q) := |N(s) \cap Q| + \frac{|N(s) \cap T_Q|}{|T_Q| + 1}$$

Page Queries



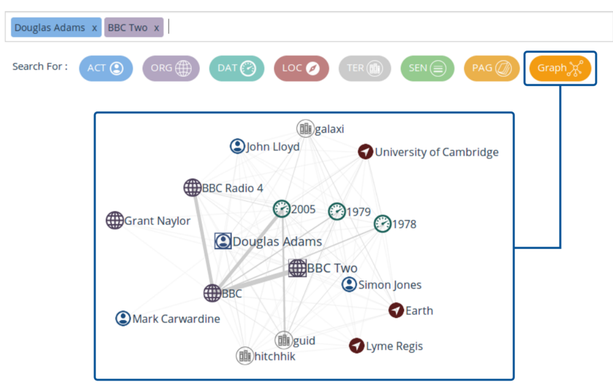
Entities or sets of entities can be linked to informative pages in the document collection that describe them well. Since we use Wikipedia as a document collection, we link entities to Wikipedia pages.

How It Works:

Ranking scores are computed for sentences s based on the input entities and then propagated to the pages p.

$$r(p|Q) := \max_{s \in p}\{|N(s) \cap Q|\} + \frac{1}{s_{max}} \sum_{s \in p} r(s|Q)$$

Subgraph Queries



For exploratory purposes, subgraphs can be extracted around the set of query entities to highlight the immediate neighbourhood. Here, we consider the three highest ranked entities of each type for the subgraph construction. Due to the density of neighbourhoods, this extraction tends to be slower than ranking queries.

References

[1] A. Spitz and M. Gertz: **Terms over LOAD: Leveraging Named Entities for Cross-Document Extraction and Summarization of Events**. *SIGIR'16*, 2016

[2] J. Strötgen and M. Gertz: **Multilingual and Cross-domain Temporal Tagging**. *Language Resources and Evaluation*, 47(2): 269–298, 2013

[3] F. Mahdisoltani, J. Biega and F. Suchanek: **YAGO3: A Knowledge Base from Multilingual Wikipedias**. *CIDR'14*, 2014

Contact Information:

Andreas Spitz

spitz@informatik.uni-heidelberg.de

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

This demonstration was presented at the 26th International World Wide Web Conference (WWW '17), April 3-7, 2017, Perth, Australia.