# Knowledge-based Relation Discovery in Cultural Heritage Knowledge Graphs

Eero Hyvönen[1,2] and Heikki Rantala[1,2]

[1] Semantic Computing Research Group (SeCo), Aalto University, Finland and
[2] HELDIG – Helsinki Centre for Digital Humanities, University of Helsinki, Finland
http://seco.cs.aalto.fi, http://heldig.fi
firstname.lastname@aalto.fi

**Abstract.** This paper presents a new knowledge-based approach for finding serendipitous semantic relations between resources in a knowledge graph. The idea is to characterize the notion of "interesting connection" in terms of generic ontological explanation patterns that are applied to an underlying linked data repository to instantiate connections. In this way, 1) semantically uninteresting connections can be ruled out effectively, and 2) natural language explanations about the connections can be created or the end-user. The idea has been implemented and tested based on a knowledge graph of biographical data extracted from life stories of 13 000 prominent historical persons in Finland, enriched by data linking to collection databases of museums, libraries, and archives. The demonstrator is in use as part of the BiographySampo portal of interlinked biographies.

## 1 Research Problem and Hypothesis

Semantic Web (SW)[3] and Linked Data (LD) technologies [3] facilitate cross-disciplinary and cross-organizational data integration. By representing and processing data using shared semantics, based on description and rule logics [6], SW data models and other standards, such as RDF, OWL, SKOS, and SWRL, data integration and reasoning can be applied to the data in a well-defined way. This is useful for semantic data interoperability, enrichment, validation, exploration, visualization, and knowledge discovery. A most promising and challenging interdisciplinary research and application area in this field has been Cultural Heritage (CH) [7] and Digital Humanities (DH) [14]. Data in this domain is heterogeneous, multi-topical, multi-lingual, comes in large quantities, is strongly contextualized by time and place, has to be personalized for users, and is created collaboratively. These challenges can be attacked using semantic web technologies.

**Research Problem** Within this area of research, this paper focuses on the problem of discovering serendipitous relations (a.k.a connections, associations) in semantically rich interlinked CH datasets, i.e., *Knowledge Graphs* (KG).

[3] For this international effort, see http://www.w3.org/standards/semanticweb/.

Serendipitous[4] knowledge discovery [1] is one of the grand promises and challenges of the Semantic Web. However, there is surprisingly little research about it. A reason for this may be that the notion of serendipity is conceptually complicated to model and measure [2] and the lack of high quality densely interlinked datasets, which are needed for finding novel connections in data.

In particular, we focus on the problem of finding "interesting" [16] connections between the resources in a KG, such as persons, places, and other named entities. Here the *query* consists of two or more resources, and the task is to find semantic relations, i.e., the *query results*, between them that are of interest to the user.

**Related Works** This problem has been addressed before in different domains. The approaches reported in the literature [2] differ in terms of the query formulation, underlying KG, methods for finding connections, and representation of the results. Some sources of inspiration for our paper are shortly reviewed below. In [15] the idea is applied for association finding in national security domain. Within the CH domain, CultureSampo[5] [8, 13] contains an application perspective where connections between two persons were searched using a breath-first algorithm, and the result was a list of arcs (such as *student-of*, *patron-of*, etc.), connecting the persons based on the Getty ULAN[6] knowledge graph of historical persons. In RelFinder[7] [12, 5, 4], based on the earlier "DBpedia Relationship Finder" [11], the user selects two or more resources, and the result is a minimal visualized graph showing how the query resources are related with each other, e.g., how is Albert Einstein related to Kurt Gödel in DBpedia/Wikipedia—both gentlemen, e.g., worked at the Princeton University. In WiSP [17], several paths with a relevance measure between two resources in the WikiData KG[8] can be found, based on different weighed shortest path algorithms. The query results are represented as graph paths.

From a methodological perspective, the main challenge in these systems is how to select and rank the interesting paths, since there are exponentially many possible paths between the query resources in a KG. This problem can be approached by focusing only on "simple paths" that do not repeat nodes, on only restricted node and arc types in the graph (e.g., social connections between persons), and by assuming that shorter, possibly weighted paths are more interesting than longer ones. For weighting paths, measures such as page rank of nodes and commonness of arcs, can be used. The problem of finding interesting connections has also addressed in the field of recommender systems [9]. However, here the goal is to find recommended interesting resources related to a given resource. For example, if a user is interested in a movie, then other movies can be recommended based on, e.g., similarity to other movies, or on statistical collaborative filtering.

[4] Serendipity means 'happy accident' or 'pleasant surprise', even 'fortunate mistake'.
[5] http://www.kulttuurisampo.fi
[6] http://www.getty.edu/research/tools/vocabularies/ulan/
[7] http://www.visualdataweb.org/relfinder.php
[8] http://wikidata.org

**Research Hypothesis** The graph-based works above make use of generic traversal algorithms that are application domain agnostic. In contrast, this paper suggests an alternative, *knowledge-based* approach to finding interesting connections in a KG. The idea is to formalize the notion of 'interestingness' [16] in the application domain using general explanation patterns that can be instantiated in a KG by using graph traversal queries, e.g., SPARQL[9]. For example, the pattern "Person $X$ created an artwork $Y$ with place $Z$ as a subject" can be used to find numerous instances of relation $< X, Y, Z >$ between $X$ (any person instance), $Y$ (any book, painting, piece of music, etc.), and $Z$ (any place) sharing the same explanation and the connection type. The benefits of this approach are: 1) Non-sense relations between the query resources can be ruled out effectively, and 2) the explanation patterns can be used for creating natural language explanations for the connections, not only graph paths to be interpreted by the end user. The price to be paid is the need for crafting the patterns and queries manually, based on application domain knowledge, as customary in knowledge-based system. The amount of work needed depends on the underlying knowledge graph, especially its versatility in different kind of arcs and nodes and how easily they can be generalized in the patterns. We believe that the number of ways in which resources can potentially be connected is in many cases manageable, which makes the approach feasible.

In the following, a case study of applying this approach is presented in the Cultural Heritage domain by using a KG of biographical data. In conclusion, lessons learned are discussed, and further research suggested.

## 2 Case Study: Semantic Relations in a Biographical Knowledge Graph

In historical research one is often interested to find out relations between things, such as people, say Leonardo da Vinci, and places, say Florence. Or perhaps the researcher is interested to find out about more general relations between certain types of things, such as Finnish novelists, and larger areas, such as South America. Our tool, Faceted Relator, can be used for solving such problems.

Faceted Relator combines ideas of faceted search [18] and relational search. The idea is to transform a KG into a set of instances of interesting relations for faceted analysis. A relation instance has the following core properties: 1) a literal natural language expression that explains the connection in a human readable form. 2) a set of properties that explicate the resources that are connected. For example, the following illustrative example of a tertiary relation $< X, Y, Z >$ connects *Leonardo da Vince* to *Vince* and to year 1452 based on the explanation "Person $X$ was born in place $Y$ in $Z$" for birth events:

```
:c123 a :BirthConnection;
  :explanation "Leonardo da Vinci was born in Vince in 1452";
  :place :vince;
  :time 1452;
```

[9] https://www.w3.org/TR/sparql11-overview/

```
    :person :Leonardo_da_Vince .

  :BirthConnection rdfs:label "Person X was born in place Y in time Z" .
```

Relation instances like this can be searched for in a natural way using faceted search, where the facets are based on the properties of the instances, that can often be organized hierarchically. In this case, there would be a facet for explanation types (such as `:BirthConnection`), and facets for places (in a partonomy), persons (that may be organizaed into a hierarchy based on, e.g., occupation or nationality), and times (in a partonomy). By making selections on the facet hierarchies, the result set is filtered accordingly and hit counts in facets recalculated. In this way, one could filter out with two clicks, e.g., the different ways in which artists are related to Italy. The query results would include Leonardo being born in Vince, but also Canaletto having painted a picture that depicts Venice, and so on.

The transformation of a KG into relation instances can be created dynamically/virtually while querying or in a separate preprocessing compilation phase. If the number of interesting connections to be pre-computed is not too large, preprocessing makes sense since it speeds up the querying substantially and makes its easier to debug the system, because all interesting connection are explicated and can be checked. For graph transformations, SPARQL CONSTRUCT queries can be used effectively. In our demonstrator, for example, each explanation pattern has a corresponding SPARQL CONSTRUCT query that extracts the corresponding connection instances to be use in faceted search. Based on ten patterns, 40 000 connections in total were found and could be generated in a preprocessing phase fairly easily.

The method outlined above was tested in the context of BiographySampo[10], a linked data service and semantic portal aggregating and serving biographical data. The knowledge graph of this system includes several interlinked datasets:

1. Biographical data extracted in RDF form from 13 144 Finnish biographies, including, e.g., 51 937 family relations, 4953 places, 3101 occupational titles, and 2938 companies.
2. HISTO ontology[11] of Finnish history including more than one thousand historical events. Data for the events includes people and places related to the event. The data was available in RDF format.
3. The Fennica National Bibliography[12] is an open database of Finnish publications since 1488. The metadata includes, among other things, the author of the book and the subject matter of the book, which can include places.
4. BookSampo[13] data covering virtually all Finnish fiction literature in RDF format, maintained by the Finnish Public Libraries consortium Kirjastot.fi.

[10] https://seco.cs.aalto.fi/projects/biografiasampo/
[11] https://seco.cs.aalto.fi/ontologies/histo/
[12] https://www.kansalliskirjasto.fi/en/services/conversion-and-transmission-services-of-metadata/open-data
[13] https://www.ldf.fi/dataset/kirjasampo/index.html

5. The Finnish National Gallery[14] has published the metadata about the works of art in their collections. The metadata is described using Dublin Core standard and was available in JSON and XML format.
6. The collected works of the J. V. Snellman portal[15] includes the texts written by J. V. Snellman, the national philosopher of Finland. The data includes, e.g., 1500 letters. We transformed the data into RDF.

The focus in our demonstrator is on finding relation instances describing connections between people and places in Finnish cultural history. However, the system can be extended to cover other kind of relations, too. The relation graph was created using SPARQL CONSTRUCT queries in two main steps. Firstly, data was extracted from sources using SPARQL CONSTRUCT queries that aim to be general and that would ideally work with different data describing similar relations as long as the data is expressed with similar patterns. In the second phase, separate SPARQL CONSTRUCT queries are used to replace the original person and place entities with entities corresponding to the ontologies used in this implementation. URIs and human readable labels for the relations are also created at this phase. In cases where the structured data was not available in RDF form, such as in the case of the data from the Finnish National Gallery, the connection instances were crated from available data using Python RDFLib-library[16]. The types of relations extracted from the data ware based on both what information was readily available and on a subjective evaluation of the interestingness of the relation types in the biographical cultural heritage context.

Relation instances were created for the ca. 13 000 core people in BiographySampo. Places were limited to those having a match in the YSO places ontology[17] that was used as the place facet. A limited set of places helps in keeping the number of relations manageable and it hopefully helps in creating more interesting relations in average by concentrating on the relations to the places generally consider interesting from a Finnish perspective, and that have therefore been selected to the YSO places ontology. This more limited set of places also helps in limiting the cases where it would be necessary to disambiguate between places and people or other things that might have the same name as a place. A simple ontology of relation types was also created, with a hierarchy expressed using the SKOS vocabulary[18]. This ontology is also available as a facet to focus search on selected relation types.

The natural language explanations of the relations were created by using simple literal templates: Each relation type (class) has a generic template to describe it, where names and other variables, such as person names or dates, are inserted to appropriate places. For example, the following template is used for explaining artistic creation relations related to painting collections:

[14] https://www.kansallisgalleria.fi/en/avoin-data/
[15] http://snellman.kootutteokset.fi/
[16] https://github.com/RDFLib/rdflib
[17] https://finto.fi/yso-paikat/en/
[18] https://www.w3.org/TR/skos-reference/

"< *personname* > has has created a work of art called < *paintingname* > in the year < *year* > that depicts < *placename* >."

The Finnish language has complicated rules for the conjugation of words, but the templates could be formulated in such a way that conjugating the variables was not necessary. The sentences generated are fully understandable Finnish, but may sometimes feel a bit artificial in style.

| **Type of Relation** | **# of Relations** |
|---|---|
| Historical event in a place | 345 |
| Letter sent from | 575 |
| Letter received from | 124 |
| Text describes a place | 881 |
| Received an award in a place | 2528 |
| Died in | 7349 |
| Painting depicts a place | 1091 |
| Novel depicts a place | 290 |
| Born in | 7182 |
| Career is related to a place | 20536 |
| In total | 40901 |

**Table 1.** The types of the relations.

Table 1 list the relations types and the number of found relation instances in our demonstrational system.

## 3 Demonstrator at Work

Faceted Relator was published as part of the BiographySampo portal, and is in use online[19] as a separate application perspective. Figure 1 depicts the user interface of the application. The data and interface are in Finnish, but there is a Google Translate button in the right upper corner of the interface for foreign users available.

In this case study, Faceted Relator can be used for filtering relations with selections in four facets seen on the left: 1) person names, 2) occupations, 3) places, and 4) relation types. The system shows a hit list of the relation instances that fit the selected filtering criteria in the facets. Each instance is represented in a row that shows first a natural language explanation of the relation, then the related person, place, and data source as links to further information, and finally the relation type. Different types of relations are highlighted in different colors and have their own symbols in order to give the user a visual overview of relations found. At any point, the distribution of the hit counts in categories along each facet can be visualized using a pie chart—one of them can be seen in the left upper corner of figure 1.

[19] http://biografiasampo.fi/yhteyshaku/

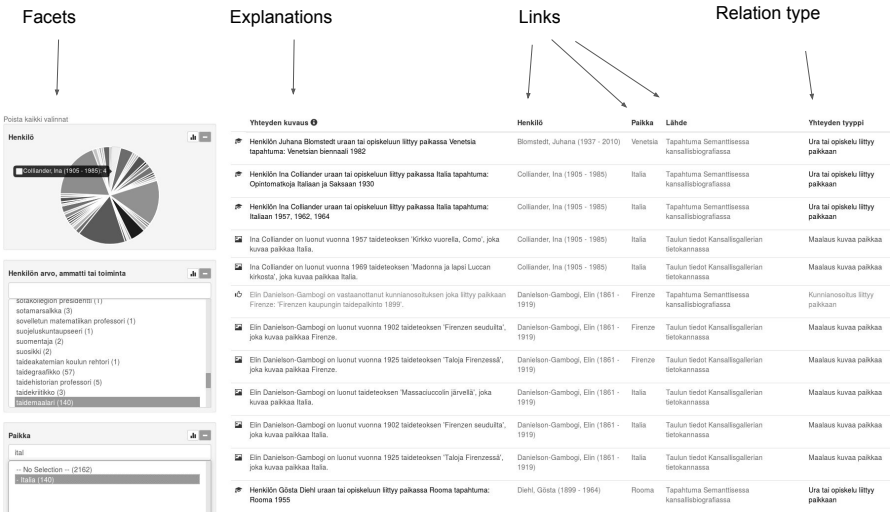


**Fig. 1.** View of the user interface

For example Kalle Päätalo (1919–2000) is an author well known in Finland for his autobiographical novels. If the user selects Kalle Päätalo from the person facet (s)he is shown a hit list of 39 relations between Päätalo and a place. 32 of those relations are of the type "novel depicts a place". Two places occurring most times in the results are Tampere and Taivalkoski. The other types of relations found concern the birth and death place of Päätalo and events in his career. Also here relations between Päätalo and Tampere and Taivalkoski are frequent. The system works as is expected and shows only relevant relations. Kalle Päätalo has described places in many of his books, and one can clearly see from the relations that the places that he depicts are also related to his own life events.

Another example can be taken from the field of art. The person with most relations of the type "created a painting that depicts a place" turns out to be Werner Holmberg with over 200 connections of that type. This is as expected, because according to his biography in BiographySampo, he was a most important Finnish landscape painter. When searching for relations of people with the occupation "painter", a couple of cities outside of Finland stand out having the most relations: Paris is clearly an important place for Finnish painters as one would expect. Also Stockholm and Saint Petersburg stand out, which should be expected as they are large cities near Finland. Other foreign cities with over 20 relations for painters include, e.g., Florence and Düsseldorf. Florence is a well known city of culture, but the connection of painters to Düsseldorf may not be so obvious. Düsseldorf was, however, an important place of study for many Finnish painters during the 19th century, including aforementioned Werner Holmberg.

The facets show hit counts for each possible selection, and hides the selections that would not yield any results. These numbers can be used not only for guid-

ing next steps in faceted search but are interesting information by themselves, especially when visualizing the relative numbers with the pie chart option. For example, by selecting the relation type "received an award in a place" in the relation type facet, and "Germany" in the place facet, one can see that not only in total 234 awards have been given to the 13 000 persons in the person facet, but also for each persons how many awards (s)he received. From the pie chart visualization it is easy to see that the person with most German awards is Carl Gustav Mannerheim, the marshal of the Finnish army in the Second World War, with 8 medals.

The demonstrator is based on an architecture with the server side consisting of a Apache Jena Fuseki[20] graph store and the client side consisting of an application written with AngularJS[21]. The faceted search was implemented with the SPARQL Faceter[22] [10] tool.

## 4 Lessons Learned and Future Research

An informal initial evaluation and testing of the demonstrator showed that it works as expected in test cases, and that a layman can potentially learn new information by using the system. However, more testing is needed to find out how interesting and surprising the results are for an expert of CH and how a system like this can be used for DH research. We also found out needs to improve the usability of the system. For example, the demonstrator now sorts results based on firstly the name of the person and secondly on the name of the place. The user should probably be offered the possibility to sort the relations freely along any facet.

When applying faceted search to relations, one should keep in mind that the hit counts refer to relation instances found. For example, if type "novel depicts place" is selected, the occupations facet shows number 48 after the occupation "professor". This does not mean that that selection would limit the search to 48 professors, like some user might expect. Instead that selection would limit the relations to 48. This might mean, for example, that there are two professors who wrote 24 novels each that depicted a place, or it might even mean that one professor wrote only one novel that depicts 48 places. Both of these scenarios would generate 48 unique relations between a person and a place.

Preprocessing the data gives benefits of speed at query time but at the cost of time needed for preprocessing and greater memory requirement for the server. In our case, the preprocessing phase could be computed reasonably fast. To give one example, extracting out the 1000 relations based on the large Fennica SPARQL end point data took about 10 seconds to compute. However the exact speed varies greatly based on, for example, the type of the relations and the source.

When using the Faceter tool, multiple hierarchical facets may create heavy SPARQL queries when used together. In the public web service, the number of

[20] https://jena.apache.org/documentation/fuseki2/
[21] https://angularjs.org/
[22] https://github.com/SemanticComputing/angular-semantic-faceted-search

hierarchical facets has therefore been limited to avoid slowing down that might arise when many people use the system at the same time. The file size for the relation graph in our case is 28.5 megabytes, and 373 000 triples. The graph size may become much larger in other applications, but current triple stores are able to handle routinely KGs that contain billions of triples. If needed, the graph size can be made smaller and the preprocessing time for generating the relation instances shorter by generating the relations dynamically in query time.

**Acknowledgements** Our research was supported by the Severi project[23], funded mainly by Business Finland.

## References

1. Christopher Baker and Kei-Hoi Cheung, editors. *Semantic Web—Revolutionizing Knowledge Discovery in the Life Sciences*. Springer–Verlag, 2007.
2. Gong Cheng, Fei Shao, and Yuzhong Qu. An empirical evaluation of techniques for ranking semantic associations. *IEEE Transactions on Knowledge and Data Engineering*, 29(11):1, 08 2017.
3. T. Heath and C. Bizer. *Linked Data: Evolving the Web into a Global Data Space (1st edition)*. Morgan & Claypool, Palo Alto, California, 2011.
4. Philipp Heim, Sebastian Hellmann, Jens Lehmann, Steffen Lohmann, and Timo Stegemann. Relfinder: Revealing relationships in rdf knowledge bases. In *Proceedings of the 4th International Conference on Semantic and Digital Media Technologies (SAMT 2009)*. Springer–Verlag, 2009.
5. Philipp Heim, Steffen Lohmann, and Timo Stegemann. Interactive relationship discovery via the semantic web. In *Proceedings of the 7th Extended Semantic Web Conference (ESWC 2010)*, volume 6088, Berlin/Heidelberg, 2010. Springer–Verlag.
6. Pascal Hitzler, Markus Krötzsch, and Sebastian Rudolph. *Foundations of Semantic Web technologies*. Springer–Verlag, 2010.
7. Eero Hyvönen. *Publishing and Using Cultural Heritage Linked Data on the Semantic Web*. Morgan & Claypool, Palo Alto, California, 2012.
8. Eero Hyvönen, Eetu Mäkelä, Tomi Kauppinen, Olli Alm, Jussi Kurki, Tuukka Ruotsalo, Katri Seppälä, Joeli Takala, Kimmo Puputti, Heini Kuittinen, Kim Viljanen, Jouni Tuominen, Tuomas Palonen, Matias Frosterus, Reetta Sinkkilä, Panu Paakkarinen, Joonas Laitio, and Katariina Nyberg. CultureSampo – Finnish culture on the Semantic Web 2.0. Thematic perspectives for the end-user. In *Museums and the Web 2009, Proceedings*. Archives and Museum Informatics, Toronto, 2009.
9. Dietmar Jannach, Markus Zanker, Alexander Felfernig, and Gerhard Friedrich. *Recommender Systems. An introduction*. Cambridge University Press, Cambridge, UK, 2011.
10. Mikko Koho, Erkki Heino, and Eero Hyvönen. SPARQL Faceter – Client-side faceted search based on SPARQL. In *Joint Proceedings of the 4th International Workshop on Linked Media and the 3rd Developers Hackshop*. CEUR Workshop Proceedings, vol. 1615, 2016.
11. Jens Lehmann, Jörg Schüppel, and Sören Auer. Discovering unknown connections—the DBpedia relationship finder. In *Proc. of the 1st Conference on Social Semantic Web (CSSW 2007)*, volume 113 of *LNI*, pages 99–110. GI, 2007.

[23] http://seco.cs.aalto.fi/projects/severi

12. Steffen Lohmann, Philipp Heim, Timo Stegemann, and Jürgen Ziegler. The relfinder user interface: Interactive exploration of relationships between objects of interest. In *Proceedings of the 14th International Conference on Intelligent User Interfaces (IUI 2010)*, pages 421–422. ACM, 2010.
13. Eetu Mäkelä, Tuukka Ruotsalo, and Hyvönen. How to deal with massively heterogeneous cultural heritage data—lessons learned in CultureSampo. *Semantic Web – Interoperability, Usability, Applicability*, 3(1), 2012.
14. Willard McCarty. *Humanities Computing*. Palgrave, London, 2005.
15. A. Sheth, B. Aleman-Meza, I. B. Arpinar, C. Bertram, Y. Warke, C. Ramakrishnan, C. Halaschek, K. Anyanwu, D. Avant, F. S. Arpinar, and K. Kochut. Semantic association identification and knowledge discovery for national security applications. *Journal of Database Management on Database Technology*, 16(1):33–53, 2005.
16. Avi Silbershactz and Alexander Tuzhilin. On subjective measures on interestingness in knowledge discovery. In *Proceedings of KDD-1995*. AAAI Press, 1995.
17. Gonzalo Tartari and Aidan Hogan. WiSP: Weighted shortest paths for RDF graphs. In *Proceedings of VOILA 2018*. CEUR Workshop Proceedings, vol. 2187, 2018.
18. D. Tunkelang. Faceted search. *Synthesis Lectures on Information Concepts, Retrieval, and Services*, 1(1):1–80, 2009.