

Linked Data Entity Summarization

Zur Erlangung des akademischen Grades eines
Doktors der Ingenieurwissenschaften
(Dr.-Ing.)

von der Fakultät für Wirtschaftswissenschaften
des Karlsruher Instituts für Technologie (KIT)

genehmigte
DISSERTATION

von

Dipl.-Inf. Univ. Andreas Thalhammer

Tag der mündlichen Prüfung: 8. Dezember 2016
Referent: Prof. Dr. Rudi Studer
Korreferentin: Prof. Dr. Dunja Mladenić

Karlsruhe 2016

This document was created on February 2, 2017

To my mother, Berta Thalhammer, who taught me to finish the things that I start.

Abstract

In recent years, the availability of structured data on the Web has grown and the Web has become more and more entity-focused. An entity can be a person, a book, a city, etc. In fact, all of these entities are connected in a large knowledge graph. In consequence, a lot of data is often available for single entities. However, in its complete form, the data is not always useful for humans unless it is presented in a concise manner.

The task of entity summarization is to identify facts about entities that are particularly notable and worth to be shown to the user. A common usage scenario of entity summarization is given by knowledge panels that are presented on search engine result pages. For producing summaries, search engine providers have a large pool of data readily available in the form of query logs, click paths, user profiles etc. This data is not openly available and emerging open approaches for producing summaries of entities can not rely on such background data. In addition, at the point of presentation, summaries are usually strongly tied to the user interfaces of the specific summary providers. This makes it difficult to compare and exchange summaries of entities. On top of that, the majority of current entity summarization approaches rely only on one knowledge base (that is often proprietary). When entity summaries are presented to the user, issues and discussions about trust, notability, data quality, and objectivity have come up in the recent years. To address this issue, it is necessary to fuse entity data from different (Web) sources.

In this work, we address the above-mentioned challenges with three main contributions:

1. We propose two lightweight entity summarization approaches that require minimal background knowledge.
2. We introduce a common API for publishing and consuming entity summaries.
3. We propose an entity-centric data fusion approach that enables an alignment of facts about entities from multiple open Web sources in a schema-agnostic way.

We evaluated the contributions individually in accordance to state-of-the-art evaluation setups, implemented prototypes, and made different research datasets publicly available. The outcomes of our experiments lead us to conclude that 1) minimal background knowledge can be leveraged for producing state-of-the-art entity summaries (as exemplified with Web link structure and usage data); 2) entity summaries share many characteristics that make a common entity summarization API feasible (demonstrated with the introduction of an API, a proof-of-concept implementation, and an empirical analysis); 3) for Web-scale entity data fusion, two factors can enable robustness against the use of different vocabularies and modeling granularities: entity-centricity and the use of path features.

Acknowledgements

Similar to entities, it is very hard to summarize all the support that I received in almost six years. This is an attempt:

I'm deeply indebted to Rudi Studer, who gave me the opportunity to work in this great environment and who interprets his role as *Doktorvater* (German for “doctoral adviser”) literally: his guidance has always been built on values, freedom, and trust.

I would like to thank Dunja Mladenčić and York Sure-Vetter for their valuable feedback on this thesis and for being part of my dissertation committee.

There are various collaborators who supported me in parts of my research and helped to push the topic: Nelia Lasierra, Achim Rettinger, Steffen Stadtmüller, Steffen Thoma, Andreas Harth, Magnus Knuth, Harald Sack, Antonio J. Roa-Valverde, Ioan Toma, Dieter Fensel, Thimo Britsch, Kalpa Gunaratna, and Gong Cheng. I would like to thank all of them for the interesting discussions and the good collaboration.

Thanks also to Pascal Hitzler and his group for hosting me at the Wright State University in Ohio. I was very warmly welcomed there and it is my wish to give this back at some day.

Further, I would like to thank all colleagues from the Rudiverse who made this ride a truly enjoyable experience. The brightness of this environment and the general team spirit has left some signatures in the development of the Semantic Web and I'm very confident that some more will follow. Also, I'm grateful to the nice people that I met in and around the University of Innsbruck.

I would also like to thank my friends from childhood, the friends from the University of Passau and the University of Glasgow, all the flat mates, and everyone else who stayed in touch somehow.

Finally, I would like to thank my siblings, Regina and Johannes, my parents, Berta and Johann, and all my family for their support and unconditional love.

Contents

Abstract	v
Acknowledgements	vii
List of Figures	xiii
List of Tables	xvii
List of Listings	xix
List of Abbreviations	xxi
1. Introduction	1
1.1. Motivation	2
1.1.1. Scenario: Annotated Hypertext	6
1.1.2. Problem Statement	8
1.2. Research Questions and Contributions	8
1.3. Previous Publications	10
1.4. Impact	12
1.5. Guide to the Reader	13
2. Foundations and State of the Art	15
2.1. The Semantic Web	15
2.1.1. The Resource Description Framework	15
2.1.2. RDF Schema	19
2.1.3. SPARQL Query Language	20
2.1.4. Linked Data	22
2.1.5. RDF Knowledge Bases	26
2.2. State of the Art	29
2.2.1. Entity Summarization	29
2.2.2. Entity Presentation Based on Class Summaries	35
2.2.3. Ontology Summarization	37
2.2.4. Semantic Search	38
2.2.5. Faceted Search	39
2.2.6. Ranking RDF Data	40
2.2.7. Entity Recommendation	41
2.2.8. Ranking and Summarization in Databases	42
2.2.9. Automatic Natural Language Text Summarization/Generation	42

2.2.10. Question Answering over Linked Data	43
2.2.11. Pathfinding in Knowledge Bases	45
2.2.12. Summary	46
3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities	49
3.1. LinkSUM: Using Link Analysis for Entity Summarization	49
3.1.1. Introduction	50
3.1.2. Approach: LinkSUM	52
3.1.3. Experiments	57
3.1.4. Discussion	64
3.1.5. Related Work	67
3.1.6. PageRank Variants	69
3.1.7. Conclusions	81
3.2. UBES: Leveraging Usage Data for Entity Summarization	82
3.2.1. Introduction	82
3.2.2. Approach: UBES	84
3.2.3. Experiments	88
3.2.4. Discussion	97
3.2.5. Related Work	98
3.2.6. Conclusions	99
4. Interfacing Entity Summarization	101
4.1. Introduction	101
4.2. Approach: SUMMA API	103
4.2.1. SUMMA Vocabulary	105
4.2.2. RESTful Web Service	107
4.3. Experiments	109
4.3.1. Evaluation	110
4.3.2. Implementation	112
4.4. Discussion	116
4.5. Related Work	118
4.6. Conclusions	121
5. Towards Entity Data Fusion	123
5.1. Introduction	123
5.2. Approach: Entity Data Fusion	127
5.2.1. Record Linkage	127
5.2.2. Data Retrieval	128
5.2.3. Feature Extraction	128
5.2.4. Clustering	130
5.2.5. Cluster Merging	131
5.2.6. Representative Selection	132
5.2.7. Filtering / Ranking	133
5.3. Experiments	134
5.3.1. Dataset	135

5.3.2. Baseline: Sig.ma	136
5.3.3. System Configuration	137
5.3.4. Evaluation Setup	138
5.3.5. Evaluation Results	140
5.4. Discussion	141
5.5. Related Work	143
5.6. Conclusions	144
6. Conclusions	147
6.1. Discussion of Contributions	147
6.1.1. Integration of Contributions	149
6.2. Overall Conclusions	151
6.3. Outlook	152
Bibliography	155
A. Appendix	179
A.1. UBES	179

List of Figures

1.1.	High-level view on entity summarization: The input involves the knowledge graph containing the target entity and all of its (incoming and outgoing) relations (the relations are not ranked—the connecting edges have the same length). The output involves the target entity and a ranked top- k subset of its relations (shorter edges indicate higher importance of the relation for the target entity).	4
1.2.	Screenshot of qSUM, a program that shows entity summaries for semantically annotated texts.	6
1.3.	Overview of the contributions of this thesis: link-analysis-based (1 – Research Question 1.1) and usage-data-based (2 – Research Question 1.2) entity summarization, a common application programming interface (API) for entity summarization (3 – Research Question 2), entity data fusion (4 – Research Question 3). The dashed border of the box of Contribution 4 indicates that this step is optional.	10
2.1.	Subset relation between resources and information resources.	24
2.2.	An early draft on keyword search results augmented with semantic data [GMM03].	31
2.3.	Screenshots of the Google Knowledge Graph summaries of the Japanese (left, https://google.jp , retrieved 2016-03-26) and the US (right, https://google.com , retrieved 2016-03-26) versions of the entity “John Travolta”. The Japanese version covers different features than the US version (e.g., the body height).	34
2.4.	Screenshots of entities in Yahoo’s Knowledge Graph (https://www.yahoo.com , retrieved 2016-04-07). The presented attributes of the entities are similar to selections that are produced in accordance to class summaries.	37
2.5.	Screenshot of RelFinder [HLS10] for relations between John Travolta and Samuel L. Jackson on Wikidata. Filters are active for <code>wdt:P161</code> (i.e., cast member) and for hop-2 relations (i.e., the distance via one connecting node in between).	46
3.1.	Overview of the contributions of this thesis. In this part, we focus on LinkSUM, an entity summarization approach that utilizes links between entities.	50
3.2.	Screenshot of a Google Knowledge Graph summary of the entity “Pulp Fiction” (http://g.co/kg/m/0f4_1 , retrieved 2016-07-30).	51

List of Figures

3.3.	Web links (black, solid) and semantic relations (blue, dashed) between “Quentin Tarantino” and “Pulp Fiction”.	53
3.4.	LinkSUM (SPO) average <i>Quality</i> scores with different settings for α and different relation selection approaches for top-5 summaries.	60
3.5.	LinkSUM (SPO) average <i>Quality</i> scores with different settings for α and different relation selection approaches for top-10 summaries.	60
3.6.	Excerpt of the interface for qualitative evaluation for the entity “The Cosby Show”. The users could choose whether they prefer the summary of LinkSUM (left) or FACES (right) in a SERP setting.	63
3.7.	Results of the qualitative evaluation. The x-axis denotes the respective entities and the y-axis accounts for the number of user votes per system.	65
3.8.	Activity diagram for computing PageRank scores. A set of Web pages (with links) serves as an initial input.	69
3.9.	Transitive resolution of a redirect in Wikipedia. <i>A</i> and <i>C</i> are full articles and <i>B</i> is called a “redirect page”, <i>PL</i> are page links, and <i>PL^R</i> are page links marked as a redirect (e.g., #REDIRECT [[United Kingdom]]). The two page links from <i>A</i> to <i>B</i> and from <i>B</i> to <i>C</i> are replaced by a direct link from <i>A</i> to <i>C</i> .	72
3.10.	Overview of the contributions of this thesis. In this part, we focus on UBES, an entity summarization approach that leverages usage data.	82
3.11.	Example: The feature <code>:director :Quentin_Tarantino</code> that the movie “Pulp Fiction” shares with its <i>k</i> -nearest neighbors is considered more important than the feature <code>:genre :Black_Comedy</code> that “Pulp Fiction” shares with a non-neighboring movie.	85
3.12.	Screenshots of <i>WhoKnows?Movies!</i> exemplifying <i>One-To-One</i> and <i>One-To-N</i> questions and their correct or incorrect answers (for better readability, the IRIs were changed).	93
4.1.	Overview of the contributions of this thesis. In this part, we focus on the SUMMA API, an interaction mechanism for sharing and exchanging entity summaries.	101
4.2.	The SUMMA Vocabulary. Mandatory parameters in grey.	105
4.3.	RESTful interaction mechanism for entity summaries. Messages for first interaction: white. Messages for second interaction: grey	109
4.4.	Screenshot of the GKG representation of the “Ramones”: 1) Specific predicates such as the type and the Wikipedia description are always there (Property Restriction). 2) Several statements are gathered in a group named “Songs” (Statement Groups). 3) N-ary relations—in this case title, year, and album—are supported (Multi-hop Search Space). 4) The summary is offered in multiple languages (Languages)	110
4.5.	Basic summaries with Wikidata. This RDF knowledge base offers a high coverage of labels in different languages (in the presented case: English, Japanese, Spanish, and Chinese).	113
4.6.	Screenshot: Two example summaries with the same configuration but different systems (top). Example summary with restriction to two predicates (bottom left) and a different language and $topK = 3$ (bottom right).	114

4.7.	Automatically annotated excerpt of a Wikipedia article and the <i>summa-Client</i> knowledge panel with a summary by LinkSUM.	115
4.8.	Summary of <code>dbr:Barack_Obama</code> (left) and the ranked list of statements with <code>dbo:birthPlace</code> and <code>dbr:Hawaii</code> (right).	119
5.1.	Overview of the contributions of this thesis. In this part, we focus the fusion of duplicate/similar facts that are provided as Linked Data.	123
5.2.	Mock-up of a trustable knowledge panel (based on a Google screenshot). The colors of the buttons implement a traffic light scheme for the trustability of the presented fact. By clicking on such a button, a pop up would open that provides direct reference to documents which cover the presented fact as well as each document's individual presentation.	125
5.3.	Data fusion processing pipeline.	127
5.4.	Output of the data retrieval step: an RDF graph that contains a forest of trees, each with a reference IRI as a root.	129
5.5.	Number of different sources for each entity (the ticks on the x-axes each represent one entity of the TREC dataset).	135
5.6.	Number of path features and clusters before/after the merging step (the ticks on the x-axes each represent one entity of the TREC dataset).	135
5.7.	Number of clusters with more than one source (the ticks on the x-axes each represent one entity of the TREC dataset).	136
6.1.	Overview of the internally integrated contributions of this thesis.	150
6.2.	Screenshot of the ELES demonstrator, a combination of internally and externally integrated components. The service is available online at http://people.aifb.kit.edu/ath/ELES/ (as of July 2016).	151

List of Tables

2.1.	Example for a SPARQL result.	21
2.2.	Overview of entity summarization approaches and related fields.	47
3.1.	Example: Top-20 resources that have a semantic relation to “Pulp Fiction” ranked by their individual PageRank scores (DBpedia version 3.9).	54
3.2.	Resources (in no particular order) that have a semantic relation to “Pulp Fiction” and that are connected with Wikipedia Backlinks (DBpedia version 3.9).	55
3.3.	Example: Top-10 resources that have a semantic relation to “Pulp Fiction” ranked in accordance to the combined score (with $\alpha = 0.9$).	56
3.4.	Agreement among the experts.	61
3.5.	Overall <i>Quality</i> results of the quantitative evaluation and their respective standard deviation (SD). Best results are bold.† compared to the best, difference is significant ($p < 0.05$) ; ‡ compared with each of the other two settings, difference is significant ($p < 0.05$).	64
3.6.	Number of links per link graph. Duplicate links were removed in all graphs (except in ATL-RP where multiple occurrences have different positions).	78
3.7.	Number of mutually covered entities (the colors are used for better readability and have no further meaning).	78
3.8.	Correlation: Spearman’s ρ (the colors are used for better readability and have no further meaning).	79
3.9.	Correlation: Kendall’s τ on a sample of 1 000 000 (the colors are used for better readability and have no further meaning).	79
3.10.	The top-50 rankings of SubjectiveEye3D (< 0.3 , above are: Wiki, HTTP 404, Main Page, How, SDSS), DBP 2015-04, and ATL-RP.	80
3.11.	Example: User-item matrix created as an abstraction of usage data of resources.	83
3.12.	Example: The 20-nearest neighbors of the entity <code>fb:en.pulp_fiction</code> (the resource <code>fb:source.allocine.ca.film.53879</code> corresponds to the movie “Kill Bill: Volume 2” and occurs twice due to an error in a previous matching step).	86
3.13.	Top-10 features: Pulp Fiction	89
3.14.	Top-10 features: Beauty and the Beast	89
3.15.	Top-10 features: The Naked Gun - From the Files of Police Squad!	90
3.16.	Top-10 features: Bridget Jones’s Diary	90
3.17.	Example: Triples used to create a question and according answer options (for better presentation, we use the labels of the resources).	94

List of Tables

3.18. Example: Facts about Pulp Fiction that were played at least three times sorted by average correctness.	95
3.19. <i>WhoKnows?Movies!</i> predicates sorted by average answer correctness (all movies).	96
3.20. Performance of the predicate ranking.	97
3.21. Performance of the actor ranking.	97
4.1. Requirements per interface. The checked features are supported by the specific interface, the crossed ones are not required.	112
5.1. Example: Cluster statistics, cluster representative, and source representatives of a cluster of the entity “Montana State University”.	139
5.2. Results for our approach and Sig.ma: the number of produced GKG facts, GKG coverage, number of type 1 errors, number of type 2 errors, precision, recall, and f-measure at different thresholds for the number of sources. The # symbol should be read as “number of”.	140
A.1. White list of covered Freebase predicates by <i>WhoKnows?Movies!</i>	180

List of Listings

2.1.	Example for a SPARQL SELECT query.	21
2.2.	Shortened excerpt of the QALD-6 “Multilingual question answering over RDF data” task (other language versions of the question were omitted for brevity).	44
3.1.	Example: SPARQL query for retrieving the top-10 semantically related resources of “Pulp Fiction” in the order of their PageRank scores.	54
3.2.	Example: SPARQL query for retrieving resources with a Backlink to “Pulp Fiction”.	55
3.3.	Example: SPARQL query on DBpedia for retrieving top-100 universities sorted by their PageRank scores.	70
3.4.	SPARQL query: retrieving outgoing predicate-object pairs shared with at least one of the 20-nearest neighbors of <code>fb:en.pulp_fiction</code>	87
3.5.	SQL query: retrieving the Freebase identifiers given an IMDb identifier.	88
3.6.	Property chain axiom for creating direct “hasActor” relationships via OWL2 RL reasoning (Turtle syntax, the round brackets define an RDF collection [FPSH14]).	92
4.1.	Example for using vRank for ranking an RDF Statement.	106
4.2.	Example for a summary request that is sent via POST.	106
4.3.	Example response in Turtle.	108
4.4.	Full HTML example of the three interaction modes with the jQuery (UI), DBpedia Spotlight, and the SUMMA client libraries included.	117
A.1.	Used property chain rules.	179

List of Abbreviations

<i>ABL</i>	Abstract Links	74
<i>ALL</i>	All Links	74
<i>API</i>	Application Programming Interface	8
<i>ARWU</i>	Academic Ranking of World Universities	73
<i>ATL</i>	Article Text Links	74
<i>ATL – RP</i>	Article Text Links with Relative Position	74
<i>DBP</i>	DBpedia PageRank	75
<i>DBP – U</i>	DBpedia PageRank Unredirected	75
<i>DEF</i>	DBpedia Extraction Framework	71
<i>DSC</i>	Description	57
<i>ELC</i>	Entity List Completion	135
<i>EXC</i>	Exclusivity	57
<i>FRQ</i>	Frequency	57
<i>FSL</i>	Fresnel Selector Language	120
<i>GKG</i>	Google Knowledge Graph	94
<i>GWAP</i>	Game With A Purpose	84
<i>HTML</i>	HyperText Markup Language	15
<i>HTTP</i>	Hypertext Transfer Protocol	7
<i>IMDb</i>	Internet Movie Database	25
<i>IRI</i>	Internationalized Resource Identifiers	15
<i>ITS</i>	Internationalization Tag Set	116
<i>MQL</i>	Metaweb Query Language	88
<i>NLP</i>	Natural Language Processing	72
<i>OWL</i>	Web Ontology Language	23
<i>RDF</i>	Resource Description Framework	15
<i>RDFS</i>	RDF Schema	19
<i>REF</i>	Related Entity Finding	135
<i>REST</i>	Representational State Transfer	101
<i>RFC</i>	Request for Comments	106
<i>SERP</i>	Search Engine Result Page	5
<i>SO</i>	Subject–Object	60
<i>SPARQL</i>	SPARQL Protocol and RDF Query Language	20
<i>SPO</i>	Subject–Predicate–Object	60
<i>SUB</i>	SubjectiveEye3D	75
<i>TEL</i>	Template Links	74
<i>TF – IDF</i>	Term Frequency-Inverse Document Frequency	84
<i>TimBL</i>	Tim Berners-Lee	125
<i>TOWR</i>	The Open Wikipedia Ranking	75

List of Abbreviations

<i>TOWR – H</i>	The Open Wikipedia Ranking - Harmonic centrality	76
<i>TOWR – I</i>	The Open Wikipedia Ranking - Indegree	76
<i>TOWR – PR</i>	The Open Wikipedia Ranking - PageRank	75
<i>TOWR – PV</i>	The Open Wikipedia Ranking - Page views	76
<i>TREC</i>	Text REtrieval Conference	135
<i>TSV</i>	Tab-Separated Values	58
<i>Turtle</i>	Terse RDF Triple Language	16
<i>UBES</i>	Usage-Based Entity Summarization	49
<i>URI</i>	Uniform Resource Identifiers	16
<i>W3C</i>	World Wide Web Consortium	15
<i>Web</i>	World Wide Web	1
<i>YAGO</i>	Yet Another Great Ontology	27

1. Introduction

Decades before the invention of the World Wide Web (in short “Web”), Vannevar Bush expressed his concerns on how relevant information is lost in the noise of all information; in his words:

“[...] truly significant attainments become lost in the mass of the inconsequential.” [Bus45]

In order to address this issue, he suggests an appliance called “memex”: its main ideas were later reflected by the concepts of Web sites and hyperlinks. Later in his work, he continues by explaining that the access of the “common record” (i.e., the world’s collected knowledge) would also encompass a new challenge:

“Thus far we seem to be worse off than before—for we can enormously extend the record; yet even in its present bulk we can hardly consult it. This is a much larger matter than merely the extraction of data for the purposes of scientific research; it involves the entire process by which man profits by his inheritance of acquired knowledge.” [Bus45]

This led Bush to elaborate on “selection devices” (as they were at the time) and how he envisioned them for the future. It is interesting that, indeed, for retrieving information from the Web we make use of modern selection devices—**search engines**.

Soon after the first Web sites went online people started to create indexes and catalogs of Web sites. The first of these efforts “The WWW Virtual Library”¹ is still available online. However, with rapid growth of the Web also these indexes became hard to maintain and navigate. In 1994, this led to the practices of Web crawling (in order to maintain the index) and keyword search (in order to enable the direct retrieval of matching documents) which—in their core principles—have not changed since then and are still in place in most modern search engines.

Thus, many of the indexes of current search engines are based on textual representations. This naturally also accounts for keyword queries. This circumstance was pointed out as a flaw by Peter Norvig (Google’s current Director of Research)² in 2006:

“The only other comparable expansion started in 1456, with the introduction of the printing press. Fifty years and 15 million books later, the theologian Sebastian Brant wrote ‘There is nothing nowadays that our children... fail to know.’

¹The WWW Virtual Library – <http://vlib.org/>, retrieved 2016-07-13.

²As of July 13, 2016.

1. Introduction

Today, 12 years into the era of search engines, we still have not made good on Brant's boast. Search engines deliver relevance but knowledge requires human work.” [Nor06]

Examining the type of queries that current search engines receive, it becomes clear what Norvig meant: More than 40% of queries that Yahoo receives are focused on one particular **entity**³ [PMZ10]. The “knowledge” that Bush and Norvig addressed in their works is the knowledge about entities. It involves all their aspects and relations to other entities and is naturally organized in a graph—the **knowledge graph**.⁴ This structure, as Norvig pointed out, is not adequately handled by the text indexing and keyword matching paradigms that try to optimize the relevance of the retrieved Web sites in which users can seek for the requested knowledge.

Nonetheless, the knowledge graph has been growing in recent years. There are three main reasons for that development: First, the capabilities of computers to interpret natural language text have been strongly extended (e.g., [CBK⁺10, DGH⁺14a]). Second, the limiting boundaries of relational database schemes are breaking up with the introduction of graph structures (e.g., [ABK⁺07, BEP⁺08, SR13]). Third, newly created information is typically more fine-grained (e.g., context and provenance information are getting increasingly important).

The information covered by the knowledge graph is extremely versatile. It covers all kinds of language/cultural aspects, all levels of importance, and even opinions. Thus, zooming into a single entity node provides a multitude of individual and shared relations. Moreover, the knowledge does not stand by itself: its final interpretation by humans, that are primed by their personal experience (which is not covered by the knowledge graph), creates an even more unique perspective.

Both, the constant growth of the knowledge graph as well as its versatility pose new challenges to current information retrieval systems.

1.1. Motivation

Nowadays, when a user searches for an entity with a keyword query (which could be “pulp fiction”), search engines identify the meaning of the words as an entity within the knowledge graph (the node that represents the movie “Pulp Fiction”) and directly zoom into the respective node. They present the entity and its properties and relations to the user in so-called “knowledge panels” that are similar to the infoboxes⁵ of Wikipedia. However, facing the large amount of knowledge that is available about single entities, search engines

³We use the term “entity” to address real world objects or abstract concepts, more specifically: anything that can be identified (see Section 2.1.1).

⁴Here, we intentionally pitch the universality of the knowledge graph.

⁵Wikipedia infoboxes – <https://en.wikipedia.org/wiki/Help:Infobox>, retrieved 2016-07-14.

need to address the problem of “selecting general knowledge about an entity”. This is a non-trivial task that leads us to define our **initial problem**:

“How do we present knowledge graph entities?”

The naïve solution: A straight-forward solution to this problem is to define fixed property-patterns (or lenses [PBKL06]) for the presentation of entities specific to each entity type. An example for such a pattern is:

Movie: title, release date, genre, director, actor

With this method, the movie “Pulp Fiction” would be presented as follows:

- Title: *Pulp Fiction*
- Release date: *October 14, 1994*
- Genre: *Crime*
- Director: *Quentin Tarantino*
- Actor: *John Travolta, Samuel L. Jackson, Uma Thurman, Bruce Willis, ...*

Unfortunately, this encompasses a variety of problems: First, this solution is very static. Once it has been decided that the production company should not be presented for movies, it will not be shown for *any movie*. This also affects movies where the production company is an important property, for example many animation movies produced by the Walt Disney Company. Second, especially in cross-domain scenarios, the challenge to produce patterns for every type can be challenging. Eventually, we could define a pattern for the type “chicken recipe” which includes the property “chicken part”. Third, some entities have multiple types. An example is Arnold Schwarzenegger, who could be classified either as a bodybuilder, actor, or politician. With defined patterns, it is difficult to present these entities in a comprehensive way. Forth, some entity properties can have a high number of values. For example, Pulp Fiction has more than 47 actors in total; the knowledge panel would be too full if it presented all actors. A ranking of actors with respect to Pulp Fiction is not defined in the knowledge graph and the plain pattern-based approach can either present all actors, or a random selection. We summarize the problems of the naïve approach as follows.

Problems:

1. The patterns are very static and do not reflect the individual particularities of entities.
2. A pattern needs to be created for each type: automatic approaches are not straight forward and manual methods may not scale well⁶.
3. Some entities are of multiple (almost) distinct types with unclear main type.
4. Some of the properties can have many values for which no ranking or cut-off is defined.

1. Introduction

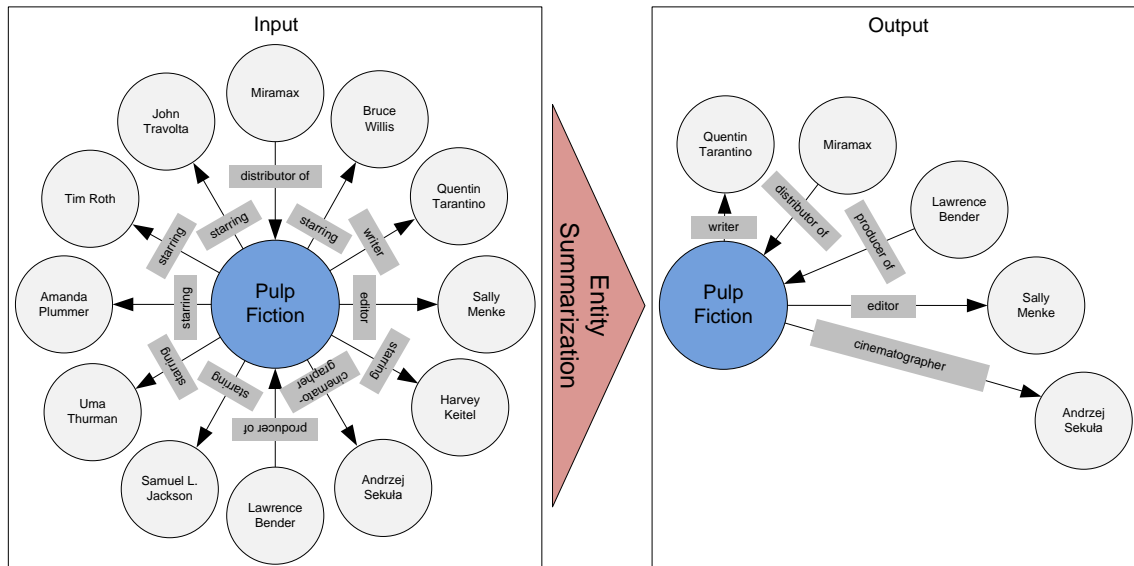


Figure 1.1.: High-level view on entity summarization: The input involves the knowledge graph containing the target entity and all of its (incoming and outgoing) relations (the relations are not ranked—the connecting edges have the same length). The output involves the target entity and a ranked top- k subset of its relations (shorter edges indicate higher importance of the relation for the target entity).

Entity Summarization: The problems of the “naïve solution” are addressed by a new subfield of semantic search⁷, called “entity summarization”. The main idea is to adapt the presentation of entities towards their individual properties. This approach avoids type-based patterns and their accompanying deficits and introduces ranking scores for each fact about an entity with respect to its individual importance for the entity. With this, we can present entities without considering their type(s), enabling to focus on each entity individually. The general process of entity summarization is presented in Figure 1.1.

Our main motivation for working in this field is driven by the need for presenting structured data about entities in a concise manner. Currently, often myriads of facts are presented in tables or graphs and often multiple navigational steps are needed in order to retrieve relevant information. The selection and ranking of facts about an entity can depend on a number of factors such as the context of a user (e.g., time or location), their preferences (e.g., if a user likes movies, the presented entity information could relate to movies in some way), or terms of a search query that complement an entity (e.g., “Pulp Fiction actors”). Summarization systems that consider one or more of these factors rely on specific background data such as session information, user profiles, and query logs and are often heavyweight and highly customized. Although this data is available to big search engine providers, the need for summaries while lacking such specific—and often privacy-sensitive—information states an important problem for content providers such as news publishers, online stores, and other portals. With this thesis, we aim to address this gap and focus on effective entity summarization with lightweight background information.

⁶Depending on the use case and the data model, the number of entity types can get very high.

⁷Semantic search itself is a subfield of information retrieval.

We further specify the task of entity summarization along the following points:

1. **Structured data** The summaries of entities take the form of structured feature lists that are covered by their relations in the knowledge graph. In particular, natural language summaries are not in the scope of this work.
2. **Entity focus** Each summary should focus on the individual particularities of the respective entity. This means that patterns that focus on the classes of the entities are not considered.
3. **Informative/General** The produced summaries should be of general nature and should provide information about the entity. Specializations on users, contexts, or tasks are not considered.

With these assumptions, we are able to border our notion of entity summarization from the following related fields:⁸

1. **Textual Entity Summarization** Next to fact-based summaries, entity descriptions can also be created in a textual way. This can be done by a variety of methods that may or may not use structured data as a basis or support.
2. **Entity presentation according to type** The goal of this task is to create presentation patterns for individual types (this is what we previously described as the “naïve solution”). These presentation patterns mostly include the selection and ranking of properties with respect to a specific type.
3. **User/context/task-specific entity summarization** The goal of this topic is to select features that are relevant to the user as an individual, the user’s context, or a specific task the user has to solve.

Background: Summaries of entities in context to the Semantic Web were first mentioned in [GMM03]. In that work, the authors emphasize on the importance “[...] to determine what relevant data to pull from the Semantic Web.” In May 2012, Google announced a new product they named the “Knowledge Graph” [Sin12]. In that announcement “get the best summary” is pointed out as one of three main contributions of the new Google product. Shortly after Google, other search engine providers like Bing and Yahoo also introduced own knowledge bases. Commonly, all three search engine providers present facts about entities in a side panel of their search engine result pages (SERPs). Although little is known about the details how the summaries of Google, Bing, and Yahoo are generated, the description in [Sin12] indicates that the search queries of their users play a significant role for Google:

“How do we know which facts are most likely to be needed for each item? For that, we go back to our users and study in aggregate what they’ve been asking Google about each item. For example, people are interested in knowing what books Charles Dickens wrote, whereas they’re less interested in what books Frank Lloyd Wright wrote, and more in what buildings he designed.” [Sin12]

⁸We provide a more complete overview in Section 2.2.

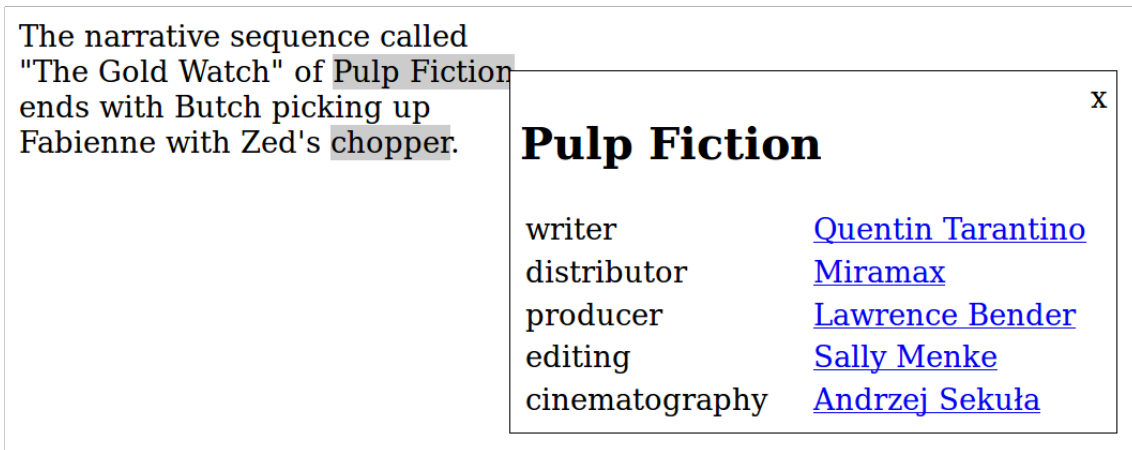


Figure 1.2.: Screenshot of qSUM, a program that shows entity summaries for semantically annotated texts.

In general, the systems of Google, Bing, and Yahoo appear to be highly customized towards the search scenario even though, since August 2015, Bing is moving towards a “knowledge and action graph API” [Pal15]. The search engine providers often use a variety of data signals in order to provide summaries [Sin12]. On top of that, the tight coupling between the actual summaries and the user interfaces make it difficult to distinguish between customized parts from automatic summaries in the panel. Current research approaches often only consider the structured data itself as the main driver for producing summaries (e.g., [SPS10, CTQ11]). In addition, the produced summaries of the research approaches are often not publicly available (see for example [SPS10, CTQ11]). This prevents comparison between entity summarization approaches. According to Dong et al., Google’s knowledge vault only makes small use of structured data on the Web for knowledge integration and discovery: the portion that is used is restricted to 14 `schema.org` predicates that are manually mapped to the Freebase ontology [DGH⁺14a]. Thus, the full integration of structured entity data on the Web—towards *the* knowledge graph—is an open challenge.

In the following, we describe a use case scenario in order to motivate the need for flexible and lightweight entity summarization approaches (Section 1.1.1). Based on the scenario and the presented background, we identify the central set of problems that are targeted by this work (Section 1.1.2).

1.1.1. Scenario: Annotated Hypertext

In our scenario, we consider hypertexts that are semantically annotated with named entities. This means that it was manually (or automatically) defined which parts of the text refer to which entities. As an example, with annotations, we know that the part “chopper” in Figure 1.2 refers to the entity `dbr:Chopper_(motorcycle)`.

A client-side application programmed in JavaScript (e.g., qSUM⁹—see Figure 1.2) retrieves

⁹See Chapter 4 for more information on qSUM.

all annotations and sends requests for summaries to a summarization server in case the user hovers the mouse pointer over one of the entities. The server provides the client with summaries of the respective entity. The server has limited information about the retrieval context and the user. However, as a central element, it has access to a knowledge base that covers information about the entity:

- **Knowledge bases** consist of factual information about real world entities. This could be encyclopedic data, product information, user profiles, etc. Knowledge bases can be considered as fractions of the knowledge graph. Entities from within a knowledge base are the target of the summarization approach (i.e., the input). The fact

```
dbr:Pulp_Fiction    dbo:editing    dbr:Sally_Menke .
```

is an example for a fact about the target entity `dbr:Pulp_Fiction`.

Further, more background information around the entities can be accumulated. We informally define background information as follows:

- **Background information** is supplementary data about the entities that is commonly not directly covered by the *knowledge bases*, such as Web links, usage data, and further external data. This information can be mined and utilized in order to determine the relevance of facts about the targeted entities. An example for such background data is the link structure of Wikipedia with respect to the article `dbr:Pulp_Fiction`.

In our scenario we focus on two types of background information that cover multiple aspects of Web mining (link structure and usage data analysis).

1. In addition to the available structured data, the entities can be described in Web documents (like in Wikipedia). The Web documents of the entities can refer to each other via hyperlinks. These links can serve as indicators for human associations which are often subtle—and therefore not directly covered by the knowledge bases.
2. Multiple entities can be consumed by the same user. Parameters of the Hypertext Transfer Protocol (HTTP), Cookies, and IP addresses can help to identify user sessions with consumed entities. However, as the server does not receive data about the annotated texts it is only possible to detect co-occurrences on the session level but not on the document level.

These descriptions are only two examples of background information that can be used to summarize entities. Some providers of summarization systems may have more (non-public) data available while others could focus on specific aspects that are more relevant to the context in which they are being used. The two examples of background information are suitable to demonstrate the modularity and extensibility of the approach of this thesis.

1. Introduction

1.1.2. Problem Statement

This work covers various aspects around the problem of entity summarization. We present solutions to the following three sets of problems:

1. *How can we provide generic summaries of entities with limited background information?*

Entity summarization systems often lack sufficient information about the users and their context. Similar to the cold start problem in recommender systems [AT05] there is a point where user interactions (such as search queries) are missing for producing more precise summaries. Still, summaries are necessary and good baselines need to be established, for example via Web mining [Liu11]. It is unclear which easily accessible data sources can serve as means for producing entity summaries at a high quality level.

2. *How can we interface entity summaries to machines and users?*

Similar to automatic summarization of text, that is independent from font size or style, automatic summaries of entities can be decoupled completely from presentation. As a matter of fact, summaries in their most basic form are composed by an entity-centric ranking of triples that is independent of all presentation-related matters, even the language. As such, summaries can be exchanged, remixed, and tested by machines with common application programming interfaces (APIs). To develop standard exchange mechanisms, it needs to be defined which parameters are essential to produce summaries of entities, how entity summarization systems receive these parameters, and which data model should be used for the answers of entity summarization systems. In addition, common user interaction mechanisms need to be supported.

3. *How can we integrate data about entities from different Web sources?*

Information about many entities is published in accordance to the Linked Data paradigm. In many cases the same information about the same entity can be found at different sites. However, the data is often modeled with different vocabularies which offer different data representation granularities (i.e., whether more context is available or not). The entity-centric fusion of structured data available on the Web modeled with different vocabularies at different granularities poses a new and important challenge.

1.2. Research Questions and Contributions

In the following, we introduce three main research questions that will be addressed in this work. Each of the research questions focus on one of the problems mentioned in Section 1.1.2.

Research Question 1. *How can we effectively summarize entities with limited background information?*

This research question divides into two sub-questions:

Research Question 1.1. *How can we use link analysis effectively in order to derive summaries of entities?*

Link analysis is used in many areas in order to gain additional insights about the importance of Web sites [Liu11]. We try to answer the question whether this also applies for entity summarization. In particular, we aim to analyze the link structure of Wikipedia (which covers many entities) in order to gain additional insights on entities, their importance, and their relevance for another.

Research Question 1.2. *How can we use usage data analysis effectively in order to derive summaries of entities?*

The analysis of usage data is wide-spread in the field of Web mining [LMN11] with applications for recommender systems and popularity measures. We investigate whether techniques from recommender systems (i.e., k -nearest neighbors) can be utilized to identify important facts about entities. The main idea is that characteristic facts could be shared between neighboring entities.

The next research question focuses on sharing entity summaries:

Research Question 2. *Is there a minimum set of re-occurring/common features of entity summarization systems that allows us to provide a generic API?*

Although a variety of entity summarization systems has been developed, few operate as a Web service. We identify patterns that are reoccurring in online published summaries and define an abstract interface that enables efficient sharing and reuse of entity summaries.

The final research question focuses on mining similar facts across RDF knowledge bases:

Research Question 3. *How can we align duplicate/similar facts about Linked Data entities on the Web?*

Much of the information about single entities is reoccurring as structured data on the Web. Thus, an important step for entity summarization is to identify similar or redundant data about the same entity. We propose an entity-centric processing pipeline that enables to identify similar facts stated in different sources (even if they are modeled in different ways).

The contributions of this work are centered around the three research questions. In the following, we will outline the paths of the proposed solutions for the individual research questions.

Contribution 1 – Research Question 1.1 : We investigate different link analysis methods that have proven effective for ranking and exploratory search in the past. We demonstrate that a combination of these methods outperforms state-of-the-art techniques in entity summarization.

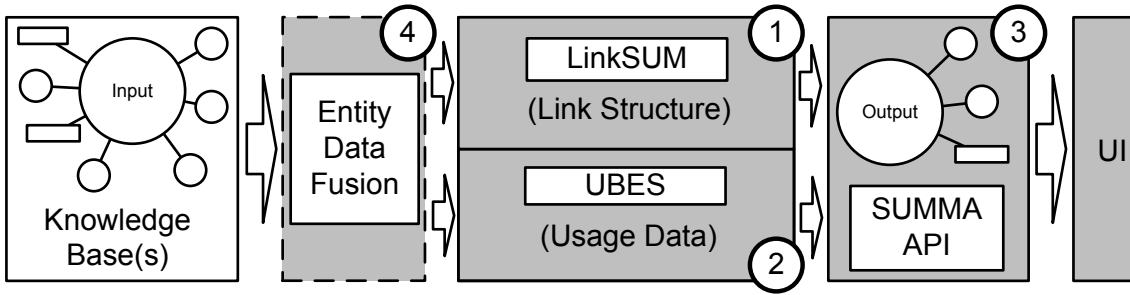


Figure 1.3.: Overview of the contributions of this thesis: link-analysis-based (1 – Research Question 1.1) and usage-data-based (2 – Research Question 1.2) entity summarization, a common application programming interface (API) for entity summarization (3 – Research Question 2), entity data fusion (4 – Research Question 3). The dashed border of the box of Contribution 4 indicates that this step is optional.

Contribution 2 – Research Question 1.2 : We demonstrate that data about linked data usage, such as consumption or rating, can help to establish intrinsic similarities between entities of the same type (nearest neighbors). This knowledge can be used to acquire information about the most common predicate-object pairs that are shared between an entity and its neighbors.

Contribution 3 – Research Question 2 : We provide requirement analysis, a data model, and an interaction model for the RESTful exchange of Linked Data entity summaries. We verify the feasibility of the presented data and interaction models in an empiric study.

Contribution 4 – Research Question 3 : We provide a system that enables the automatic investigation of different sources for Linked Data facts about entities, align these facts, identify their most common representation, and provide an estimate about how important these facts are for the target entity with a measure about redundancy and reliability across sources.

Figure 1.3 visualizes the four contributions and their interplay. The two entity summarization approaches (Contribution 1 and Contribution 2) in the center take input from knowledge bases and output the summaries through a uniform API (Contribution 3). As different knowledge bases often contain the same information, a data integration approach can help to identify redundancies for multi-source entity summarization approaches (Contribution 4). In case the entity summarization approaches rely only on a single source, this step is not necessary (indicated in the figure with a dashed border).

1.3. Previous Publications

Several parts of this work have been published before.

The following works relate to the generation of entity summaries (i.e., Research Question 1):

- Andreas Thalhammer, Ioan Toma, Antonio J. Roa-Valverde, and Dieter Fensel. Leveraging Usage Data for Linked Data Movie Entity Summarization. In *Proceedings of the 2nd International Workshop on Usage Analysis and the Web of Data (USEWOD 2012) held in conjunction with the 21st International World Wide Web Conference (WWW 2012), Lyon, France, April 17th, 2012*, volume abs/1204.2718, 2012. [workshop]
- Andreas Thalhammer, Magnus Knuth, and Harald Sack. Evaluating Entity Summarization Using a Game-Based Ground Truth. In *The Semantic Web – ISWC 2012*, volume 7650 of *Lecture Notes in Computer Science*, pages 350–361. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. [conference]
- Andreas Thalhammer, Nelia Lasierra, and Achim Rettinger. LinkSUM: Using Link Analysis to Summarize Entity Data. In *Web Engineering: 16th International Conference, ICWE 2016, Lugano, Switzerland, June 6-9, 2016. Proceedings*, volume 9671 of *Lecture Notes in Computer Science*, pages 244–261. Springer International Publishing, Cham, 2016. [conference]
- Andreas Thalhammer and Achim Rettinger. PageRank on Wikipedia: Towards General Importance Scores for Entities. In *The Semantic Web: ESWC 2016 Satellite Events, Heraklion, Crete, Greece, May 29 – June 2, 2016, Revised Selected Papers*, pages 227–240. Springer International Publishing, Cham, October 2016. [workshop]

The following works provide contributions to machine-readable representations and interfaces for entity summarization (i.e., Research Question 2).

- Antonio J. Roa-Valverde, Andreas Thalhammer, Ioan Toma, and Miguel-Angel Sicilia. Towards a formal model for sharing and reusing ranking computations. In *Proceedings of the 6th International Workshop on Ranking in Databases (DBRank 2012) held in conjunction with the 38th Conference on Very Large Databases (VLDB 2012)*, 2012. [workshop]
- Andreas Thalhammer and Achim Rettinger. Browsing DBpedia Entities with Summaries. In *The Semantic Web: ESWC 2014 Satellite Events*, *Lecture Notes in Computer Science*, pages 511–515. Springer International Publishing, Cham, 2014. [demo]
- Andreas Thalhammer and Steffen Stadtmüller. SUMMA: A Common API for Linked Data Entity Summaries. In *Engineering the Web in the Big Data Era*, volume 9114 of *Lecture Notes in Computer Science*, pages 430–446. Springer International Publishing, Cham, 2015. [conference]
- Andreas Thalhammer and Achim Rettinger. ELES: Combining Entity Linking and Entity Summarization. In *Web Engineering: 16th International Conference, ICWE 2016, Lugano, Switzerland, June 6-9, 2016. Proceedings*, volume 9671 of *Lecture Notes in Computer Science*, pages 547–550. Springer International Publishing, Cham, 2016. [demo]

1. Introduction

Further, the author contributed to the following publications that should be considered as additional reading (i.e., they directly relate to the topic of this thesis):¹⁰

- John Domingue, Nelia Lasierra, Anna Fensel, Tim Kasteren, Martin Strohbach, and Andreas Thalhammer. *New Horizons for a Data-Driven Economy: A Roadmap for Usage and Exploitation of Big Data in Europe*, chapter Big Data Analysis, pages 63–86. Springer International Publishing, Cham, 2016. [book chapter]
- Kalpa Gunaratna, Gong Cheng, Andreas Thalhammer, and Qingxia Liu. Results of the 2016 ENtity Summarization Evaluation Campaign (ENSEC 2016). In *Proceedings of the 2nd International Workshop on Summarizing and Presenting Entities and Ontologies (SumPre 2016) co-located with the 13th Extended Semantic Web Conference (ESWC 2016), Anissaras, Greece, May 30, 2016.*, volume 1605 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2016. [workshop]

1.4. Impact

The work presented in this thesis has been used successfully in real-world scenarios. For example, the xLiMe project¹¹ was using entity summaries in their semantic search interface.

The author of this thesis was also involved in organizing the successful workshop series “International Workshop on Summarizing and Presenting Entities and Ontologies (SumPre)”. This workshop series covers different topics around entity summarization. The past proceedings of this workshop series are [CGT⁺16] (SumPre 2015) and [TCG16] (SumPre 2016).

The work [TS15] was nominated for the Best Paper Award at the International Conference on Web Engineering 2015 (ICWE 2015).¹² The work [TR16a] was awarded the Best Demo Award at the International Conference on Web Engineering 2016 (ICWE 2016).¹³

The author’s work on the “DBpedia PageRank”¹⁴ dataset (see Section 3.1.6) has received attention by the research community. This has led to various adoptions of the dataset (documented in [Kul15, RSP15, DVBV⁺16, vEMP⁺16] and others). In addition, since DBpedia version 2015-04, the DBpedia PageRank scores are included in the official DBpedia SPARQL endpoint. The dataset was also used for teaching purposes at the

¹⁰Note: Both of the referenced works did not undergo a strict peer review process.

¹¹xLiMe project – <http://xlime.eu/>, retrieved 2016-07-12.

¹²ICWE 2015, Best Paper Candidates – <http://icwe2015.webengineering.org/program/best-paper-candidates/>, retrieved 2016-07-12.

¹³ICWE 2016, Best Demo Award – <http://icwe2016.webengineering.org/program/posters.html>, retrieved 2016-07-12.

¹⁴DBpedia PageRank – http://people.aifb.kit.edu/ath/#DBpedia_PageRank, retrieved 2016-07-12.

Massive Open Online Courses “Knowledge Engineering with Semantic Web Technologies 2015”¹⁵ and “Linked Data Engineering”¹⁶ that were both conducted by Harald Sack.

1.5. Guide to the Reader

This thesis is structured in six chapters that include the introduction, one chapter with foundations/state of the art, three chapters with the main contributions, and one chapter that concludes this work.

- Chapter 2 provides foundations for the main concepts that are used in this work.
This chapter is split into two main parts: 1) the foundations; 2) the state of the art.
- Chapter 3 introduces two different entity summarization approaches based on the respective background information settings.
The chapter is split into two parts: 1) the contribution to link-analysis-based entity summarization; 2) the contribution to usage-data-based entity summarization.
- Chapter 4 introduces our work on a common API for entity summarization.
The chapter focuses on client-server interaction as well as exemplified user interfaces and according implementations.
- Chapter 5 presents our work on entity data fusion.
We demonstrate how we derive, analyze, and mine facts from different sources and create clusters of similar facts (from different sources) about specific entities.
- Chapter 6 concludes this work and provides an overview about the integration of the individual contributions as well as open topics.

¹⁵Knowledge Engineering with Semantic Web Technologies 2015 – <https://open.hpi.de/courses/semanticweb2015/>, retrieved 2016-07-12.

¹⁶Linked Data Engineering – <https://open.hpi.de/courses/semanticweb2016/>, retrieved 2016-11-17.

2. Foundations and State of the Art

In this chapter, we present the fundamental technologies and the state of the art with respect to the overall context of this thesis.

We introduce the main idea of the Semantic Web and its implementing recommendations and best practices in Section 2.1. Section 2.2 covers the definition and the state of the art of entity summarization, its related fields (and their differences to entity summarization), and an overview at the end.

2.1. The Semantic Web

The vision of the Semantic Web was born and has been carried on since the mid 1990s [DB95, LSR96, FDES98, SBF98, Las98, FHLW00, BLHL01]. In 2001, Berners-Lee et al. described the vision of the Semantic Web as follows:

“The Semantic Web is not a separate Web but an extension of the current one, in which information is given well-defined meaning, better enabling computers and people to work in cooperation.” [BLHL01]

The main idea of the Semantic Web is to annotate existing Web structures (i.e., HTML pages) with semantic markup in order to enable a better understanding of their content for machines (such as search engines) that, in turn, would help users with the retrieval and filtering of information. In its original form, methods and systems for deriving new facts from already existing knowledge on the Web were given a key role (i.e., ontologies, formal specifications, and reasoners) [FDES98]. These approaches later moved to the background in favor of efforts that attempted to “[...] bootstrap the Web of Data by identifying existing data sets that are available under open licenses, converting these to RDF according to the Linked Data principles, and publishing them on the Web.” [BHBL09] (i.e., the Linked Data movement).

2.1.1. The Resource Description Framework

The following description of RDF should be regarded with reference to the World Wide Web Consortium (W3C) recommendations [CWL14] and [FPSH14].

The Resource Description Framework (RDF) is an abstract data model that is used to describe knowledge in a graph-based way. The main components of RDF are nodes; in particular Internationalized Resource Identifiers (IRIs) [DS05], literals, and blank nodes.

2. Foundations and State of the Art

The RDF language—that is defined upon these three types of nodes—resembles a strongly simplified version of natural language: its structure includes the terms subject, predicate (also called property), and object. Subjects are IRIs or blank nodes, predicates are IRIs, and objects are IRIs, blank nodes, or literals. A single connection of subject, predicate, and object is called a triple. An example for a triple is as follows:

```
Subject: http://dbpedia.org/resource/Pulp_Fiction
Predicate: http://dbpedia.org/ontology/releaseDate
Object: 1994-10-14 (http://www.w3.org/2001/XMLSchema#date)
```

A set of triples forms an RDF graph where IRIs and blank nodes can be used multiple times either in the subject or object position. The IRIs of predicates can also occur in multiple triples (in all positions).

RDF is an abstract data model because it allows for a variety of concrete RDF syntaxes, for example RDF/XML [GS14] or the Terse RDF Triple Language (Turtle) [PC14]. These syntaxes are also called serializations. In many serializations, like Turtle and N-Triples, full IRIs are put into pointed brackets “<” and “>” and literals are surrounded by double quotes “””. In these syntaxes, triples are terminated by a full stop.

Like URIs (Uniform Resource Identifiers) [BLFM05], IRIs are used in order to identify resources but, in contrast to URIs, also allow for direct inclusion of international characters in the respective identifiers (IRIs are a generalization of URIs: IRIs additionally support Unicode characters while URIs are restricted to ASCII characters). An example for an IRI is as follows:

```
http://dbpedia.org/resource/Pulp_Fiction
```

We define resources (that are identified by IRIs/URIs) in accordance to Berners-Lee et al.:

“A resource can be anything that has identity. Familiar examples include an electronic document, an image, a service (e.g., ‘today’s weather report for Los Angeles’), and a collection of other resources. Not all resources are network ‘retrievable’; e.g., human beings, corporations, and bound books in a library can also be considered resources.” [BLFM05]

We will discuss the difference between network-retrievable resources and those which are not network-retrievable in Section 2.1.4. At the current point, we want to assert that a resource can be any identifiable thing. Blank nodes are used for producing anonymous nodes when coining a new IRI is not intended by the creator.

The term “entity”—one of the main themes in this work—is related to resources and triples: “[...] an entity is synonymous with the Subject of an RDF Triple.”¹ When we speak about entity summarization we mean a summary of an RDF graph that is formed around an entity. As a matter of fact, in these cases, an entity often occurs in the subject position of

¹Definition of “Entity” in the Linked Data Glossary – <https://www.w3.org/TR/2013/NOTE-ld-glossary-20130627/#entity>, retrieved 2016-05-27.

the involved triples. However, as the given directionality of a triple is usually arbitrary,² we extend the above definition such that the IRI that identifies an entity can also occur in the object position of a triple. Therefore, we use the term “entity” synonymously with “resource” (see above for a definition of “resource”).

Literal values are information snippets that have a data type, for example string, date, integer, etc. (as known from programming languages). An example for a literal is the date

1994-10-14 (<http://www.w3.org/2001/XMLSchema#date>).

Literals consist of two parts: their lexical form and their data type IRI. To define the data type by using IRIs provides the option for defining own data types. However, in active use are mostly the data types defined by XML Schema [PGM⁺12]. If the data type is of type language string (i.e., <http://www.w3.org/2001/XMLSchema#langString>), a third component, a language tag in accordance to [PD09] is added to the literal. Language tags, for example “en”, describe the language in which the human reader should interpret the lexical form. For example, the word “chef” has different meanings in English and German. In some syntaxes, neither data type IRI nor language tags are necessary for correct parsing. In such cases, the data type IRI can then be inferred to be of data type string.³ In the Turtle and N-Triples serializations, literals can be denoted as follows:

- "John Travolta" (has data type string)
- "John Travolta"@fr (has data type langString, is in French language)
- "1994-10-14"^^<<http://www.w3.org/2001/XMLSchema#date>>

Often IRIs that are defined by a single creator (a person or a organization) start with a common part. When referring to this common part of the IRIs the term “namespace” has been established.⁴ In different RDF serializations, the namespaces can be defined to be reduced to prefixes that can be used together with the rest of the IRI separated by a colon. For example if the prefix `dbr` is defined for <http://dbpedia.org/resource/> the above IRI can be denoted as `dbr:Pulp_Fiction`. The same holds for `dbo` that stands for <http://dbpedia.org/ontology/>. An example for a triple, denoted in Turtle, making use of the according namespaces is stated as follows:

```
dbr:Pulp_Fiction dbo:director dbr:Quentin_Tarantino .
```

Blank nodes are denoted in a similar way, but with an underscore “_” instead of a namespace; for example `_:S`.

A set of IRIs with a common namespace is called a vocabulary, particularly in cases where the nodes defined by the IRIs help to structure the represented knowledge (e.g., by defining IRIs for predicates). The RDF standard provides a vocabulary to provide the basic structures of the data model [FPSH14]. Additional vocabularies such as, RDFS [BG14]

²Tim Berners-Lee: “Backward and Forward links in RDF just as important” – <http://dig.csail.mit.edu/breadcrumbs/node/72>, retrieved 2016-06-12.

³i.e., <http://www.w3.org/2001/XMLSchema#string>.

⁴Namespaces in RDF were adopted from the according XML specifications [BHLT06].

2. Foundations and State of the Art

(see Section 2.1.2), are commonly used for the further organization of knowledge. In more formal, logic-oriented contexts vocabularies are also called ontologies. In this work we treat the terms vocabulary and ontology synonymously.

One of the main features of RDF is that the data is self-describing. This means that the vocabulary elements (e.g., predicates) are themselves described in RDF. As such, the predicate of the above-given example is used in the subject position in other triples where more information about its meaning is retrievable (e.g., a human-readable description of the predicate in one or more natural languages).

2.1.1.1. Complex Relations in RDF

RDF serves the natural intuition of expressing direct relations between any two nodes (i.e., resources/literals/blank nodes under consideration of the mentioned restrictions, for example a literal can not be in the subject position of a triple). However—in many real-world settings—relations can involve more than two nodes, in particular when additional context is provided. An example for such a context is:

John Travolta played the role Vincent Vega in the movie Pulp Fiction.

There exist multiple ways to model complex relations in RDF. The three most important ones are the following:

Reification (vocabulary) The definition of RDF includes multiple terms for reification [BG14]. These are: `rdf:Statement`, `rdf:subject`, `rdf:predicate`, and `rdf:object`. With this vocabulary, it is possible to identify an RDF statement with an own node (IRI or blank node) and include the resource that identifies the statement in other statements. In consequence, with the reification vocabulary, the above example can be modeled with the following four triples:

```
_:S rdf:subject :Pulp_Fiction .
_:S rdf:predicate :actor .
_:S rdf:object :John_Travolta .
_:S :role :Vincent_Vega .
```

N-ary relations In a W3C Working Group Note⁵ it is outlined that relations that involve more than two nodes could be modeled with an additional node that represents the relation itself (i.e., the starring relation) that connects all involved binary relations in an unambiguous way. In consequence, with n-ary relations, the above example can be modeled with the following three triples:

```
:Pulp_Fiction :starring _:S .
_:S :actor :John_Travolta .
_:S :role :Vincent_Vega .
```

⁵Defining N-ary Relations on the Semantic Web – <http://www.w3.org/TR/2006/NOTE-swbp-n-aryRelations-20060412/>, retrieved 2015-10-20.

By using the blank node `_:S`, it is made clear that the respective node describes a relation which connects multiple resources. However, in many implementations these nodes receive static IRIs and are marked as “relation nodes”. In [EGK⁺14], the process of introducing an additional node that identifies the relation/statement is referred to as “reification”. As such, n-ary relations can be regarded as a reification option.

Named graphs RDF triples form a graph. If additional context needs to be added, the graphs in which specified sets of triples occur can be given names. This leads to the notion of “named graphs”.⁶ A triple that is contained in a named graph is extended by a fourth component, the graph name. The resulting structures are called “quadruples” or—in short—“quads”. The name of a graph can be an IRI or a blank node. An RDF dataset consists of (zero or more) named graphs and one default graph that does not have a name. In consequence, with named graphs, the above example can be modeled with the following quad/triple combination:

```
:Pulp_Fiction :actor :John_Travolta _:S .
_:S :role :Vincent_Vega .
```

In this case, `_:S` is the name of the graph in which the triple `:Pulp_Fiction :actor :John_Travolta` is located.

Nguyen et al. suggest an additional method for modeling complex relations that is called “Singleton Properties” in [NBS14]. The main idea is to add the additional context with the help of a newly defined predicate per n-ary relation. A comprehensive overview about modeling complex relations is provided in [HHK15].

2.1.2. RDF Schema

RDF Schema (RDFS) [BG14] is a vocabulary that enables to introduce more structure for knowledge that is represented in RDF. The namespace of RDFS (used with the prefix `rdfs`) is:

```
http://www.w3.org/2000/01/rdf-schema#
```

The main concept that is introduced by RDFS is `rdfs:Class`. It enables grouping of resources that belong to one class. Basic examples for classes are *films*, *actors*, and *persons*. All things that belong to one class are called its “instances”. In order to describe that something is an instance of a class we use the `rdf:type` predicate. An example for the use of classes is as follows:

```
:Film rdf:type rdfs:Class .
:Person rdf:type rdfs:Class .
:Actor rdfs:subClassOf :Person .
```

⁶RDF 1.1: On Semantics of RDF Datasets – <https://www.w3.org/TR/2014/NOTE-rdf11-datasets-20140225/>, retrieved 2016-05-11.

2. Foundations and State of the Art

```
:Pulp_Fiction rdf:type :Film .  
:John_Travolta rdf:type :Actor .
```

The example shows that the `rdfs:subClassOf` predicate enables to define hierarchies of classes: every instance of the subclass is also an instance of the super class. As this knowledge is not stated explicitly—but can be inferred—there exist different reasoning systems that derive new knowledge from according RDFS definitions.

More semantics can be defined for predicates, in particular their domains and ranges. For example the predicate `:starring` has the following domain and range:

```
:starring rdfs:domain :Film  
:starring rdfs:range :Actor
```

Another important extension is provided by the predicate `rdfs:label`. This predicate is used for providing human-readable representations of IRIs and blank nodes. An example for the use of `rdfs:label` is as follows:

```
:John_Travolta rdfs:label "John Travolta" .
```

Labels are literals that are usually of data type string or langString. Labels are commonly used for the human-readable rendering of data modeled in RDF.

2.1.3. SPARQL Query Language

The SPARQL Protocol and RDF Query Language (SPARQL) [W3C13] defines a protocol and language for querying RDF data. While the protocol covers many additional parts—such as an interaction protocol, service discovery, and federated queries—we lay our focus on the main aspects⁷ of the query language. The SPARQL query language [HS13] enables to retrieve data from an RDF graph with patterns that are matched against it. It provides four types of queries:

“SELECT – Returns all, or a subset of, the variables bound in a query pattern match.

CONSTRUCT – Returns an RDF graph constructed by substituting variables in a set of triple templates.

ASK – Returns a boolean indicating whether a query pattern matches or not.

DESCRIBE – Returns an RDF graph that describes the resources found.” [HS13]

These SPARQL query types are commonly supported by special graph databases that support RDF storage. These databases are called triplestores. Commonly, triplestores support the above query types that use RDF concepts such as IRIs, prefixes, literals, literal data types, and language tags for the specification of query patterns. Variables in patterns start with a question mark, for example `?s`. As an example, the query pattern

```
:Pulp_Fiction :starring ?o .
```

⁷With the introduction of SPARQL 1.1, new features such as aggregation and subqueries were introduced in the query language. The reader is kindly referred to [HS13].

Listing 2.1: Example for a SPARQL SELECT query.

```

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

SELECT ?l ?d WHERE {
  wd:Q104123 wdt:P161 ?o .           # starring in Pulp Fiction
  ?o wdt:P569 ?d .                 # their dates of birth
  ?o rdfs:label ?l .              # their labels
  FILTER (LANG(?l) = "en" ) .     # filter labels for English
}

```

Table 2.1.: Example for a SPARQL result.

l	d
"Bruce Willis"@en	"1955-03-19T00:00:00Z"
"Quentin Tarantino"@en	"1963-03-27T00:00:00Z"
"John Travolta"@en	"1954-02-18T00:00:00Z"
"Steve Buscemi"@en	"1957-12-13T00:00:00Z"
"Rosanna Arquette"@en	"1959-08-10T00:00:00Z"
...	...

matches every triple that has subject `Pulp_Fiction` and predicate `:starring` while

`?s ?p ?o .`

matches all triples. Multiple patterns are separated by dots (“.”) that combine them in an AND logic (i.e., all defined patterns need to match). Next to pattern restrictions, also filter conditions that focus on one or more specific variables of the query can be applied. Each result that matches the set of triple patterns is also validated against the constraints defined by the filters. Often, filters are applied for matching (parts of) literals, their data types, or languages. An example for a filter is:

```
FILTER REGEX(str(?o), "Quentin") .
```

This filter rule for the (previously assigned) variable `?o` matches IRIs or literals that contain the string “Quentin”. The used regular expression language is based on regular expressions of XML Schema [PGM⁺12]. A typical SPARQL SELECT query is exemplified in Listing 2.1 and its result is exemplified in Table 2.1.⁸

The results of a SELECT query are of tabular format where the selected variables form the heading. In contrast, CONSTRUCT queries define graph templates with the defined variables either to extract a subgraph of the current RDF graph or to create a new graph to specify new connections between the selected nodes. Illegal RDF triples, such as literals as a subject, can not be constructed (and, as such, are not included in the resulting RDF graph). An example for a SPARQL CONSTRUCT query is as follows:

⁸The query can be executed at <https://query.wikidata.org>, retrieved 2016-04-13.

2. Foundations and State of the Art

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
```

```
CONSTRUCT { ?s rdf:type foaf:Agent }
WHERE { ?s rdf:type foaf:Organization }
```

The query manually performs the inference step that—in an automatic system—would be triggered by the following triple of the FOAF Vocabulary:⁹

```
foaf:Organization rdfs:subClass foaf:Agent .
```

The ASK query type returns a single boolean value, that is either TRUE or FALSE. For example, the following query checks whether any triple exists in the triplestore:

```
ASK {?s ?p ?o}
```

Similar to CONSTRUCT, the SPARQL DESCRIBE query type retrieves a graph. However, this query type returns information about specified resources and/or about resources that match a defined pattern. An example query for describe is:

```
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
```

```
DESCRIBE dbr:Pulp_Fiction ?s
WHERE {?s rdfs:label "Quentin Tarantino"@en}
```

This query describes the resource `dbr:Pulp_Fiction` as well as every resource `?s` that is matched by the provided pattern. Descriptions can also include triples that do not involve the specified or matching resource as a subject or object. The decision about which triples should belong to the graph that describes a resource is not formally specified and depends on the implementation of the triplestore. Generic implementations do not rely on any predefined vocabulary or context and commonly use a variant of Concise Bounded Descriptions¹⁰ [HS13].

2.1.4. Linked Data

Linked Data is RDF data that is published in accordance to specified principles. These principles are described in a design issue note by Tim Berners-Lee that was first published in 2006. In their latest form, the Linked Data principles are stated as follows:

1. *“Use URIs as names for things*
2. *Use HTTP URIs so that people can look up those names.*
3. *When someone looks up a URI, provide useful information, using the standards (RDF*, SPARQL)*

⁹FOAF Vocabulary Specification – <http://xmlns.com/foaf/spec/>, retrieved 2015-11-17.

¹⁰Concise Bounded Description – <http://www.w3.org/Submission/CBD>, retrieved 2015-11-17.

4. *Include links to other URIs. so that they can discover more things.*” [BL06]

Since RDF 1.1 [CWL14], instead of URIs the more general IRIs are used in the data model. We adopt this generalization also for Linked Data. The idea of Linked Data is to identify resources with IRIs and to publish RDF descriptions of the resources at the locations of these IRIs. The RDF information that is retrieved about the resources usually involves triples that contain unknown IRI nodes (different from the requested one) that are themselves retrievable via HTTP. In this context, particular semantics are involved when the description of one resource asserts that it is the same as another resource. For this, the following predicate is commonly in use to make such statements:

```
http://www.w3.org/2002/07/owl#sameAs11
```

Statements using this predicate indicate that the IRI of the subject can be used interchangeably with the IRI of the object and vice versa (the resources are equal).

Linked Data uses the established Web technology HTTP as a vehicle to share RDF-structured data.¹² Two main questions need to be addressed in this context:

1. How does Linked Data comply with the established structures of the World Wide Web (WWW), where most users expect to retrieve HTML documents or images rather than RDF by looking up IRIs?
2. How can we distinguish between the IRIs of RDF documents and those of the real-world entities, that they describe?

There exist multiple solutions for both questions. The first question can be solved by content negotiation [FR14], more specifically with the HTTP protocol header field “Accept”. With this field, the requesting client can specify the preferred media type of the server’s answer. As such, common Web browsers usually request the `text/html` media type from a server. An application for which structured RDF data is more useful than HTML can use the HTTP accept header in order to ask the server for example for a Turtle [PC14] representation of the resource by setting

```
Accept: text/turtle
```

in the request header. In fact, with this field, a client can give multiple options to the server (separated by a semicolon), commonly starting with the most preferred one. As an alternative (or in addition), the server could deliver RDF data inside an HTML presentation, for example with hidden `span` or `script` tags. For that, currently multiple structured data markup languages such as RDFa¹³, microdata¹⁴, or JSON-LD [SKL14] are actively

¹¹The predicate stems from the Web Ontology Language (OWL) that is a description-logics-driven extension of RDF and RDFS.

¹²We also refer to Linked Data as “structured data on the Web” throughout this thesis.

¹³RDFa 1.1 Primer - Third Edition – <http://www.w3.org/TR/2015/NOTE-rdfa-primer-20150317/>, retrieved 2015-11-24.

¹⁴HTML Microdata – <http://www.w3.org/TR/2013/NOTE-microdata-20131029/>, retrieved 2015-11-24.

2. Foundations and State of the Art

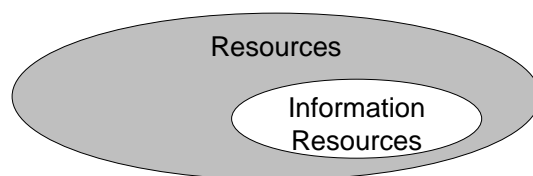


Figure 2.1.: Subset relation between resources and information resources.

used and supported by major search engines.¹⁵ The following example is an HTML fragment annotated with schema.org¹⁶ microdata:

```
<div itemscope itemtype="http://schema.org/Movie">
<span itemprop="https://schema.org/name">Pulp Fiction</span>
<span itemprop="https://schema.org/director"
itemscope itemtype="http://schema.org/Person">
<span itemprop="https://schema.org/name">Quentin Tarantino
</span></span></div>
```

The structured data markup languages enable to deliver RDF and HTML data with a single response in an HTML document while the content negotiation mechanism delivers one of the two. However, the two methods are orthogonal and can be simultaneously implemented by a Linked Data server.

The second question aims at the distinction between real-world objects/abstract concepts and RDF documents that are describing them. In general, as the term “Internationalized Resource Identifier” suggests, everything can be a resource. This includes things in the physical world (like the person “Quentin Tarantino”) or abstract concepts (like the concept of a “movie”). In addition, every RDF document is a resource itself. In this context, RDF documents are resources of which “[...] all of their essential characteristics can be conveyed in a message”; in consequence they are called “information resources” [JW04]. Figure 2.1 visualizes the subset relation between resources and information resources. Resources that are not information resources are called “non-information resources”. The W3C Interest Group Note “Cool URIs for the Semantic Web”¹⁷ emphasizes this difference and proposes two solutions that enable a clear distinction between information resources and non-information resources. The first solution includes the introduction of hash IRIs¹⁸. While the information-resource (i.e., the RDF document that describes “Pulp Fiction”), can be found at

<http://example.com/movies/2517>¹⁹

¹⁵“Focusing on microdata seemed like a pragmatic decision at the time. For some time now we have been supporting multiple syntaxes, specifically including RDFa and JSON-LD.” Source: <https://schema.org/docs/faq.html#14>, retrieved 2015-11-24.

¹⁶schema.org – <http://schema.org>, retrieved 2016-07-28.

¹⁷Cool URIs for the Semantic Web – <http://www.w3.org/TR/2008/NOTE-cooluris-20081203/>, retrieved 2015-11-30.

¹⁸In the referenced publication the authors refer to URIs. We generalize to IRIs in order to comply to RDF 1.1.

¹⁹The top-level domain <http://example.com> is reserved for documentation purposes (see <http://www.iana.org/domains/reserved>, retrieved 2016-07-03). We use it for that in the course of

the non-information resource (the IRI that means the movie itself) would be

```
http://example.com/movies/2517#pf.
```

At the point of retrieval, the two IRIs are the same as the part that starts with the hash symbol would be truncated by any HTTP client. The retrieved RDF content can include information about the information-resource, for example:

```
<http://example.com/movies/2517>  
<http://schema.org/license>  
<https://creativecommons.org/publicdomain/zero/1.0/> .
```

This means that the information resource licenses all RDF data that it offers as public domain. Additional RDF triples retrieved from the IRI can include triples about non-information resources, for example:

```
<http://example.com/movies/2517#pf>  
<http://schema.org/director>  
<http://example.com/persons/270363#qt> .
```

As such, an information resource may include triples about itself but may also provide triples about non-information resources. In this scenario, the IRIs of the non-information resources include the #-symbol and are called “hash IRIs”.

The second option for distinguishing between the describing document and the actual thing is to use HTTP 303, See Other redirects. The idea is to identify, for example, the non-information resource “Pulp Fiction” with `http://example.com/id/2517` that, if requested by the client in Turtle [PC14] syntax (see above for content negotiation), returns a 303 redirect to the information resource `http://example.com/ttl/2517` where a description of the movie is available in Turtle format (and potentially also information about the information resource). In this case, the redirect of the server tells the client that the requested resource is a non-information resource and that a description of it can be found at the provided location (i.e., an information resource). If the provided information resource is requested directly, the response code would be 200, OK and the client can infer that the retrieved resource (in this case `http://example.com/ttl/2517`) is an information resource. The IRIs of the non-information resources do not include the #-symbol and are called “slash IRIs”.

2.1.4.1. Linked Data Adoption and Conformance

Linked Data has found adoption in different commercial and non-commercial sectors. As an example, different Linked Data publishers provide IRIs for the entity “Quentin Tarantino”:

- **IMDb**²⁰: `http://www.imdb.com/name/nm0000233`

this work and also introduce the according prefix: PREFIX ex: <http://example.com/>.

²⁰Internet Movie Database (IMDb) – `http://www.imdb.com/`, retrieved 2016-07-01.

2. Foundations and State of the Art

- **The New York Times:** <http://data.nytimes.com/19079500517022208763>
- **DBpedia:** http://dbpedia.org/resource/Quentin_Tarantino
- **Wikidata:** <http://www.wikidata.org/entity/Q104123>

However, with respect to the adoption of the Linked Data technologies the recommendations described in the current section are not always implemented. In particular, sites such as IMDb and `schema.org` do not distinguish between information and non-information resources. Similarly, The New York Times implements an individual solution that suggests that the non-information resource is an information resource. It is noticeable that this accounts mostly for non-information resources that have slash IRIs. In such cases, a 303-redirect strategy would be necessary. One reason for not implementing 303-redirects might be practicability, as—according to the deprecated standard [FGM⁺99]—a HTTP 303, See Other “[...] response MUST NOT be cached [...]”. Although this standard, and likewise the according prohibition, was obsoleted in June 2014 by [FR14], many HTTP clients and servers still implement this rule. This may become an issue for a server when many clients are consuming the same type of data (e.g., in the case of a vocabulary like `schema.org`) repeatedly within a short time frame. Accordingly, the respective servers could be overloaded. Another reason for not implementing the 303-redirect strategy might be the complexity of the workflow, especially if it is combined with content negotiation. In consequence, Linked Data clients should not rely on clear distinctions between information resources and non-information resources. However, either with or without redirects, the finally delivered RDF document should provide triples that include the originally requested IRI in the subject or object positions.

A complete survey on linked data conformance, focused on the core principles, is provided by Hogan et al. [HUH⁺12].

2.1.5. RDF Knowledge Bases

The Oxford dictionary defines the term “knowledge base” with the following two options:

- 1 *A store of information or data that is available to draw on.*
- 2 *The underlying set of facts, assumptions, and rules which a computer system has available to solve a problem.*²¹

Both definitions could fit our case (as they are intersecting in the “making sense of data” part), but we use the term more in the second sense, as we do not emphasize on the technical aspects of storing the information (which could be any triplestore) but rather use facts (or, more correctly factoids)—materialized in an RDF graph—and according rules to solve a problem. We distinguish (RDF) knowledge bases from the knowledge graph (motivated in Chapter 1) in the sense that we interpret the knowledge graph as the union of

²¹Oxford dictionary definition of “knowledge base” – <http://www.oxforddictionaries.com/definition/english/knowledge-base>, retrieved 2016-08-01.

all externalized knowledge, while individual (RDF) knowledge bases only cover parts of it.

Before and around the time of the inception of the Linked Data movement in 2006, a number of approaches were started in order to create and maintain cross-domain RDF data. The approaches could be coarsely separated into extraction-based and user-generated ones but, in general, such knowledge bases are maintained in a semiautomatic manner. We will briefly introduce the main aspects and the short histories of DBpedia, Freebase, and Wikidata. These knowledge bases are used and referenced in the further chapters of this work. There exist further knowledge bases such as Yet Another Great Ontology (YAGO) and OpenCyc. For an extensive survey and an in-depth comparison we refer the reader to Färber et al. [FBMR17].

DBpedia The DBpedia knowledge base [ABK⁺07] was originally introduced in a joint research effort by Auer et al. in the beginning of 2007.²² The idea of DBpedia was to create an RDF knowledge base by using mappings in order to extract structured knowledge from Wikipedia, in particular from its infoboxes²³ [AL07]. The vocabulary (the “DBpedia ontology”) and the according mappings are maintained manually in a community effort. As the data is mostly created from table-like structures, the DBpedia ontology does not make use of concepts like reification or n-ary statements.

As of version 2015-10, the DBpedia knowledge base extracted from English Wikipedia includes 1.1 billion RDF triples and the data from all utilized Wikipedia projects (other language editions and projects such as Wikipedia Commons) together involves 8.8 billion triples.²⁴ DBpedia includes many links to other cross-domain and domain-specific knowledge bases such as Wikidata and GeoNames²⁵ (among others). The DBpedia project operates a public endpoint at <http://dbpedia.org/sparql> where DBpedia can be queried with SPARQL (see Section 2.1.3). DBpedia publishes its data in accordance to the Linked Data principles: it offers content negotiation as well as RDFa inside the HTML; it distinguishes information resources from non-information resources with a 303-redirect strategy (see Section 2.1.4). Commonly used DBpedia prefixes and according namespaces include:²⁶

```
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX dbp: <http://dbpedia.org/property/>
```

²²First email in the “dbpedia-discussion” mailing list by Sören Auer on March 09, 2007 – <https://sourceforge.net/p/dbpedia/mailman/dbpedia-discussion/thread/45F0A6D1.20908%40informatik.uni-leipzig.de/#msg1377189>, retrieved 2016-04-29.

²³“An infobox is a fixed-format table designed to be added to the top right-hand corner of articles to consistently present a summary of some unifying aspect that the articles share and sometimes to improve navigation to other interrelated articles.” Source: <https://en.wikipedia.org/w/index.php?title=Help:Infobox&oldid=713200930>, retrieved 2016-05-02.

²⁴Source: <http://wiki.dbpedia.org/dbpedia-dataset-version-2015-10>, retrieved 2016-05-02.

²⁵GeoNames – <http://www.geonames.org/>, retrieved 2016-05-03.

²⁶DBpedia SPARQL endpoint, predefined namespace prefixes – <http://dbpedia.org/sparql?nsdecl>, retrieved 2016-05-02.

2. Foundations and State of the Art

Freebase The development of Freebase [BEP⁺08] was announced in March 2007 by the company Metaweb Technologies which was founded by William Daniel Hillis, Robert Cook, and John Giannandrea in 2005.²⁷ The principle of Freebase was to maintain an open knowledge base via a combination of active users and bots. Later, in 2008 during his keynote at the 7th International Semantic Web Conference in Karlsruhe,^{28,29} co-founder John Giannandrea announced Freebase's RDF service as a new data source published in accordance to the Linked Data principles.³⁰ The used namespace was:

```
PREFIX fb: <http://rdf.freebase.com/ns/>
```

In addition to the direct RDF data export, Freebase offered a query service that was based on its own Metaweb Query Language.³¹ The main difference to the effort of DBpedia was that Freebase was created to be a directly writable knowledge base for users and included versioning at the entity level. In addition, further facts such as data from Wikipedia or GeoNames were included and maintained by a bot infrastructure. Unlike Wikipedia (implicitly DBpedia) and Wikidata, Freebase did not have a notability policy.^{32,33,34} The Freebase data model made strong use of n-ary statements (in Freebase called “Compound Value Type”).³⁵ As of May 2016, the Freebase data export consists of 1.9 billion triples.³⁶ On December 16, 2014 the Google Knowledge Graph team announced the discontinuation of Freebase in order to encourage the use of Wikidata³⁷ and on May 2, 2016 the Freebase site and the Freebase APIs were shutdown.³⁸

Wikidata Wikidata was started as a new Wikimedia project at the German branch of the Wikimedia foundation (Wikimedia Deutschland) on October 30, 2012 and was

²⁷The New York Times on March 09, 2007: Start-Up Aims for Database to Automate Web Searching – <http://www.nytimes.com/2007/03/09/technology/09data.html>, retrieved 2016-05-03.

²⁸7th International Semantic Web Conference, keynote announcement – <https://km.aifb.kit.edu/conferences/iswc2008/program/information-on-keynotes/john-giannandrea/index.html>, retrieved 2016-05-03.

²⁹AV recording of John Giannandrea's keynote at the 7th International Semantic Web Conference – http://videlectures.net/iswc08_giannandrea_fowdw/, retrieved 2016-05-03.

³⁰Email by Yves Raimond to the mailing list of the W3C Semantic Web Education and Outreach Interest Group Community Project – <https://lists.w3.org/Archives/Public/public-lod/2008Oct/0047.html>, retrieved 2016-05-03.

³¹Metaweb Query Language – <http://wiki.freebase.com/wiki/MQL>, retrieved 2016-05-03.

³²Wikipedia Notability – <https://en.wikipedia.org/w/index.php?title=Wikipedia:Notability&oldid=715150424>, retrieved 2016-05-03.

³³Wikidata Notability – <https://www.wikidata.org/w/index.php?title=Wikidata:Notability&oldid=313507467>, retrieved 2016-05-03.

³⁴“Freebase has no ‘notability’ requirement: topics about any subject are welcome, independent of how widely known it is.” Source: <http://wiki.freebase.com/wiki/Notability>, retrieved 2016-05-03.

³⁵Freebase's Compound Value Types – http://wiki.freebase.com/wiki/Compound_Value_Type, retrieved 2016-05-03.

³⁶Source: – <https://developers.google.com/freebase/>, retrieved 2016-05-04.

³⁷Google Knowledge Graph team on Google Plus – <https://plus.google.com/109936836907132434202/posts/bu3z2wVqcQc>, retrieved 2016-05-03.

³⁸Email by Jason Douglas to the “Freebase Discuss” mailing list – <https://groups.google.com/d/msg/freebase-discuss/WEnyO8f7xOQ/VucJKFVhBAAJ>, retrieved 2016-05-02.

initially funded by the Allen Institute for Artificial Intelligence, the Gordon and Betty Moore Foundation, and Google [VK14]. Similar to Freebase, Wikidata targets users as key contributors for factual knowledge. In addition several bots and tools are in place that support the creation and curation of facts (e.g., the Primary Sources Tool described in [PTVS⁺16]). Wikidata follows a notability policy.²⁹ As of August 2015, the Wikidata knowledge base has 66 million RDF triples [PTVS⁺16]. The used data model makes use of n-ary relations (in the referenced work by Erxleben et al. called “reification”) but also includes simplified, direct relations [EGK⁺14].

Since September 2015, Wikidata officially maintains a query service that is available at <https://query.wikidata.org/>.³⁹ The most relevant namespaces are:

```
PREFIX wd: <http://www.wikidata.org/entity/>
PREFIX wdt: <http://www.wikidata.org/prop/direct/>
PREFIX wikibase: <http://wikiba.se/ontology#>
PREFIX p: <http://www.wikidata.org/prop/>
PREFIX ps: <http://www.wikidata.org/prop/statement/>
PREFIX pq: <http://www.wikidata.org/prop/qualifier/>
```

All resources described in Wikidata are published online in accordance to the Linked Data principles [EGK⁺14] with content negotiation and 303 redirects (see Section 2.1.4).

2.2. State of the Art

In this part, we will first describe entity summarization (Section 2.2.1) before we move to the commonalities and differences to the related fields (sections 2.2.2 to 2.2.9). In addition, the chapters 3 to 5 include further related-work subsections that focus on the specific context of the respective contributions.

2.2.1. Entity Summarization

The idea to provide human users with interfaces for browsing and editing RDF data exists since RDF and RDFS became W3C standards (e.g., [NFM00]) and also before, when other knowledge representation languages were in use (an overview about such efforts is provided by Duineveld et al. [DSW⁺00]). The field of entity summarization has emerged from different research efforts and contributions. It mainly has its roots in the field of semantic search where Guha et al. introduced the concept of “augmenting search with data” [GMM03] in context to the TAP knowledge base, built by researchers of the Stanford University, IBM Almaden, and W3C. Back then, in 2003, they provided a system that augmented Google search results with data from TAP. Interestingly, this interface had strong resemblance to today’s knowledge panels provided by major search engine

³⁹Email by Dan Garry to the Wikidata mailing list – <https://lists.wikimedia.org/pipermail/wikidata/2015-September/007042.html>, retrieved 2016-05-10.

2. Foundations and State of the Art

providers (see Figure 2.2). In the related publication, Guha et al. also mentioned the task of “determining what to show” [GMM03] as one of the key challenges to be solved. However, since then, the field that we now interpret as “semantic search” has strongly been focused on enhancing the text-based components of the search process with semantic counterparts (see Section 2.2.4). This led further efforts that targeted the challenge of “determining what to show” to refer to the problem of “entity summarization” [SPS10, CTQ11, TTRVF12]. Yet, we regard entity summarization as a subfield of semantic search.

We define the main characteristics of an entity summarization system as follows:

Input IRI of an entity, k —the maximum number of facts to be shown, the knowledge base(s) that include(s) a description of the entity.

Output A top- k selection of triples that involve the input entity (in the subject or object position of the triple).⁴⁰

Entity focus Summaries of an entity should focus on facts that describe the entity best. In particular, the facts of the summary should be selected under the consideration of all related concepts (other resources or literals) and connecting predicates of a specific entity. That is, we do not consider approaches that treat entity presentation on the more abstract level of classes (see Section 2.2.2 for more details).

Type Summaries can be of two basic types [Man01]:

1. **Extract:** The content of this type of summary is contained in the original source. In the case of entity summarization, this means that all presented triples of the output are also contained in the input knowledge base. This type of summary is the most common in entity summarization (and also in text summarization).
2. **Abstract:** An abstract summary contains material that is not covered by the source. In the case of entity summarization, this means that new triples are generated in the course of the entity summarization process. Abstract summaries can include aggregates such as “number of written and directed movies: 9”.

Purpose Entity summaries can be designed towards specific purposes. At the time of writing, the following purposes are documented:

1. **Informative/General:** Summaries should provide information about the input entity. A typical use case is the “Search Engine Result Page” (SERP) scenario, where summaries are presented in the form of a knowledge panel next to a search result (in such cases, the entity is typically identified as the main subject of the query). This type of entity summary has two subtypes:
 - a) **Relevance-oriented:** Summaries are focused on the values (i.e., the connected resources). The importance of the connected resource and the relevance for the target entity is prioritized. In this setting, a complete summary could involve only one predicate (in combination with different

⁴⁰We simplify with the assumption that relevant related resources and literals always have a direct link to the input entity. Complex relations (see Section 2.1.1.1) can be stated in more basic, directly-connected forms by dropping the provided context.

The screenshot shows a Google search interface with the following elements:

- W3C Logo:** Located at the top left.
- Google Logo:** The standard multi-colored logo.
- Search Bar:** Contains the text "eric miller".
- Search Buttons:** "Search" and "Search within results".
- Search Tips:** A link above the search bar.
- Search Options:** "Search WWW" and "Search w3.org".
- Search Results Header:** "Searched pages from w3.org for eric miller. Results 1 - 10 of about 3,190. Search took 0.08 seconds."
- Search Results:** A list of 10 search results, each with a title, snippet, and URL.
 - Eric Miller's Home Page:** W3C, Eric Miller. Semantic Web Activity Lead. ... Eric Miller is the Activity Lead for the W3C World Wide Web Consortium's Semantic Web Initiative. ... www.w3.org/People/EM/ - 4k - [Cached](#) - [Similar pages](#)
 - W3C Semantic Web:** ... RDF Resource Description Framework Metadata Icon Eric Miller <em@w3.org>, (W3C) Semantic Web Activity Lead Ralph Swick <swick@w3.org> (W3C) Development Lead ... www.w3.org/2001/sw/ - 28k - [Cached](#) - [Similar pages](#)
 - Semantic Search:eric miller:** Search WWW Search w3.org. Searched pages from w3.org for eric miller . Results 1 - 10 of about 2,870. ... Eric Miller RDF Model Theory, 14 February 2002. ... www.w3.org/2002/05/tap/semsearch/ - 20k - [Cached](#) - [Similar pages](#)
 - w3c-rdfcore-wg@w3.org from April 2001: Introduction: Eric Miller:** Introduction: Eric Miller. From: Eric Miller (em@w3.org) Date: Fri, Apr 27 2001; Next message: Martyn Horner: "Conference call and ... lists.w3.org/Archives/Public/w3c-rdfcore-wg/2001Apr/0063.html - 6k - [Cached](#) - [Similar pages](#)
 - w3c-rdfcore-wg@w3.org from May 2001: Re: W3C RDFCore WG 2001-05:** From: Eric Miller (em@w3.org) Date: Fri, May 18 2001; Next message: Eric Miller: "Re: W3C RDFCore WG 2001-05-18 Teleconference Minutes"; ... lists.w3.org/Archives/Public/w3c-rdfcore-wg/2001May/0124.html - 6k - [Cached](#) - [Similar pages](#)
 - w3c-rdfcore-wg@w3.org from May 2001: Re: W3C RDFCore WG 2001-05:** From: Eric Miller (em@w3.org) Date: Sat, May 19 2001; ... Previous message: Eric Miller: "Re: W3C RDFCore WG 2001-05-18 Teleconference Minutes"; ... lists.w3.org/Archives/Public/w3c-rdfcore-wg/2001May/0125.html - 7k - [Cached](#) - [Similar pages](#)
 - www-rdf-comments@w3.org from January to March 1999: RDF Model a:** From: Eric Miller (emiller@oclc.org) Date: Thu, Jan 07 1999; ... Next in thread: Eric Miller: "Resource Description Framework (RDF) Becomes a W3C Recommendation"; ... lists.w3.org/Archives/Public/www-rdf-comments/1999JanMar/0002.html - 9k - [Cached](#) - [Similar pages](#)
 - w3c-rdfcore-wg@w3.org from September 2001: PRIMER: Teleconferen:** PRIMER: Teleconference logistics. From: Eric Miller (em@w3.org) Date: Wed, Sep 26 2001; Next message: Art Barstow: "Re: Ampersand ... lists.w3.org/Archives/Public/w3c-rdfcore-wg/2001Sep/0395.html - 7k - [Cached](#) - [Similar pages](#)
 - www-rdf-comments@w3.org from January to March 1999: Resource De:** From: Eric Miller (emiller@oclc.org) Date: Wed, Feb 24 1999; ... In reply to: Eric Miller: "RDF Model and Syntax moves to W3C Proposed Recommendation"; ... lists.w3.org/Archives/Public/www-rdf-comments/1999JanMar/0016.html - 7k - [Cached](#) - [Similar pages](#)
 - www-archive@w3.org from August 2001: RDF Topicmap schema:** RDF Topicmap schema. From: Eric Miller (em@w3.org) Date: Thu, Aug 16 2001; Next message: Eric Miller: "XTM Instance Data"; Previous ... lists.w3.org/Archives/Public/www-archive/2001Aug/0026.html - 6k - [Cached](#) - [Similar pages](#)
- Related Activities:** W3C Semantic Web Activity
- Related Recommendations:** RDF, 22 February 1999. Ralph Swick, Ora Lassila
- Related W3C Working Drafts:**
 - RDF Test Cases, 15 November 2001. Dave Beckett, Art Barstow
 - RDF Primer, 19 March 2002. Eric Miller
 - RDF Model Theory, 14 February 2002. Patrick Hayes
 - Semantic Interpretation for Speech Recognition, 16 November 2001. Luc Van

At the bottom of the page, there is a navigation bar with the following text: "Google Home - Advertise with Us - Search Solutions - News and Resources - Language Tools - Jobs, Press, Cool Stuff..." and a copyright notice "©2002 Google".

Figure 2.2.: An early draft on keyword search results augmented with semantic data. Screenshot of <https://web.archive.org/web/20030106032059/http://www.w3.org/2002/05/tap/semsearch>, retrieved 2016-05-20. An annotated version of this screenshot was presented in [GMM03].

2. Foundations and State of the Art

related resources), if the respective resources are deemed more important than others with different predicates.

- b) Diversity-oriented: Summaries focus more on presenting a diverse selection of predicates (i.e., the type of relation). Repetitive lists of the same type of relation (e.g., “*starring* Uma Thurman; *starring* John Travolta; *starring*...”) are avoided in this setting. Instead, diversification of the predicates aims at providing a more complete overview of an entity.
2. User/context/task-specific: Summaries can be specific to the user’s preferences or interests, to specific user contexts such as time or location, or to specific tasks that the user needs to solve.

In the following we describe entity summarization approaches in accordance to their purpose. It has to be noted that, to the best of our knowledge, all current approaches perform extractive entity summarization.

2.2.1.1. Relevance-Oriented Entity Summarization

The PRECIS algorithm (originally by [KSI06] in application to relational databases), presented by Sydow et al. [SPSS10], uses edge weights in the knowledge base in order to compute relevance-oriented summaries. The approach is based on Dijkstra’s algorithm [Dij59] in its shortest-path tree variant: the k -sized summary of the result is filled up by the traversed shortest paths of Dijkstra’s algorithm. A small user study is provided in this work that is later extended with a comparison to the DIVERSUM method by the same authors [SPS11, SPS13] (see Section 2.2.1.2).

Cheng et al. introduce RELIN [CTQ11], an entity summarization approach that combines relatedness and informativeness-based centrality. The approach works with features that are denoted by predicate-object pairs of entities. Relatedness and informativeness are both string measures that are applied on features with respect to the combined labels of the predicate and object. The measures are used to derive probabilities for a “relational move” (relatedness) or an “informational jump” (informativeness) from one feature to another. The two probability models are then used in a random surfer model, similar to the PageRank [BP98] algorithm, for computing the rank values of individual features incrementally. The goal of RELIN is to provide entity summaries in accordance to a blend of relatedness and informativeness. The experiments include a quantitative as well as a qualitative evaluation where RELIN is compared to the baselines OntoSum [ZCQ07] and Random. The authors demonstrate that the summaries are better than the introduced baselines and conclude the work by emphasizing that “[...] the results are still far from perfect” [CTQ11] and by presenting potential directions for improvement.

In different blog posts [Sin12, Pun12, Bro12], Google researches, engineers, and managers introduce and detail on the features of the Google Knowledge Graph. In [Sin12], Singhal introduces the Google Knowledge Graph with the slogan “things, not strings”. As one of the three main features of the Google Knowledge Graph, the author describes entity summarization in the paragraph titled “get the best summary”. The description of the

summary method suggests that the system determines relevance from search engine queries and/or click feedback: “[...] we go back to our users and study in aggregate what they’ve been asking Google about each item” [Sin12]. The method of query analysis is not generally applicable for entity summarization, as the amount of user queries and click feedback available to Google forms a specific scenario. Similar to our work on usage-based entity summarization [TTRVF12], the Google summaries were enabled to explain specific relations between the target entity and related entities that are covered by a recommendation engine [Pun12] (i.e., the “People also search for” suggestion box in entity search results)⁴¹.

2.2.1.2. Diversity-Oriented Entity Summarization

Sydow et al. introduce DIVERSUM, “[...] the problem of k -limited diversified entity summarisation in knowledge graphs” in [SPS10]. The authors draw a direct relation to the field of search result diversification (i.e., [AGHI09]) and adopt the concept of maximizing the probability of providing at least one relevant document within the top- k diverse results (given an ambiguous query). The approach follows a greedy algorithm that adds triples in accordance to the following features: “[...] (first) novelty (arc label not present in the result yet), (second) popularity (arc multiplicity) and (third) importance (arc weight)” [SPS10]. Given the “novelty” feature, the method quickly exhausts the hop-1 candidate space (i.e., all triples which involve the target entity) and the method stops before k results are reached. As a solution, the author extend the candidate set to hop- m (with $m \geq 1$) consecutively until k results are reached [SPS10]. In effect, the result does not contain only triples but also paths that include multiple triples. In the preliminary experiments, the approach is exemplified to produce better summaries than a non-diverse baseline with the entity “Tom Cruise”. This work was later on extended by [SPS11, SPS13] where DIVERSUM was compared with PRECIS [SPSS10], a diversity-oblivious algorithm (see Section 2.2.1.1). The evaluation of the algorithms is shaped towards the movie domain and involved expert-based assessments as well as crowd-sourced experiments. The results suggest that the DIVERSUM algorithm was favored over the PRECIS approach.

Gunaratna et al. present FACES, another diversity-aware approach to entity summarization [GTS15]. The system has two stages: 1) partitioning the feature set and 2) ranking the features within the partitions. The main idea is to partition the predicates of triples of the target entity into semantically diverse clusters (called “facets”) of predicates by using an adaptation of the COBWEB algorithm [Fis87]. In order to determine semantic similarity for strings (i.e., the labels of the predicates), the authors use WordNet [Fel12] hyponyms. For each cluster, the set of contained triples is afterwards ranked with a tf-idf-related popularity measure on the object. The authors conduct a quantitative and a qualitative evaluation in which they demonstrate that their system provides better results than RELIN [CTQ11] and SUMMARUM [TR14]. In a later contribution [GTSC16], the authors extended the approach of FACES to FACES-E, where two main extensions were introduced: relating literal values to RDFS classes in order to include literal values in the summaries; an extended scoring approach that is also used for ranking clusters against each other (by

⁴¹As of March, 2016.

2. Foundations and State of the Art



Figure 2.3.: Screenshots of the Google Knowledge Graph summaries of the Japanese (left, <https://google.jp>, retrieved 2016-03-26) and the US (right, <https://google.com>, retrieved 2016-03-26) versions of the entity “John Travolta”. The Japanese version covers different features than the US version (e.g., the body height).

averaging the individual rank scores of the triples within a cluster). In the evaluation the authors show superiority of the FACES-E approach over RELIN [CTQ11].

2.2.1.3. User/Context/Task-Specific Entity Summarization

The Google Knowledge Graph entity summarization system was introduced in Section 2.2.1.1. After the inception of the product [Sin12], Google extended their approach by contextualizing its Knowledge Graph results with respect to the individual culture of the country where the search is triggered from [Bro12]. As an example, the body height of people is a much more sought-after feature in Japan than in the United States.⁴² As a result, when showing summaries of people, this feature is more often covered by the Japanese version of a Google knowledge panel than it is shown in the respective US version (see Figure 2.3).

Cheng et al. followed up on their initial work on RELIN [CTQ11] (see Section 2.2.1.1) and re-focused their work towards entity co-reference resolution [XCQ14, CXQ15a] and human-centered entity linking [CXQ15b]. In [XCQ14], the authors show that their approach COMPSUMM enables almost 2.7–2.9 times faster manual entity co-reference resolution in the case of two created disambiguation datasets (DBpedia–LinkedMDB and DBpedia–GeoNames) than without summaries. The COMPSUMM approach combines measures on commonalities, differences, informativeness, and diversity with an adapted heuristic approximation [YWC13] of the binary quadratic knapsack problem [Pis07]. A special focus of the contribution is laid on the effects of including/excluding the measures

⁴²Casey Newton: “How Google is taking the Knowledge Graph global” – <http://www.cnet.com/news/how-google-is-taking-the-knowledge-graph-global/>, retrieved 2016-03-16.

on commonalities and differences: it is shown that the inclusion of both measures brings advantages for manual entity co-reference resolution in terms of accuracy and speed. Cheng et al. later extended this work in [CXQ15a], where they introduce the COMPSUMM extension C3D and its different presentation variant C3D+P. Similar to COMPSUMM, C3D focuses on the combination of a set of four features: “[...] common, conflicting, characteristic, and diverse ones” [CXQ15a]. The authors report about a user study on the DBpedia–LinkedMDB and DBpedia–GeoNames datasets, where C3D and C3D+P were shown to bring major improvements over non-summarized entities (alphabetical order), RELIN [CTQ11], a C3D variant, and COMPSUMM [XCQ14] in terms of the accuracy and speed in manual entity co-reference resolution.

The task of entity linking is different from entity co-reference resolution: While in entity co-reference resolution the task is to create “same as”-relations between entities (both identified by their IRI), entity linking is the task of relating text fragments to entities (commonly identified by their IRI). Usually, in such settings, the text fragments are already recognized to represent an entity and a set of candidate entities are proposed. The selection of the appropriate candidate can be performed manually by human users. As such, Cheng et al. [CXQ15b] propose entity summarization for the task of human-centered entity linking. The approach combines characteristic, contextual, and differential summaries. The harmonic mean of characteristic and contextual summaries is incorporated with differential summaries in the form of an adapted heuristic approximation [YWC13] to the binary quadratic knapsack problem [Pis07] (similar to [XCQ14] and [CXQ15a]). The evaluation was performed in an extrinsic (time and accuracy of subjects given entity linking tasks) and intrinsic way (direct comparison of summaries of the individual methods given entity linking tasks). The results of both evaluations suggest that the described combination of the approaches outperforms individual variants, non-summarized entity descriptions, and RELIN [CTQ11].

2.2.2. Entity Presentation Based on Class Summaries

Entities can also be presented in accordance to summaries of their classes. While, in some works this is also called “entity summarization” [BUNN15, NPG⁺17], the focus on the individual particularities of an entity is not given. In consequence we consider this method as a specific type of ontology summarization (see Section 2.2.3) rather than entity summarization. The task of class-summary-based entity presentation is usually split into two subtasks:

Subtask 1: Determine the most relevant class that the target entity instantiates, for example [TSW11, BBH15].

Subtask 2: Use the top- k relations of the identified class to present the target entity, for example [BUNN15, NPG⁺17].

Subtask 1 can be regarded as entity summarization in a very specific scenario: restriction to `rdf:type` relations in a top-1 setting. Subtask 2 can be considered as a class-centric ontology summarization problem (see Section 2.2.3) in which an individual RDFS class is summarized in accordance a fixed number of predicates (to be shown). The authors

2. Foundations and State of the Art

of [BUNN15] suggest the following basic method: select the most specific class of an entity in the `rdfs:subClassOf` hierarchy that the entity has as `rdf:type`; select top- k predicates according to the number of occurrences in triples with instances of the selected class. The authors of [NPG⁺17] implement a similar approach but use the frequency of “encyclopedic knowledge patterns” (that are derived from Wikipedia) instead of the frequency of triples. Alahmari and Thom propose a similar approach by suggesting a type (given an ambiguous entity query), selecting attributes for a type, and ranking the attributes in accordance to the type [ATM14]. Similarly, Lee et al. mine different data sources in order to provide “typicality” scores for attributes (i.e., predicates) with respect to provided concepts (i.e., classes) [LWWwH13]. Homoceanu and Balke [HB15] define the concept of “typicality” with models from cognitive psychology in the field of family resemblance. Their ARES system identifies typical attributes that should be presented for an entity, given a class and an entity as input parameters. Accordingly, similar to [BUNN15] and [NPG⁺17], the system is based on a measure that uses the overlap of occurring predicates of instances of a class as an indicator for typicality. The reverse approach is taken in [SRP15]: Given an entity and a class, the approach identifies which features are not shared with other individuals of the same class. Combined with measures of Subtask 1, this approach would qualify as an entity summarization approach (see Section 2.2.1) as the selected predicates would be specific for each entity. Assaf et al. [AATC14] present a reverse engineering approach for the Google knowledge panel and derive four to six predicates that are shown in 98% of the cases with respect to a specific class. The Fresnel display vocabulary by Pietriga et al. [PBKL06] enables to specify which predicates (and the respective objects) should be presented for a class (these views are called “lenses”). In addition, formats can be defined that relate lenses to Cascading Style Sheets. An in-depth discussion of Fresnel is provided in Section 4.5.

The presentation of entities with respect to class summaries can be considered as a complementary method to entity summarization. It focuses on aspects such as “typicality” of predicates for specific classes. On the positive side, this type of system could be perceived as comforting as it presents expected predicates in relation to a class. However, there are two drawbacks that are respectively related to each of the two subtasks: For Subtask 1, it is often hard to choose the most relevant class. This is the case, for example, if a career of a person changed multiple times such as the one of Arnold Schwarzenegger, for whom the classes `:Bodybuilder`, `:Actor`, `:Politician` could each fit as the “main class”. Related to this is also the question which attributes should converge to classes and which ones should not. For example: Should `:Politician` be an own class with `rdfs:subClassOf :Person` or otherwise be modeled as a potential attribute of a `:Person` with the predicate-object pair `:profession :Politician`. This question is partially answered by Noy and McGuinness with the first rule for ontology design decisions:

“There is no one correct way to model a domain—there are always viable alternatives. The best solution almost always depends on the application that you have in mind and the extensions that you anticipate.”[NM01]

This makes Subtask 1 strongly dependent on the original use case of the vocabulary. Eventually, by using class summaries, the original use case of the vocabulary then also

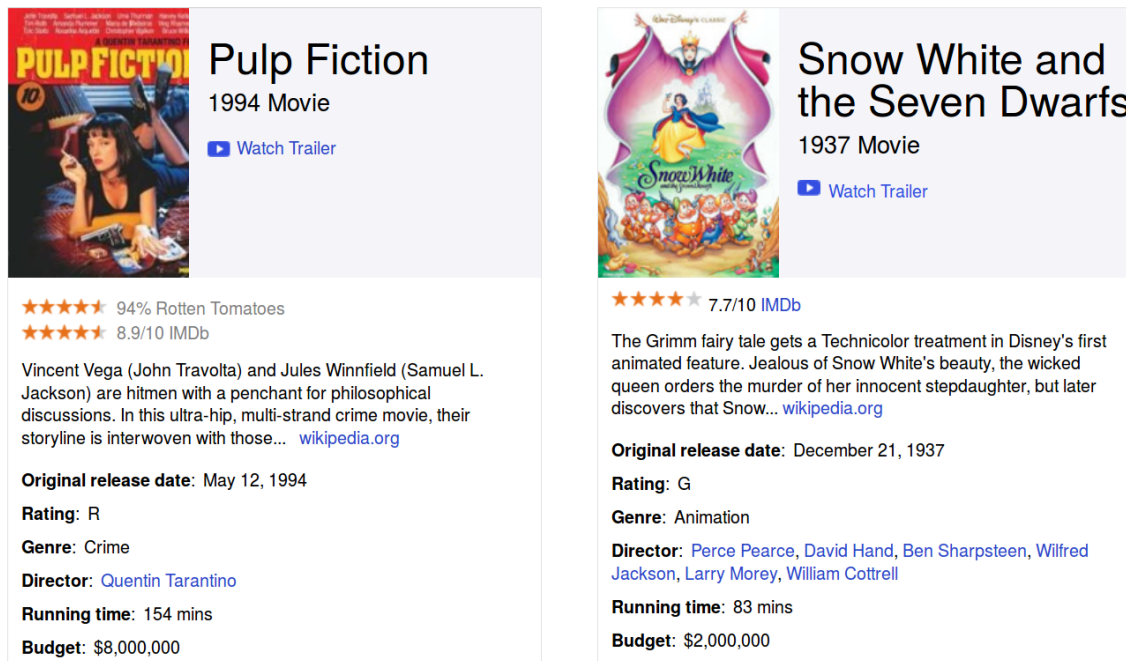


Figure 2.4.: Screenshots of entities in Yahoo’s Knowledge Graph (<https://www.yahoo.com>, retrieved 2016-04-07). The presented attributes of the entities are similar to selections that are produced in accordance to class summaries.

has an effect on the presentation of the instances of its classes. With respect to Subtask 2, once the main class is determined, the selection of attributes is fixed and no variations are possible. An example for such a fixed setting is presented in Figure 2.4, where we present two screenshots of Yahoo’s Knowledge Graph⁴³. The selected attributes for the two (very different) movies are identical and in the same order. However, it would have made sense to include the production company for “Snow White” as it was the first feature film of Walt Disney Animation Studios. In contrast, some of the presented attributes, such as budget, might not be perceived as very important by many users. Finally, a class-summary-based entity presentation needs to implement additional ranking measures in the cases where a predicate occurs with (potentially many) different objects.

2.2.3. Ontology Summarization

The field of ontology⁴⁴ summarization is mainly motivated by the need for a (fast) decision making process on the question whether an ontology is suitable for an application or not [SGPD⁺04]. The idea is to identify a few key terms (classes and predicates) of an ontology that are representative with respect to its potential application areas. In order to summarize ontologies, Zhang et al. define “RDF sentences” in order to account for

⁴³Nicolas Torzec: “Yahoo’s Knowledge Graph” – <http://semtechbizsj2014.semanticweb.com/sessionPop.cfm?confid=82&proposalid=6452>, retrieved 2016-04-07

⁴⁴Please note: we use the terms “ontology” and “vocabulary” synonymously in this work (see Section 2.1.1).

2. Foundations and State of the Art

blank-node connectedness and RDF Sentence Graphs [ZCQ07]. The connections between the RDF Sentence Graphs are then used as an input graph for centrality measures (such as PageRank [BP98]) each applied respectively with MMR [CG98] re-ranking. The final ranking enables to provide top- k summaries of ontologies. Penin et al. [PWTY08] adopt the notion of RDF Sentence Graphs and produce snippets of ontologies with respect to a given query. The authors introduce a semantic similarity measure for RDF sentences in order to reduce redundancy. Peroni et al. [PMd08] make use of a variety of different heuristic measures such as “global popularity” in order to extract key concepts of an ontology. The different measures are combined to a final score via linear combination. A more recent, instance-based approach to ontology summarization is presented by Troullinou et al. [TKDP15]: the authors combine the concepts “relevance” and “coverage” in an new algorithm that also accounts for the number and distribution of instances with respect to the vocabulary terms.

The field of ontology summarization is different from entity summarization in two main points: First, rather than descriptions of real world objects, descriptions of abstract concepts—formalized as a vocabulary—are summarized. Second, entity summarization is topic-bound (the topic is the entity) while ontology summarization is document-bound (the document contains the description of the ontology). Summaries of specific classes (see Section 2.2.2) can be regarded as a connecting piece between entity summarization and ontology summarization.

2.2.4. Semantic Search

Semantic search was originally defined to benefit Web search by addressing two main challenges [GMM03]: 1) augment Web search results with data from the Semantic Web; 2) extension of the textual retrieval components of a search engine by semantic features. The first of these challenges can be split up into three main tasks:

“Denotation: We need to determine the concept denoted by the search query, if any.

What to show: We need to determine what relevant data to pull from the Semantic Web.

Presentation: We need to appropriately format the data/triples for inclusion in the search results.” [GMM03]

The entity summarization problem addresses the second of the above tasks. However, much of the related work in the field of semantic search has been focused on the first task (denotation/entity retrieval/entity identification) and, more generally, on the second of the above challenges (extending the textual retrieval components by semantic features) [THS09]. This becomes more clear with the following statement:

“There exist a wide range of semantic search solutions targeting different tasks - from using semantics captured in structured data for enhancing document

representation [...] to processing keyword search queries and natural language questions directly over structured data [...].” [BHH⁺13]

Hence, the problem of retrieving entities or sets of entities given a keyword query has been very prominent in the field and was also the target of the Semantic Search Challenge in 2010⁴⁵ and 2011⁴⁶. A comprehensive overview about both challenges and participating systems is provided by Blanco et al. [BHH⁺13]. In general, entity retrieval systems are split up into two components: 1) internal index of the entities; 2) functions for the matching of keyword queries and the ranking of results [BHH⁺13]. For example, in Yahoo’s entity retrieval system [BMV11], Blanco et al. combine adapted versions of the BM25F [RZ09] ranking model and the MG4J [BV05] indexing approach and achieve an improvement of +42% over the best result of the Semantic Search Challenge 2010.

Although entity summarization is a subfield of semantic search [GMM03], a large fraction of semantic search research has been focused on the enhancement of text processing elements with semantic technologies [BGMD08, GMDW09, GMDW10, TMWG11]. However, in keyword search scenarios, the output of the entity retrieval step (i.e., a unique identifier for the entity: an IRI) can serve as one of the necessary input parameters of entity summarization systems.

2.2.5. Faceted Search

Faceted search enables flexible user navigation that is independent from the creator’s intentions (that are reflected by static category trees or site maps) and independent from the data’s representation in the storage layer [MB03]. The dimensions include categories that can be orthogonal (e.g., color, style), organized in hierarchies (e.g., brand, series), and single or multi-valued (e.g., mother, sibling) [EHS⁺02, YSLH03]. This way of browsing is often used in combination with traditional keyword search that act as additional filters [HEE⁺02].

With respect to the Semantic Web, a variety of faceted search interfaces have been developed. The graph structure in the storage layer and the direct use of predicate-object pairs as a notion for categories make it easy to implement according interfaces. For the MuseumFinland project, Hyvönen et al. define rules to map predicate-object pairs and according hierarchies to categories that are used for faceted browsing [HMS⁺05]. In contrast, Oren et al. propose an automatic mechanism to construct and rank facets [ODD06]. The authors use a weighted combination of predicate measures such as balance, object cardinality, and predicate frequency in order to determine their navigational value. In later research efforts with respect to facets over RDF data, the creation, selection, and inter as well as intra-ranking of facets received further attention. For example, Wagner et al. [WLT11] introduce the notion of a facet tree for RDF data and define metrics for the ranking of facets. The idea is that browsing should be supported by gradual steps and that related

⁴⁵Semantic Search Challenge 2010 – <http://km.aifb.kit.edu/ws/semsearch10/>, retrieved 2016-04-12.

⁴⁶Semantic Search Challenge 2011 – <http://km.aifb.kit.edu/ws/semsearch11/>, retrieved 2016-04-12.

2. Foundations and State of the Art

queries should result in similar sizes of the result sets. With VisiNav [Har10], Harth also defines facets for search over RDF data. An discussion of the approach is provided in Section 4.5.

Another aspect of faceted search that relates to entity summarization is the presentation of aggregated previews of facets. Similar to [SRP15] (see Section 2.2.2), Dash et al. [DRM⁺08] introduce measures on interestingness and unexpectedness. However, instead of measuring these values for specific entities in relation to their class [SRP15], the authors of [DRM⁺08] perform these analyses on aggregated values of facets, in order to dynamically select, rank, and present the facet in context to keyword queries.

Faceted search can be considered orthogonal to entity summarization: In faceted search, the facets are the central elements and entities are retrieved and ranked in accordance to the selected categories. In contrast, entity summarization is centered around specific entities and their features are ranked with respect to the target entity. For example, in entity summarization, the ranking of the actor list of “Pulp Fiction” is specific to the entity and the ranks of the individual actors can change when another movie with the same actors is browsed. However, the presentation of facts about an entity can trigger further browsing and answer questions such as “which other movies do have John Travolta as an actor?”. Such predicate-object pairs are similar to categories in faceted browsing and can be used for exploration as, for example, done in [KVV⁺07]. In addition, the dynamic aggregation approach presented in [DRM⁺08] could be extended and used as a foundation of an abstractive entity summarization approach.⁴⁷

2.2.6. Ranking RDF Data

The general ranking of RDF data is related to entity summarization. In this part, we focus on approaches that have a direct relation to entity summarization. A broader survey about ranking in RDF datasets is provided by Roa-Valverde and Sicilia [RVS14].

Query-independent ranking of RDF data can focus on two general dimensions: it can target triples or quads as a whole or their individual parts, i.e., subject, predicate, object, and context. In many cases, it is possible to transform from the ranking scores of the parts to scores for triples or quads as the scores of the parts can be normalized and then added up. In the reverse direction, ranking scores of triples or quads can be used to provide average scores for individual IRIs or literals.

One of the earlier approaches for ranking RDF data was provided by Ding et al. in context to the Swoogle system [DFJ⁺04, DPF⁺05]. The underlying ranking algorithm OntoRank is a variant of the original PageRank algorithm [BP98] where non-uniform probabilities are used in the random walk model (similar to [BYD04]): the probability of switching from one resource to another is influenced by a weight that is assigned to the connecting predicate. In addition, the ranking model of OntoRank assumes that an agent that retrieves one ontology also browses all referenced ontologies and therefore alters the probability model for such resources [DPF⁺05]. Another centrality-based measure was introduced by Hogan

⁴⁷See Section 2.2.1 for more information about the “extract” and “abstract” entity summary types.

et al. [HHD06]. The approach follows a variant of the PageRank algorithm and extends the ranking of resources by the ranking of contexts. Similarly, Harth et al. introduce the concept of “naming authority”, compute PageRank on the level of pay-level domains, and use the derived scores as a base for assigning scores to all resources [HKD09]. A similar two-layered, centrality-based approach is followed by Delbru et al. [DTC⁺10]. Franz et al. introduce TripleRank, a tensor-based approach for ranking RDF triples [FSSS09]. Similarly, also based on tensor decomposition, TOPDIS is presented by Harth and Kinsella in [HK09]. Both approaches assign ranks to full triples/quads rather than producing scores for individual resources. A more detailed overview of TripleRank and TOPDIS is provided in Section 3.1.5. Dali et al. present a variety of measures that are combined via a learning to rank approach [DFDM12]. This includes various statistics on RDF data, two centrality-based measures, and ranking scores from external sources and datasets. The authors train their model with gold standard rankings and, for the case of DBpedia and YAGO, with rankings derived from the number of views of the according Wikipedia article.

Rankings over RDF data can serve as a basic entity summarization system: considering one target entity, according triples (in which the target entity occurs) can be ranked with respect to the scores of the connected resources (as we have shown with our SUMMARUM system [TR14]). In optimal cases, however, the scores of a triple should depend on the strength of the connection where the focus should be on relevancy/diversity aspects or further context for describing the target entity (see Section 2.2.1).

2.2.7. Entity Recommendation

The field of entity recommendation in Linked Data is closely related to exploratory search systems [MG14]. That is, depending on the concrete task, entity recommendation systems can be interpreted as exploratory search systems and vice versa.

In [Pas10] Passant introduces dbrec, a linked-data-based entity recommendation system for the music domain. The main idea is a similarity measure (called “Linked Data Semantic Distance”) that measures the distance between two entities based on statistics about connecting predicates and nodes. Similarly, Waitelonis and Sack introduce different heuristics for measuring connectedness between resources in order to implement exploratory search for videos [WS11]. Both systems take one entity as an input and return a list of recommended entities as output. Two research papers with authors from Yahoo tackle the problem of entity recommendation in Web search (e.g., “related people”, “related movies”, etc.) [vZGMS10, BCMT13]. There, the Spark approach [BCMT13] is built on a learning to rank approach that considers a variety of signals that enables entity recommendation. Other search engines, such as Google, use collaborative filtering approaches for recommendations, i.e., they provide “People also search for ...” lists along with the entity summaries [Pun12].

A main difference between entity recommendation and summarization system is that recommender systems do not require a direct or indirect relation between the items (although such relations can be used for explaining recommendations [HG12, TM15] or for cross-domain recommender systems [Hox14]). In addition, the main intent of recommender

systems is to point users to items that are unknown to them but relevant to browse or click. Thus, these systems require to keep track of user profiles. In contrast, the purpose of general entity summarization systems is mainly to inform the user about an entity. We assume that entity recommendation and entity summarization are partly intersecting when entity summaries are user-specific and cross-domain entity recommendations are explained, in particular via direct relations. However, there are still distinguishing points such as the inclusion of literals (that are not covered by recommender systems).

2.2.8. Ranking and Summarization in Databases

The database research community has also addressed entity search and according summaries. Balmin et al. [BHP04] introduce ObjectRank, a system that assigns authorities to database entries in accordance to an authority flow that is predefined in a “authority transfer schema graph”. The system was designed for keyword search and the ranks are computed iteratively at runtime. PopRank by Nie et al. [NZWM05] and EntityRank by Cheng et al. [CYC07] integrate entity records from data-rich Web pages, also referred to as “Web databases”, and introduce according ranking measures. The Précis system by Koutrika et al. provides a semi-automatic approach to create a natural language descriptions of database objects by traversing multiple relations [KSI06]. The approach uses manually assigned weights on the schema for ranking the individual records.

Most similar to entity summarization are the efforts by Fakas et al. [Fak08, FC09, Fak11, FCM11, FCM14, FCM15] who approach the problem of “size- l object summaries” for relational databases. Originally introduced in [Fak08], the workflow is described as follows [Fak11]: 1) textual elements are indexed that enables to identify relevant tuples in a keyword search; 2) in accordance to the schema, the identified relevant tuples are extended in a tree structure over multiple relations; 3) affinities between relations and attributes are computed; 4) the summary is constructed; 5) the elements of the summary are ranked. In their recent work [FCM14, FCM15], Fakas et al. introduce the aspects of versatility/diversity and the automatic selection of the size parameter l .

In most cases, the approaches to ranking and summarization from the database community involve the schema terms and according distances. Therefore, the according rankings depend on the quality and consistency of the schema and are naturally restricted to specific domains. However, as relational databases are the prevalent method of storing and retrieving information, the presented efforts are a valuable improvement over manually designed, static queries. We regard Linked Data entity summarization as an improvement over these methods, as it enables summaries without limiting domain or proprietary borders.

2.2.9. Automatic Natural Language Text Summarization/Generation

Automatic text summarization systems use one or more natural language texts as an input and provide a short textual summary. The main types of systems are extractive or abstractive

(we adopted this notion for describing entity summarization systems, see Section 2.2.1). The field has a long tradition (see for example Luhn’s work published in 1958 [Luh58]) and various hundreds of research papers have been published accordingly. An in-depth textbook describing the field is provided by Mani [Man01] and a more recent survey by Lloret and Palomar covers the state of the art in text summarization [LP11]. Most related to entity summarization are the sub-fields of “survey summaries” and “query-focused” summaries, if the provided texts and queries focus on a single entity. Unfortunately, in most cases the technologies are very different from the ones used in entity summarization and a direct relation can not be drawn. However, there are different approaches that make use of semantic graphs for text summarization.

Leskovec et al. propose to extract a graph structure from textual documents and use support vector machines to extract sub-graphs that are then used for extractive text summarization [LGMF04, LMFG05]. A similar approach was later adopted by Rusu et al. [RFGM08]. More recent works by Ell and Harth [EH14] and Gerber and Ngonga Ngomo [GN14] show patterns that are learned by annotating text documents and combine the output with existing RDF knowledge bases. An example is the sentence “Tarantino is known for his masterpiece Pulp Fiction” that, with entity linking, produces the entities `dbr:Quentin_Tarantino` and `dbr:Pulp_Fiction`. The DBpedia relation between the entities is `dbo:director`. As such, if patterns of the form “ENTITY is known for his masterpiece ENTITY” occur not only once but multiple times in a corpus, it can be learned that this phrase only occurs if there is a `dbo:director` relationship between the two mentioned entities. In the reverse direction, this phrase could then be used to verbalize the triple in natural language. In effect, these approaches could be used to verbalize the output graph of entity summarization systems.

2.2.10. Question Answering over Linked Data

Closely related to semantic search, question answering is the research field where questions (formulated in natural language) are answered concisely in natural language. The idea of ontology-based question answering dates back to 2003, when Vagas-Vera et al. introduced an initial prototype of AQUA [VVMD03], a system that combines knowledge representation, reasoning, and natural language processing. Parts of this system were later used as a basis for AquaLog [LM04, LUMP07] that evolved to PowerAqua [LMU06] in 2006. In contrast to AquaLog, PowerAqua does not rely on a specific vocabulary. Dali et al. introduce a question answering system that is based on question understanding, semantic graph construction, and document summarization [DRF⁺09]. Instead of an answer that is directly derived from verbalizations of triples, the system presents a textual summary of a document that contains the answer for the input question. With this technique, additional context for the answer of the respective question is provided.

Later, in 2011, the first “Question Answering over Linked Data” (QALD)⁴⁸ evaluation campaign was held [LUCM13]. Since then, multiple editions of this campaign have been

⁴⁸QALD – <http://www.sc.cit-ec.uni-bielefeld.de/qald>, retrieved 2016-04-13.

2. Foundations and State of the Art

Listing 2.2: Shortened excerpt of the QALD-6 “Multilingual question answering over RDF data” task (other language versions of the question were omitted for brevity).⁵¹

```
{ "id": "186",
  "answertype": "resource",
  "aggregation": "false",
  "onlydbo": "true",
  "hybrid": "false",
  "question": [
    { "language": "en",
      "string": "What is the largest city in Australia?",
      "keywords": "Australia, largest city" } ],
  "query": { "sparql": "SELECT DISTINCT ?uri WHERE { <http://dbpedia.org/resource/Australia> <http://dbpedia.org/ontology/largestCity> ?uri . } " },
  "answers": [
    { "head": { "vars": [ "uri" ] },
      "results": { "bindings": [ {
        "uri": { "type": "uri",
                  "value": "http://dbpedia.org/resource/Sydney"
                }
      } ] }
    }
  ]
}
```

held and it currently⁴⁹ continues with its 6th edition.⁵⁰ The campaign releases test and training datasets for different tasks in question answering over Linked Data which are considered as the de facto standard in the research field. We provide an excerpt from the QALD-6 campaign training dataset in Listing 2.2 in order to exemplify the type of questions/answers that the field tries to address.

Question answering is a research field that is centered around the correct interpretation of natural language questions and providing according fact-based answers. In many cases, the questions involve aspects about real-world entities. However, the input of question answering systems is different from entity summarization systems: in question answering the users specify a concrete information need (e.g., “What is the largest city in Australia?”) while in entity summarization a single entity (identified via an IRI) is targeted without any specific information need (e.g., <http://dbpedia.org/resource/Sydney>). Entity summarization serves the purpose of describing an entity in a structured way. Thus, we consider question answering over Linked Data and entity summarization complementary, as the output of the former (the answer of a question is an entity or a list of entities) can in many cases be used as an input for the latter (entity summarization).

⁴⁹As of April, 2016.

⁵⁰QALD-6 challenge announcement – <https://lists.w3.org/Archives/Public/public-lod/2016Jan/0073.html>, retrieved 2016-04-13.

⁵¹Source: <http://qald.sebastianwalter.org/6/data/qald-6-train-multilingual.json>, retrieved 2016-04-13.

2.2.11. Pathfinding in Knowledge Bases

In sociology and likewise in theories about social networks, the notion of “six degrees of separation”, introduced by Milgram as the “small world problem” [Mil67], is widely known and accepted. The idea is that every node can be reached from any other node with at maximum six hops in a social graph. This was adopted in various settings, for example two measures about scientists and actors were introduced: The Erdős number is a number that computes the number of co-author hops that are needed in order to reach Paul Erdős (a famous mathematician). Similarly, for actors, the Bacon number is the number of co-starring hops that are needed in order to reach Kevin Bacon (a famous actor). For knowledge bases, this shortest distance aspect was adopted by the “Wiki Game” in 2004. The game measures the time and the number of clicks that a player needs in order to find a click path from one fixed random article to another in the Wikipedia article link graph.⁵² The human navigation patterns that resulted as datasets from different implementations of this game are used in research efforts that address hidden relations between things and according human associations, for example [WL12, SHHS15].

While the Wiki Game defines the discovery of paths as a manual challenge, paths can also be discovered automatically. The WikiBinge system⁵³ suggests paths that avoid pages with high PageRank [BP98] scores and provides connections between two Wikipedia articles via unpopular nodes. The idea is that these paths are likely to be unknown by the user and thus interesting or surprising. A similar setting has also been applied for RDF knowledge bases: The ReIFinder system by Heim et al. [HLS10] finds relations between two DBpedia items until up to a predefined number of hops. The system can be configured to consider incoming and outgoing relations and, in addition, the number of changes in the directionality within a path can be configured. Also individual predicates can be selected, see for example Figure 2.5. With “Everything is connected” [VVC⁺12], Vander Sade et al. present an approach that retrieves different media data from various sources in order to feature pathfinding in DBpedia with rich media, such as images and video, accompanied with speech synthesis for explaining the individual relations.

The task of identifying meaningful paths between any two entities is also related to entity summarization. In particular, systems that create a set of related entities for a target entity in a recommender-system-like setup can profit from identifying multi-hop paths between the involved entities. Often the relation would be a connecting resource, for example “John Travolta is related to Samuel L. Jackson, because they are both actors of Pulp Fiction and Basic” (see Figure 2.5). While pathfinding methods can help to identify (multi-hop) relations between two entities, one of the tasks of entity summarization is the identification of the most relevant relation.

⁵²The Wiki Game as explained in November 2004 – https://en.wikipedia.org/w/index.php?title=Wikipedia:Wiki_Game&oldid=7731042, retrieved 2016-04-26.

⁵³WikiBinge – <http://www.wikibinge.com/>, retrieved 2016-04-26.

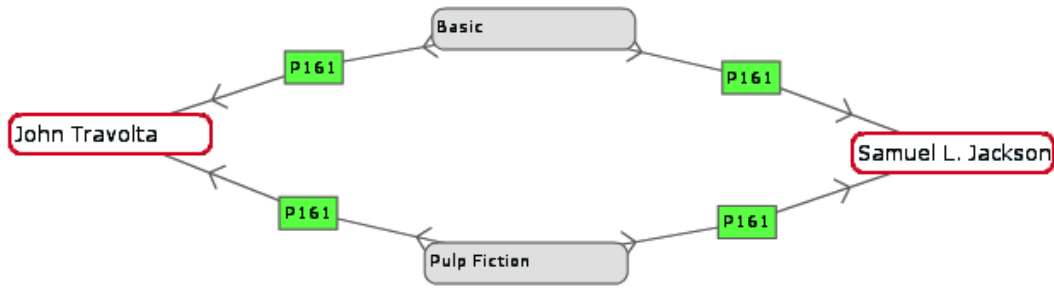


Figure 2.5.: Screenshot of RelFinder [HLS10] for relations between John Travolta and Samuel L. Jackson on Wikidata. Filters are active for `wdt:P161` (i.e., cast member) and for hop-2 relations (i.e., the distance via one connecting node in between).

2.2.12. Summary

We thoroughly discussed entity summarization, its related fields, and the individual relations. Table 2.2 provides an overview of the discussed topics. The two entity summarization approaches, that we introduce in Chapter 3 are classified in the category of relevance-oriented methods. There are currently two branches of work: approaches that only rely on the structure of the respective knowledge base [SPSS10, CTQ11] and approaches of search engines, that rely on heavyweight background information (such as query logs and user profiles [Sin12]). In the following chapter, we aim to fill the gap between these two extremes.

Table 2.2.: Overview of entity summarization approaches and related fields.

Entity Summarization		
	Relevance-oriented	Focus on the individual relevance of the related entities for the target entity (e.g., [CTQ11, Sin12]).
	Diversity-oriented	Provide a diverse selection of predicates describing the target entity (e.g., [SPS10, GTS15]).
	User/context/task-specific	Involve the respective user preferences, their context or tasks (e.g., [Bro12, XCQ14]).
Entity Presentation Based on Class Summaries	Entities are presented in accordance to a class they belong to (e.g., [PBKL06, AATC14, NPG ⁺ 17]).	
Ontology Summarization	A document that contains multiple concepts and their relations is summarized (e.g., [PMd08, TKDP15]).	
Semantic Search	Bridging the gap between traditional keyword search and the actual semantics of search requests and results (e.g., [GMM03, THS09, BMV11]).	
Faceted Search	Oriented towards specific predicates, predicate-object pairs, or other filters (e.g., [HMS ⁺ 05, Har10]).	
Ranking RDF data	Ranking of resources, predicates, and triples (e.g., [DPF ⁺ 05, HKD09, FSSS09]).	
Entity Recommendation	Recommendation of entities given a user profile and/or context (e.g., [Pas10, BCMT13, Hox14]).	
Ranking and Summarization in Databases	Approaches using the structures and relations given by relational algebra (e.g., [BHP04, CYC07, Fak08]).	
Automatic Natural Language Text Summarization/Generation	Methods that leverage knowledge bases for summarizing or generating natural language (e.g., [RFGM08, EH14, GN14]).	
Question Answering over Linked Data	Answering questions in natural language via knowledge bases (e.g., [VVMD03, LM04, DRF ⁺ 09]).	
Pathfinding in Knowledge Bases	Identifying (interesting) paths between any two entities in a knowledge base (e.g., [HLS10, SHHS15]).	

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

We address Research Question 1 in this chapter:

How can we effectively summarize entities with limited background information?

For this, we distinguish between two types of relations between entities, *explicit relations*, such as Web links, and *implicit relations*, such as usage patterns or co-mentions. We use both types of relations for producing summaries of entities. In particular, we provide two contributions: In Section 3.1, we introduce LinkSUM, a link-analysis-based approach for entity summarization: we combine the traditional PageRank measure [BP98] with the Back-link heuristic [WS11] and demonstrate that the system outperforms the FACES [GTS15] state-of-the-art system. In Section 3.2, we introduce an approach that leverages entity-neighborhood established in indirect ways. For this, we present the usage-based entity summarization (UBES) approach, a game-based method to establish a ground truth, and experiments in which we compare UBES to the Google Knowledge Graph [Sin12]. The results indicate that the game-based ground truth could be used for entity summarization and that further research is needed.

This chapter is based on methods, descriptions, figures, experiments, and according results that were previously published by the author and his collaborators in [TTRVF12], [TKS12], [TLR16], and [TR16b].

3.1. LinkSUM: Using Link Analysis for Entity Summarization

We address Research Question 1.1 in this section:

How can we use link analysis effectively in order to derive summaries of entities?

The contribution—which addresses this question—states one of the two entity summarization approaches that are introduced in this thesis (see Figure 3.1).

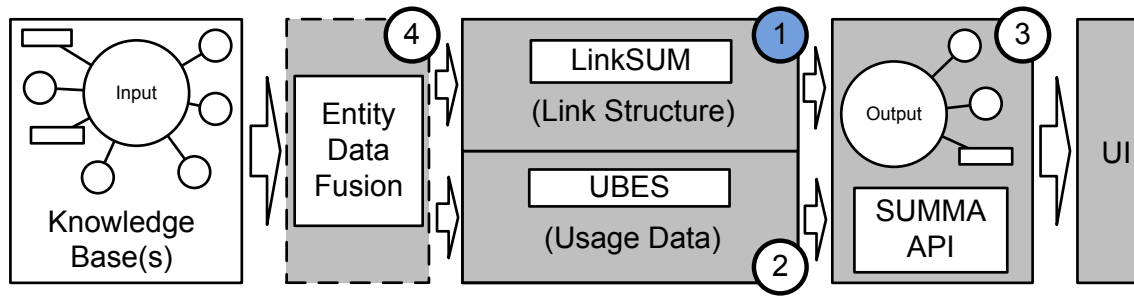


Figure 3.1.: Overview of the contributions of this thesis. In this part, we focus on LinkSUM, an entity summarization approach that utilizes links between entities.

3.1.1. Introduction

A significant part of search engine result pages (SERPs) is nowadays dedicated to knowledge panels about entities (e.g., Figure 3.2). In its complete form, data about a single entity may involve thousands of statements. This is an overloading amount for humans. For that reason, SERPs show concise summaries of the respective entities. These summaries are often presented without focusing on a specific task, user, or context. In that regard—for informative/general entity summarization—we distinguish between relevance-oriented and diversity-oriented summarization systems (see “purpose” in Section 2.2.1). In this section, we present LinkSUM, an entity summarization system that follows a relevance-oriented approach to produce general summaries to be displayed in a SERP. We present the results of a comparison of the LinkSUM system with the diversity-centered approach of FACES [GTS15]. FACES was compared to two relevance-oriented systems [CTQ11, TR14] with respect to the “[...] purpose of quick identification of an entity” [GTS15] and superiority of FACES was demonstrated. The authors concluded that FACES, and its aspect of diversity, is superior with respect to the mentioned task [GTS15]. With our work on LinkSUM, we intended to verify the findings of [GTS15] with respect to the scenario of SERPs. Both systems rely on minimal amounts of background information and therefore suit the scenario described in Section 1.1.1. We compared the two systems in a quantitative as well as qualitative evaluation setting.

With the work on LinkSUM, we aimed to address the following challenges:

1. In many settings—like in the scenario of Section 1.1.1—the amount of data about the user is limited and the current context as well as previous keyword search terms are unknown. How can Web links (i.e., minimal, commonly available background information) be leveraged to produce competitive summaries of entities?
2. For the SERP scenario, it is unclear whether relevance-oriented or diversity-oriented summaries (see Section 2.2.1) are better. Are descriptions with maximal diverse predicates perceived as better (by the users) than purely relevance-oriented summaries?
3. In many SERPs, related resources and predicates occur multiple times (e.g., see Figure 3.2). To which extent do users want to see multiple predicates/resources in entity summaries displayed in SERPs?

3.1. LinkSUM: Using Link Analysis for Entity Summarization

The screenshot shows a Google Knowledge Graph summary for the movie "Pulp Fiction". At the top, the title "Pulp Fiction" is displayed with a share icon and a movie poster. Below the title, it indicates the year "1994", the genre "Crime film/Drama film", and the duration "2h 58m". A row of three boxes shows ratings: "8,9/10 IMDb", "94 % Metacritic", and "94 % Rotten Tomatoes". The main text provides a synopsis: "Vincent Vega (John Travolta) and Jules Winnfield (Samuel L. Jackson) are hitmen with a penchant for philosophical discussions. In this ultra-hip, multi-strand crime movie, their storyline is interwoven with those of their boss, gangster Marsellus Wallace (Ving Rhames); his actress wife, Mia (Uma Th... More". Below this, it lists the "Release date: November 3, 1994 (Germany)", "Director: Quentin Tarantino", "Screenplay: Quentin Tarantino", "Executive producers: Danny DeVito, Stacey Sher, Michael Shamberg", and "Awards: Palme d'Or, more". A "Critic reviews" section includes two quotes: "Whatever you call it, Pulp Fiction is indisputably great. Full review" by Peter Travers from Rolling Stone, and "Brilliantly written and unfathomably cool, this would make a good case for most quotable crime movie of all time. Full review" by Ian Freer from Empire. The "Profiles" section shows a Facebook icon. The "Cast" section features six portraits with names and roles: Quentin Tarantino (Jimmie Dimmick), John Travolta (Vincent Vega), Uma Thurman (Mia Wallace), Samuel L. Jackson (Jules Winnfield), Bruce Willis (Butch Coolidge), and Tim Roth (Pumpkin). A "View 10+ more" link is present. The "People also search for" section shows three categories of related content: "Directed by Quentin Tarantino" (with posters for Kill Bill, Reservoir Dogs, and Pulp Fiction), "John Travolta movies" (with posters for Get Shorty, Boyz n the City, and Pulp Fiction), and "Other similar movies" (with posters for True Romance, Lethal Weapon, and Die Hard).

Figure 3.2.: Screenshot of a Google Knowledge Graph summary of the entity “Pulp Fiction” (http://g.co/kg/m/0f4_1, retrieved 2016-07-30).

4. RDF knowledge bases (see Section 2.1.5) are typically unranked; in particular, from the provided data it is not clear whether one triple or a specific entity is more worth to be presented to a user rather than another. How can we introduce a relevance-oriented setting for informative/general entity summarization in unranked RDF knowledge bases?

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

Along that line, the overall contributions of the approach are as follows:

1. We introduce LinkSUM, a lightweight link-based approach for the relevance-oriented summarization of entities in RDF knowledge bases. LinkSUM optimizes the combination of the PageRank algorithm with an adaptation of the Backlink method [WS11] together with new approaches for predicate selection. The approach is exemplified with DBpedia as a knowledge base and uses links of Wikipedia as a source for background knowledge. It can be easily applied for further data sources and suits the scenario of Section 1.1.1.
2. In our experiments we compare the LinkSUM approach to FACES [GTS15]. We show that the LinkSUM system performs significantly better than FACES in a quantitative and a qualitative evaluation setting. Moreover, in the qualitative evaluation setting, the comments of the users suggest that relevance-oriented systems should be preferred in SERP scenarios.
3. In our qualitative evaluation, we also presented summaries to the users that contain related resources and/or predicates that occur in multiple triples. We analyze the comments of the users with respect to the topic of redundancy and verify that relevance is more important than diverse selections of predicates. In addition we find that related resources should be presented only with one predicate (that has been identified as the strongest connection).
4. We present PageRank-based rankings of entities computed on the Wikipedia link graph. We change the probabilistic impact of links in accordance to their position on the page and measure the effects on the output of the PageRank algorithm [BP98]. We compare the resulting rankings and those of existing systems with pageview-based rankings and provide statistics on the pairwise computed Spearman and Kendall [Ken38] rank correlations. The individual ranks of the entities can be directly transferred to the according IRI of the entity in the respective RDF knowledge base.

The following subsections are structured as follows: We will first introduce the approach of LinkSUM in Section 3.1.2. The approach is then extensively tested with according experiments that are presented in Section 3.1.3. A discussion of the results is provided in Section 3.1.4. Related work is presented in Section 3.1.5. In Section 3.1.6 we compare different variants for computing PageRank on Wikipedia. We draw our conclusions in Section 3.1.7.

3.1.2. Approach: LinkSUM

The proposed entity summarization method includes two main stages:

Resource Selection The goal of this stage is to create a ranked list of resources that are semantically connected to the target entity. The output of this step is a set of triples, where the semantic relation is not fixed, for example *Pulp Fiction* – ?relation → *Quentin Tarantino*. One requirement for a resource to be included in the list of relevant entities is at least one existing semantic relation to the target entity.

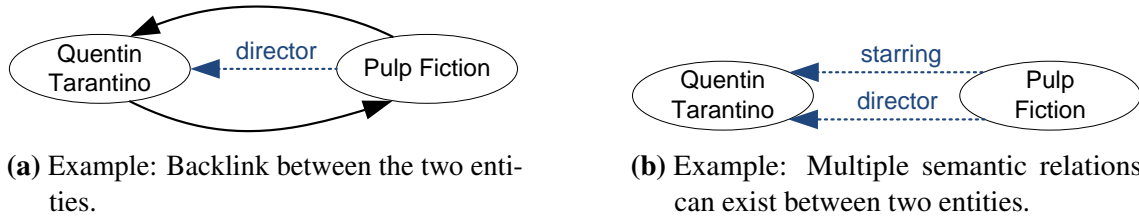


Figure 3.3.: Web links (black, solid) and semantic relations (blue, dashed) between “Quentin Tarantino” and “Pulp Fiction”.

Relation Selection This stage deals with the selection of a semantic relation that connects the resource with the target entity. This step is necessary if more than one relation exists, for example

Pulp Fiction – *starring* → *Quentin Tarantino*, and

Pulp Fiction – *director* → *Quentin Tarantino*.

In the following subsections we will explain each of the two parts. We will refer to the target entity as e (i.e., the entity that needs to be summarized).

3.1.2.1. Resource Selection

For the resource selection, we combine two link-measures: one that accounts for the importance of the connected resource (PageRank [BP98]) and one that accounts for the strength of the connection (Backlink [WS11]). We consider links between entities as a means for identifying and ranking relevant resources. The presented method covers scenarios, in which semantic relations are present in addition to textual descriptions that contain Web links to other resources.

3.1.2.1.1. Important Related Resources As a first step, we run the PageRank algorithm [BP98] on the set of all resources R with their individual directed links $link(r_1, r_2)$ with $r_1, r_2 \in R$, in particular the set of pages that link to a page $l(r) = \{r_1 | link(r_1, r)\}$ and the count of out-going links $c(r) = |\{r_1 | link(r, r_1)\}|$:

$$pr(r) = (1 - d) + d \cdot \sum_{r_n \in l(r)} \frac{pr(r_n)}{c(r_n)} \quad (3.1)$$

The variable d marks the damping factor: in the random surfer model, it accounts for the possibility of accessing a page via the browser’s address bar instead of accessing it via a link from another page. Like in [BP98], we set the damping factor to 0.85 in all our experiments. The PageRank algorithm applies the above-given formula incrementally. The number of iterations depends on the general size of the dataset as well as on the density of links. After executing the algorithm, each resource r has its own PageRank score $pr(r)$. The set of resources that have a semantic connection to e is defined as $res(e) \subseteq R$. As a matter of fact, every resource $r \in res(e)$ can be ranked in accordance to its individual PageRank. A basic popularity-based top- k summary of e can be produced with that

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

Listing 3.1: Example: SPARQL query for retrieving the top-10 semantically related resources of “Pulp Fiction” in the order of their PageRank scores.

```
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX vrank:<http://purl.org/voc/vrank#>

SELECT DISTINCT ?o ?v
FROM <http://dbpedia.org>
FROM <http://people.aifb.kit.edu/ath/#DBpedia_PageRank>
WHERE {
    { dbr:Pulp_Fiction ?p ?o } UNION { ?o ?q dbr:Pulp_Fiction }
    ?o vrank:hasRank/vrank:rankValue ?v.
}
ORDER BY DESC(?v) LIMIT 20
```

Table 3.1.: Example: Top-20 resources that have a semantic relation to “Pulp Fiction” ranked by their individual PageRank scores (DBpedia version 3.9).

o	v
dbr:Category:English-language_films	220.961
dbr:Quentin_Tarantino	13.740
dbr:John_Travolta	10.577
dbr:Miramax_Films	9.940
dbr:Category:1994_films	8.445
dbr:Bruce_Willis	7.161
dbr:Samuel_L._Jackson	6.695
dbr:Category:Films_shot_anamorphically	6.367
dbr:Christopher_Walken	5.371
dbr:Category:Films_set_in_Los_Angeles,_California	4.478
dbr:Uma_Thurman	3.977
dbr:Harvey_Keitel	3.724
dbr:Category:Miramax_Films_films	2.579
dbr:Category:Edgar_Award_winning_works	2.485
dbr:Category:Films_about_drugs	2.184
dbr:Category:Anthology_films	2.156
dbr:Ving_Rhames	1.930
dbr:Category:1990s_crime_films	1.918
dbr:The_Critic	1.830
dbr:Eric_Stoltz	1.827

information. For that, Listing 3.1 shows an example query and Table 3.1 presents the according results (for DBpedia 3.9).

3.1.2.1.2. Strongly Connected Resources PageRank focuses on the general importance of related resources. It does not provide an indication whether the two resources are important for each other. This part is addressed by the Backlink method that was first

Listing 3.2: Example: SPARQL query for retrieving resources with a Backlink to “Pulp Fiction”.

```
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX dbo: <http://dbpedia.org/ontology/>

SELECT DISTINCT ?r WHERE {
  { dbr:Pulp_Fiction ?p ?r } UNION { ?r ?p dbr:Pulp_Fiction }
  dbr:Pulp_Fiction dbo:wikiPageWikiLink ?r .
  ?r dbo:wikiPageWikiLink dbr:Pulp_Fiction .
  FILTER (?p != dbo:wikiPageWikiLink ) .
}
```

Table 3.2: Resources (in no particular order) that have a semantic relation to “Pulp Fiction” and that are connected with Wikipedia Backlinks (DBpedia version 3.9).

dbr:Quentin_Tarantino	dbr:Bruce_Willis
dbr:John_Travolta	dbr:Samuel_L._Jackson
dbr:Harvey_Keitel	dbr:Miramax_Films
dbr:Uma_Thurman	dbr:Andrzej_Seku%C5%82a
dbr:Christopher_Walken	dbr:Roger_Avary
dbr:Tim_Roth	dbr:Ving_Rhames
dbr:Amanda_Plummer	dbr:Lawrence_Bender
dbr:Sally_Menke	dbr:Maria_de_Medeiros
dbr:Rosanna_Arquette	dbr:Eric_Stoltz

described in [WS11]. In this work, the authors analyze a variety of set-based heuristics for identifying related resources in order to feature exploratory search with Linked Data. The analyzed Backlink method performs best in terms of F-measure when the results are compared to their reference dataset. In [WS11] the method is introduced as follows:

$$bl(e) = \{r | link(r, e) \wedge link(e, r), r \in R\} \quad (3.2)$$

For entity summarization, we adapt the Backlink method in order to ensure that a semantic relation exists between e and every r . The adapted formula is as follows:

$$bl(e) = \{r | link(r, e) \wedge link(e, r) \wedge r \in res(e), r \in R\} \quad (3.3)$$

Figure 3.3a shows the Backlink method and the additional requirement for a semantic relation between two resources. Backlink can not be used directly for entity summarization as it returns an unranked set of related entities and the size of this set depends on the target entity. An example for a set of resources retrieved with Backlink for the entity “Pulp Fiction” is provided in Table 3.2. Listing 3.2 provides the according query. The filter rule of the query asserts that $?p$ only matches typed links and, therefore, only resources with an actual semantic relation are contained in the result set. Note that the Wikipedia page links are not stored in the public endpoint of DBpedia but can be easily added to a local deployment.

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

Table 3.3.: Example: Top-10 resources that have a semantic relation to “Pulp Fiction” ranked in accordance to the combined score (with $\alpha = 0.9$).

Resource $r \in res(dbr:Pulp_Fiction)$	$score(dbr:Pulp_Fiction, r)$
dbr:Category:English-language_films	0.900
dbr:Quentin_Tarantino	0.156
dbr:John_Travolta	0.143
dbr:Miramax_Films	0.141
dbr:Bruce_Willis	0.129
dbr:Samuel_L._Jackson	0.127
dbr:Christopher_Walken	0.122
dbr:Uma_Thurman	0.116
dbr:Harvey_Keitel	0.115
dbr: Ving_Rhames	0.108

3.1.2.1.3. Combined Scores for Resource Selection We propose a combination of PageRank with Backlink. This enables us to select relevant resources with a tight connection to e . For this, we normalize the PageRank score of each entity in relation to the maximum and linearly combine the score with the indicator function applied on the set $bl(e)$. With $r \in res(e)$:

$$score(e, r) = \alpha \cdot \frac{pr(r)}{\max\{pr(a) : a \in res(e)\}} + (1 - \alpha) \cdot \mathbf{1}_{bl(e)}(r) \quad (3.4)$$

For a top- k summary we rank the resources $r \in res(e)$ in accordance to the defined score and cut off at k . We define a top- k list of connected resources with the function $top_k(res(e))$. An example for the top-10 related resources of Pulp Fiction is provided in Table 3.3. The α parameter is flexible and lies in the interval $0.5 \leq \alpha \leq 1$. With $\alpha = 0.5$, the top positions of a summary of e first involve all resources contained in the Backlink set $r \in bl(e)$ in the order of their PageRank scores. This listing is followed by the resources that are not in the Backlink set $r \notin bl(e)$ yet still semantically connected in the order of their PageRank scores. This is also the case if α is chosen in the interval $0 < \alpha < 0.5$. With $\alpha = 1.0$, all connected resources $r \in res(e)$ are ordered in accordance to their PageRank scores. In this case, the Backlink set does not influence the results. In Section 3.1.3.1 we present different configurations of LinkSUM with respect to α .

3.1.2.2. Relation Selection

In an RDF knowledge base, two resources can be linked through multiple semantic connections. We provide an example in Figure 3.3b which demonstrates that the entities “Pulp Fiction” and “Quentin Tarantino” are connected in multiple ways. As a matter of fact, it is very common that multiple relations between entities exist. However, in many cases, one relation is more relevant than others. In our approach, the relation selection task

3.1. LinkSUM: Using Link Analysis for Entity Summarization

identifies the most prominent connection for presentation in order to avoid redundancies among the connected resources in the top- k set.

In order to choose an optimal relation selection method for LinkSUM, the following factors were defined:

Frequency (FRQ) Ranks the candidate relations in accordance to how often a specific relation is used overall in the complete dataset. The relation that is used the most is selected as the most promising candidate.

Exclusivity (EXC) For both entities of a relation, the relation might not be exclusive. For example a movie has commonly more than one starring actor while also an actor is usually starring in more than one movie (N:M). This measure considers the exclusivity of a relation in context to e and $r \in res(e)$ respectively. For both resources, e and r , we add up the number of times the relation is used with each (N+M). We use the inverse of this number $1/(N + M)$, in order to get the exclusivity score (the more exclusive, the better). The upper bound of EXC is 0.5 (for a 1:1 relation).

Description (DSC) Relations are represented by RDF predicates. Those predicates are commonly described with domains, ranges, and labels in different languages. The sum $|labels| + |ranges| + |domains|$ forms a basic method for estimating the quality of the description of the predicate. The relation with the highest quality is chosen.

For each related resource in $r \in top_k(res(e))$, combinations of the above-presented relation selection mechanisms identify the most relevant connection to e .

3.1.3. Experiments

In the following, we present a set of experiments that were conducted in order to configure and evaluate LinkSUM effectively.

3.1.3.1. Configuration

As reported in [GTS15], the FACES system (to which we compare) was tuned to its best performance by setting the cut-off level of the cluster hierarchies to 3. Also LinkSUM can be configured with respect to various parameters. First, the α -value for resource selection is flexible in the range of 0.5 to 1 (see Section 3.1.2.1.3). Second, the relation selection method can be adjusted or replaced in order to fit one or another scenario (see Section 3.1.2.2). For finding the best configuration, we considered variants of the following parameters:

α -value We tested different settings for α in the range of 0.5 to 1 with 0.1 steps.

Relation Selection We tested different relation selection mechanisms. We considered only combinations via product¹ based on frequency as it has been shown to be a robust popularity measure [SPS13]. The following setups were considered as promising candidates:

- FRQ – relations are selected by their frequency in the dataset.
- FRQ*EXC – relations are chosen by the product of frequency and exclusivity.
- FRQ*DSC – relations are selected by the product of frequency and description.
- FRQ*EXC*DSC – relations are chosen by the product of frequency, exclusivity, and description.

3.1.3.1.1. DBpedia PageRank We computed the PageRank [BP98] scores for each DBpedia entity for multiple versions of DBpedia (versions 3.8, 3.9, 2014, 2015-04, and 2015-10) in English and—depending on the version—also in other languages (French, German, Italian, Spanish, Russian, and Chinese). As a basis for this, we used DBpedia’s page links dataset.² This dataset contains triples of the form “Wikipedia page A links to Wikipedia page B”. We only use these extracted Web links, in particular we do not make use of semantic links (e.g., `dbo:birthPlace`) for the computation of PageRank. For our computations we used the following parameters: 40 iterations, damping factor 0.85, start value 0.1, non-normalized. The PageRank scores were made available online in Turtle (using the vRank vocabulary [RVTTTS12]) and in tab-separated values (TSV) format.³ The dataset is licensed under the *Creative Commons Attribution-ShareAlike*⁴ license. Since DBpedia version 2015-04, the DBpedia PageRank scores are regularly included in the official DBpedia SPARQL endpoint.

3.1.3.1.2. Configuration Setup We configured LinkSUM to use the DBpedia 3.9 dataset in English language. As additional input for our system we used the DBpedia PageRank scores and, for the Backlink method, we used DBpedia’s Wikipedia page links dataset (both also in English language DBpedia version 3.9; see also Section 3.1.3.1.1). We configured the system to feature outgoing connections only in order to fit the scenario of the used reference dataset (see below).

Our experimental setup further involves a reference dataset as well as measures for computing the agreement and similarity. We use a similar evaluation setup as the FACES approach [GTS15] as we directly compare LinkSUM with the FACES system (see Section 3.1.3.2).

¹We chose to combine the scores via product with the same intuition as it is done in tf-idf (e.g., predicate frequency multiplied with their exclusivity in the specific context of use).

²DBpedia page links datasets – <http://wiki.dbpedia.org/services-resources/documentation/datasets#pagelinks>, retrieved 2016-06-15.

³DBpedia PageRank dataset – http://people.aifb.kit.edu/ath/#DBpedia_PageRank, retrieved 2016-06-14.

⁴Creative Commons Attribution-ShareAlike 3.0 Unported – <https://creativecommons.org/licenses/by-sa/3.0/>, retrieved 2016-06-15.

Reference Dataset We use the same reference dataset as [GTS15].⁵ The dataset provided by FACES involves DBpedia version 3.9 in English language and features outgoing connections only [GTS15]. From this dataset, the authors randomly selected 50 entities. The reference data contains at least seven top-5 and seven top-10 reference summaries per entity that were created by 15 experts of the Semantic Web field [GTS15]. For each entity, these references describe outgoing connections to other resources. The average number of these relations is 44. In addition, several relations, such as `dcterms:subject` and Wikipedia related links, were removed as they do not contain sufficient semantic information [GTS15].

The dataset provided in [BWS16] would also have served as reference for evaluation. Unfortunately, we could not obtain summaries of the FACES system for the entities covered by [BWS16].

Measures For computing the agreement and for comparing the produced summaries with the reference dataset, we use the same similarity measures as [CTQ11] and [GTS15]:

$$Agreement(e) = \frac{2}{n(n-1)} \sum_{i=1}^n \sum_{j=i+1}^n |Sum_i^E(e) \cap Sum_j^E(e)| \quad (3.5)$$

$$Quality(Sum(e)) = \frac{1}{n} \sum_{i=1}^n |Sum(e) \cap Sum_i^E(e)| \quad (3.6)$$

The variable n denotes the number of experts. The expert summaries are denoted as $Sum_i^E(e)$. The agreement measure estimates the agreement of the experts about a top- k summary of the entity e (i.e., the average overlap between the expert summaries). The *Quality* measure estimates the average overlap of the produced summary $Sum(e)$ with all expert summaries. Both values are computed for all entities in the reference dataset and afterwards averaged.⁶ The upper and lower bounds for both measures are $0 \leq Agreement(e) \leq k$ and $0 \leq Quality(Sum(e)) \leq k$ in the top- k setting. When we reproduced the setting of FACES, we found that our results did not match the values provided in [GTS15]: our *Quality* values for FACES were lower than the provided ones. In order to reproduce the reported values for the FACES system in [GTS15], we found out that only the last part of the IRI was used for matching automatically generated summaries with expert summaries for all tested systems. Unfortunately, this setting matches DBpedia predicates with different namespaces (i.e., `dbp` and `dbo`, see Section 2.1.5) in an arbitrary way. As an example, on the one hand, `dbp:party` and `dbo:party` are matched while, on the other hand, `dbp:placeOfBirth` and `dbo:birthPlace` remain unmatched because the last parts of the IRI are syntactically not the same. As a consequence, we decided not to adopt this basic ontology alignment approach and to apply the two following measures instead:

⁵FACES reference dataset – http://knoesis.wright.edu/researchers/kalpa/faces_evaluation.zip, retrieved 2016-06-10.

⁶ k is fixed to the same value for all summaries, expert and automatically generated ones, before applying the measures.

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

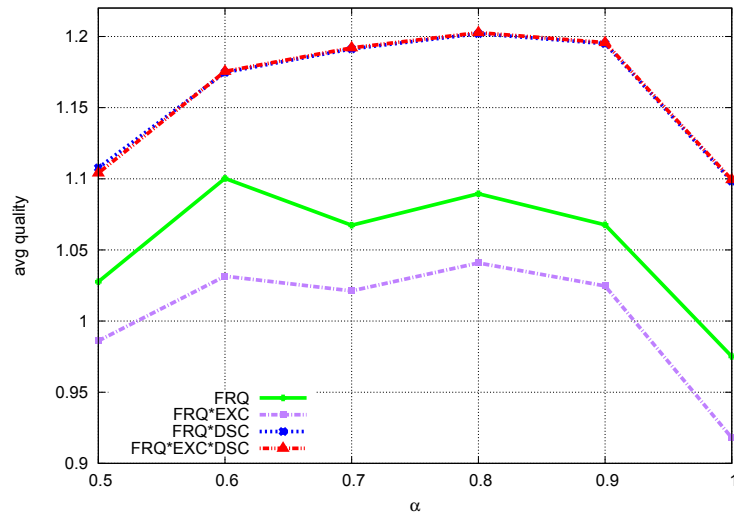


Figure 3.4.: LinkSUM (SPO) average *Quality* scores with different settings for α and different relation selection approaches for top-5 summaries.

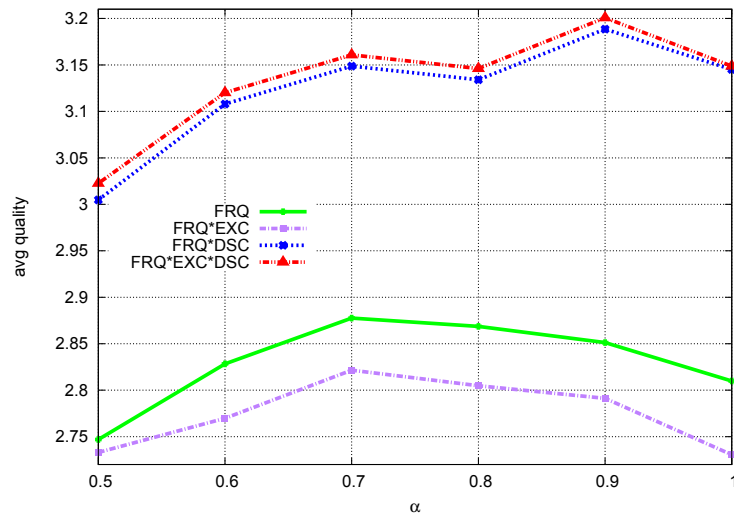


Figure 3.5.: LinkSUM (SPO) average *Quality* scores with different settings for α and different relation selection approaches for top-10 summaries.

- Subject–Object (SO): This measure treats a summary as a set of tuples containing only subjects and objects while ignoring the predicate. The full IRI of the subject and the object are used respectively. As a matter of fact, the relation selection method has no impact on this measure.
- Subject–Predicate–Object (SPO): This measure treats summaries as sets of triples. For representing a triple we use the full IRI of each, the subject, the predicate, and the object. This measure also estimates the performance of the relation selection approach.

Table 3.4.: Agreement among the experts.

	SO	SPO
top-5	2.14	1.64
top-10	5.14	3.92

3.1.3.1.3. Configuration Results In the following, we report about the results of the LinkSUM configuration: In [GTS15], the reported agreement among the experts is 1.92 for top-5 and 4.64 for top-10 respectively. Those values were computed with the aforementioned basic ontology alignment approach. We recomputed the values for SO and SPO respectively. The results are displayed in Table 3.4. The agreement among the experts is not particularly high. According to [GTS15], this can be explained by the high number of facts that were presented to the experts for each entity (in average 44 facts per entity). Although—technically—the average agreement is not an upper bound for the performances of the tested systems, the values can serve as reference points.

In the SO setting, the best achieved scores of LinkSUM are 1.89 (top-5, $\alpha = 0.8$) and 4.82 (top-10, $\alpha = 0.9$) respectively. The results of the SPO settings are shown in Figure 3.4 and Figure 3.5. The FRQ measure provides a good baseline for both, top-5 and top-10. While the combination of FRQ with DSC improves the *Quality* in both settings, the combination with EXC damps the impact of FRQ. In the top-10 setting, the combination of the three measures (FRQ*EXC*DSC) provides best values. In the top-5 settings, FRQ*DSC and FRQ*EXC*DSC provide equally good results. The values for α are best at 0.8 for top-5 and 0.9 for top-10. The impact of the Backlink method on the rankings ($\alpha < 1.0$) in comparison to PageRank-only ($\alpha = 1.0$) is evident. In addition, it is noticeable that strictly prioritizing all results of the Backlink method (ranked in accordance to their respective PageRank scores) does also not yield best results ($\alpha = 0.5$). The full blend between importance and strong connectivity produces the best outcomes.

Summarizing, the following configurations performed best for top-5 and top-10 summaries respectively:

config-1 (top-5): $\alpha = 0.8$, FRQ*EXC*DSC

config-2 (top-10): $\alpha = 0.9$, FRQ*EXC*DSC

3.1.3.2. Evaluation

In our evaluation setting, we compare LinkSUM with the FACES entity summarization system [GTS15]. FACES focuses on the diversification of the relation types (i.e., no semantically similar predicates should occur in the result summary). The system has two stages: partitioning the feature set and ranking the features. The main idea is to partition the semantic links of an entity into semantically diverse clusters of predicates. For resource selection, the approach uses a tf-idf-related popularity measure for the object. In contrast, in our approach we follow the objective to identify the most relevant object first and then select the predicate. In their evaluation, the authors demonstrate that their system provides better results than [CTQ11] and [TR14]. For 50 DBpedia entities, the

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

authors published the results of FACES for top-5 and top-10 summaries (along with the reference dataset described in Section 3.1.3.1.2).⁷ The used DBpedia version is 3.9.

We compare LinkSUM and FACES in two evaluation settings, a quantitative and a qualitative one. In the following, we will first describe the experimental setup and afterwards the obtained results.

3.1.3.2.1. Evaluation Setup

Quantitative Analysis For evaluating the two methods quantitatively, we chose the same setup as described in Section 3.1.3.1.2 which means the same reference dataset and the same evaluation measures that were used for the evaluation of the FACES system [GTS15]. For comparison, we use the average *Quality* value of each method. In addition, in order to prevent influence of strong outliers, we also use the *Quality* value of each of the 50 entities per system for computing significance. As a significance test, we use the Wilcoxon Signed-Rank Test with two tails as recommended in [Dem06]. We compare the best configurations of LinkSUM for top-5 and top-10 respectively (see Section 3.1.3.1.3) with the published results of FACES.

Qualitative Analysis For the qualitative evaluation we sent a call for participation to more than 60 people and asked them to compare summaries of different entities. In this setup, we evaluated the top-10 setting with LinkSUM@config-2 (which turned out to perform best for the top-10 setting in the configuration, see Section 3.1.3.1.3). We chose a set of ten entities out of the 50 provided summaries of FACES with respect to their types. The types of the selected entities involve the following classes: person, country, football club, TV series, movie, and company. The selection between the entities of a specific type was random.

For each entity, we displayed the summaries of the two systems next to each other (see Figure 3.6) without giving indications about which system produced the summaries. The summaries produced by LinkSUM were displayed on the left side in 50% of the cases with random choice. Below each summary, we provided a radio button for the users to choose their preferred summary. Every user had one vote either for LinkSUM or FACES. We used two 5-point Likert scale questions in order to enable participants to provide information about their previous knowledge about the entity and the confidence with their choice:

- “*I know a lot about this entity*” – [Strongly agree; Agree; Neither agree, nor disagree; Disagree; Strongly disagree]
- “*I am sure that I prefer the chosen summary over the other*” – [Very confident; Confident; Neutral; Not very confident; Not at all confident]

⁷FACES summaries – http://knoesis.wright.edu/researchers/kalpa/faces_evaluation.zip, retrieved 2016-06-10.

The Cosby Show

I know a lot about this entity:

Strongly agree
 Agree
 Neither agree, nor disagree
 Disagree
 Strongly disagree

language	English language	network	NBC
network	NBC	language	English language
genre	Sitcom	opening theme	Bobby McFerrin
camera	Multiple-camera setup	composer	Bill Cosby
starring	Bill Cosby	executive producer	Tom Werner
opening theme	Bobby McFerrin	distributor	Paramount Domestic Television
related	A Different World	genre	Sitcom
executive producer	Tom Werner	creator	Bill Cosby
starring	Malcolm-Jamal Warner	company	Viacom Productions
starring	Lisa Bonet	format	480i

>> <<

I am sure that I prefer the chosen summary over the other:

Very confident
 Confident
 Neutral
 Not very confident
 Not at all confident

Please provide reasons why you prefer the chosen summary over the other (optional):

e.g.
- reason 1
- reason 2

Figure 3.6.: Excerpt of the interface for qualitative evaluation for the entity “The Cosby Show”. The users could choose whether they prefer the summary of LinkSUM (left) or FACES (right) in a SERP setting.

Besides we provided an optional field where comments about their choice could be given. We included the following introductory text in order to instruct the users on how to proceed with the evaluation:

“You have been searching on a Web search engine for an entity. The search engine result page (SERP) is displayed with a picture of the entity, a short textual description, and a box with facts about the entity. For the following ten entities, it is your task to decide which fact box you would like to see in a SERP.”

In addition, we asked the participants to assume that all displayed data is correct. This was to avoid influence of data quality on the results. Finally, for statistical classification, we requested the participants to provide the following information: gender, age, whether or not the participants had a background in computer science, and the time taken for evaluation.

3.1.3.2.2. Evaluation Results

Quantitative Analysis In Table 3.5, we present the overall *Quality* results of the quantitative evaluation. In average, both configurations of LinkSUM achieve better results than FACES in the described settings (top-5/top-10, SO/SPO). LinkSUM@config-2 performs significantly better than FACES in all settings ($p < 0.05$). LinkSUM@config-1 is significantly ($p < 0.05$) better than FACES in three of four settings while the level of significance is not fully reached at SPO, top-10.

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

Table 3.5.: Overall *Quality* results of the quantitative evaluation and their respective standard deviation (SD). Best results are bold. † compared to the best, difference is significant ($p < 0.05$); ‡ compared with each of the other two settings, difference is significant ($p < 0.05$).

	SO (top-5)	SPO (top-5)	SO (top-10)	SPO (top-10)
LinkSUM@config-1	1.89 (SD 0.55)	1.20 (SD 0.57)	4.78 (SD 1.05)	3.15 (SD 0.89)
LinkSUM@config-2	1.84 (SD 0.60)	1.20 (SD 0.60)	4.82 (SD 1.06)	3.20 (SD 0.87)
FACES	1.66 [‡] (SD 0.57)	0.93 [‡] (SD 0.54)	4.33 [‡] (SD 1.01)	2.92 [†] (SD 0.94)

Qualitative Analysis From the invited people, a total of 20 participated in the qualitative analysis. 75% of the participants were between 25 and 35, and 25% were between 35 and 45 years old. 75% were male and 25% were female. 95% of the participants had a computer science background. The average time taken for the evaluation was 11 minutes and 27 seconds. In total, 13 participants used the option to comment about their choice. With respect to these characteristics, we did not find any significant difference within the distribution of the votes. The distribution of the votes is visualized in Figure 3.7. 73% of all votes were given to LinkSUM, 27% of the votes were received by FACES. Out of ten entities, LinkSUM system was clearly chosen with more than 15 votes in the case of five entities. For another 2 entities, the LinkSUM system was chosen with votes in the interval 13 to 14. The votes for the remaining three entities were distributed in the interval of 9 to 11 for both systems. Both systems each received in total ten low-confidence votes (“*Not very confident*” or “*Not at all confident*”). This means that 10 out of 146 votes in the case of LinkSUM, and 10 out of 54 votes in the case of FACES were low-confidence votes. With respect to the total number of votes for each system, this means a disproportionate low number of low-confidence votes for LinkSUM. The amount of knowledge of the participants did not influence the preference for either system: the values for high or low knowledge were both in line with the total distribution of the votes.

Another interesting part of the results of the evaluation are the comments of the participants. We group the comments into two categories depending on hints about the decision-making process of the participants. In many cases, the participants gave reason why they selected a summary and/or why they rejected the other. The most-provided reasons for selection/rejection were as follows:

Selection the presented resources are relevant for the entity (e.g., “I like to see Turing machine mentioned for Alan Turing”).

Rejection redundancy (e.g., “The same thing twice once with prize and once with award”), the presented resources do not characterize the entity (e.g., “I do not care about technical aspects such as format”).

3.1.4. Discussion

To select the most relevant facts that characterize an entity is, in many cases, a subjective task. Thus, to produce a generic summary not tailored to any specific background or

3.1. LinkSUM: Using Link Analysis for Entity Summarization

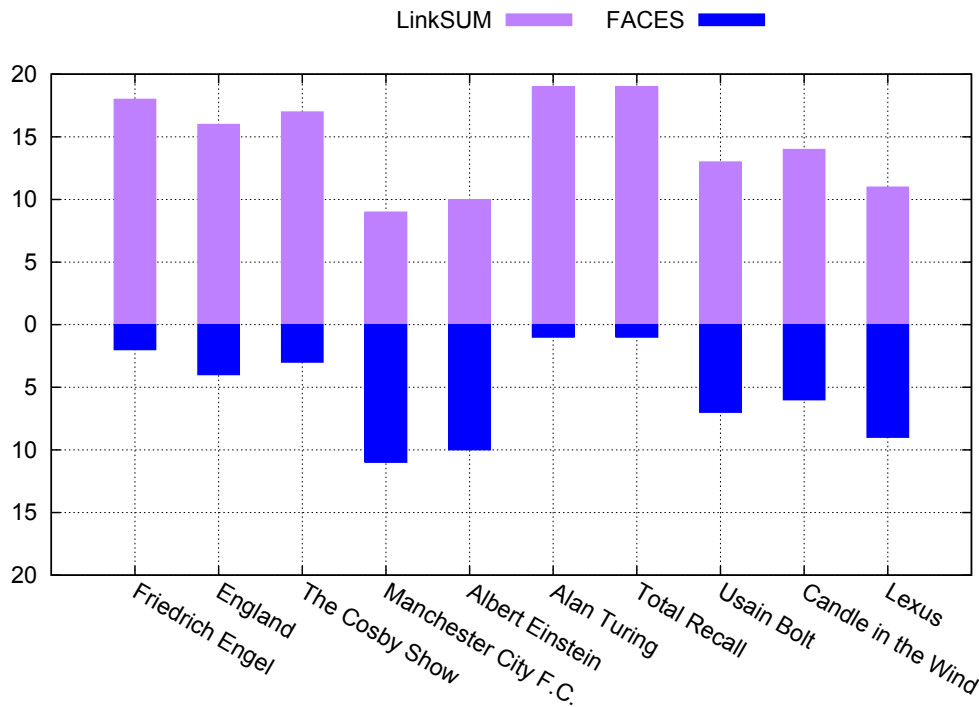


Figure 3.7.: Results of the qualitative evaluation. The x-axis denotes the respective entities and the y-axis accounts for the number of user votes per system.

context the user might have is a challenging task that involves the identification of facts that are deemed important by the majority of the users. In order to address this challenge, the LinkSUM method combines and optimizes methods that enable to select relevant facts about entities and at the same time reduce the amount of redundant information. In our experiments and evaluation we assessed and analyzed the efficiency of the mentioned aspects of the LinkSUM method. In a quantitative as well as qualitative setting we compared LinkSUM to the FACES system. In both setups, we demonstrated that LinkSUM exhibits significantly better results than FACES. The comments of the participants of our qualitative experiment suggest that the relevance of the related resources should be of importance and at the same time characterize an entity. We cover this by the combination of PageRank with Backlink. Our experiments with the SO-measure demonstrate that the produced *Quality* values are close to the agreement of the expert summaries (see Table 3.4).

We have tested four different methods for relation selection. The combination of the frequency of the relation, its exclusivity, and its description has been shown to perform best in the top-10 setting, while in the top-5 setting the exclusivity score did neither contribute positively, nor negatively in that setup. The introduced measures should be considered as baselines for future evaluation settings in context to the relation selection step.

With regard to the qualitative evaluation, in the cases of the entities “Manchester City F.C.”, “Albert Einstein”, and “Lexus” we could not find any clear majority for either of the two systems. In the case of “Lexus” the set of presented facts has a very high

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

overlap between the systems (with different ordering). In the case of “Manchester City F.C.” and “Albert Einstein” the choices are subjective as the provided comments suggest: some users liked the listing of players (“Manchester City F.C.”) or children (“Albert Einstein”) while others stated they did not. Contrary to the claims in [GTS15], we could not find evidence that repetitive relations have a negative impact on the quality of the summaries. For example, the entity “The Cosby Show” contains a listing of various actors with the “starring”-relation in the LinkSUM summary while in the output of FACES this information is missing (see Figure 3.6). This led to 17 (LinkSUM) vs. three (FACES) votes. In this case many of the participants provided the “inclusion of the actors” in the LinkSUM method as the main reason for their choice. The FACES system does not filter redundancies on the object level: it happened that the set of relations was diverse while on the object side, a connected resource was re-occurring multiple times (linked through different relations). An example is the entity “Total Recall (1990 film)” where FACES included the following information: *director Jerry Goldsmith; Artist Jerry Goldsmith; music Jerry Goldsmith; music composer Jerry Goldsmith; screenplay David Cronenberg; writer David Cronenberg*. Those and similar repetitions in the summaries of other entities were commented as “redundant” by a high number of participants (in total ten out of 13 participants with comments mentioned redundancy as a problem). In the context with places, also LinkSUM produces redundancies. For example in many cases, the “birthplace” of a person is provided twice, once with a link to the country and once with a link to the city. This issue can be addressed by increasing the quality of RDF knowledge bases and the introduction of logic RDF compression techniques (e.g., [JHD13]) that remove the facts that can be inferred (in this case the relation to the country can be inferred).

We did not compare the runtime of the system (according numbers are reported in [GTS15]) as some of the computations that are needed for producing a summary with LinkSUM are performed offline (i.e., PageRank). The runtime of LinkSUM depends mostly on the performance of the underlying triplestore. In our tests, we used DBpedia 3.9 and Virtuoso 7.1.0 on a Virtual Machine with 8 cores and 80 GB of RAM. With this setting, the system commonly produced summaries in less than 0.5 seconds.

We demonstrated applicability of the LinkSUM method for the DBpedia and Wikipedia datasets and provide results that significantly improve the state of the art. The LinkSUM system is relevant to many other tasks, like for example

Semantic MediaWiki Semantic MediaWiki (SMW) [KVV⁺07] is a popular extension of the MediaWiki software (used by Wikipedia). In SMW, (hyper-) textual information about entities is combined with structured information about them. Using the hyperlinks of the MediaWiki articles in combination with the semantic links of the SMW, LinkSUM can be used to provide structured summaries of entities in SMW.

Microdata/RDFa The number of Web pages that include semantic information about entities is on the rise [MPB14]. In many sites that focus on specific entities, hyperlinks and semantic links are occurring side by side. A prominent example for such co-occurrence is IMDb. Applied in a Web data setting, LinkSUM can use plain hyperlinks in combination with the hidden semantic information for providing structured summaries of entities that occur on the Web.

LinkSUM is applicable to both of the above-mentioned scenarios and it remains a technical task to implement prototypes. With respect to research, the DBpedia/Wikipedia setting is the most suitable scenario for evaluation as other researchers can also use the same datasets for providing their own summaries and compare them to LinkSUM.

The experiments demonstrated that the use of LinkSUM is particularly beneficial when the relevance of the presented facts about the entity are of high importance. By default, LinkSUM has no mechanisms included that consider predicate diversity or user profiles, contexts, or tasks. However, LinkSUM can serve as a foundation for such approaches. LinkSUM only addresses facts that involve related entities and does not consider facts with literals. The task to include literals is part of the future work.

3.1.5. Related Work

Various previous approaches for entity summarization and ranking in the Semantic Web are related to LinkSUM.

The authors of [CTQ11] introduce RELIN, a summarization system that supports quick identification of entities. The approach applies a “goal directed surfer” which is an adapted version of the random surfer model that is also used in the PageRank algorithm. The main idea of the contribution is to combine textual notions of informativeness and relatedness for the ranking of features. As a major effect, the concise presentation of retrieved entities for quick identification by users after search is one of the scenarios that RELIN supports. In [GTS15], the system is shown to perform significantly worse than FACES.

Google “Knowledge Graph” [Sin12] is an example for an entity search system. The main idea is to enrich search results with summarized information about named entities. While the details of the approach are not public, Amit Singhal, the author of [Sin12] outlines that, for summarization, the system goes back to user data in order to “... study in aggregate what they’ve been asking Google about each item”. This indicates, that Google uses additional data sources for the summaries (i.e., the queries of the users). In addition, this also provides reason to assume that the analysis focuses on informal and partial statements of the subject+predicate or subject+object kind. Examples for such queries could be “Pulp Fiction starring”, for getting more information on the missing objects, or “Pulp Fiction John Travolta” in order to find more information about the relationship between entities. Making assumptions about such decoupled statements involves disambiguation of the missing part. Our approach is similar to this methodology and follows the pattern of identifying important objects first and then select a predicate.

The approach presented by Harth and Kinsella in [HK09] introduces TOPDIS, a tensor-based ranking approach for searching and navigating graph-structured data. Their method called “TOP” treats quadruples as a whole but, being based on a fixpoint iteration, each component of the quadruple is regarded in the context to combination of the three other components. In another step, the “DIS” method, the authors present different methods to combine the ranks, acquired for each component of the quadruple, to a final score. Similar to [HK09], Franz et al. introduce another tensor-based method. The approach uses the PARAFAC tensor decomposition method for deriving authority and hub scores as

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

well as information about the importance of the link type [FSSS09]. In contrast to these ranking mechanisms, in our contribution, we separate the steps of deriving importance of the resource and the importance of the link as we put additional focus on the context that the target entity brings (while TOPDIS and TripleRank address a more general ranking of triples). However, our general PageRank importance scores can be easily augmented or replaced by the scores produced by the TOPDIS or TripleRank methods.

The authors of [SPS13] discuss the notion of diversity for graphical entity summarization. Two algorithms are introduced, of which one is diversity-oblivious (called “PRECIS”) and the other is diversity-aware (called “DIVERSUM”). The evaluation of the algorithms was shaped towards the movie domain and involved expert-based assessments as well as crowd-sourced experiments. The results suggest that the DIVERSUM algorithm was favored over the PRECIS algorithm. A drawback of the method, compared to LinkSUM, is that both algorithms treat the predicate-object pairs in accordance to the predicates only—without any measures on the object.

Also with respect to diversity, Schäfer et al. detect anomalies about entities in accordance to their different types [SRP15]. At the current state, the system needs also the specific type as an input. However, if the main type of an entity is detected reliably, the method can be regarded as an entity summarization system that points out hidden or interesting facts.

Blanco et al. introduce Yahoo’s Spark system [BCMT13], an entity recommendation system that suggests related entities based on a learning approach that employs gradient boosted decision trees. The utilized features range from co-occurrence information over popularity features (such as the click frequency) to graph-theoretic features (such as PageRank). The system focuses on related entity recommendation in the domains of locations, movies, people, sports, and TV shows. The types of entities as well as the type of their relation play an important role in the recommendation process. Connecting predicates are not considered by Spark. The system is currently applied in the Yahoo search system.

Nuzzolese et al. describe the analysis of the DBpedia page links dataset in order to derive Encyclopedic Knowledge Patterns (EKPs) [NPG⁺17]. While this work is mainly based on schema-level information, our method focuses on the actual entities. Similar to our approach, the notion of importance of one class to another is derived by measures that utilize Wikipedia page links. While parts of this work could be reused for producing summaries of entities it focuses on solving the problem at the schema level. This is different from the entity-centric perception of summarization we apply.

Waitelonis and Sack explain in their paper [WS11] how different heuristics can be used for discovering related entities in order to support exploratory search. The tested Backlink heuristic achieves the best results in terms of F-measure. In our contribution, we adopted this method and adapted it in order to fit the scenario of entity summarization. Like all tested heuristics of [WS11], Backlink provides an unranked set of related entities that is not directly usable in top- k settings. As a consequence, for the resource selection approach of LinkSUM, we combine Backlink with PageRank [BP98].

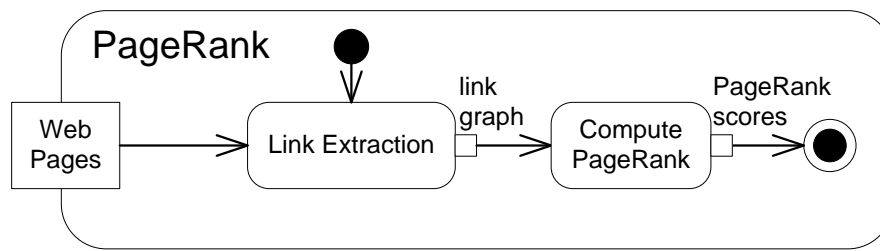


Figure 3.8.: Activity diagram for computing PageRank scores. A set of Web pages (with links) serves as an initial input.

With the work on LinkSUM, we extended on the state of the art in the field of lightweight relevance-oriented entity summarization systems and fact ranking in general.

3.1.6. PageRank Variants

The computed PageRank scores (see Section 3.1.3.1.1) for the different Wikipedia language editions can be used in many different ways. For example, in combination with DBpedia, it is possible to derive rankings such as “the top 100 universities in accordance to their Wikipedia PageRank scores” with a single query (see Listing 3.3).⁸ Similar ranking scenarios are also possible for organizations or persons. These queries typically provide reasonable output rankings although, in some cases, they seem to be obscure. For example, when ranking scientists (Listing 3.3 with `dbo:Scientist` instead of `dbo:University`), the entity “Carl Linnaeus” (512) has a much higher PageRank score than “Charles Darwin” (206) and “Albert Einstein” (184) together.⁹ The reason is easily identified by examining the Wikipedia articles that link to the article of “Carl Linnaeus”:¹⁰ Most articles use the template `Taxobox`¹¹ that defines the field `binomial_authority`. It becomes evident that the page of “Carl Linnaeus” is very often linked because Linnaeus classified species and gave them binomial names in his work “*Systema naturae per regna tria naturae: secundum classes, ordines, genera, species, cum characteribus, differentiis, synonymis, locis*” [LS58]. In general, we found that entities that give structure to the geographic and biological domains have distinctively higher PageRank scores than most entities from other domains. While, given the high inter-linkage of these domains, this is expected to some degree, according to our computations articles such as “Bakhsh” (1914), “Powiat” (1408), “Chordate” (1527), and “Lepidoptera” (1778) are occurring in the top-50 list of all things in Wikipedia (see Table 3.10, column “DBP 2015-04”). These observations led us to the question whether modifications on the input graph can improve these rankings.

In this section, we present the results of our experiments in which we investigated different

⁸An in-depth work on deriving university rankings with the Wikipedia link graph analysis was done by Lages et al. [LPS16].

⁹DBpedia PageRank version 2015-04, see Section 3.1.3.1.1.

¹⁰Articles that link to “Carl Linnaeus” – https://en.wikipedia.org/wiki/Special:WhatLinksHere/Carl_Linnaeus, retrieved 2016-06-15.

¹¹Template:Taxobox – <https://en.wikipedia.org/wiki/Template:Taxobox>, retrieved 2016-06-15.

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

Listing 3.3: Example: SPARQL query on DBpedia for retrieving top-100 universities sorted by their PageRank scores.

```
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX vrank:<http://purl.org/voc/vrank#>

SELECT ?e ?r
FROM <http://dbpedia.org>
FROM <http://people.aifb.kit.edu/ath/#DBpedia_PageRank>
WHERE {
    ?e rdf:type dbo:University .
    ?e vrank:hasRank/vrank:rankValue ?r .
} ORDER BY DESC(?r) LIMIT 100
```

link extraction¹² methods and alternative PageRank variants with the aim to address the root causes of the observed effects. With reference to Figure 3.8, we tested multiple methods for link extraction and we experimented with alternative PageRank variants. Both parameters change the final ranking produced by PageRank. In a more general context, we focus on the question whether some links—based on their context/position in the wikitext—can be deemed more important than others. In our variants, we change the probabilistic impact of links in accordance to their context/position and measure the effects on the output of the PageRank algorithm. We compare the output of these variants with previously computed rankings and with page-view-based rankings and provide statistics on the pairwise computed Spearman and Kendall rank correlations.

The following subsections are structured as follows: We first introduce further background on PageRank, the Wikipedia link graph, and related work in Section 3.1.6.1. Then we present the datasets, the measures, the experimental setup, and the results in Section 3.1.6.2.

3.1.6.1. Background

In the following we provide additional background on PageRank variants, Wikipedia links, and related approaches.

3.1.6.1.1. PageRank Variants For our computations, we use two variants of the PageRank algorithm: The traditional original PageRank Formula [BP98] (see Formula 3.1) and a modified version called “Weighted Links Rank” (WLRank) introduced by Baeza-Yates and Davis in [BYD04]. The latter enables us to account for the relative position of a link within a Wikipedia article. For this, we adapt Formula 3.1 and introduce link weights for PageRank. The idea is that the random surfer is likely not to follow every link on the page with the same probability but may prefer those that are at the top of a page. The WLRank of a resource $r_0 \in R$ (represented by its Wikipedia page) is computed as follows

¹²With “link extraction” we refer to the process of parsing the wikitext of a Wikipedia article and to correctly identify and filter hyperlinks to other Wikipedia articles.

3.1. LinkSUM: Using Link Analysis for Entity Summarization

(with all resources R , their individual directed links $link(r_1, r_2)$; $r_1, r_2 \in R$, and the set of pages that link to a page $l(r) = \{r_1 | link(r_1, r)\}$ —see Formula 3.1):

$$wlr(r_0) = (1 - d) + d \cdot \sum_{r_n \in l(r_0)} wlr(r_n) \cdot \frac{lw(link(r_n, r_0))}{\sum_{r_m} lw(link(r_n, r_m))} \quad (3.7)$$

In [BYD04], a function for the “relative position of a link” is indicated but a clear definition of the measure is missing. We assume that it was defined in a similar way as the following link weight function:

$$lw(link(r_1, r_2)) = 1 - \frac{first_occurrence(link(r_1, r_2), r_1)}{|tokens(r_1)|} \quad (3.8)$$

In order to form a correct probability model, the individual link weight is normalized in accordance to the link weights of all outgoing links of a page in Formula 3.7. If we set the link weight of every incoming link to the same value (e.g., 1) we obtain the original PageRank formula (see Formula 3.1). The used helper functions of Formula 3.8 can be described as follows:

- $first_occurrence(link(r_1, r_2), r_1)$ – the token number of the first occurrence of a $link(r_1, r_2)$ at the respective Wikipedia page of r_1 . The token numbering starts at 1 (i.e., the first word/link in the wikitext).
- $tokens(r_1)$ – the total number of tokens of the Wikipedia page of r_1 .

Tokenization is performed as follows: we split the article text in accordance to white spaces but do not split up links (e.g., `[[brown bear|bears]]` is treated as one token). PageRank and WLRank are iteratively applied until the scores converge. In both formulas, the variable d marks the damping factor (see Section 3.1.2.1.1).

As in all of our computations, we use the non-normalized version of PageRank (and also WLRank). In contrast to the normalized version, the sum of all computed PageRank scores is the number of articles (instead of 1) and, as such, does not reflect a statistical probability distribution. However, normalization has no impact on the final ranking and the resulting relations between the scores.

3.1.6.1.2. The Wikipedia Link Graph In order to extract the link graph from Wikipedia, we need to clarify which types of links are considered. Up to this point, we have made use of the DBpedia page links dataset, a link graph that is constructed by the DBpedia Extraction Framework¹³ (DEF). The DEF bases its extraction on Wikipedia database backup dumps¹⁴ that contain the non-rendered wikitexts of Wikipedia articles and templates. From these sources, DEF builds a link graph by extracting links of the form `[[article|anchor text]]`. We distinguish two types of links with respect to templates:¹⁵

¹³DBpedia Extraction Framework – <https://github.com/dbpedia/extraction-framework>, retrieved 2016-06-15.

¹⁴Wikipedia dumps – <http://dumps.wikimedia.org/>, retrieved 2016-06-15.

¹⁵Template inclusions are marked by double curly brackets: `{{ and }}`.

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

$$A \xrightarrow{PL} B \xrightarrow{PL^R} C \qquad A \xrightarrow{PL} C$$

Figure 3.9.: Transitive resolution of a redirect in Wikipedia. A and C are full articles and B is called a “redirect page”, PL are page links, and PL^R are page links marked as a redirect (e.g., #REDIRECT [[United Kingdom]]). The two page links from A to B and from B to C are replaced by a direct link from A to C .

1. Links that are defined in the Wikipedia text but do not occur within a template, for example `[[brown bear|bears]] outside {{ and }}`.
2. Links that are provided as (a part of) a parameter to a template, for example `[[brown bear|bears]] inside {{ and }}`.

DEF considers only these two types of links and not any additional ones that result from the rendering of an article. It also has to be mentioned that DEF does not consider links from category pages. This mostly affects links to parent categories as the other links that are presented on a rendered category page (i.e., all articles of that category) do not occur in the wikitext. As an effect, the accumulated PageRank of a category page would be transferred almost 1:1 to its parent category. This would lead to a top-100 ranking of things with mostly category pages only. In addition, DEF does not consider links in references (denoted via `<ref>` tags). With respect to redirects, DBpedia offers two types of page link datasets:¹⁶ one in which the redirects are resolved and one in which they are contained. In principle, also redirect chains of more than one hop are possible but, in Wikipedia, the MediaWiki software is configured not to follow such redirect chains (that are called “double redirect” in Wikipedia)¹⁷ and various bots are in place to remove them. Thus, we assume that only single-hop redirects are in place. However, as performed by DBpedia, also single-hop redirects can be resolved (see Figure 3.9). Alternatively, for various applications—in particular those addressing natural language processing (NLP)—it can make sense to keep redirect pages as they can also have a high number of inlinks in various cases (e.g., “Countries of the world”)¹⁸. However, with reference to Figure 3.9 and assuming that redirect pages only link to the redirect target, B passes most of its own accumulated PageRank score directly to C (note that the damping factor is in place).

In order to create new Wikipedia link datasets, we originally aimed to extend the DEF with more general Wikipedia link extraction methods. Unfortunately, in this respect, DEF exhibited certain inflexibilities as it processes Wikipedia articles line by line. This made it difficult to regard links in the context of an article as a whole (e.g., in order to determine the relative position of a link). In consequence, we reverse-engineered the link extraction parts of DEF and created the SiteLinkExtractor¹⁹ tool. The tool enables to execute multiple

¹⁶DBpedia page links – <http://wiki.dbpedia.org/Downloads2015-04>, retrieved 2016-06-15

¹⁷Wikipedia: Double redirects – https://en.wikipedia.org/wiki/Wikipedia:Double_redirects, retrieved 2016-06-15.

¹⁸Inlinks of “Countries of the world” – https://en.wikipedia.org/wiki/Special:WhatLinksHere/Countries_of_the_world, retrieved 2016-06-15.

¹⁹SiteLinkExtractor – <https://github.com/TBritsch/SiteLinkExtractor>, retrieved 2016-06-15.

extraction methods in a single pass over all articles and can also be extended by additional extraction approaches. In its current implementation it does not resolve redirects.

3.1.6.1.3. Related Work There are two common types of Wikipedia rankings: one is based on measures on the link graph and the other is based on consumption (e.g., page views). In the following, we briefly introduce the state of the art in both Wikipedia ranking methods.

Measures on the Wikipedia link graph: The work of Eom et al. [EAL⁺15] investigates the difference between 24 language editions of Wikipedia with PageRank, 2DRank, and CheiRank rankings. The analysis focuses on the rankings of the top-100 persons in each language edition. We consider this analysis as seminal work for investigation on mining cultural differences with Wikipedia rankings. This is an interesting topic as different cultures often use the same language edition of Wikipedia (e.g., United Kingdom and the United States use English). Similarly, the work of Lages et al. provide rankings of universities of the world in [LPS16]. Again, 24 language editions were analyzed with PageRank, 2DRank, and CheiRank. PageRank is shown to be efficient in producing similar rankings like the “Academic Ranking of World Universities (ARWU)” (that is provided yearly by the Shanghai Jiao Tong University). The Open Wikipedia Ranking project by the Laboratory for Web Algorithmics of the Università degli Studi di Milano provide Wikipedia rankings in accordance to PageRank, indegree, page views, and harmonic centrality [BV14] in a Web interface²⁰ for direct comparison.

The above approaches vary the graph measures (PageRank, 2DRank, CheiRank, indegree, harmonic centrality) but do not vary the link extraction methods. In this section, we experiment with both, different input graphs and a combination of a new weighted input graph and WLRank.

Wikipedia consumption patterns: The official page view statistics of various Wikipedia projects are publicly available as dumps²¹ or as a Web API²². Paul Houle aggregated the Wikipedia page views of the years from 2008 to 2013 with different normalization factors (particularly considering the dimensions articles, language, and time). The resulting dataset, called “SubjectiveEye3D”,²³ reflects the aggregated chance for a page view of a specific article in the interval years (2008 to 2013). Similar to non-normalized PageRank, the scores need to be interpreted in relation to each other (i.e., the scores do not reflect a proper probability distribution as they do not add up to one). In the Github project documentation of SubjectiveEye3D Paul Houle reports about Spearman and Kendall rank correlations between SubjectiveEye3D and our published PageRank computations (see Section 3.1.3.1.1).

²⁰The Open Wikipedia Ranking – <http://wikirank.di.unimi.it/>, retrieved 2016-06-16.

²¹Page view statistics for Wikimedia projects – <https://dumps.wikimedia.org/other/pagecounts-raw/>, retrieved 2016-06-16.

²²Wikipedia Pageview API – <https://wikitech.wikimedia.org/wiki/Analytics/PageviewAPI>, retrieved 2016-06-16.

²³SubjectiveEye3D – <https://github.com/paulhoule/telepath/wiki/SubjectiveEye3D>, retrieved 2016-06-16.

3.1.6.2. Experiments

In our experiments, we first computed PageRank and WLRank on link graphs that were produced with different link extraction methods. We then measured the pairwise rank correlations (Spearman's ρ and Kendall's τ)²⁴ between these rankings and the reference datasets (of which three are also based on PageRank and two are based on page-view data of Wikipedia). With the resulting correlation scores, we investigated whether the resulting correlations could be used to support the following statements:

Statement 1: Links in templates are created in a “please fill out” manner and rather negatively influence on the general salience that PageRank scores should represent.

Statement 2: Links that are mentioned at the beginning of articles are more often clicked and correlate with the number of page views that the target page receives.

Statement 3: The practice of resolving redirects does not strongly impact on the final ranking in accordance to PageRank scores.

3.1.6.2.1. Link Graphs We implemented five Wikipedia link extraction methods that enable to create different input graphs for the PageRank/WLRank algorithm. In general we follow the example of DEF and consider type 1 and 2 links for extraction. The following extraction methods were implemented:

All Links (ALL) This extractor produces all type 1 and 2 links. This is the reverse-engineered DEF method. It serves as a reference.

Article Text Links (ATL) This measure omits links that occur in Wikipedia templates (i.e., includes type 1 links, omits type 2 links). The relation to ALL is as follows: $ATL \subseteq ALL$.

Article Text Links with Relative Position (ATL-RP) This measure extracts all links from the Wikipedia text (type 1 links) and produces a score for the relative position of each link (see Formula 3.8). In effect, the link graph ATL-RP is the same as ATL but uses edge weights based on each link's position.

Abstract Links (ABL) This measure extracts only the links from Wikipedia abstracts. We chose the definition of DBpedia which defines an abstract as the first complete sentences that accumulate to less than 500 characters.²⁵ This link set is a subset of all type 1 links (in particular: $ABL \subseteq ATL$).

Template Links (TEL) This measure is complementary to ATL and extracts only links from templates (i.e., omits type 1 links, includes type 2 links). The relation to ALL is as follows: $TEL \subseteq ALL$.

²⁴Both measures have a value range from -1 to 1 and are specifically designed for measuring rank correlation.

²⁵DBpedia abstract extraction – <https://github.com/dbpedia/extraction-framework/blob/110d9ba882539e3140d733dbaeacdacf2d02559/core/src/main/scala/org/dbpedia/extraction/mappings/MissingAbstractsExtractor.scala#L203>, retrieved 2016-06-16.

Redirects are not resolved in any of the above methods. We executed the introduced extraction mechanisms on a dump of the English Wikipedia that was chosen in accordance to the input of DEF with respect to DBpedia version 2015-04.²⁶ Table 3.6 provides an overview of the number of links per graph.

The following parameters were used to compute the PageRank/WLRank scores on the introduced link graphs: non-normalized, 40 iterations, damping factor 0.85, start value 0.1.

3.1.6.2.2. Reference Datasets The following datasets are ranked in accordance to different criteria. We use them as references: We used the following rankings as reference datasets:

DBpedia PageRank (DBP) The scores of DBpedia PageRank (see Section 3.1.3.1.1) are based on the DBpedia page links dataset (i.e., Wikipedia page links as extracted by DEF, redirected). For our comparisons, we used the following versions of DBP scores based on English Wikipedia: 2014, 2015-04.

DBpedia PageRank unredirected (DBP-U) This dataset is computed in the same way as DBP but uses the DBpedia page links unredirected dataset.²⁷ As the name suggests, Wikipedia redirects are not resolved in this dataset (see Section 3.1.6.1.2 for more background on Wikipedia links and redirects). We use the 2015-04 version of DBP-U.

SubjectiveEye3D (SUB) As described in Section 3.1.6.1.3, Paul Houle aggregated the Wikipedia page views of the years 2008 to 2013 with different normalization factors (particularly considering the dimensions articles, language, and time). As such, SubjectiveEye3D reflects the aggregated chance for a page view of a specific article in the interval years 2008 to 2013. However, similar to unnormalized PageRank, the scores need to be interpreted in relation to each other (i.e., the scores do not reflect a proper probability distribution as they do not add up to one).

The Open Wikipedia Ranking (TOWR) The TOWR project is maintained by the Laboratory for Web Algorithmics of the Università degli Studi di Milano (as described in Section 3.1.6.1.3). It provides Wikipedia rankings in accordance to different ranking methods in a Web interface²⁸ for direct comparison. They provide the following measures:²⁹

TOWR-PR PageRank computed on the Wikipedia link graph with the parallel Gauß-Seidel method [KCN06] of the LAW³⁰ library.

²⁶2015-02-05, Source: DBpedia 2015-04 dump dates – <http://wiki.dbpedia.org/services-resources/datasets/dataset-2015-04/dump-dates-dbpedia-2015-04>, retrieved 2016-06-19.

²⁷DBpedia page links unredirected – <http://wiki.dbpedia.org/services-resources/documentation/datasets#pagelinksunredirected>, retrieved 2016-06-15.

²⁸The Open Wikipedia Ranking – <http://wikirank.di.unimi.it/>

²⁹For their 2015 edition (that we analyze), the link-graph-based measures are applied on an English Wikipedia extract of April 3, 2015. Links in infoboxes were not considered.

³⁰LAW – <http://law.di.unimi.it/>, retrieved 2016-07-19.

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

TOWR-H Harmonic centrality as introduced in [BV14] computed on the Wikipedia link graph.

TOWR-I Indegree, ranks Wikipedia pages in accordance to their number of incoming links.

TOWR-PV Page views, ranks Wikipedia pages in accordance to “the number of page views in the last year”³¹.

The two page-view-based rankings (i.e., SUB and TOWR-PV) serve as a reference in order to compare the different graph-based rankings. We show the mutual overlap of entities covered by the individual rankings in Table 3.7.

We used MATLAB for computing the pairwise Spearman’s ρ and Kendall’s τ correlation scores. The Kendall’s τ rank correlation measure has $\mathcal{O}(n^2)$ complexity and takes a significant amount of time for large matrices. In order to speed this up, we sampled the data matrix by a random selection of 1 million rows for Kendall’s τ . The pairwise correlation scores of ρ and τ are reported in Table 3.8 and Table 3.9 respectively. The results are generally as expected: For example, the page-view-based rankings correlate strongest with each other. Also DBP-U 2015-04 and ALL have a very strong correlation (these rankings should be equal).

3.1.6.2.3. Results *Statement 1* seems to be supported by the data as the TEL PageRank scores correlate worst with any other ranking. However, ATL does not correlate better with SUB and TOWR-PV than ALL. This indicates that the reason for the bad correlation might not be due to the “bad semantics of links in the infobox”. With random samples on ATL—which produced similar results—we found that the computed PageRank values of TEL are mostly affected by the low total link count (see Table 3.6). With respect to the initial example, the PageRank score of “Carl Linnaeus” is reduced from 512 to 217 in ATL. From a subjective perspective, this suggests a positive influence. However, objectively, better performance of ATL is not noticeable with respect to the comparison to SUB and TOWR-PV. We assume that PageRank on DBpedia’s RDF data results in similar scores as TEL as DBpedia extracts its semantic relations mostly from Wikipedia’s infoboxes.

The scores of ABL and ATL-RP are indicators for the support of *Statement 2*. However, similar to TEL, ABL does not produce enough links for a strong ranking. ATL-RP, in contrast, produces the strongest correlation with SUB. This is an indication that—indeed—articles that are linked at the beginning of a page are more often clicked. This is supported by related findings where actual HTTP referrer data was analyzed [DSLS16].

With respect to *Statement 3*, we expected DBP-U 2015-04 and DBP 2015-04 to correlate much stronger but DEF does not implement the full workflow of Figure 3.9: although it introduces a link $A \rightarrow C$ and removes the link $A \rightarrow B$, it does not remove the link $B \rightarrow C$. As such, the article B occurs in the final entity set with the lowest PageRank score of 0.15 (as it has no incoming links). In contrast, these pages often accumulate PageRank scores of 1000 and above in the unredirected datasets. If B would not occur in the final ranking of

³¹Source: <http://wikirank-2015.di.unimi.it/more.html>, retrieved 2016-07-19.

DBP 2015-04, it would not be considered by the rank correlation measures. This explains the comparatively weak correlation between the redirected and unredirected datasets.

Further observations: Another surprising result is the rather weak correlation of TOWR-PR with all the other PageRank-based rankings. As the Wikipedia dump date of DBpedia 2015-04 (that we also used for our measures, see Section 3.1.6.2) is only two months apart from the dump date used by TOWR, we expected much stronger correlations here. This is amplified by the observation that TOWR-PR correlates stronger with older DBP versions. However, Table 3.7 already suggests a clear difference with respect to the number of covered entities. Therefore, we assume that the preprocessing of the link graph performed by TOWR induces this bias. This is also supported by the strong correlations between the link-graph-based TOWR measures (i.e., TOWR-PR, TOWR-H, and TOWR-I) visible in Table 3.8 and Table 3.9.

In addition to ATL-RP, also the link-graph-based TOWR measures exhibit a stronger correlation with SUB than the other PageRank-based measures. However, with respect to Table 3.7 it becomes clear that their overlap with SUB is 949 603 entities less than the one of ATL-RP (or -19% relative to the overlap of ATL-RP and SUB). With this difference, the correlation scores are not directly comparable.

3.1.6.2.4. Summary Whether links from templates are excluded or included in the input link graph does not impact strongly on the objective quality of rankings produced by PageRank. WLRank with links weighted by their relative position produced best results with respect to the correlation to page-view-based rankings. In general, although there is a correlation, we assume that link and page-view-based rankings are complementary. This is supported by Table 3.10 which contains the top-50 scores of SUB, DBP 2015-04, and ATL-RP: The PageRank-based measures are strongly influenced by articles that relate to locations (e.g., countries, languages, etc.) as they are highly interlinked and referenced by a very high fraction of Wikipedia articles. In contrast, the page-view-based ranking of SubjectiveEye3D covers topics that are frequently accessed and mostly relate to pop culture or important historical figures or events. We assume that a strong and more objective ranking of entities is probably achieved by combining link-based and consumption-based rankings on Wikipedia. For applications that deal with NLP, we recommend to use the unredirected version of DBpedia PageRank.

The main findings of our PageRank-related experiments can be summarized as follows:

1. Removing template links has no general influence on the PageRank scores.
2. The results of WLRank with respect to the relative position of a link indicate a better correlation to page-view-based rankings than other PageRank methods.
3. If redirects are resolved, it should be done in a complete manner as otherwise entities get assigned artificially low scores. We recommend using an unredirected dataset for applications in the NLP context.
4. Link-based and consumption-based rankings have individual biases. The two signals should be considered in combination for a more neutral ranking of resources.

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

Table 3.6.: Number of links per link graph. Duplicate links were removed in all graphs (except in ATL-RP where multiple occurrences have different positions).

ALL	ATL	ATL-RP	ABL	TEL
159 398 815	142 305 605	143 056 545	32 887 815	26 460 273

Table 3.7.: Number of mutually covered entities (the colors are used for better readability and have no further meaning).

TOTAL	DBP 3.8	DBP 3.9	DBP 2014	DBP 2015-04	DBP-U 2015-04	ALL	ATL	ATL-RP	ABL	TEL	TOWR-PR	TOWR-H	TOWR-I	TOWR-PV	SUB
23035755	17082708	18172871	19437352	20473313	20473371	18493968	17846024	17846024	12319754	5028217	4853042	4853042	4853042	4853042	6211717
17082708	17082708	16553538	16084755	15814436	15814433	14501459	14119610	14119610	10238803	4086481	4082009	4082009	4082009	4082009	4899380
18172871	16553538	18172871	17528557	17183483	17183460	15682785	15241442	15241442	10880926	4339961	4316452	4316452	4316452	4316452	5234094
19437352	16084755	17528557	19437352	18923198	18923198	17151451	16613563	16613563	11639177	4676614	4612952	4612952	4612952	4612952	5193106
20473313	15814436	17183483	18923198	20473313	20473309	18479125	17833498	17833498	12310229	5026674	4781197	4781197	4781197	4781197	5235341
20473371	15814433	17183460	18923126	20473209	20473371	18479281	17833616	17833616	12310235	5026723	4781197	4781197	4781197	4781197	5235318
18493968	14501459	15682785	17151451	18479125	18479281	18493968	17845902	17845902	12311648	5028094	4780590	4780590	4780590	4780590	4936935
17846024	14119610	15241442	16613563	17833498	17833616	17845902	17846024	17846024	12311477	4382197	4779031	4779031	4779031	4779031	4936085
17846024	14119610	15241442	16613563	17833498	17833616	17845902	17846024	17846024	12311477	4382197	4779031	4779031	4779031	4779031	4936085
12319754	10236803	10880926	11639177	12310235	12310235	12311648	12311477	12311477	12319754	4062460	4739103	4739103	4739103	4739103	4425820
5028217	4086481	4339961	4676614	5026674	5026674	5028094	4382197	4382197	4062460	5028217	3320432	3320432	3320432	3320432	2913541
4853042	4082009	4316452	4612952	4781197	4781197	4780590	4779031	4779031	4739103	3320432	4853042	4853042	4853042	4853042	3986482
4853042	4082009	4316452	4612952	4781197	4781197	4780590	4779031	4779031	4739103	3320432	4853042	4853042	4853042	4853042	3986482
4853042	4082009	4316452	4612952	4781197	4781197	4780590	4779031	4779031	4739103	3320432	4853042	4853042	4853042	4853042	3986482
4853042	4082009	4316452	4612952	4781197	4781197	4780590	4779031	4779031	4739103	3320432	4853042	4853042	4853042	4853042	3986482
6211717	4899380	5234094	5193106	5235341	5235318	4936935	4936085	4936085	4425820	2913541	3986482	3986482	3986482	3986482	6211717

Legend 30000000 0

Table 3.8.: Correlation: Spearman's ρ (the colors are used for better readability and have no further meaning).

	DBP 3.8	DBP 3.9	DBP 2014	DBP 2015-04	DBP 2015-04	ALL	ATL	ATL-RP	ABL	TEL	TOWR-PR	TOWR-H	TOWR-I	TOWR-PV	SUB
DBP 3.8	1.000	0.965	0.930	0.885	0.686	0.689	0.692	0.646	0.672	0.295	0.832	0.736	0.777	0.624	0.541
DBP 3.9	0.965	1.000	0.960	0.910	0.707	0.699	0.701	0.653	0.685	0.289	0.872	0.768	0.810	0.638	0.537
DBP 2014	0.930	0.960	1.000	0.941	0.719	0.709	0.712	0.661	0.700	0.278	0.904	0.796	0.836	0.648	0.502
DBP 2015-04	0.885	0.910	0.941	1.000	0.771	0.756	0.758	0.708	0.770	0.164	0.772	0.697	0.723	0.654	0.551
DBP-U 2015-04	0.696	0.707	0.719	0.771	1.000	1.000	0.985	0.945	0.792	0.344	0.773	0.695	0.726	0.657	0.582
ALL	0.689	0.699	0.709	0.756	1.000	1.000	0.985	0.945	0.788	0.346	0.782	0.707	0.731	0.661	0.565
ATL	0.692	0.701	0.712	0.758	0.985	1.000	0.958	1.000	0.958	0.294	0.792	0.711	0.732	0.658	0.551
ATL-RP	0.646	0.653	0.661	0.708	0.945	0.945	0.958	1.000	0.794	0.315	0.794	0.714	0.736	0.646	0.642
ABL	0.672	0.685	0.700	0.770	0.792	0.788	0.797	0.794	1.000	0.263	0.542	0.441	0.535	0.499	0.455
TEL	0.295	0.289	0.278	0.164	0.344	0.346	0.294	0.315	0.263	1.000	0.487	0.425	0.522	0.419	0.407
TOWR-PR	0.832	0.872	0.904	0.772	0.773	0.782	0.792	0.794	0.542	0.487	1.000	0.859	0.889	0.645	0.593
TOWR-H	0.736	0.768	0.796	0.697	0.695	0.707	0.711	0.714	0.441	0.425	0.859	1.000	0.809	0.677	0.614
TOWR-I	0.777	0.810	0.836	0.723	0.728	0.731	0.732	0.736	0.535	0.522	0.889	0.809	1.000	0.668	0.616
TOWR-PV	0.624	0.638	0.648	0.654	0.657	0.661	0.658	0.646	0.499	0.419	0.645	0.677	0.668	1.000	0.857
SUB	0.541	0.537	0.502	0.551	0.582	0.565	0.551	0.642	0.455	0.407	0.593	0.614	0.616	0.857	1.000



Table 3.9.: Correlation: Kendall's τ on a sample of 1 000 000 (the colors are used for better readability and have no further meaning).

	DBP 3.8	DBP 3.9	DBP 2014	DBP 2015-04	DBP 2015-04	ALL	ATL	ATL-RP	ABL	TEL	TOWR-PR	TOWR-H	TOWR-I	TOWR-PV	SUB
DBP 3.8	1.000	0.931	0.879	0.798	0.611	0.606	0.604	0.548	0.571	0.209	0.695	0.569	0.625	0.455	0.383
DBP 3.9	0.931	1.000	0.924	0.826	0.627	0.620	0.618	0.556	0.583	0.205	0.740	0.598	0.658	0.464	0.379
DBP 2014	0.879	0.924	1.000	0.862	0.647	0.637	0.633	0.565	0.598	0.199	0.785	0.623	0.686	0.471	0.354
DBP 2015-04	0.798	0.826	0.862	1.000	0.761	0.743	0.725	0.632	0.689	0.116	0.615	0.524	0.563	0.473	0.392
DBP-U 2015-04	0.611	0.627	0.647	0.761	1.000	0.990	0.948	0.837	0.680	0.254	0.615	0.521	0.565	0.474	0.413
ALL	0.606	0.620	0.633	0.743	0.990	1.000	0.951	0.839	0.675	0.256	0.623	0.532	0.569	0.478	0.400
ATL	0.604	0.618	0.633	0.725	0.948	0.951	1.000	0.859	0.686	0.207	0.642	0.538	0.572	0.476	0.389
ATL-RP	0.548	0.556	0.565	0.632	0.837	0.839	0.859	1.000	0.689	0.222	0.633	0.540	0.573	0.464	0.463
ABL	0.571	0.583	0.598	0.689	0.680	0.675	0.686	0.689	1.000	0.198	0.405	0.321	0.408	0.363	0.328
TEL	0.209	0.205	0.199	0.116	0.254	0.256	0.207	0.222	0.198	1.000	0.360	0.313	0.397	0.304	0.294
TOWR-PR	0.695	0.740	0.785	0.615	0.615	0.623	0.642	0.633	0.405	0.360	1.000	0.687	0.743	0.467	0.425
TOWR-H	0.569	0.598	0.623	0.524	0.521	0.532	0.538	0.540	0.321	0.313	0.687	1.000	0.647	0.494	0.443
TOWR-I	0.625	0.668	0.686	0.563	0.565	0.569	0.572	0.573	0.408	0.397	0.743	0.647	1.000	0.500	0.457
TOWR-PV	0.455	0.464	0.471	0.473	0.474	0.478	0.476	0.464	0.363	0.304	0.467	0.494	0.500	1.000	0.695
SUB	0.383	0.379	0.354	0.392	0.413	0.400	0.389	0.463	0.328	0.284	0.425	0.443	0.457	0.695	1.000



3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

Table 3.10.: The top-50 rankings of SubjectiveEye3D (< 0.3, above are: Wiki, HTTP 404, Main Page, How, SDSS), DBP 2015-04, and ATL-RP.

Rank	SUB	DBP 2015-04	ATL-RP
1	YouTube	Category:Living people	United States
2	Searching	United States	World War II
3	Facebook	List of sovereign states	France
4	United States	Animal	United Kingdom
5	Undefined	France	Race and ethnicity in the United States Census
6	Lists of deaths by year	United Kingdom	Germany
7	Wikipedia	World War II	Canada
8	The Beatles	Germany	Association football
9	Barack Obama	Canada	Iran
10	Web search engine	India	India
11	Google	Iran	England
12	Michael Jackson	Association football	Latin
13	Sex	England	Australia
14	Lady Gaga	Australia	Russia
15	World War II	Arthropod	China
16	United Kingdom	Insect	Italy
17	Eminem	Russia	Japan
18	Lil Wayne	Japan	Village
19	Adolf Hitler	China	Moth
20	India	Italy	World War I
21	Justin Bieber	English language	Romanize
22	How I Met Your Mother	Poland	Spain
23	The Big Bang Theory	London	Romanization
24	World War I	Spain	Europe
25	Miley Cyrus	New York City	Romania
26	Glee (TV series)	Catholic Church	Soviet Union
27	Favicon	World War I	London
28	Canada	Bakhsh	English language
29	Sex position	Latin	Poland
30	Kim Kardashian	Village	New York City
31	Australia	Counties of Iran	Catholic Church
32	Rihanna	Provinces of Iran	Brazil
33	Steve Jobs	Lepidoptera	Netherlands
34	Selena Gomez	California	Greek language
35	Internet Movie Database	Brazil	Category:Unprintworthy redirects
36	Sexual intercourse	Romania	Scotland
37	Harry Potter	Europe	Sweden
38	Japan	Soviet Union	California
39	New York City	Chordate	Species
40	Human penis size	Netherlands	French language
41	Germany	New York	Mexico
42	Masturbation	Administrative divisions of Iran	Genus
43	September 11 attacks	Iran Standard Time	United States Census Bureau
44	Game of Thrones	Mexico	Turkey
45	Tupac Shakur	Voivodeship (Poland)	New Zealand
46	1	Sweden	Census
47	Naruto	Powiat	Middle Ages
48	Vagina	Gmina	Paris
49	Pornography	Moth	Communes of France
50	House (TV series)	Departments of France	Switzerland

3.1.7. Conclusions

We presented LinkSUM, a generic relevance-oriented method for entity summarization. LinkSUM works with a lightweight two-stage approach in order to produce summaries for entities. In the first step, the method identifies relevant connected resources. In the second step, the system selects the most promising semantic relation for each of the connected resources. We also investigated the most effective configuration parameters for LinkSUM and tested different variants of computing PageRank on Wikipedia.

The results of our quantitative and qualitative evaluation, where we compared LinkSUM to the state-of-the-art system FACES [GTS15], and the experiments on Wikipedia PageRank lead us to the following conclusions:

1. The incorporation of background knowledge for entity summarization is common practice (e.g., [Sin12]). Web links can serve as lightweight background knowledge for effective entity summarization.
2. For SERP scenarios, summarization systems should primarily focus on the relevance and the strength of the connection to the related resources. As a second factor, the selection of an appropriate semantic relation is of importance.
3. Redundancies in the set of related resources should be avoided (e.g., see Figure 3.2). Commonly, if two entities are related, there is one relation that is more relevant to be mentioned. Summaries should focus on this relation and then present relations to other interesting resources.
4. Ranking in RDF knowledge bases is an important problem. Link and consumption-based measures should be considered in combination for providing maximal neutral resource rankings.

The application field of LinkSUM is not limited to SERPs. As the availability of structured data is growing, applications for different domains and purposes emerge. Examples include business intelligence, e-learning, health information systems, news pages, data sheets, recipes etc. In fact, this includes all domain-specific cases where it is necessary for users to efficiently comprehend large information resources. In addition, entity summarization systems may adapt to user-context factors such as geolocation, cultural background, or time. As entities are retrieved without a specific information demand (in contrast to question answering) the full personalization/contextualization of entity summaries remains an open challenge.

LinkSUM provides high-quality summaries and improves on the performance of existing solutions in the literature. There are interesting open points that can be addressed by future systems that are based on LinkSUM :

- The evaluation of LinkSUM was focused on the case of generic search in the Web. The performance of the LinkSUM method could also be evaluated in specific domain settings (e.g., health information).

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

- LinkSUM can be combined with a learning-to-rank approach (similar to [DFDM12]) with respect to the α -value and different linear combinations of the predicate selection methods.
- In future versions, LinkSUM could include literal values—such as strings or dates—as descriptors of the entities. The blending of entity-literal and entity-entity relations into a single summary is a problem of high interest.

3.2. UBES: Leveraging Usage Data for Entity Summarization

We address Research Question 1.2 in this section:

How can we use usage data analysis effectively in order to derive summaries of entities?

The contribution—which addresses this question—states one of the two entity summarization approaches that are introduced in this thesis (see Figure 3.10).

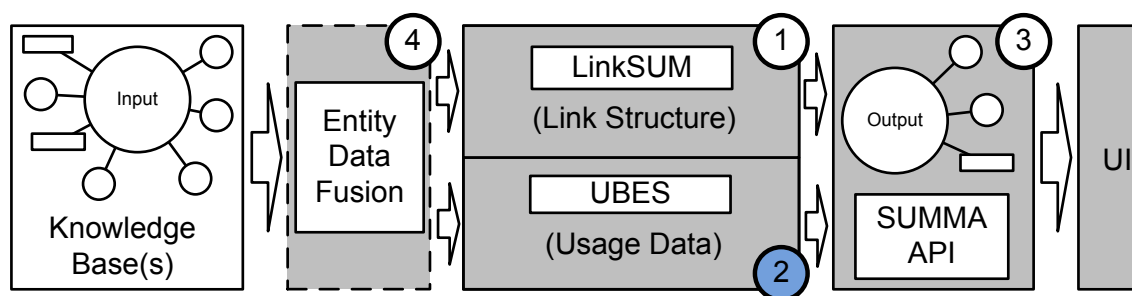


Figure 3.10.: Overview of the contributions of this thesis. In this part, we focus on UBES, an entity summarization approach that leverages usage data.

3.2.1. Introduction

In recent years, a lot of efforts in the Linked Data community have been focused on interfaces (e.g., Pubby,³² Semantic MediaWiki [KVV⁺07], Wikibase,³³ etc.) or visualizations [DR11]. Usage data, such as server logging data, from instances of these and further interfaces were published in anonymized form (e.g., [LRABH15]). Usage data is commonly gained from user sessions which, in the case of Semantic Web resources, can have the following shape [LMN11]:³⁴

³²Pubby – <http://www4.wiwiwiss.fu-berlin.de/pubby/>, retrieved 2016-06-23.

³³Wikibase – <http://wikiba.se/>, retrieved 2016-06-23.

³⁴Abstracted from typical server logs in which it is typically more difficult to identify a single user and their click-paths because of the use of proxy servers, client-side caching, etc. (see [LMN11]).

3.2. UBES: Leveraging Usage Data for Entity Summarization

Table 3.11.: Example: User-item matrix created as an abstraction of usage data of resources.

	User1	User2	User3	User4	...
dbr:Pulp_Fiction	1	0	0	0	...
dbr:Sin_City_(film)	1	0	0	1	...
dbr:Quentin_Tarantino	0	1	0	0	...
...

```
User1, dbr:Pulp_Fiction; 2016-05-24 14:34:21 UTC
User1, dbr:Sin_City_(film); 2016-05-24 14:35:43 UTC
User2, dbr:Quentin_Tarantino; 2016-05-25 11:42:41 UTC
User4, dbr:Sin_City_(film); 2016-05-25 22:15:03 UTC
...
```

This data can be used directly for analyzing click paths [SHHS15] or further processed and used at different levels of abstraction. We coarsely distinguish between two different types of abstraction:

Page views The number of times a resource was consumed within a specified time frame:

```
dbr:Pulp_Fiction; 34 clicks; 2016-05-24 14:00-15:00 UTC
```

This enables popularity-based ranking of resources that is comparable (but orthogonal) to PageRank measures (see Section 3.1.6).

User-item matrix A (typically sparse [AT05]) two-dimensional matrix (in [LMN11] called user-pageview matrix) in which one dimension stands for a user while the other dimension represents items (in our case entities/resources). In such matrices, the entries usually indicate the strength of the preference of the user (in a custom format, for example from 0 to 5) for the respective item but can also be in binary format (1 for *consumed*, 0 for *not consumed*). Note that the user-item matrix abstracts from the temporal dimension and the ordering of the requests (we do not know whether User1 browsed `dbr:Pulp_Fiction` before or after `dbr:Sin_City_(film)`). An example for a binary user-item matrix is provided in Table 3.11.

The use of popularity signals that are similar to page view counts (i.e., PageRank) for entity summarization was discussed in Section 3.1. In this section we address the task of informative/general entity summarization with the help of usage data: we investigate co-consumption patterns of entities with the help of a user-item matrix. We produce summaries of entities by combining usage patterns with knowledge from RDF knowledge bases. With the work on UBES, we aimed to address the following challenges:

1. In many settings—like in the scenario of Section 1.1.1—the amount of data about the user is limited and the actual content of a site as well as previous keyword search terms are unknown. This leads us to the question: How can we derive entity summaries from co-consumption patterns?

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

2. Entity summarization is a novel field of research and established ground truths are missing. To motivate users to help for providing data, annotations for data, and helping to establish a reference is a challenging task [SH08]. How can we efficiently produce a ground truth for evaluating entity summarization?
3. The new entity summarization approach UBES needs to be evaluated. How can we evaluate the UBES entity summarization approach with respect to a reference and to which other systems can we compare?

Accordingly, we provide the following contributions:

1. We introduce UBES, a usage-based approach to summarize entities within RDF knowledge bases. UBES uses the k -nearest neighbors (kNN) technique for entities of the same type and an adaptation of the term frequency-inverse document frequency (tf-idf) method for identifying important predicate-object pairs. We exemplify the applicability of the approach on instances of type film/movie³⁵ of the Freebase RDF knowledge base.
2. We provide a scenario for establishing a reference dataset for entity summarization by creating a game with a purpose (GWAP). For this, we adapt an existing linked data quiz game (that was originally introduced for linked data quality assessment and improvement in [WLKS11] and [KHS12]) in order to cover the movie domain of the Freebase knowledge base. With this approach we follow the idea that ideal summaries of entities should cover commonly known facts about them.
3. We compare the UBES entity summarization system with the Google Knowledge Graph [Sin12] by using the established game data. We use standard rank correlation measures for measuring the similarity to the game data and draw conclusions for the quality of UBES as well as for the usability of the game data.

In the following subsections we first introduce the UBES approach in Section 3.2.2. After that, we report on our experiments in Section 3.2.3. The results of these experiments are discussed in Section 3.2.4. After covering the related work in Section 3.2.5 we conclude the work with Section 3.2.6.

3.2.2. Approach: UBES

The main idea of the UBES approach can be phrased as follows: predicate-object pairs (in this work also called “features”) that an entity shares with its k -nearest neighbors are more relevant than predicate-object pairs that are shared with entities that are not in the k -nearest neighbors range. Figure 3.11 provides an example for such a situation: the predicate-object pair `:director :Quentin_Tarantino` can be considered as important for the movie “Pulp Fiction” as its direct neighbors also have the same predicate-object pair.

³⁵We use these terms interchangeably in this thesis.

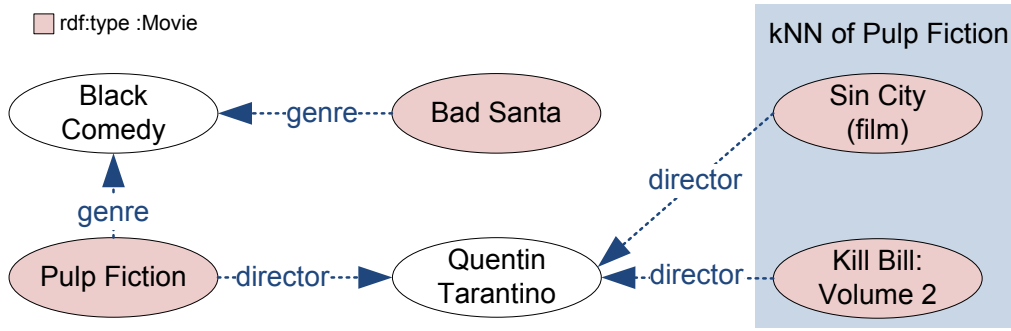


Figure 3.11.: Example: The feature `:director :Quentin_Tarantino` that the movie “Pulp Fiction” shares with its k -nearest neighbors is considered more important than the feature `:genre :Black_Comedy` that “Pulp Fiction” shares with a non-neighboring movie.

An important question that needs to be addressed is “how do we establish the neighborhood of an entity?” For this, we adopt a method from item-based collaborative filtering [SKKR01]: we use the row-vectors of the user-item matrix (exemplified in Table 3.11) that correspond to a specific `rdf:type` (e.g., `movie`) and compare the vector of the focus entity with the others via standard similarity measures—such as cosine similarity or the log-likelihood ratio [Dun93]. The top- k neighbors with highest similarity are then chosen as the entity’s neighborhood. In that respect, our method is also flexible and the entity neighborhood can be based on different data sources (e.g., co-mentions) and can also be established with different methods (such as machine learning models).

We provide a summary of a given entity e of type t in the set of all resources R ($e, t \in R, e \text{ rdf:type } t$). e has a “feature set” $FS(e)$ that includes all predicate-object pairs of e in the RDF knowledge base. The approach is based on usage data and includes six steps:

1. Generate the user-item matrix with $C = \{r | r \in R, r \text{ rdf:type } t, r \neq e\}$ (the subset of all resources that have the same RDF type as e).
2. Compute the similarity between e and all other resources $r \in C$ and identify a set $N_{k,e} \subseteq C$ of k -nearest neighbors of e .
3. For each feature $f \in FS(e)$ collect the resources $A_{e,f} \subseteq N_{k,e}$ in the entity’s neighborhood that share the same feature.
4. For each feature $f \in FS(e)$ collect the resources $B_{e,f} \subseteq C$ in the set of all resources of the same type that share the same feature.
5. The score of a feature $s_e(f), f \in FS(e)$ is computed in accordance to the following tf-idf-related ratio: $s_e(f) = |A_{e,f}| \cdot \log \frac{|C|}{|B_{e,f}|}$.
6. The features $f \in FS(e)$ are ordered descending according to their given score $s_e(f)$. Select the top- n ³⁶ most relevant features as a summary of e .

³⁶In this section, k is in use for k -nearest neighbors. Therefore, we refer to the size of the summary as top- n .

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

Table 3.12.: Example: The 20-nearest neighbors of the entity `fb:en.pulp_fiction` (the resource `fb:source.allocine.ca.film.53879` corresponds to the movie “Kill Bill: Volume 2” and occurs twice due to an error in a previous matching step).

Score	Neighbor
0.998	<code>fb:en.fight_club_1999</code>
0.998	<code>fb:en.seven_samurai</code>
0.998	<code>fb:en.the_silence_of_the_lambs</code>
0.998	<code>fb:en.memento</code>
0.998	<code>fb:en.the_usual_suspects</code>
0.998	<code>fb:en.twelve_monkeys</code>
0.998	<code>fb:en.trainspotting</code>
0.998	<code>fb:en.sin_city_2005</code>
0.998	<code>fb:en.fargo_1996</code>
0.997	<code>fb:en.american_beauty</code>
0.997	<code>fb:source.allocine.ca.film.53879</code>
0.997	<code>fb:source.allocine.ca.film.53879</code>
0.997	<code>fb:en.a_clockwork_orange_1971</code>
0.997	<code>fb:en.the_big_lebowski</code>
0.997	<code>fb:en.the_matrix</code>
0.997	<code>fb:en.forrest_gump</code>
0.997	<code>fb:en.full_metal_jacket</code>
0.997	<code>fb:en.terminator_2_judgment_day</code>
0.997	<code>fb:en.la_confidential</code>

The concept of a user-item matrix (Step 1) is a well-known principle in the field of recommender systems [SKKR01, AT05]. Each column of the matrix represents a single user and each row represents a single entity. The entries of the matrix are either one, if the user has consumed the resource, or zero/empty, if the user has not consumed a particular entity (which is the standard case). The column or row vectors can be used to compare users or entities with each other respectively.

Several similarity measures have been introduced for comparing column/row vectors (Step 2). Cosine similarity and Pearson correlation (comparing the vectors with regard to their angular distance) are the most common techniques [AT05]. In order to be more robust against sparsity, we apply the log-likelihood ratio score [Dun93] for computing entity similarities (i.e., the row vectors of the example in Table 3.11). In this context, the ratio combines four parameters: the number of users who consumed both entities, the number of users who consumed the first but not the second entity and vice versa, and the number of users who consumed none of the two entities. The similarity measure works with binary data, in particular Web usage data (consumed or not consumed). With the similarity scores it is possible to identify a set of k -nearest neighbors for a given entity. The 20-nearest neighbors of the entity `fb:en.pulp_fiction` are provided in Table 3.12.

Listing 3.4 exemplifies a SPARQL query that is used for the retrieval of common outgoing³⁷ features between the entity (`fb:en.pulp_fiction`) and its 20 nearest neighbors

³⁷We focus on outgoing features for reasons of clarity. Similar queries can be created for incoming features.

Listing 3.4: SPARQL query: retrieving outgoing predicate-object pairs shared with at least one of the 20-nearest neighbors of `fb:en.pulp_fiction`.

```
PREFIX fb: <http://rdf.freebase.com/ns/>
PREFIX ex: <http://example.com/>

SELECT (COUNT(?s) as ?c) ?p ?o WHERE {
  fb:en.pulp_fiction ?p ?o .
  en.pulp_fiction ex:knn20 ?s . # or ?s rdf:type fb:film.film .
  ?s ?p ?o .
  FILTER((?s != en.pulp_fiction) && (?p != ex:knn20)) .
} GROUP BY ?p ?o
```

(Step 3).³⁸ For identifying feature overlaps between the entity and all resources of the same type in the dataset (Step 4), the line

```
en.pulp_fiction ex:knn20 ?s .
```

needs to be replaced by

```
?s rdf:type fb:film.film .
```

For each of the two queries (Step 2 and Step 3), the predicate-object pairs are counted by occurrence (variable `?c`). The filter rule of the query removes predicate-object pairs that stem from the given entity and also removes relationships that origin from the nearest neighbors approach.

In the result set of Step 3, many features occur frequently—such as the following predicate-object pair:

```
fb:film.film.country fb:en.united_states
```

If the scoring would involve only counting, features like the above would be considered as highly relevant for many movies. However, as these features do not only occur frequently in the k -nearest neighbors set but also with all entities of the same type, they are less important (or individual). For reducing the importance, we adopt the classic tf-idf method (Step 5). In our case a “term” is a single feature and the term frequency is the frequency of the feature in the nearest neighbors set. The inverse document frequency is the logarithm of the ratio between the size of C (i.e., the number of resources of type t —in our case movies) and the number of movies within C that share a feature with e (see Step 4). Applying this measure, a feature that is common not only among the nearest neighbors but also among all entities of the same type receives a lower score. After this step, every feature that is shared with at least one of the k -nearest neighbors has an assigned score.

In the last step, we select the n most relevant predicate-object pairs in accordance to their score.

³⁸The 20-nearest-neighbors relationship is modeled with the predicate `ex:knn20`.

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

Listing 3.5: MQL query: retrieving the Freebase identifiers given an IMDb identifier.

```
{
  "id"= null,
  "imdb_id"="ttIMDb_ID",
  "type"= "/film/film"
}
```

3.2.3. Experiments

Our experiments were separated into different steps: 1) select datasets and a knowledge base and perform initial experiments; 2) identify a way to produce a ground truth and implement it; 3) select measures, baselines, systems, and their configuration as the evaluation setup; 4) perform measures and present results.

3.2.3.1. Dataset, Setup, and Initial Experiments

As a dataset, we combined the usage data of the HetRec2011 MovieLens2k dataset [CBK11] with Freebase [BEP⁺08]. The HetRec2011 MovieLens2k contains ratings of 2113 users for 10 197 movies and is an extension to the MovieLens10M dataset [HK15]. In addition, it provides additional metadata such as directors, actors, countries, and locations. Although this dataset already contains material with which we could perform tests without making use of Freebase, the search space for predicates and objects is still restricted. In particular, 26 predicates (such as genre, year, Spanish title, rotten tomatoes³⁹ rating, etc.) are opposed to more than 240 predicates that Freebase covers for movies. Also, the range in Freebase is much broader: for example, more than 380 different genres (`fb:film.film.genre`) are covered in contrast to 20 fixed genres contained in the HetRec2011 MovieLens2k dataset. The HetRec2011 MovieLens2k dataset includes IMDb⁴⁰ identifiers for each movie. This enabled us a straight forward linking to Freebase by running a query in Metaweb Query Language (MQL) at the Freebase query service⁴¹ (see Listing 3.5). With this query, we were able to match more than 10 000 of the 10 197 movies.⁴² For performance reasons, we retrieved the respective movie descriptions from Freebase and stored them in a local triplestore.

With the usage data, we computed the 20-nearest neighbors for each movie with Apache Mahout⁴³ and stored the results with an synthetic `knn` predicate in the triplestore together with the movie descriptions. Thus, in addition to the movie descriptions, the triplestore contained triples of the following form:

```
fb:en.pulp_fiction ex:knn20 fb:en.sin_city_2005 .
```

³⁹<http://www.rottentomatoes.com/>, retrieved 2016-07-01.

⁴⁰<http://www.imdb.com/>, retrieved 2016-07-01.

⁴¹See Section 2.1.5 for more information on Freebase.

⁴²Unmatched items are mostly TV series that do not match the pattern `"type"="film/film/"`.

⁴³Apache Mahout – <http://mahout.apache.org/>, retrieved 2016-07-06.

3.2. UBES: Leveraging Usage Data for Entity Summarization

Table 3.13.: Top-10 features: Pulp Fiction

Score	predicate	object
21.58	fb:film.film.directed_by	fb:en.quentin_tarantino
19.75	fb:film.film.genre	fb:en.crime_fiction
19.10	fb:user.robert.default_domain.rated_film.ew_rating	92
16.94	fb:film.film.rating	fb:en.r_usa
16.38	fb:film.film.featured_film_locations	fb:en.los_angeles
14.12	fb:film.film.written_by	fb:en.quentin_tarantino
13.72	fb:film.film.film_collections	fb:en.afis_100_years_100_movies
13.48	fb:film.film.edited_by	fb:en.sally_menke
13.31	fb:film.film.film_production_design_by	fb:en.david_wasco
12.39	fb:film.film.produced_by	fb:en.lawrence_bender

Table 3.14.: Top-10 features: Beauty and the Beast

Score	predicate	object
39.56	fb:film.film.genre	fb:en.fantasy
29.40	fb:film.film.rating	fb:en.g_usa
19.23	fb:film.film.production_companies	fb:en.the_walt_disney_company
16.89	fb:film.film.music	fb:en.howard_ashman
13.31	fb:film.film.music	fb:en.alan_menken
12.86	fb:film.film.subjects	fb:en.fairy_tale
9.14	fb:film.film.film_casting_director	fb:en.albert_tavares
8.04	fb:film.film.written_by	fb:en.linda_woolverton
7.75	fb:film.film.produced_by	fb:en.don_hahn
7.30	fb:film.film.genre	fb:en.costume_drama

This triple means that `fb:en.sin_city_2005` is one of the 20-nearest neighbors of `fb:en.pulp_fiction`. Using SPARQL queries (like in Listing 3.4) we were able to retrieve common predicates between single movies and neighboring/non-neighboring movies. In our experiments we tried multiple configurations of k (with respect to the k -nearest neighbors approach) in which $k = 20$ proved as the most stable one. An example neighborhood for the entity `fb:en.pulp_fiction` is given in Table 3.12. A particularity about this neighborhood is that the movie

`fb:source.allocine.ca.film.53879`

(that corresponds to “Kill Bill: Volume 2”) occurs twice in the list. The reason for this lies in the previous matching steps that were performed for creating the HetRec2011 MovieLens2k dataset—it contains several duplicates with different identifiers.

With the neighborhood and the according measures on the dataset, we were able to extract the 10 most important features for each entity. Each of the tables 3.13, 3.14, 3.15, 3.16 provide an example for a top-10 movie entity summary produced with UBES. Most of the presented examples have `fb:film.film.genre` as one of their best-scoring predicates. In that regard, the fine granularity of Freebase poses a strong advantage: genres

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

Table 3.15.: Top-10 features: The Naked Gun - From the Files of Police Squad!

Score	predicate	object
27.77	fb:film.film.written_by	fb:en.jim_abrahams
26.00	fb:film.film.written_by	fb:en.pat_proft
22.59	fb:film.film.written_by	fb:en.jerry_zucker
22.04	fb:film.film.written_by	fb:en.david_zucker
18.92	fb:film.film.music	fb:en.ira_newborn
18.44	fb:media_common. netflix_title.netflix_genres	fb:en.comedy
16.89	fb:film.film.film_series	fb:m.0dl08h
16.38	fb:film.film.featured_film_locations	fb:en.los_angeles
16.12	fb:film.film.genre	fb:m.02kdv5l
15.97	fb:film.film.genre	fb:en.parody

Table 3.16.: Top-10 features: Bridget Jones’s Diary

Score	predicate	object
29.67	fb:film.film.genre	fb:en.romantic_comedy
29.39	fb:film.film.written_by	fb:en.richard_curtis
19.40	fb:film.film.country	fb:en.united_kingdom
18.43	fb:film.film.film_casting_director	fb:en.michelle_guish
16.75	fb:film.film.produced_by	fb:en.eric_fellner
16.50	fb:film.film.produced_by	fb:en.tim_bevan
13.05	fb:user.robert.default_ domain.rated_film.ew_rating	69
12.79	fb:film.film.film_format	fb:en.super_35_mm_film
12.51	fb:film.film.production_companies	fb:en.universal_studios
9.14	fb:film.film.story_by	fb:en.helen_fielding

such as “costume drama”, “crime fiction” or “parody” are missing in the HetRec2011 MovieLens2k dataset and many other datasets. It is interesting that the predicate

`fb:film.film.written_by`

has an impact for all of the presented movies. One of the main aspects of entity summarization systems is the coverage of particularities (i.e., what is special about this particular entity). In that regard, in the results, the movie “Bridget Jones’s Diary” shares with its neighbors that they were filmed in the United Kingdom while Walt Disney as the production company is surely important for the movie “Beauty and the Beast”. It is also worth to mention that, according to our results, the movie “Pulp Fiction” is strongly influenced by its director Quentin Tarantino.

To this point, the presented summaries do not cover n-ary relations⁴⁴ as they occur in Freebase. In theory, it would be possible to deal with n-ary relations via according queries but these involve multiple joins and, therefore, were often too hard to process for the used triplestores. For our evaluation we circumvented this issue by introducing manually-created reasoning axioms that enabled direct relations for the specific domain of movies (see Section 3.2.3.2.1). Another issue was the problem of data quality and the constant

⁴⁴See Section 2.1.1.1 for an introduction to n-ary relations.

evolution of data in Freebase (that could be edited by everyone). As visible in Table 3.13 and Table 3.16, this introduced some noise and, therefore, our results included predicates like `fb:user.robert.default_domain.rated_film.ew_rating`.

Similar to the high requirements with respect to the quality of the knowledge base, also the availability and correct processing/interpretation of usage data is a central aspect of UBES. For this matter, we kindly refer the reader to [LMN11].

3.2.3.2. Towards a Ground Truth

A ground truth was necessary to evaluate the UBES approach in a neutral way. In general, it is difficult to establish such a ground truth—so we introduced a specific setting that makes the task feasible:

- Focus on the movie domain.
- Use popular movies to facilitate the task for the users.
- Select Freebase as a single data source.
- Resolve n-ary relations.

Even in this setting, it was difficult to generate summaries of a significant amount of movies under the consideration that many of them are described with multiple hundreds of features. In order to address this issue, we introduced a game-based approach to generate a ground truth. The main idea was to use a quiz game in order to determine which facts are commonly known about entities, which facts are known by some users but not by all, and which facts are commonly unknown. We assumed that, with this information, it is possible to establish a ground truth for entity summaries.

3.2.3.2.1. Data sources We focused on movie entities from Freebase in our evaluation. Freebase contains a large amount of openly available data and—in contrast to DBpedia and the Linked Movie Database (LinkedMDB)⁴⁵—very detailed and well curated information about movies. Large parts of this dataset were also used by Google for its summaries [Sin12]. For the evaluation, we randomly selected 60 movies of the IMDb Top 250 movies⁴⁶ (as of April/May 2012) and derived the Freebase identifiers by querying Freebase for the predicate `imdb_id`⁴⁷ (see Listing 3.5). For reasons of efficiency we restricted the number of movies to 60, because otherwise it would have been difficult to achieve the necessary number of game participants for sufficient coverage. We downloaded RDF descriptions of the movies (that were available via content negotiation) and stored them in an OWLIM⁴⁸ triplestore with OWL2 RL [MGH⁺09] reasoning enabled. This

⁴⁵LinkedMDB – <http://www.linkedmdb.org/>, retrieved 2016-07-03.

⁴⁶IMDB Top 250 – <http://www.imdb.com/chart/top>, retrieved 2016-07-03.

⁴⁷List of selected movies – http://yovisto.com/labs/iswc2012/selected_movies_freebase.txt, retrieved 2016-07-03.

⁴⁸OWLIM (now Ontotext GraphDB™) – <http://www.ontotext.com/owlim>, retrieved 2016-07-03.

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

Listing 3.6: Property chain axiom for creating direct “hasActor” relationships via OWL2 RL reasoning (Turtle syntax, the round brackets define an RDF collection [FPSH14]).

```
<http://example.com/hasActor>
<http://www.w3.org/2002/07/owl#propertyChainAxiom> (
  <http://rdf.freebase.com/ns/film.film.starring>
  <http://rdf.freebase.com/ns/film.performance.actor>
) .
```

enabled us to resolve the n-ary relations (such as movies/actors/roles) of Freebase. We were able to transform these to normal triples with property-chain reasoning rules. An example such an axiom is provided in Listing 3.6.⁴⁹ We created such direct links for actors, achieved awards, budgets, and durations.

3.2.3.2.2. *WhoKnows?Movies!* – Concept and Realization We adopted the GWAP *WhoKnows?*, an online quiz game in the style of “Who Wants to Be a Millionaire?”. It was originally developed for DBpedia data quality checking and curation [WLKS11, KHS12]. Our version *WhoKnows?Movies!*⁵⁰ was rebuilt towards the Freebase movie scenario and—in contrast to quality and curation—the goal was to obtain a reference for the relevance of facts. However, the principle of the GWAP was left unchanged: it presents multiple choice questions to the player that were generated from the respective facts about a fixed selection of entities. In our case we used the dataset as described in Section 3.2.3.2.1. The players could score points by answering single questions correctly within a limited period of time and lose points and lives when providing none (in case of a time out) or incorrect answers.

As an example, Figure 3.12a shows the question “John Travolta is the actor of ...?” and Figure 3.12b exemplifies the expected answer “Pulp Fiction”. The question stems from the triple

```
fb:en.pulp_fiction ex:hasActor fb:en.john_travolta .
```

and was composed reversing the triples’ order, for example “object is the predicate of: subject1, subject2, subject3”. The remaining options were selected from entities that apply the same predicate at least once but are not linked to the object in question (see Table 3.17). In this way we assured that only wrong answers were presented as alternative choices. The questions were constructed in two variants: *One-To-One* (see Figure 3.12a and Figure 3.12b) where exactly one answer was correct and *One-To-N* (see Figure 3.12c and Figure 3.12d) where one or more answers were correct.

When a player answered a question correctly they scored points and increased the level. With an incorrect answer the players were penalized by losing points and a life. The earned score depended on the correctness of the answer and the time needed for providing the answer. With higher levels, the number of options was raised and, thus, correct answers were becoming increasingly harder to guess. When submitting an answer, the user received

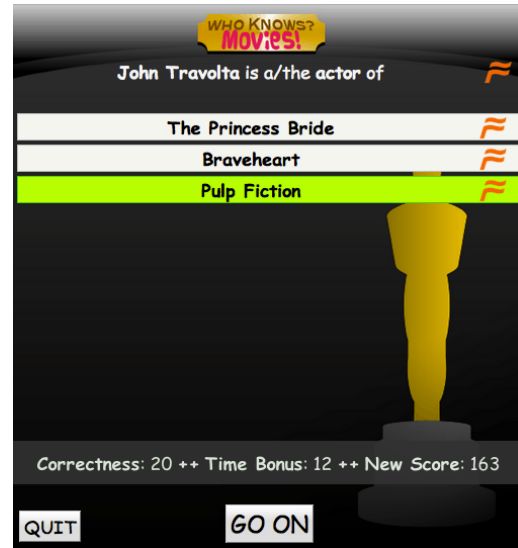
⁴⁹All used property-chain reasoning rules are provided in Appendix A.1.

⁵⁰*WhoKnows?Movies!* – <http://141.89.225.43/whoknowsmovies/>, retrieved 2016-07-06.

3.2. UBES: Leveraging Usage Data for Entity Summarization



(a) Example: *One-To-One* question generated from `:Pulp_Fiction :hasActor :John_Travolta`.



(b) Example: Correct answer for the *One-To-One* question on the left side (Figure 3.12a).



(c) Example: *One-To-N* question generated from `:Star_Wars_IV :director :Dianne_Crittenden`.



(d) Example: Incorrect answer for the *One-To-N* question on the left side (Figure 3.12c).

Figure 3.12.: Screenshots of *WhoKnows?Movies!* exemplifying *One-To-One* and *One-To-N* questions and their correct or incorrect answers (for better readability, the IRIs were changed).

immediate feedback about the correctness of their answer in the result panel where their choices were shown once again and the correct answer was highlighted (see Figure 3.12b and Figure 3.12d). The provided answers were logged for later traceability and the triple's statistics were updated accordingly. The game finished when the player lost all of their five lives.

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

Table 3.17.: Example: Triples used to create a question and according answer options (for better presentation, we use the labels of the resources).

Subject	Predicate	Object
Pulp Fiction	actor	John Travolta
	actor	Uma Thurman
	actor	...
Braveheart	actor	Mel Gibson
	actor	Sophie Marceau
	actor	...
The Princess Bride	actor	Robin Wright
	actor	Annie Dyson
	actor	...

For the game configuration the applied dataset included the 60 described movies from Freebase. We selected 25 different predicates (see Appendix A.1) that were relevant for movies, resulting in 2829 distinct triples.⁵¹ For each triple a set of false answers was preprocessed and stored to a database. When generating a question for a specific triple, a number of false subjects was randomly selected from this set.

3.2.3.2.3. Results: Game Data As of September 2012, in total, the quiz was played 690 times by 217 players. A majority of 135 players played only once. All 2829 triples were played at least once and 2314 triples were played at least three times. In total 8308 questions were replied of which 4716 were answered correctly. For each of the 60 movies, this enabled a direct ranking of the facts by the average correctness of provided answers to quiz questions about them. An example for such a ranking is provided in Table 3.18 for the movie “Pulp Fiction”. In this case, the set of top-ranked predicate-object pairs overlaps with summaries of “Pulp Fiction” (as often presented by entity summarization systems).

In Table 3.19 we provide an aggregate with respect to the different predicates of the triples. With respect to that, it was generally more easy for the players to provide a correct answer with the predicates “prequel”, “film series”, or “sequel”. This is due to the fact, that these three predicates often occurred with facts, where the object and the subject are similar (e.g., “Star Wars V is the sequel of...? – Star Wars IV”).

3.2.3.3. Evaluation Setup

We compared the rankings of UBES with the rankings of the dataset established with the game *WhoKnows?Movies!*. We introduced summaries of Google Knowledge Graph (GKG) and random summaries (RANDOM) as competing baselines and, accordingly, the setup of UBES, GKG, and RANDOM.

⁵¹Initially we also included role names but these were dropped later-on as they added a large amount of mostly unimportant information for the price of a strongly negative impact on the player’s experience (with many unknown role names asked in a game session the users stopped playing).

3.2. UBES: Leveraging Usage Data for Entity Summarization

Table 3.18.: Example: Facts about Pulp Fiction that were played at least three times sorted by average correctness.

Predicate	Object	Correctness
fb:film.film.rating	fb:en.r_usa	100.00%
ex:hasActor	fb:en.amanda_plummer	100.00%
fb:film.film.directed_by	fb:en.quentin_tarantino	100.00%
ex:hasActor	fb:en.samuel_l_jackson	100.00%
fb:film.film.written_by	fb:en.quentin_tarantino	100.00%
fb:film.film.film_festivals	fb:en.1994_cannes_film_festival	100.00%
ex:hasActor	fb:en.quentin_tarantino	100.00%
ex:hasActor	fb:en.john_travolta	100.00%
ex:hasActor	fb:en.ani_sava	100.00%
ex:hasActor	fb:en.don_blakely	75.00%
fb:film.film.featured_film_locations	fb:en.los_angeles	66.67%
fb:film.film.edited_by	fb:en.sally_menke	66.67%
fb:film.film.genre	fb:en.crime_fiction	66.67%
ex:hasActor	fb:m.0bhh4y8	66.67%
ex:hasActor	fb:en.stephen_hibbert	66.67%
ex:hasActor	fb:en.alexis_arquette	66.67%
ex:hasActor	fb:en.harvey_keitel	66.67%
ex:hasActor	fb:en.susan_griffiths	66.67%
ex:hasActor	fb:en.rosanna_arquette	66.67%
fb:film.film.film_casting_director	fb:en.ronnie_yeskel	50.00%
ex:hasActor	fb:en.lawrence_bender	33.33%
ex:hasActor	fb:en.steve_buscemi	33.33%
ex:hasActor	fb:en.rich_turner	33.33%
ex:hasActor	fb:en.burr_steers	33.33%
ex:hasActor	fb:en.emil_sitka	33.33%
ex:hasActor	fb:en.dick_miller	33.33%
ex:hasActor	fb:en.brenda_hillhouse	33.33%
ex:hasActor	fb:en.devon_richardson	33.33%
ex:hasActor	fb:en.tim_roth	33.33%
ex:hasActor	fb:en.julia_sweeney	33.33%
ex:hasActor	fb:en.maria_de_medeiros	25.00%
ex:hasActor	fb:en.jerome_patrick_hoban	20.00%
fb:film.film.film_casting_director	fb:en.gary_m_zuckerbrod	0.00%
fb:film.film.cinematography	fb:en.andrzej_sekula	0.00%
ex:hasActor	fb:en.paul_calderon	0.00%
ex:hasActor	fb:en.bronagh_gallagher	0.00%

The configuration and setup of the respective systems/baselines was as follows:

UBES We computed the 20-nearest neighbors for each of the 60 selected movies with the log-likelihood ratio [Dun93]. After applying the main steps of UBES, we retrieved a ranked output for each of the 60 movies. The output then was filtered with the white list⁵² of 25 predicates in order to fit with the predicates covered by the game.

GKG The 60 movie summaries by Google were processed in a semi-automatic way to fit with the Freebase IRIs. The first step was to retrieve the summaries of all 60 movies and storing the according HTML files. While the Freebase IRIs for predicates such as “director” had to be entered manually, most objects could be linked to Freebase automatically.⁵³ Google’s ordering of the five main facts was interpreted in a top to

⁵²See Appendix A.1.

⁵³Email by Andreas Thalhammer to the mailing list of the W3C Semantic Web Interest Group – <http://lists.w3.org/Archives/Public/semantic-web/2012Jun/0028.html>, retrieved 2016-07-06.

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

Table 3.19.: *WhoKnows?Movies!* predicates sorted by average answer correctness (all movies).

Predicate	Correct
fb:film.film.prequel	95.39%
fb:film.film.film_series	95.16%
fb:film.film.sequel	85.33%
fb:base.parody.parodied_subject.parodies	76.47%
fb:media_common.adaptation.adapted_from	74.32%
fb:film.film.subjects	73.91%
fb:film.film.genre	65.14%
fb:film.film.initial_release_date	65.14%
fb:film.film.directed_by	63.51%
fb:film.film.rating	61.61%
fb:film.film.written_by	61.61%
fb:film.film.featured_song	60.00%
fb:film.film.featured_film_locations	60.00%
fb:film.film.production_companies	56.10%
ex:hasRunningTime	54.52%
fb:film.film.music	54.11%
ex:hasAward	53.41%
ex:hasActor	52.86%
fb:film.film.story_by	51.18%
fb:film.film.edited_by	50.00%
fb:fictional_universe.work_of_fiction.events	50.00%
fb:film.film.cinematography	44.20%
ex:hasBudget	42.78%
fb:film.film.film_festivals	42.27%
fb:film.film.film_casting_director	41.32%

bottom order and the ranking of cast members was interpreted with declining rank from left to right.

RANDOM The random summaries were generated in accordance to the white list⁵⁴. In order to prevent positive/negative outliers from having an impact, we created 100 random rankings and used the average scores of these for our comparison.

For measuring the similarity between the ranked output of the systems/baselines and the game data, we applied Kendall’s τ rank correlation coefficient [Ken38]. In that regard, we evaluated the respective systems/baselines with respect to two main aspects:

Predicate Ranking To evaluate the ranking of predicates for a single movie, we determined the ranking of predicates according to the correct answer ratio. The GKG movie representation listed general facts in an ordered manner, whereas the cast of the movie was displayed separately. Accordingly, only the remaining 24 predicates were used for this evaluation. Predicates that did not occur in the systems’ results were jointly put to the bottom position.

Actor Ranking The three systems/baselines output rankings of the objects for the members of the cast that are connected via the `ex:hasActor` predicate. We used the rankings of the actors for each movie in order to determine the performance of the individual approaches relative to the game data.

⁵⁴See Appendix A.1.

Table 3.20.: Performance of the predicate ranking.

	Kendall's τ_{avg}	Kendall's τ_{min}	Kendall's τ_{max}
UBES	0.045	-0.505 (The Sixth Sense)	0.477 (Reservoir Dogs)
GKG	0.027	-0.417 (The Big Lebowski)	0.480 (Reservoir Dogs)
RANDOM	0.031	-0.094 (American Beauty)	0.276 (Monsters Inc)

Table 3.21.: Performance of the actor ranking.

	Kendall's τ_{avg}	Kendall's τ_{min}	Kendall's τ_{max}
UBES	0.121	-0.405 (The Princess Bride)	0.602 (Indiana Jones – Last Crusade)
GKG	0.124	-0.479 (The Princess Bride)	0.744 (The Matrix)
RANDOM	0.013	-0.069 (Fargo)	0.094 (Good Will Hunting)

3.2.3.4. Evaluation Results

In the following, we present the results of the evaluation.

Predicate Ranking For each movie Kendall's τ was determined over the set of its predicates. Table 3.20 shows the average, minimum, and maximum values of Kendall's τ . It can be seen, that the three systems/baselines performed equal in average. For each system/baseline, for about half of the movies the correlation was negative which means that the orderings were partly reverse compared to the ordering of the derived dataset. In general, none of the UBES and GKG rankings differed significantly from a random ranking.

Actor Ranking In Table 3.21, we present the average, minimum, and maximum resulting Kendall's τ correlation scores for the ranking of actors. The results for the actor ranking were fairly equal for both systems (GKG and UBES) in the average case. The average Kendall τ value respectively differed from random scores. We estimated that the difference to the random ranking was significant ($p < 0.05$) for both systems. It has to be noted that, in some cases, the UBES approach provided none or very few proposals due to the required actor overlap with the 20-nearest neighbors.

3.2.4. Discussion

In our results we found evidence, that there exists a correlation between quiz-game-based rankings (i.e., what people generally know about an entity) and facts that are typically shown in entity summarization (i.e., guessing what people generally would like to know). With reference to the game data, the UBES ranking performed similarly well as the GKG ranking (which is based on more background data—the Google query logs, see Section 2.2.1.1 for more details). However, the weak results of GKG and UBES suggest further investigation. In particular, the following points are worthy to be tested in future studies:

- It needs to be investigated whether a negative influence is introduced with the aggregation on objects (with respect to the ranking of the predicates, see Table 3.20).

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

- Although providing indications, the study did not entirely clarify whether and in which way the established data from the *WhoKnows?Movies!* game is suitable for evaluating entity summarization systems.
- It is also possible that the tested entity summarization systems both just did not perform very well (with respect to the researched context—stated by the movies).
- The influence of the sparsity of the game data needs to be investigated (many of the facts were played only three times or less).

A likely case could be a mix of some of the above points. However, the exploration of each of these points forms a new contribution. The results of the actor ranking provide an indication that the relative importance of predicate-object pairs can be captured by the statistics established through the *WhoKnows?Movies!* game. The actor rankings of UBES and GKG are significantly different from RANDOM and correlate stronger with the game data. Eventually, given the available data, it was not possible to determine whether UBES or GKG performed better.

The data established with *WhoKnows?Movies!*, the Google summaries, the respective UBES and GKG rankings, and further information are available online.⁵⁵

3.2.5. Related Work

This section relates to different topics in the fields of Web usage mining in combination with Linked Data, frequency and co-occurrence analyses over RDF data, and the use of GWAPs in context of the Semantic Web.

The idea of combining areas of Semantic Web and Web usage mining was initially presented by the works of Berendt et al. [BHS02] and Oberle et al. [OBHG03]. The authors present the idea of (not explicitly stated/provided) relations that can be inferred by analyzing consumption patterns from usage data [BHS02] and personalization in accordance to previously consumed knowledge [OBHG03]. Later works by Möller et al. [MHC⁺10] and Kirchberg et al. [KKL11] elaborate on measures for usage data analysis with respect to machine-readable data in regard to different portals offering Linked Data. Their analyses focus on Linked Data resources as well as SPARQL query logs. In [FMG11], Fortuna et al. combine usage data with semantically enriched content data in an efficient model that is demonstrated to be suitable for machine learning in order to gain deep insights on user interest and content. To the best of our knowledge, UBES is the first approach that utilizes patterns mined from usage data for ranking semantic information, in particular predicate-object pairs with respect to a selected entity.

Oren et al. introduce “Simple Algorithms for Predicate Suggestions Using Similarity and Co-occurrence” [OGD07]. The authors present two approaches: 1) predicate suggestions via the similarity of resources that use them; 2) predicate suggestions via their co-occurrence. The first approach relates strongly to UBES as predicates are ranked indirectly via the previously determined similarity of resources (that divides all resources into

⁵⁵Data of the experiments – <http://yovisto.com/labs/iswc2012/>, retrieved 2016-07-12.

the sets: similar; not similar). The ranking involves a ratio that is similar to tf-idf. The second approach follows a co-occurrence analysis based on association rule mining [AIS93]. Our UBES approach relates mostly to the first approach while we do not only consider predicates, but full features (predicate-object pairs). Furthermore, we compute similarity with additional background knowledge, while resource similarity is established via the use of similar predicates in [OGD07]. Similar to [OGD07], Paulheim and Bizer [PB13] address the topic of type inference via statistical models over RDF data via analyzing the usage of RDF predicates in combination to existing classes. Di Noia et al. [DNMO⁺12] use the tf-idf measure for movie recommendation: the similarities of the movies are estimated with a vector space model that includes tf-idf weights for predicate-object pairs. In their model, the term frequency is binary (either a movie has a feature, or it has not). In this way, the particularity of a predicate-object pair can be estimated (a very common predicate-object pair has a low weight in the feature vector). With UBES, we also apply tf-idf weighting on predicate-object pairs. However, in contrast to [DNMO⁺12], our aim is to identify their importance for an entity. In our k -nearest-neighbors model we identified “the number of nearest neighbors that share a feature” analogously to term frequency.

The idea of using GWAPs in the context of the Semantic Web and according examples (from their earlier work) were comprehensively presented by Siorpaes and Hepp [SH08]. Most related to our work on *WhoKnows?Movies!* is the “RISQ!” game [WKOS11] by Wolf et al.—an early approach that focuses on ranking facts with respect to a specific entity. A similar approach also was followed by the works of Hees et al. with their games “BetterRelations” [HRBB⁺12] and the “Knowledge Test Game” [HKB⁺13]. We follow the design of *WhoKnows?* [WLKS11] [KHS12] and—in contrast to directly asking for “what is more important” [HRBB⁺12] or “what do you associate with ...” [HKB⁺13]—identify commonly known/unknown facts about entities in a more subtle way. In a later effort to obtain evaluation data for the task of “entity-centric fact ranking” (i.e., entity summarization) Bobić et al. implemented a Web interface for a crowd sourcing effort [BWS16]. Similar to *WhoKnows?Movies!*, the obtained reference data was published online.

3.2.6. Conclusions

We presented UBES, an entity summarization approach based on usage data, and the GWAP *WhoKnows?Movies!*, a quiz game designed with the intention to establish a reference dataset for entity summarization. UBES works with the k -nearest neighbors method on usage data and utilizes the tf-idf approach for identifying important predicate-object pairs. *WhoKnows?Movies!* introduces a quiz game with which we could distinguish commonly known from commonly unknown facts. We assumed that this information could correlate with the information requirements in the field of entity summarization. We tried to verify this by using the game data for comparing UBES to Google’s entity summarization method and to a random baseline in the film/movie domain.

3. Entity Summarization: Analyzing Explicit and Implicit Relations Between Entities

Our conclusions are as follows:

1. Consumption patterns derived with Web usage mining techniques [LMN11] can provide important information about the particularities of entities. We introduced an efficient and effective approach to leverage these patterns for entity summarization.
2. We introduced a quiz game that enables us to identify the level of knowledge that users generally have about single facts of entities. The positive correlation to UBES and GKG on the actor ranking task leads us to conclude that a GWAP is at least partially suitable for generating a reference dataset for entity summarization. In order to verify this, further experiments are necessary.
3. With respect to the ranking of actors, our experiments showed that UBES and GKG produce better rank correlations to the game data than random. The performance levels of UBES and GKG are very similar in the used test bed. Further experiments are necessary in order to provide a clear distinction.

Following these contributions there are still open points that need further attention. Thus, for future work on this topic, we suggest the following problems:

- Item similarities can only be established if items are consumed. This is especially difficult to achieve for new entities (i.e., the new item problem of collaborative recommender systems [AT05]). This problem could be addressed by blending our results with other entity summarization systems.
- For literal values we only consider exact matches. This is difficult to achieve as literal values often describe individual facts of entities (such as the name, birth date, price, etc.). In general, it is hard to include very specific predicates (e.g., one-to-one relations such as spouse). In order to mitigate this effect, we plan to blend the results of UBES with other approaches such as LinkSUM (see Section 3.1).
- With respect to *WhoKnows?Movies!*, we plan to extend the game with respect to images and other media in order to help players to identify persons/music/etc. related to a movie, or other composed information artifacts. The main idea here is to activate subconscious knowledge.
- Since the collection of the game data and the execution of the experiments, the *WhoKnows?Movies!* game has been online for more than four years. It would be interesting to repeat the presented experiments with updated game statistics.

4. Interfacing Entity Summarization

We address Research Question 2 in this chapter:

Is there a minimum set of re-occurring/common features of entity summarization systems that allows us to provide a generic API?

The contribution—which addresses this question—states an interaction mechanism for sharing and exchanging entity summaries (see Figure 4.1).

This chapter is based on methods, descriptions, figures, experiments, and according results that were previously published by the author and his collaborators in [TR14], [TS15], and [TR16a].

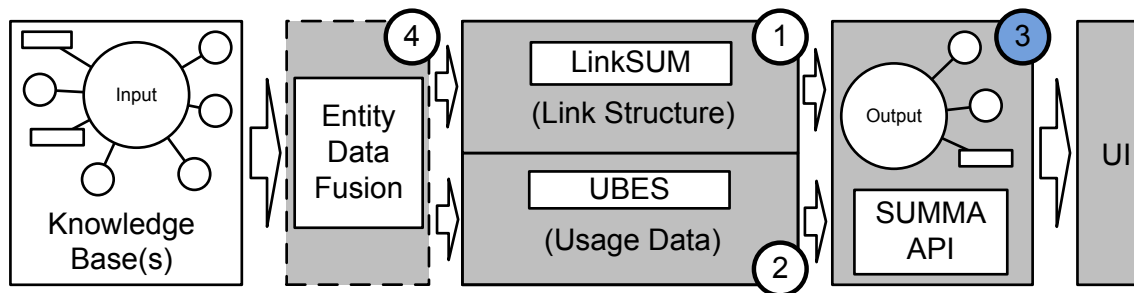


Figure 4.1.: Overview of the contributions of this thesis. In this part, we focus on the SUMMA API, an interaction mechanism for sharing and exchanging entity summaries.

4.1. Introduction

The amount of commercial systems that offer entity summaries are on the rise (e.g., [Pal15, Sin12]). Due to their proprietary nature, these systems tightly couple their user interface and back end in accordance to their specific requirements. Also the data sources, from which these commercial summaries are derived, are usually not publicly available. As a consequence, it becomes hard to exchange, evaluate, and compare the output of summarization systems in an objective manner. In order to facilitate accessibility of entity summaries it is necessary to identify the principal properties of entity summarization systems, create a corresponding data model, and to adhere to the best practices of Web APIs.

To enable clients to easily consume the summaries of entities from different summarization services we propose SUMMA, a uniform lightweight interface design based on a request/response vocabulary and the Representational State Transfer (REST) interaction paradigm.

4. Interfacing Entity Summarization

The approach enables to combine a diverse selection of summarization approaches on a single Web site and to switch from one service to another even during user navigation. The proposed API aligns with the Linked Data interaction model. Our approach treats the summarization approach¹ itself as a black box while preserving the possibility to define the required parameters of an entity summarization system in a uniform manner. Thus clients can easily substitute or combine the employed entity summarization system in a plug-and-play fashion.

With the introduction of the SUMMA API, we aim to address the following challenges:

1. The quantitative evaluation of entity summarization is difficult as the existing systems are typically strongly tied to their user interfaces. How can we model entity summaries in order to enable direct quantitative comparison without the overhead of mapping string literals back to the original data source and format (e.g., matching the string “prize” to `dbo:award`)?
2. For qualitative entity summarization—in particular for direct comparison—the interfaces of entity summarization systems have to be adapted and unified in a way so that graphical and style elements do not influence the users’ decisions. How can we present the output of multiple entity summarization systems through a uniform user interface for qualitative comparison?
3. A/B testing needs a clear separation of the style/presentation and content. What is the best separation of these concepts and how do we achieve it in the best flexible way?

Therefore, the contributions of the SUMMA API are as follows:

1. SUMMA enables consumers to retrieve summaries of entities in their most pure form; that is a ranked list of RDF statements. Thus, for quantitative evaluation, reverse engineering tasks such as disambiguating strings to IRIs are not needed for automatic comparison of different approaches.
2. In qualitative evaluation settings for entity summarization, multiple systems are often placed next to each other and users are asked to choose one (or more). To support this, a SUMMA client can present summaries of multiple different summarization systems in a uniform way. In this way it can be ensured that style elements (such as pictures, borders, colors, etc.) do not play a significant role in the users’ decision making process.
3. Evaluation with A/B testing is commonly applied in industry settings. SUMMA enables to change the entity summarization system while the user interface stays the same (and vice versa). By tracking the interaction with each variant it is possible to compare the effects of changing/modifying the entity summarization approach.

In an empirical evaluation, we measure the overlap of our established requirements with the features of real-world systems. This study includes interfaces of well known search engines like Bing, Google, and Yahoo as well as entity presentations of well-known news

¹Note: we present two approaches for entity summarization in Chapter 3.

portals. For our approach, we also provide an open-source reference implementation and deployment. The source code of the reference implementation as well as different deployments are available online.

The remainder of this chapter is organized as follows: In Section 4.2 we present a requirement analysis for a uniform entity summarization API as well as the API itself. In Section 4.3 we present experiments that consist of an evaluation (Section 4.3.1) and a reference implementation (Section 4.3.2). We discuss the approach in Section 4.4. In Section 4.5 we analyze the most related approaches and outline how SUMMA differs from them. Section 4.6 concludes the chapter and provides an overview of open points.

4.2. Approach: SUMMA API

In its most basic form, a summary of an entity can be produced by two given parameters:

IRI An IRI that identifies the entity.

k A number k that defines an upper limit of how many facts about the entity should be presented.

While it is obvious that there is a need for an unambiguous reference to the entity, it could be argued that a summary could also be specified by a given compression level. For example, we could specify that 30% of all facts about the given entity should be contained in the summary. In this respect, we would like to point out that concise presentations (for which we are aiming) are better declared with an upper limit rather than a given percentage. This is due to the fact that knowledge bases commonly cover well documented entities as well as a long tail of sparsely documented ones: in this respect, 30% could mean 20 000 facts for some entities and only three for others.

When defining a uniform interface for entity summarization, various specifics that are inherent to the definition of RDF itself have to be considered as well. This ranges from the possibility to have multiple labels for vocabulary or data items to the more complex summaries that consider n-ary relations² or enable full property chains. Next to these features, other requirements include the grouping of statements and the restriction to a predefined set of selected predicates. In the following we present an overview of all further requirements of the API:

Languages In many knowledge bases, labels in different languages for resources and predicates are commonly available. In order to avoid multiple requests or queries to different knowledge sources (e.g., in order to retrieve labels for predicates of the RDF or RDFS vocabularies) we find it necessary to include labels of one or more languages in the output of the summary.

²See Section 2.1.1.1 for an introduction to n-ary relations.

4. Interfacing Entity Summarization

Multi-hop Search Space It might be necessary (e.g., with n-ary relations or reification) or interesting to include statements in the summary that do not directly involve the targeted entity but are connected through one or more hops. For example, a “max hop” parameter of one (default) only considers statements where the entity is either in the subject or object role, while a “max hop” of two could cover facts that are still about the entity but are modeled via a n-ary relation. Further hops are possible.

Property Restriction A summary can be targeted to a predefined set of selected predicates. An example would be to restrict the summary of a movie to `{dbo:starring}` or `{dbo:starring, dbo:director}`. This feature is very useful if the interface has reserved space for specific features such as a map presenting geolocations or pictures. These features can be retrieved in a separate API request.

Statement Groups Rather than ranking statements only individually, the system could form groups or clusters of statements and, if applicable, provide names for these groups. An example for such a group could be biographic information about a person, such as birth place, birth date, alma mater, etc.

These features and their compositions enable very specific views on entities although they are still abstract enough to be applicable to any knowledge base, be it encyclopedic or proprietary. In general, also the following considerations have to be taken into account:

Resources/Literals Linking to other resources (i.e., IRI-identified entities) supports exploration aspects while textual information (represented as literals) satisfies more the information need about the specific entity. For visualization purposes any resource IRI included in a summary has to be accompanied by a literal description which enables a user-friendly rendering of the resource. Clients consuming the summary can therefore ignore the resource IRIs and only use literals for presentation.

Outgoing/Incoming Links For any unidirectional relation `:x :link :y` a second relation can be established in the way `:y :link_by :x`. In many cases displaying such a relation in a summary of `:y` makes sense as it covers information about it. Knowledge bases such as DBpedia, Freebase, and Wikidata enable to retrieve incoming links from other resources of the respective knowledge base with queries. For Linked Data in general, many incoming links can be retrieved with crawls as provided for example by the Billion Triples Challenge (BTC) dataset [KH14].

Our approach consists of two main components with a strong interplay:

- The **SUMMA Vocabulary** can be used to frame summary requests, which can be submitted to a summarization engine. Servers can interpret the given parameters in the request and produce result sets with the vocabulary that are in accordance to the provided parameters.
- The description of the **RESTful Web Service** provides a clear guideline for the interplay between summary consumers and producers.

In the following, we first introduce the *SUMMA Vocabulary* and thereafter the *RESTful Web Service* interaction guideline.

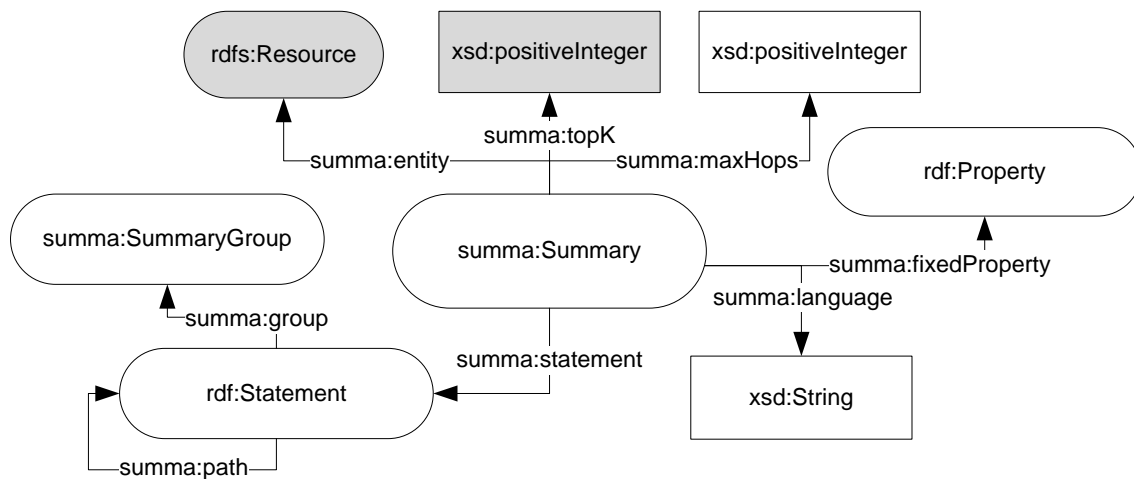


Figure 4.2.: The SUMMA Vocabulary. Mandatory parameters in grey.

4.2.1. SUMMA Vocabulary

The *SUMMA Vocabulary* offers various parameters that help to configure and represent a summary. During the design of the vocabulary we took the above considerations into account. An overview of the vocabulary is depicted in Figure 4.2. In the following we introduce all classes and predicates:

Summary This class describes the abstract concept of a summary of an entity. The IRIs of instances of this class are constructed with all query parameters.

SummaryGroup This class describes a group of statements. The entity summarization system does not necessarily have to produce groups. If groups are formed, it is completely up to the summarization system what is meant by them or if they come with a label in the desired language.

entity This predicate with domain `Summary` and range `rdfs:Resource` points to the entity that is summarized. As an example, the object of this predicate could be a DBpedia, Wikidata, or Freebase entity. This predicate is mandatory for the API.

topK This predicate defines the maximum number of statements that are being returned. This predicate is mandatory for the API.

statement This predicate with domain `Summary` and range `rdf:Statement` attaches statements to a summary in the response context.

maxHops This predicate defines the maximum number of hops in the graph the interface is able to represent. The default value is set to one, which means that all outgoing/incoming predicate-object pairs of the focused entity are being considered.

path This predicate enables to include the full paths in the returned statements of the summary. For each statement that is included in the summary that does not directly involve the focused entity, a path that shows how the current statement relates to the entity needs to be provided. This situation can occur if the maximum of hops

4. Interfacing Entity Summarization

Listing 4.1: Example for using vRank for ranking an RDF Statement.

```
@prefix vrank: <http://purl.org/voc/vrank#>.
@prefix dbr: <http://dbpedia.org/resource/>.
@prefix dbo: <http://dbpedia.org/ontology/>.
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema#>.

[ rdf:type rdf:Statement ;
  rdf:subject dbr:Barack_Obama ;
  rdf:predicate dbo:birthDate ;
  rdf:object "1961-08-04"^^xsd:date ;
  vrank:hasRank [ vrank:rankValue "3213.101"^^xsd:float ] ]
```

Listing 4.2: Example for a summary request that is sent via POST.

```
@prefix : <http://purl.org/voc/summa/>.
@prefix dbr: <http://dbpedia.org/resource/>.
@prefix dbo: <http://dbpedia.org/ontology/>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema#>.

[ a :Summary ;
  :entity dbr:Barack_Obama ;
  :topK "2"^^xsd:positiveInteger ;
  :language "en" ;
  :maxHops "2"^^xsd:positiveInteger ;
  :fixedProperty dbo:birthDate ;
  :fixedProperty dbo:birthPlace . ]
```

is greater than one. For more than two hops, this relation is needed multiple times until the object statement of path includes a triple that contains the focused entity.

language This predicate defines the languages in which the output literals should be available. We recommend to use a fixed vocabulary like Request for Comments (RFC) 4646³ for this.

group The group predicate enables summaries to form groups of statements. Attaching a group directly to a statement enables clients to ignore the predicate if present but not supported.

fixedProperty If there is already some background knowledge on the summarizer’s side about the underlying data structure it can request predicates that it wants to show in any case. Multiple different predicates can be defined in this way and thereby restricting the output to the defined set of predicates.⁴

Next to this vocabulary, we make use of the vRank vocabulary [RVTTS12], XSD [PGM⁺12], and OWL [W3C12]. The vRank vocabulary is necessary to include the computed scores

³RFC 4646 – <http://www.ietf.org/rfc/rfc4646.txt>, retrieved 2016-07-25.

⁴For the naming of this attribute we chose “property” rather than “predicate”. This decision was made to highlight the entity focus (a *property* of an entity rather than a *predicate* of a triple).

of each statement by the summarization service. A summary typically includes more than one `rdf:Statement`. Although, in some syntaxes, constructs such as

```
... [ a rdf:Statement; ... ], [ a rdf:Statement; ... ] .
```

could be mistaken for ordered lists, the group of statements is returned as a set. To determine an order between the statements additional information is required. In this respect, we choose to use `vRank` rather than `rdf:List` to enable summarization systems to publish the ranking scores. Listing 4.1 exemplifies the use of the `vRank` vocabulary in combination with a reified `rdf:Statement`.

The *SUMMA Vocabulary* is published at <http://purl.org/voc/summa/>. Exemplary usages of the vocabulary terms are shown in Listing 4.2 and Listing 4.3: The former states an input POST request for a summary of size two for the entity “Barack Obama” and involves further parameters (e.g., language, maxHops, etc.). The latter states an example output for the summary request of Listing 4.2 and includes two statements with ranking information, all necessary labels in English, a statement group, and further information about the summary (in particular all parameters that were sent via the POST request).

4.2.2. RESTful Web Service

Our interaction guideline is based on established combinations of RESTful architectures and Linked Data, in particular Richardson’s maturity model [RR07]. Stadtmüller et al. [SSHS13] summarize its main points as follows:

- *“The use of URI-identified resources.*
- *The use of a constrained set of operations, i.e., the HTTP methods, to access and manipulate resource states.*
- *The application of hypermedia controls, i.e., the data representing a resource contains links to other resources. Links allow a client to navigate from one resource to another during his interaction.”* [SSHS13]

The use of IRI-identified⁵ entities and their interlinkage are also direct consequences from the Linked Data design principles (see Section 2.1.4). Therefore, several existing approaches have already recognized the value of combining RESTful services and Linked Data (e.g., [BL09, VSD⁺11, SH11, SSHS13]).

We adopt these notions for our approach in order to enable a uniform interface to summarized entities that aligns with the standard Linked Data interaction model. The interaction of a client to retrieve the summary of an entity according to our approach is depicted in Figure 4.3 and works as follows:

1. A client can send a summary request for an entity to a server offering a summarization service via an HTTP POST request.

⁵We generalize from URIs to IRIs in order to comply to RDF 1.1.

4. Interfacing Entity Summarization

Listing 4.3: Example response in Turtle.

```
@prefix : <http://purl.org/voc/summa/>.
@prefix vrank: <http://purl.org/voc/vrank#>.
@prefix dbr: <http://dbpedia.org/resource/>.
@prefix dbo: <http://dbpedia.org/ontology/>.
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>.
@prefix xsd: <http://www.w3.org/2001/XMLSchema#>.

<http://ex.com/summary?entity=dbr:Barack_Obama&topK=2&language=en&
  maxHops=2&fixedProperty=dbr:birthDate,dbr:birthPlace> a :Summary ;
:entity dbr:Barack_Obama ;
:topK "2"^^xsd:Integer ;
:language "en" ;
:maxHops "2"^^xsd:Integer ;
:fixedProperty dbo:birthDate ;
:fixedProperty dbo:birthPlace ;
:statement

[ rdf:type rdf:Statement ;
  rdf:subject dbr:Barack_Obama ;
  rdf:predicate dbo:birthDate ;
  rdf:object "1961-08-04"^^xsd:date ;
  :group <http://example.com/group/12> ;
  vrank:hasRank [ vrank:rankValue "3213.101"^^xsd:float ]
] ,

[ rdf:type rdf:Statement ;
  rdf:subject dbr:Honolulu ;
  rdf:predicate dbo:areaCode ;
  rdf:object "808"@en ;
  vrank:hasRank [ vrank:rankValue "2323.433"^^xsd:float ] ;
:path [ rdf:type rdf:Statement ;
  rdf:subject dbr:Barack_Obama ;
  rdf:predicate dbo:birthPlace ;
  rdf:object dbr:Honolulu ]
] .

dbr:Barack_Obama rdfs:label "Barack Obama"@en .
dbo:birthDate rdfs:label "birth date"@en .
dbr:Honolulu rdfs:label "Honolulu"@en .
dbo:areaCode rdfs:label "area code"@en .
dbo:birthPlace rdfs:label "birth place"@en .
<http://example.com/group/12> rdfs:label "Important Dates"@en .

<http://ex.com/summary?entity=dbr:Barack_Obama&topK=2&language=en&
  maxHops=2&fixedProperty=dbr:birthDate,dbr:birthPlace#id> owl:sameAs
  dbr:Barack_Obama .
```

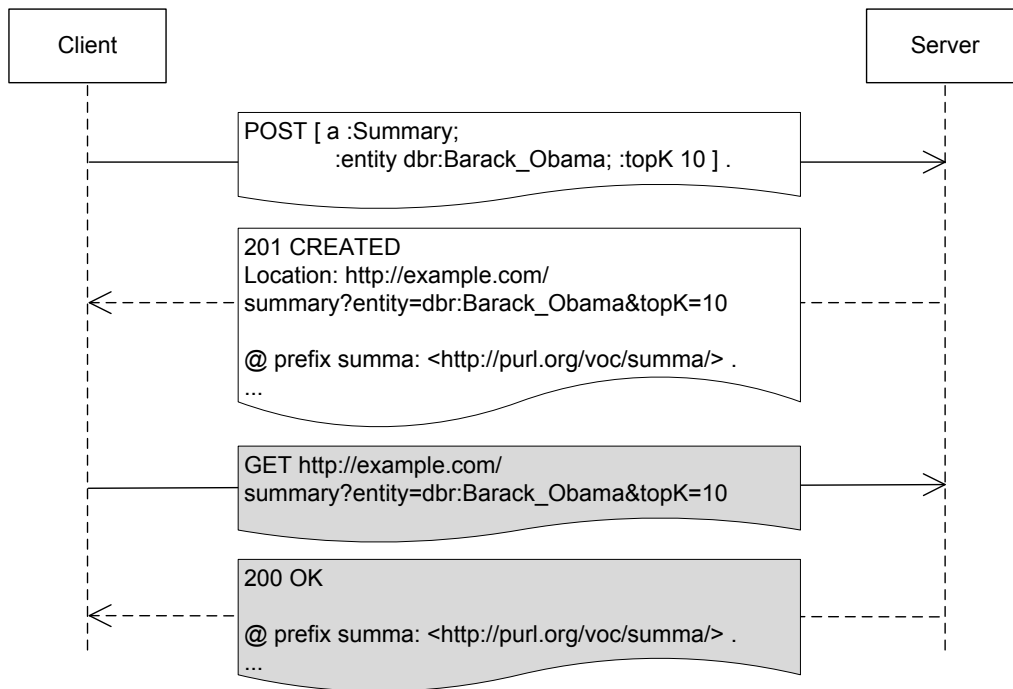


Figure 4.3.: RESTful interaction mechanism for entity summaries. Messages for first interaction: white. Messages for second interaction: grey

2. The response to the request contains the summarized entity in its payload, as well as a IRI in the location header field that identifies the created summary.
3. The client can use the IRI of the summary for further lookups of the summary via HTTP GET.

Since summaries can be looked up via HTTP GET, we enable simple caching mechanisms for the clients. The IRI of the summary also enables to include direct links to the summary within other Web resources. To construct the IRI that identifies a given summary, we adopt the approach of [SH11] where the IRI contains key/value pairs that correspond to the predicates in the original summary request. Note that the server does not have to store the created summaries for allowing the direct lookup but it can produce the summary on-the-fly by interpreting the key/value pairs of the IRI (in the case of GET requests).

A client can also skip the first interaction via POST and anticipate how the IRI of a summary would look like as the lookups are computed in the same way as the original POST request. However, we keep both interaction schemes in place in order to enable a clear formulation of a request and a clean cacheable lookup.

4.3. Experiments

Our experiments cover an empirical evaluation of the established requirements of Section 4.2 as well as reference implementations of the server and client components. Both, the

4. Interfacing Entity Summarization

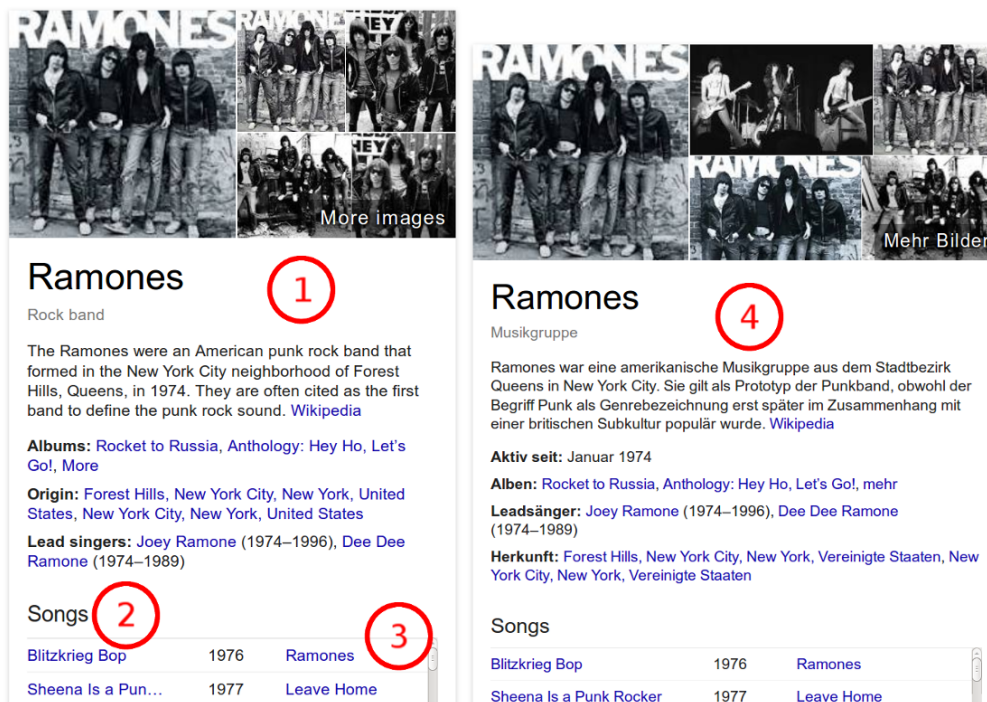


Figure 4.4.: Screenshot of the GKG representation of the “Ramones”: 1) Specific predicates such as the type and the Wikipedia description are always there (Property Restriction). 2) Several statements are gathered in a group named “Songs” (Statement Groups). 3) N-ary relations—in this case title, year, and album—are supported (Multi-hop Search Space). 4) The summary is offered in multiple languages (Languages)

evaluation and the reference implementations demonstrate the feasibility and effectiveness of the SUMMA API definition.

4.3.1. Evaluation

In our evaluation we inspected interfaces from well-known providers such as the Google Knowledge Graph (GKG) [Sin12], Microsoft Bing Satori/Snapshots [Qia13], or Yahoo Knowledge [Tor14]. We assessed whether the expressibility of these interfaces could be served via the SUMMA API. Thus, we provide empirical evidence about the general applicability of the API for different kinds of RDF entity summaries.

For our evaluation, we selected the entity summarization systems of the three major search engines (mentioned above) as well as systems from the Alexa Top News sites⁶ that offered factual knowledge about entities. We selected two of the top 25 news portals offering infoboxes about entities. These were Forbes⁷ and BBC news⁸. Our hypothesis was that the

⁶Alexa Top News sites – <http://www.alexa.com/topsites/category/Top/News>, retrieved 2016-07-25.

⁷Forbes, e.g., <http://www.forbes.com/profile/dirk-nowitzki/>, retrieved 2016-07-25.

⁸BBC news, e.g., <http://www.bbc.com/news/world-europe-17299607>, retrieved 2016-07-25.

defined API could serve all of these interfaces, thus potentially enabling them to switch between different entity summarization services without changing their layout. For this, we focused on five entities from diverse domains: Spain, Dirk Nowitzki, Ramones, SAP, Inglourious Basterds. These entities are representatives for a country, a person (or athlete), a band, a company (or organization), and a movie. We have to note that, as of March 2015, BBC only supported summaries of countries, Forbes only supported summaries of persons and organizations, and Yahoo only supported persons and movies. For these systems our insights were focused on the supported types. Some of the analyzed systems also used fixed schema patterns or a combination of entity-specific summaries and schema patterns. We assumed that, even by using only fixed schema patterns, the requirements for the interface would be the same. This still suited our evaluation scenario as our main goal was to decouple summary and presentation: the way in which the summaries were generated is not relevant (black box). We also tried to include research prototypes into our evaluation: unfortunately, although research in this field has been very active in recent years (e.g., [GTS15, CXQ15b, CXQ15a, GTSC16]), except LinkSUM (see Section 3.1), none of these prototypes was made available as an online system.

In the following we will analyze for each of the above-mentioned interfaces on whether they would be able to consume data from the API without changing their layout. We assume both, the IRI of the entity and the maximum number of facts (topK) as standard parameters. Figure 4.4 demonstrates the analysis of the interfaces.

Google Knowledge Graph For some facts, GKG used contexts about the data items (e.g., Wikipedia abstracts, population numbers, dates of marriage, release year of album, role names, etc.). In RDF, these contexts are represented as n-ary relations. Our API supports summaries over such constructs with the multi-hop search space. Further, certain predicates such as entity names, pictures, or types were always present in GKG. Not considering the result of the dynamic ranking, these predicates can be addressed with a separate summary request with `fixedProperty`. Further, GKG supported special groups of statements, such as the group of albums of a band. We support this feature by enabling to add a group to each statement by the entity summarization system. GKG was able to adapt the interface to different languages. This is supported by RDF (multilingualism of `rdfs:label`, that is literals) and by a parameter for the entity summarization system.

Bing Satori/Snapshots Bing Snapshots also supported features similar to the GKG (i.e., context, special predicate selection, grouping, multiple languages). Bing enabled tables like “Career vs. Season” statistics in their summaries. Even these statistics can be broken down to triples and represented in our output format. How the triples are arranged in the end, in a table style or just sequential is a matter of choice on the client side. Certain patterns in the output (e.g., multiple numerical values with the same predicate but varying context) suggest table-style presentation.

Yahoo Knowledge At the time of writing, Yahoo displayed factual knowledge about persons and movies. The output for movies was very similar to the aforementioned summarization systems of Google and Microsoft. Similar to Bing, the output for Dirk Nowitzki included various sport statistics. Like in Bing, this data can be covered by our output model. Entities representing other persons are very similar to the standard

4. Interfacing Entity Summarization

Table 4.1.: Requirements per interface. The checked features are supported by the specific interface, the crossed ones are not required.

Features (SUMMA)	Google	Bing	Yahoo	Forbes	BBC
Languages	✓	✓	✗	✗	✗
Multi-hop Search Space	✓	✓	✓	✓	✓
Property Restriction	✓	✓	✓	✓	✓
Statement Groups	✓	✓	✓	✓	✗

output of Google and Bing. Yahoo, as of March 2015, did not offer summaries in multiple languages.

Forbes The interface showed basic attributes of persons and companies via predicate-object pairs. Selected predicates—such as a depiction—were present for any entity. Similar to GKG, for some predicate-object pairs the context was added, e.g., “As of June 2014”. For companies, Forbes formed two groups: “At a Glance” and “Forbes Lists”. All these features are supported by our defined data model. Like Yahoo, Forbes did not offer their summaries in different language versions.

BBC news The BBC news portal included summaries of countries only. Like in Forbes, this data contained mainly key-value pairs and can be easily represented with our output format. Also presenting multi-hop information was needed, as the presented images had a caption that was also shown. BBC did not define groups of facts and did not offer other languages than English.

The complete results of our analysis are presented in Table 4.1. Overall we found that all the requirements (that these interfaces needed in order to offer all their functionality) were fulfilled by the proposed SUMMA API.

4.3.2. Implementation

The SUMMA API definition is based on Web standards such as the HTTP protocol and RDF. Summary producers as well as consumers can be implemented in a variety of programming languages. However, in order to demonstrate feasibility and to facilitate adoption, we provide a reference implementation based on Java Jersey⁹ (server) and JavaScript (client).

4.3.2.1. SUMMA Server

The *summaServer* application is an Apache Tomcat server application that fully implements the SUMMA API. It provides a basic summarization method for DBpedia entities. It ranks objects (only outgoing links are considered) based on their DBpedia PageRank scores (see Section 3.1.3.1.1), similar to [TR14]. All necessary information (including the DBpedia

⁹Java Jersey – <https://jersey.java.net/>, retrieved 2016-07-25.

Pulp Fiction		パルプ・フィクション	
country of origin	United States of America	本国	アメリカ合衆国
original language of work	English	原語	英語
narrative location	Los Angeles	撮影地	ロサンゼルス
instance of	film	以下の実体	映画
genre	comedy film	ジャンル	コメディ映画
genre	thriller	ジャンル	スリラー
main subject	organized crime	著作物の主題	組織犯罪
genre	independent film	ジャンル	自主映画
screenwriter	Quentin Tarantino	脚本家	クエンティン・タランティーノ
production company	Miramax Films	制作会社	ミラマックス
Pulp Fiction		低俗小説	
país de origen	Estados Unidos	起源国	美国
idioma original	inglés	原語言	英語
lugar de filmación	Los Ángeles	拍摄地点	洛杉矶
instancia de	película	性质	电影
género	comedia	艺术流派(类型)	喜劇電影
género	suspense	艺术流派(类型)	驚悚
tema principal de la obra	crimen organizado	作品主题	黑社會
género	cine independiente	艺术流派(类型)	美國獨立電影
director	Quentin Tarantino	导演	昆汀·塔伦蒂诺
empresa productora	Miramax	製作商	米拉麥克斯影業

Figure 4.5.: Basic summaries with Wikidata. This RDF knowledge base offers a high coverage of labels in different languages (in the presented case: English, Japanese, Spanish, and Chinese).

PageRank scores) is available via the official DBpedia SPARQL endpoint. The source code of *summaServer* application is published at

<https://github.com/athalhammer/summaServer>

and dual-licensed with the MIT License and GPLv3.¹⁰

Deployments of the *summaServer* application and LinkSUM (see Section 3.1), both implementing the SUMMA API, can be found at the following addresses:

- <http://km.aifb.kit.edu/summaServer>
- <http://km.aifb.kit.edu/services/link>

As of July 2016, first experiments with Wikidata¹¹ as a knowledge base have already been conducted. A main advantage of Wikidata over DBpedia is the high availability of multi-lingual labels (see Figure 4.5). For enabling summaries with Wikidata, the implementation of *summaServer* only required minor adaptations.

¹⁰License of *summaServer* – <https://raw.githubusercontent.com/athalhammer/summaServer/master/LICENSE>, retrieved 2016-07-25.

¹¹See Section 2.1.5 for more information on Wikidata.

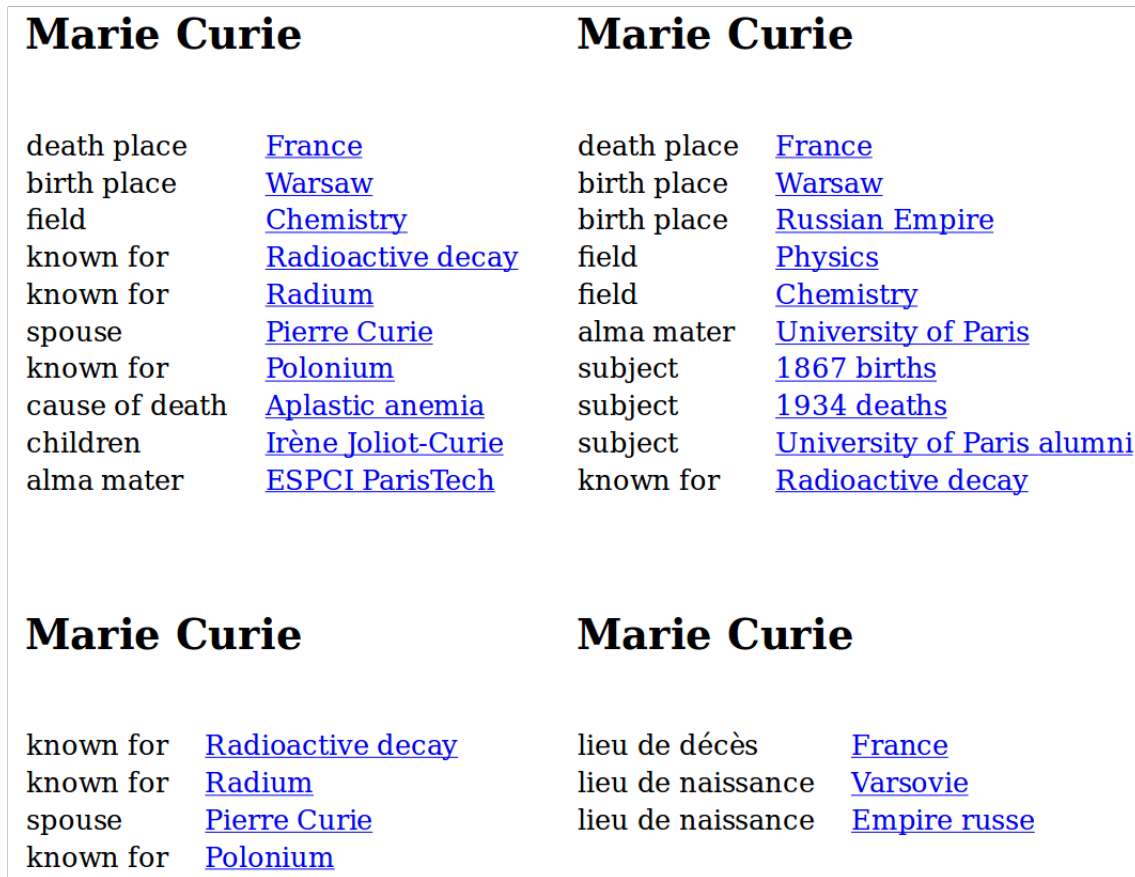


Figure 4.6.: Screenshot: Two example summaries with the same configuration but different systems (top). Example summary with restriction to two predicates (bottom left) and a different language and $topK = 3$ (bottom right).

4.3.2.2. SUMMA Client

The *summaClient* library is a lightweight JavaScript application that interacts with servers that implement the SUMMA API. It builds on jQuery¹² and jQuery UI¹³ and enables visualization and interaction with the results of multiple summarization engines within a single Web page and can be implemented with different styles (see Figure 4.5 and Figure 4.6). The source code of the *summaClient* library is published at

<https://github.com/athalhammer/summaClient>

and dual-licensed with the MIT License and GPLv3.¹⁴ A deployment of the *summaClient* library can be found at the following address:

<http://athalhammer.github.io/summaClient/>

¹²jQuery – <https://jquery.org/>, retrieved 2016-07-25.

¹³jQuery UI – <http://jqueryui.com/>, retrieved 2016-07-25.

¹⁴License of *summaClient* – <https://raw.githubusercontent.com/athalhammer/summaClient/gh-pages/LICENSE>, retrieved 2016-07-25.

On 25 September 2007, Merkel met the **14th Dalai Lama** for "private and informal talks" in the Chancellery in **Berlin** amid protest from **China**. **China** afterwards cancelled separate talks with **German** officials, including talks with **Justice Minister Brigitte Zypries**.

One of Merkel's priorities was strengthening transatlantic economic relations - she signed the agreement for the **Transatlantic Economic Council** on 30 April 2007 at the **White House**. The Council co-chaired by an EU and a US official, aims at removing barrier free-trade area. This pro **White House** left-wing politician **Jean-Citizens to multinationals American** foreign policy

White House

tenant [President of the United States](#)
 tenant [Barack Obama](#)
 style [Neoclassical architecture](#)
 style [Palladian architecture](#)
 location [Northwest, Washington, D.C.](#)

Der Spiegel reported that **Barack Obama** eased during Commenting on a **White Spiegel** stated, "Of course keep up with [Obama's]. "good natured" diplomat Obama's sister in **Heidelberg**, making it clear that she had read his autobiography".

Summary by <http://km.aifb.kit.edu/services/link>

Figure 4.7.: Automatically annotated excerpt of a Wikipedia article and the *summaClient* knowledge panel with a summary by LinkSUM (Source of the annotated text: https://en.wikipedia.org/w/index.php?title=Angela_Merkel&oldid=709980123, retrieved 2016-07-25).

The *summaClient* library supports three main interaction modes that support different levels of automation:

summa This method enables to place a summary into a specific HTML element (typically an empty `div` element) that is identified with an `id`. The method takes six parameters:

1. **entity**: the IRI of the entity.
2. **topK**: the number k of maximal facts.
3. **language**: the language in RFC 4646.
4. **fixedProperty**: a comma separated list of IRIs.
5. `element id`: the identifier of the HTML element where the summary is placed.
6. `summarization service`: the IRI of the SUMMA API server.

The parameters denoted in bold directly implement their SUMMA API counterparts. The `maxHops` parameter is not provided as it depends on the client library itself how many hops it supports. In the case of the *summaClient*, we use the setting `maxHops = 1`.

4. Interfacing Entity Summarization

qSUM This method detects annotations with `its-ta-ident-ref` (from the W3C Recommendation “Internationalization Tag Set (ITS) Version 2.0” [FML⁺13]) within any —typically `span`—HTML elements and registers `summa` (see above) summaries as `mouseover` events (see also the scenario of Section 1.1.1). As the entities are directly provided via the annotations and the summaries are not permanently shown, the `entity` and the `element id` parameters are not needed. The remaining four parameters are used as in `summa` (see above). The provided summaries can be also be used for browsing. For this, the user can fix the respective knowledge panel with a single mouse click on the annotated text.

ELES This way of operation combines the `qSUM` method (see above) directly with an the DBpedia Spotlight [DJHM13] entity linking approach. For this, we extended the DBpedia Spotlight jQuery plugin in order to enable ITS 2.0 output.¹⁵ The system uses a DBpedia Spotlight deployment in order to annotate one or more text paragraphs with entities from the DBpedia knowledge base. The `qSUM` method is then used within a hook that automatically recognizes when the DBpedia Spotlight annotation service has finished and the annotations are ready (via the modification of the document object model subtree).

The three interaction modes are exemplified in Listing 4.4. A screenshot of the `qSUM`/`ELES` interface in combination with `LinkSUM` (see Section 3.1) is provided in Figure 4.7. It has to be noted that, in Section 1.1.1, the `qSUM`/`ELES` interaction modes are presented as the main introductory scenario of this work.

4.4. Discussion

There are different points about the `SUMMA` API that deserve further discussion.

A modeling decision, that we took, was to provide the possibility to restrict the summary in accordance to selected predicates. However, in some settings it may also be useful to specify, on the client side, that certain predicates should be omitted. This can be useful in cases where many high-ranked relations via one predicate exist (such as incoming `dbo:birthPlace` relations for countries). Thus, we consider the possibility for omitting relations as a promising option for extending `SUMMA` API for future releases.

Another point that deserves rethinking is the provision of context that comes with `n`-ary relations. As we outlined in our descriptions, the `summa:path` predicate enables this feature. However, to this point, it is unclear whether the “principal” triples and triple chains from the top- k list should be distinguished from triples that add additional context to one of these. To make this more clear, we revisit the example of Section 2.1.1.1:

John Travolta played the role Vincent Vega in the movie Pulp Fiction.

¹⁵ITS 2.0 for DBpedia Spotlight – <https://github.com/dbpedia-spotlight/demo/pull/5>, retrieved 2016-07-25.

Listing 4.4: Full HTML example of the three interaction modes with the jQuery (UI), DBpedia Spotlight, and the SUMMA client libraries included.

```

<!DOCTYPE html><html><head><title>Example</title>
<style>span {background-color:#AAAAAA}</style>
<link rel="stylesheet" type="text/css"
href="http://athalhammer.github.io/summaClient/css/summaClient.css" />
<script src="http://code.jquery.com/jquery-2.2.1.min.js"></script>
<script src="http://code.jquery.com/ui/1.11.4/jquery-ui.js"></script>
<script src="http://dbpedia-spotlight.github.io/demo/dbpedia-spotlight
-0.3.js"></script>
<script src="http://athalhammer.github.io/summaClient/js/summaClient.js
"></script>
<script>
$(document).ready(function() {

    /*** summa ***/
    summa("http://dbpedia.org/resource/Quentin_Tarantino", 5, "en",
        "http://dbpedia.org/ontology/director",
        "pf-summary", "http://km.aifb.kit.edu/services/link/sum");

    /*** qSUM ***/
    qSUM(5, "en", null, "http://km.aifb.kit.edu/services/link/sum");

    /*** ELES ***/
    var select = ".annotate";
    $(select).bind("DOMSubtreeModified", function() {
        qSUM(5, "en", null, "http://km.aifb.kit.edu/services/link/sum")
    });
    // DBpedia Spotlight
    var settings = { "endpoint" : "http://spotlight.sztaki.hu:2222/rest
", "its" : "yes",
        "spotter" : "Default" };
    $(select).annotate(settings);
    $(select).annotate("best");
});
</script></head><body>

<div id="pf-summary"></div>

<div>
The narrative sequence called "The Gold Watch" of <span its-ta-ident-
ref="http://dbpedia.org/resource/Pulp_Fiction"> Pulp Fiction</span>
ends with Butch picking up Fabienne with Zed's <span its-ta-ident-
ref="http://dbpedia.org/resource/Chopper_(motorcycle)">chopper</
span>.
</div>

<div class="annotate">Uma Thurman was nominated for the Best Actress in
a Supporting Role at the Academy Awards 1995.</div>
</body></html>

```

4. Interfacing Entity Summarization

When modeled with n-ary relations

```
:Pulp_Fiction :starring _:S .  
_:S :actor :John_Travolta .  
_:S :role :Vincent_Vega .
```

and both facts, the actor and the role, are provided by the SUMMA API for a summary of Pulp Fiction, it is unclear whether the role only provides additional context for the “main fact” (that John Travolta acted in Pulp Fiction) or it counts as a own top- k fact. In case of the former, additional extensions to SUMMA are necessary in order to mark “context facts” accordingly.

The next aspect that we want to discuss is related to the presentation: On the left hand side of Figure 4.6 it is recognizable that facts with the same predicate (in this case “known for”) are ranked at different positions with facts that have a different predicate between them. This is a particularity of the SUMMA API that was directly transferred to *summaClient*. However, in other clients, we group facts together that have the same predicate. This is done simply by averaging the scores of facts with the same predicate and reordering them accordingly in the interface (note that the order between the facts with the same predicate stays the same). This makes the interface more clear and removes the cluttered appearance that the summaries of figures 4.5 and 4.6 transmit. However, what follows directly from this step is the question about the completeness of the provided facts.¹⁶ While—as per definition—we present only a selection of facts, the visual grouping of facts with the same predicate makes the user to raise questions like “are these the only actors in this movie?”. Unfortunately, from what has been defined in the SUMMA API, the SUMMA client can not directly provide an answer to this question. One option to mitigate this is to provide a “see all” button for each predicate of the summary, with which further subject-predicate combinations can be browsed. Another option could be to extend SUMMA with predicate annotations that can be either “complete” (all known facts with this predicate are contained in this summary) or “incomplete” (further facts with this predicate exist). We implemented the complementary approach to a “see all” button with our SUMMARUM system [TR14] (an initial prototype of LinkSUM, see Section 3.1) in which we adopted the interaction mechanism of Semantic MediaWiki [KVV⁺07], where predicate-object combinations can be browsed with a “plus lens” sign (see Figure 4.8).

4.5. Related Work

For the related work we distinguish between two kinds of approaches: systems that add an additional layer between a SPARQL endpoint and data consumers (as such serving as direct data providers) and approaches that introduce formalisms that enable ranked views on Linked Data.

¹⁶See [DRPN16] for a related discussion on completeness in RDF knowledge bases.

Barack Obama			birth place Hawaii		
Subject	Living people	± 🔍	birth place of	Barack Obama	± 🔍
birth place	United States	± 🔍	birth place of	Nicole Kidman	± 🔍
party	Democratic Party (United States)	± 🔍	birth place of	Presidency of Barack Obama	± 🔍
region	Illinois	± 🔍	birth place of	Daniel Inouye	± 🔍
religion	Christianity	± 🔍	birth place of	Nicole Scherzinger	± 🔍
incumbent of	President of the United States	± 🔍	birth place of	Lois Lowry	± 🔍
leader name of	Puerto Rico	± 🔍	birth place of	Bernice Pauahi Bishop	± 🔍
predecessor	George W. Bush	± 🔍	birth place of	Tia Carrere	± 🔍
birth place	Hawaii	± 🔍	birth place of	Michelle Wie	± 🔍
alma mater	Columbia University	± 🔍	birth place of	Israel Kamakawiwo'ole	± 🔍

Figure 4.8.: Summary of `dbr:Barack.Obama` (left) and the ranked list of statements with `dbo:birthPlace` and `dbr:Hawaii` (right).

Pubby¹⁷ is used to add an intuitive interface to SPARQL endpoints. It enables to consume entities and ontologies on a per-concept basis directly in various formats. For entities, it considers attached literal values in all available languages as well as all incoming and outgoing relations. In general, Pubby implements the following pattern for resources described by their IRI:

```
SELECT * WHERE { { <IRI> ?p ?v . } UNION { ?v ?p <IRI> . } }
```

This may result in a large set of facts that are directly related to the currently browsed entity. For machines as well as for human consumers all information about an entity is provided. In our approach we extend this mechanism by various configurable properties (e.g., maximum number of statements) that enable client interfaces to retrieve distilled versions of entities in a uniform way.

The Linked Data API¹⁸ adds a RESTful layer on SPARQL endpoints. It enables developers who are not familiar with SPARQL or RDF in general to access SPARQL endpoints in a RESTful manner. As an example, it enables to represent selectors and filter options as request parameters in the following form:

```
http://example.com/university?country=UK&min-noStudents=10
```

Potential response formats include JSON, XML, RDF/XML, and Turtle. The Elda¹⁹ system provides a reference implementation for the Linked Data API definition. The Linked Data API and SUMMA both add an additional RESTful layer on top of SPARQL endpoints. However, the rationales of both approaches are complementary: while the Linked Data API tries to make part of the SPARQL feature set more intuitively accessible using REST, we are focusing on defining a uniform RESTful interface that enables multiple services to provide concise views on the same entity in a uniform way.

Pietriga et al. define Fresnel [PBKL06],²⁰ a vocabulary for selecting and formatting RDF

¹⁷Pubby – <http://wifo5-03.informatik.uni-mannheim.de/pubby/>, retrieved 2016-07-25.

¹⁸Linked Data API – <https://github.com/UKGovLD/linked-data-api/blob/wiki/Specification.md>, retrieved 2016-07-20.

¹⁹Elda – <https://github.com/epimorphics/elda>, retrieved 2016-07-25.

²⁰Fresnel – <http://www.w3.org/2005/04/fresnel-info/manual/>, retrieved 2016-07-25.

4. Interfacing Entity Summarization

data. The vocabulary is supported by RDF browsers such as Longwell²¹, Piggy Bank²², or IsaViz²³. It is divided into two main components, lenses and formats. While the lenses help on selecting which content should be presented the formats define the style in which the selected content should be presented. Our work is mostly related to Fresnel Lenses: The predicates `fresnel:instanceLensDomain` and `fresnel:classLensDomain` define the levels on which the lenses can be applied. The predicates `fresnel:showProperties` and `fresnel:hideProperties` define which properties of the instance or class are commonly shown and in which order. The order is defined with `rdf:List`. Moreover, the Fresnel Selector Language (FSL)²⁴ enables to define further restrictions, for example which properties of connected entities should be shown (e.g., `foaf:name`). The predicate `fresnel:instanceLensDomain` in combination with the predicate `fresnel:showproperties` predicate and FSL enable quite particular decisions on which triples are included in the output and which are not. Eventually, however, covering specific triples for the output with Fresnel involves complex FSL patterns and, more importantly, still only provides a description of which information should be presented but not the information itself. Summarizing entities with respect to their individual particularities is possible but the lens descriptions would already cover much of the actual data. The remaining information such as the objects and all labels would have to be gathered at a different place. In other words, SUMMA provides access to entity-specific data while Fresnel, more abstractly, was designed to operate on the class level and to provide views. In fact, there are efforts to identify the most common predicates per DBpedia class with surveys and crowd sourcing and to publish them as Fresnel lenses [AATC14]. The SUMMA API could be used for interpreting such class-level lenses and for delivering the respective content accordingly. In addition, the SUMMA API explicitly enables entity-specific summaries that are beyond the scope of Fresnel.

Federated SPARQL queries [PBA13] offer the possibility to query knowledge bases distributed over multiple endpoints with a single query. Summaries that are computed offline could be stored at one endpoint while the actual summarized knowledge base that contains further information (such as labels) is available at a different endpoint. A single federated query would retrieve triples specific to an entity while the SPARQL `LIMIT` clause would enable different summary sizes. As in our approach, the endpoint for the summary can be easily exchanged. Summaries that are computed online (e.g., depending on the user's geolocation, language, the time of the day, etc.) can get too complex in order to be retrieved with SPARQL queries of any kind. Intermediately storing the result in an endpoint in order to make it retrievable with SPARQL adds significant overhead to a process that needs to be performed in a range of few 100 milliseconds.

Roa-Valverde et al. introduce a vocabulary for sharing ranking computations over RDF data [RVTT12]. This enables to provide detailed information about ranking computations

²¹Longwell – <http://web.archive.org/web/20140829055659/http://simile.mit.edu/wiki/Longwell>, retrieved 2016-07-25.

²²Piggy Bank – http://web.archive.org/web/20140930172921/http://simile.mit.edu/wiki/Piggy_Bank, retrieved 2016-07-25.

²³IsaViz – <https://www.w3.org/2001/11/IsaViz/>, retrieved 2016-07-25.

²⁴Fresnel Selector Language (FSL) – <http://www.w3.org/2005/04/fresnel-info/fsl/>, retrieved 2016-07-25.

in RDF. Properties include ranking values and time stamps as well as algorithm descriptions and configurations. We use the vRank vocabulary in order to provide ranking values to the client interface.

Harth introduces VisiNav [Har10], a system that allows for new interaction principles within the Web of Data. The system is based on four key concepts that support search and navigation: *Keyword Search*, *Object Focus*, *Path Traversal*, and *Facet Selection*. Our API clearly supports *Object Focus* as it is specifically designed to deliver entity-specific summaries. We also support *Path Traversal* and *Facet Selection*. However, the two concepts become quite similar if one does not distinguish between incoming and outgoing connections. More specifically, we slightly reinterpret the *Facet Selection* concept as we form the union rather than the intersection (“... the user can reformulate the query and obtain increasingly specific result sets” [Har10]). Like our approach VisiNav also provides ranked views on data. VisiNav strongly couples the user interface and the back end. As such, the rankings and views on the data can only be displayed with the VisiNav system. In this chapter, we provided a way to enable decoupling of the interfaces and their respective ranking back end.

In conclusion, we can state that the idea of browsing Linked Data with concise presentations is well established and real-world applications are taking up this idea [Sin12, Qia13, Tor14]. To the best of our knowledge, all previous research approaches for presenting RDF data in a concise way are based on schema patterns and do not provide the data itself. In this chapter, we introduced a novel approach that supports the evaluation and exchange of entity summaries in a lightweight way.

4.6. Conclusions

We introduced an API that enables entity summarization systems to publish summaries in a uniform way. Further, it enables consumers to access summaries of Linked Data entities from a multitude of summarization services through a single lookup mechanism. Our empirical evaluation shows that the SUMMA API could be applied to existing commercial systems while the reference implementations provide evidence for feasibility and facilitate adoption. Existing approaches can easily adopt the SUMMA API as their current user interfaces can be augmented with the RESTful access mechanism.

Our main conclusions are as follows:

1. The SUMMA API provides a mechanism for the uniform exchange of entity summaries between systems. We have demonstrated that our design decisions, in particular the inclusion of the ranking scores, enable direct comparison between summaries. This also includes reference datasets and thus, enables to fully automatize quantitative evaluation processes.
2. With the implementation of multiple SUMMA servers that are accessed via a single interface, we have demonstrated that summaries of different systems can be seamlessly arranged in a single interface for direct qualitative comparison. This will

4. *Interfacing Entity Summarization*

enable future comparative qualitative evaluation efforts to focus on the evaluation aspect itself rather than accompanying technicalities of providing unified interfaces.

3. The lightweight JavaScript approach of our *summaClient* reference implementation enables to change summarization approaches (and only the according ranking) during live interaction. This can support companies and researchers to gain deep insights about the used summarization approaches and their effects on user interaction.

Following these contributions, the SUMMA API was designed in a flexible way for future extension (and to enable backwards compatibility). Therefore, the following problems can be addressed in and along future versions of SUMMA:

- Next to restraining a summary towards a set of predicates, the SUMMA API could be extended by a mechanism for specifying (on the client side) omission of a set of predicates.
- The top- k summaries currently also include facts that provide additional context for others. It is unclear whether these should be counted as one of k or if they should be in a separate category. In the latter case, additional predicates for marking such facts are necessary.
- For the predicates of an entity summary, it is unclear whether there exist further objects in the knowledge base that are not shown. This information could be transmitted with an additional flag on the provided predicates.
- For every existing summarization system, wrappers to SUMMA can be implemented. The implementation of such adapters remains an open (technical) task.
- We envision a portal where different entity summarization services are gathered and described also in accordance to their non-functional properties, for example response time and availability.
- The SUMMA API does not address any context and or personalization factors. Implementing such measures may lead to significant extensions of the SUMMA API.

5. Towards Entity Data Fusion

We address Research Question 3 in this chapter:

How can we align duplicate/similar facts about Linked Data entities on the Web?

The contribution, in which we address this question, states an entity data fusion approach that enables multi-source summaries of Linked Data entities. This is an optional step (indicated by the dashed border of the box in Figure 5.1): as demonstrated in Chapter 3, each of the two presented entity summarization approaches can also work with only one knowledge base.

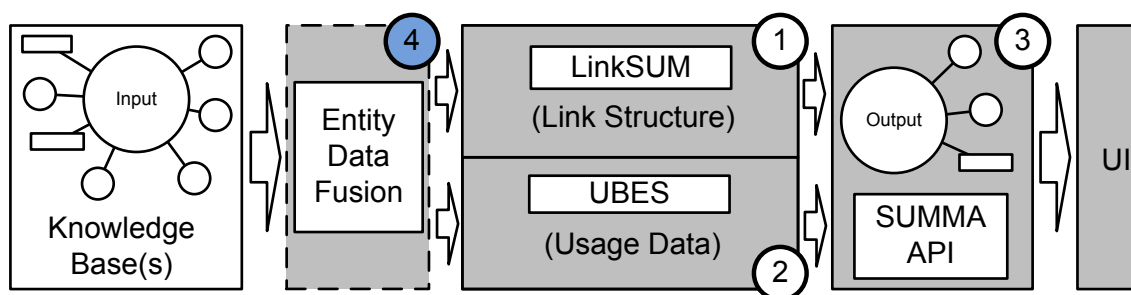


Figure 5.1.: Overview of the contributions of this thesis. In this part, we focus the fusion of duplicate/similar facts that are provided as Linked Data.

5.1. Introduction

Between December 2014 and December 2015 the percentage of Web sites that include semantic markup (i.e., schema.org) has risen from 22% to 31.3% [GBM15]. Thus, current content management systems, electronic shops, transparent government initiatives, non-profit organizations, and commercial sources publish information in accordance to the Linked Data principles (see Section 2.1.4). Large-scale retrieval systems and Web mashups (e.g., search engines, social networks) utilize the provided information and often present all or part of the data to their users, for example in knowledge panels. However, as of July 2016, many of the facts in knowledge panels do not provide a sufficient amount of (authoritative) sources that support the stated facts and doubts about correctness or notability were claimed (e.g., [FG16, Cav16]). In this chapter, we tackle the problem of fact alignment in order to enable the scenario of a trustable knowledge panel (see Figure 5.2). The idea of the panel serves two main purposes: 1) it enables users to verify the source(s) of a provided fact; 2) the provided number of different sources for a fact can serve as a support or justification

5. Towards Entity Data Fusion

for notability (an important aspect of entity summarization). In addition, the alignment of facts enable consolidation of sources and facts, that are commonly acknowledged but not covered by one (or more) of the sources, can be identified. In essence, we tackle the problem of identifying when multiple sources make the same claim about a specific entity in different structured ways (i.e., by using different vocabularies/granularities). We define “entity data fusion” as a method that addresses this problem.

Linked Data deals with a collection of individual triples that constitute very concise information units. As of 2013, more than 17 billion of such triples exist on the Web [MPB14]. This leads us to the following challenge: The sheer amount of data makes it difficult to identify statements that make the same or a similar claim. Different vocabulary terms are used to describe the same resources and relations. In particular, the aforementioned 17 billion triples use more than 15 thousand classes and more than 170 thousand relations [MPB14]. In our contribution, we address this challenge by introducing an entity-centric view on the data. We use agreement among different sources as a key concept and enable to address a range of problems that are associated with knowledge panels that rely on single sources:

Trust Data sources are not referenced by knowledge panels. The sources should be referenced (providing provenance) in order to enable users a clear decision on whether they want to trust a stated fact or not.

Notability On the Web, facts about an entity are not ranked and are either present or not. The notability in accordance to each entity is not known in such binary settings. We assume that some data sources cover fewer details than others what can help to estimate the notability of each fact.

Reliability Single data sources may contain false, dirty, or outdated information. Fusing data from a variety of sources can help to identify common information as well as its most common representation.

Objectivity Certain aspects of an entity might be covered by one data source but not by another. Multiple sources of data can help to identify commonly known facts about an entity and mitigate each data source’s individual bias.

Availability A knowledge panel that relies on a single data source will depend on it. It depends not only on quality aspects (see Reliability) but also on the availability of the data source itself.

These issues do not stand by themselves but are intersecting in the question “which sources cover similar facts?”.

While structured data on the Web is commonly published in RDF format [GBM15], there exist a number of different identifiers and vocabularies. For example, as of July 2016, it is not made explicit in either of the two resources

<http://www.imdb.com/name/nm3805083> and

http://dbpedia.org/resource/Tim_Berners-Lee


that they describe the same entity. This problem is commonly solved by generic or customized record linkage algorithms or according heuristics [HHD07, HMBT13]. The

Tim Berners-Lee³

Computer scientist ²

Sir Timothy John Berners-Lee OM KBE FRS FREng FRSA FBCS, also known as TimBL, is an English computer scientist, best known as the inventor of the World Wide Web. ¹

Born: June 8, 1955 (age 60) ⁴, London, United Kingdom ³



```

<http://dbpedia.org/ontology/birthDate> "1955-06-08" .
(http://dbpedia.org/data/Tim\_Berners-Lee.ttl)

<http://www.wikidata.org/prop/direct/P569> "1955-06-08" .
(https://www.wikidata.org/wiki/Special:EntityData/Q80.ttl)

<http://rdf.freebase.com/ns/people.person.date_of_birth> "1955-06-08" .
(https://www.googleapis.com/freebase/v1/rdf/m/07d5b)

<http://schema.org/birthDate> "1955-6-8" .
(http://www.imdb.com/name/nm3805083)

```

Figure 5.2.: Mock-up of a trustable knowledge panel (based on a Google screenshot). The colors of the buttons implement a traffic light scheme for the trustability of the presented fact. By clicking on such a button, a pop up would open that provides direct reference to documents which cover the presented fact as well as each document’s individual presentation.

focus of this work is to get from mapped identifiers (of a single entity) to completely mapped RDF triples (and chains of triples). For this, it is necessary to map the respective vocabulary terms and involved entities while accounting for different (non-trivial) modeling decisions. In the following, we present an example for modeling a single fact in RDF in multiple different ways: we want to state (in English language) that the entity “Tim Berners-Lee” (TimBL) has “Web developer” as an occupation. The following three sets of triples transmit this fact at different levels of granularity:

1. `ex1:TimBL ex1:occ "Web developer"@en .`
2. `ex2:TimBL ex2:job ex2:WebDev .`
`ex2:WebDev rdfs:label "Web developer"@en .`
3. `ex3:TimBL ex3:occ ex3:Work4 .`
`ex3:Work4 ex3:work ex3:WebDev .`
`ex3:Work4 ex3:since "1989-03" .`
`ex3:WebDev rdfs:label "Web developer"@en .`

In (1.), only a non-clickable string would be displayed for “Web developer”. With (2.) and (3.), a link to `ex2:WebDev/ex3:WebDev` can be provided where potentially more information about the profession can be retrieved. However, if we also want to model “since when Tim Berners-Lee has been a Web developer”, we need to make use of n-ary relations¹ as shown in (3.). We create an individual connecting node (`ex3:Work4`) in order to combine the information that “Tim Berners-Lee has been a Web developer since

¹See Section 2.1.1.1 for an introduction to n-ary relations.

5. Towards Entity Data Fusion

March 1989”². While some vocabularies (such as schema.org or the Open Graph Protocol²) commonly use the more coarse-grained variants of (1.) and (2.) in their modeling, Web knowledge bases such as Freebase [BEP⁺08] and Wikidata [VK14] enable fine-grained modeling with n-ary relations (context/qualifiers) as exemplified in (3.). In general it is the authors’ decision which level of detail they want to address with the data they publish on the Web.

We summarize the above-mentioned points in three main challenges:

1. There are billions facts directly retrievable from the Web as structured data [MPB14]. Many of these facts occur multiple times with different vocabularies. How do we align duplicate/similar facts at high precision?
2. Depending on the source, a single fact can be modeled at different granularity levels. This poses a difficult problem that—to the best of our knowledge—has not been directly targeted by fact/ontology alignment approaches [Ehr06, TCC⁺10, SAS11]. This leads to the question: how can we align facts across different modeling granularity levels?
3. Fact alignment is a novel and complex task. How do we create a reproducible evaluation scenario that covers the different aspects and how do we compare our approach to existing baselines?

In our approach, we tackle these problems by introducing a pipelined, entity-centric approach that retrieves facts from different Web sources; extracts path features; performs hierarchical clustering; refines clusters; and selects representatives. The approach builds on retrieving data from different sources that provide information about a specific entity. The fusion method performs the complex alignment of different model granularities and automatically moves similar facts (and chains of facts) to the same clusters. We compare our approach to a baseline established by Tummarello et al. in [TCC⁺10].

The contributions of this work are as follows:

1. We introduce an entity-centric approach that enables the fact alignment without prior knowledge about the used vocabularies.
2. As, to the best of our knowledge, the first existing fusion approach, we enable the alignment of facts from multiple sources while also taking into account different modeling granularities.
3. In our experiments we apply different measures on entity-centric, multi-sourced facts and demonstrate superiority over the Sig.ma baseline [TCC⁺10] along the scenario of a trustable knowledge panel.

The rest of this chapter is organized as follows: In Section 5.2, we introduce our fact fusion approach with a walk-through example. In Section 5.3, we provide an in-depth evaluation along the scenario of a trust-able knowledge panel. We then discuss the results of the evaluation in Section 5.4. In Section 5.5 we introduce related approaches and discuss, how our method is different. Finally, we conclude this chapter with Section 5.6.

²Open Graph Protocol – <http://ogp.me/>, retrieved 2016-07-28.



Figure 5.3.: Data fusion processing pipeline.

5.2. Approach: Entity Data Fusion

The data fusion method consists of a processing pipeline that is outlined in Figure 5.3. It includes seven steps:

1. Discover a set of IRIs that represent a specific entity (i.e., record linkage).
2. Retrieve RDF data for each identifier and their connected resources (concentrically up to a certain depth).
3. Extract a set of path features from the RDF data.
4. Run agglomerative hierarchical clustering on the set of path features.
5. Refine clusters by merging.
6. Identify facts as cluster representatives.
7. Use different cluster features for ranking or filtering.

The main focus of this work is on steps 2 to 6 but we also provide general information on 1 (e.g., what kind of input do we expect from the record linkage step) and 7 (e.g., what kind of output does the system produce and how it can be used).

The key principle of the approach is the combination of multiple aspects: 1) the entity-centricity strongly reduces ambiguity when applying string similarity measures for the clustering approach; 2) the use of path features (and their individual string representations) enables the alignment across different modeling granularities; 3) the clustering and the cluster merging steps enable the natural grouping of similar facts across sources.

In the following we explain each of the involved steps in more detail.

5.2.1. Record Linkage

The topic of record linkage has had a long tradition in statistics and different subfields of computer science, including databases and information retrieval [KSS06]. While it had a variety of different names,³ the main idea is to retrieve different files, entries, or identifiers that refer to the same entity (e.g., a specific person). This problem has also been explored in the (Semantic) Web context [HMBT13, HHD07]. With the use of explicit equivalence (e.g., by using `http://schema.org/sameAs` or `owl:sameAs`), the availability of a variety of algorithms (e.g., [GDS14] for a recent work), and the availability of systems that

³This concept has many other names in computer science research (e.g., entity linking/matching/consolidation or entity/record resolution).

5. Towards Entity Data Fusion

offer record linkage as a service (e.g., <http://sameas.org>), we regard this problem as sufficiently addressed. The record linkage approach is expected to take one IRI for an entity as an input (e.g., http://dbpedia.org/resource/Tim_Berners-Lee) and then produce an extended set R of reference IRIs that all describe the same entity, for example:

$$R = \{ \text{http://dbpedia.org/resource/Tim_Berners-Lee}, \\ \text{http://www.imdb.com/name/nm3805083}, \\ \text{http://www.wikidata.org/entity/Q80} \}$$

These reference IRIs can be used to retrieve different descriptions of the same entity. HTTP GET requests on the mentioned reference IRIs provide both, human-readable and machine-processable data. While, for human readers the layout looks different (although the resources provide HTML), for machines the vocabularies and the modeling of the provided data is different (although the resources provide RDF). For brevity, in the following examples we use the following set of reference IRIs:

$$R = \{ \text{ex1:TimBL}, \text{ex2:TimBL} \}$$

5.2.2. Data Retrieval

For each IRI that was retrieved by the record linkage approach, we aim to retrieve RDF data. If one of the IRIs offers structured data, the crawler performs a breadth-first search around the IRI (up to a certain depth). For example, if the triple $[\text{ex2:TimBL} \text{ ex2:job} \text{ ex2:WebDev}]$ is contained in the retrieved dataset, it also tries to retrieve RDF data from ex2:WebDev . In addition, the crawler also retrieves information about the used predicates; in this case it also retrieves data from ex2:job . We retrieve this information up to a certain depth around the entity. During this process, the crawler stores the complete path to the finally delivering IRI for each IRI call. If RDF data is returned, this IRI is used as a context for all retrieved triples. As cross-references are common in Linked Data, the crawler does not follow links that point to a known reference IRI, for example if ex2:TimBL is known to represent the same entity, we do not follow the link in $[\text{ex1:TimBL} \text{ ex1:seeAlso} \text{ ex2:TimBL}]$ even if the second resource has not yet been targeted at that point. The result of this step is RDF data that contains a forest of trees that each have one reference IRI for the entity (from the record linkage) as a root. Figure 5.4 shows an example for such a forest.

5.2.3. Feature Extraction

We produce path features from each tree in the forest that was created by the data retrieval step. In the following we consider paths in the tree from the root to a leaf. In accordance to the definition of RDF, each tree can have two types of leaves: resource nodes or literal nodes. However, leaves that are resource nodes do not provide sufficient information as

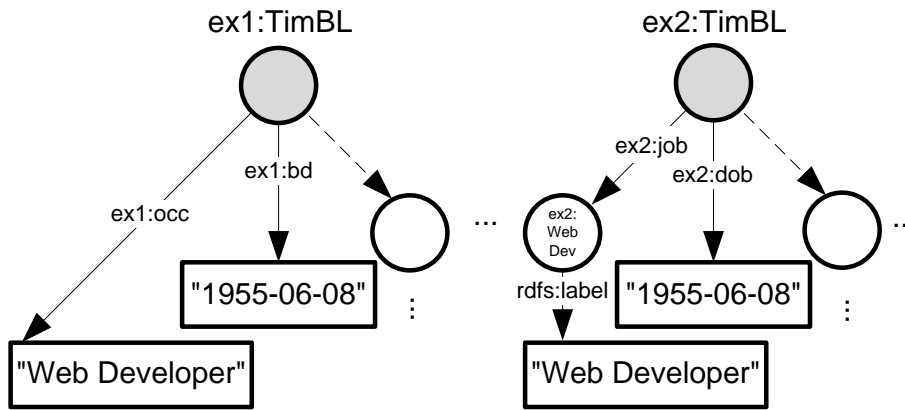


Figure 5.4.: Output of the data retrieval step: an RDF graph that contains a forest of trees, each with a reference IRI as a root.

the node itself is a IRI that was not retrieved. The system only knows that it exists. For example, if we crawl `ex2:TimBL` only with depth 0 (i.e., `ex2:TimBL` and according predicates are retrieved) the system knows that the node `ex2:WebDev` exists but we do not get the label “Web Developer” if the IRI is not retrieved. Therefore, we only consider paths that end with a literal node. We refer to path features that involve multiple triples as “multi-hop” and path features, that are constituted by only one triple, as “single-hop” path features. In the case of Figure 5.4, if we only consider the depicted facts, the following path features are created:

1. [`ex1:TimBL ex1:bd "1955-06-08"`]
2. [`ex1:TimBL ex1:occ "Web developer"`]
3. [`ex2:TimBL ex2:dob "1955-06-08"`]
4. [`ex2:TimBL ex2:job ex2:WebDev`] $\circ\circ$ ⁴
`[ex2:WebDev rdfs:label "Web developer"]`

We can present path features as linked lists of strings by removing all resource nodes and by using the `rdfs:label` for the predicates. Note that different vocabularies often provide different labels. For example `ex1:bd` may provide “birth date” while `ex2:dob` may have “date of birth” as a label. If a predicate has more than one label in a language we create an additional representation for the path feature, for example `ex2:job` may have two labels, “occupation” and “profession”. To account for that we add another string representation for the path feature. We collect all string representations in a (multi-valued) map L :

$$L = [\begin{array}{l} ("birth\ date" \rightarrow "1955-06-08", [1]); \\ ("occupation" \rightarrow "Web\ developer", [2]); \\ ("date\ of\ birth" \rightarrow "1955-06-08", [3]); \\ ("occupation" \rightarrow "label" \rightarrow "Web\ developer", [4]); \\ ("profession" \rightarrow "label" \rightarrow "Web\ developer", [4]) \end{array}]$$

⁴The $\circ\circ$ symbol is used to denote the connection of multiple triples that form a path feature.

For all text-based literals and labels we fix the language. In practice best results can be achieved with English as vocabularies often provide labels only in that language.

5.2.4. Clustering

We cluster path features in accordance to their string representations. At this point, the key feature of the approach—the **entity centrality**—mitigates the occurrence of ambiguities and unwanted fusions. For example, the string “Apple” has only one reasonable meaning in the vicinity of the entity `ex: iPhone5` while in the whole Web graph there are many different meanings for this term.

Similarity. In order to compare the string representations with each other, we use string similarity functions as they are proposed for ontology alignment [CH13]. For two given string representations we compare the head h (i.e., the label of the first predicate) and the tail t (i.e., the leaf nodes) of each list $l_i \in L$ respectively. As such, our similarity is defined as follows:

$$\text{sim}(l_1, l_2) = \frac{\text{strSim}(h(l_1), h(l_2)) + \text{strSim}(t(l_1), t(l_2))}{2} \quad (5.1)$$

The string similarity function incorporates basic tokenization (to) and normalization steps. We distinguish between single-token and multi-token strings:

$$\text{strSim}(s_1, s_2) = \begin{cases} \text{jw}(s_1, s_2) & \text{if } |to(s_1)| = 1 \\ & \& |to(s_2)| = 1 \\ \text{ja}(to(s_1), to(s_2)) & \text{otherwise} \end{cases} \quad (5.2)$$

Single-token strings use the Jaro-Winkler similarity metric (jw) and multi-token strings use Jaccard similarity (ja). These measures are recommended in [CH13] for achieving high precision. For both string similarity measures, a value of 0 means no similarity and 1 is an exact match.

Clustering. We compute a similarity matrix for all string representations as an input for agglomerative hierarchical clustering⁵. The clustering is based on two steps: in the beginning, the linkage of all elements is computed and afterwards the clusters are formed by a cut-off.⁶ The linkage starts with clusters of size 1 and uses the similarity matrix in order to link two clusters. This is done in accordance to the smallest Euclidean distance of any two elements in the respective clusters. The elements are represented as column vectors. We repeat this step until all clusters are linked. The linkage is then used to determine a cut-off level that produces n or fewer clusters. Under the assumption that all

⁵MATLAB hierarchical clustering – <http://mathworks.com/help/stats/hierarchical-clustering.html>, retrieved 2016-10-09.

⁶Detailed information on the number of comparisons and a discussion on the scalability are provided in Section 5.3 and Section 5.4 respectively.

resources in R provide RDF data and that each covers the same amount of information, the value of n can be set to $\left\lceil \frac{|L|}{|R|} \right\rceil$.⁷ In our running example n would be $\left\lceil \frac{5}{2} \right\rceil = 3$.

After the clustering, we use the map L to move back from the string representation level to the path feature level. The clusters are then represented as follows:

- **Cluster 1:** { [ex1:TimBL ex1:bd "1955-06-08"],
[ex2:TimBL ex2:dob "1955-06-08"] }
- **Cluster 2:** { [ex1:TimBL ex1:occ "Web developer"],
[ex2:TimBL ex2:job ex2:WebDev]↔↔
[ex2:WebDev rdfs:label "Web developer"] }
- **Cluster 3:** { [ex2:TimBL ex2:job ex2:WebDev]↔↔
[ex2:WebDev rdfs:label "Web developer"] }

In accordance to the defined similarity measure, the items of Cluster 2 have a perfect match (similarity between the strings "occupation"→"label"→"Web developer" and "occupation"→"Web developer" is 1). The items of Cluster 1 have a high similarity as the literal values match perfectly and the predicates have a partial match. The most dissimilar item is Path Feature 4 with its alternative label "profession" for ex2:job. This item ends up in its own cluster (as the number of total clusters is predefined with 3, see above).

5.2.5. Cluster Merging

After the clustering, similar string representations of path features are in the same cluster but some information is also dispersed. For example, Cluster 2 and Cluster 3 represent similar information. The data retrieval step (see Section 5.2.2) also retrieves path features that include information about related entities. For example, if we also cover the birth place of the entity "Tim Berners-Lee", via [ex1:TimBL ex1:bp ex1:London] we produce a lot of path features that differ only in factual information about London. ex2 might cover similar facts and its information might be gathered in the same clusters as the facts from ex1. This could lead to clusters like the following:

```
{ [ex1:TimBL ex1:bp ex1:London]↔↔
  [ex1:London ex1:long "-0.127"],
  [ex2:TimBL ex2:pob ex2:London]↔↔
  [ex2:London ex2:longitude "-0.1275"] }
```

We perform a merging step of this cluster with all other clusters that contain information about London, for example its label, the longitude, foundation year, etc. Relevant for the merging step is the first triple of the path features. The first triples of the path features of Cluster 2 and Cluster 3 are as follows:

⁷ R is defined as the set of all reference URIs in Section 5.2.1, L is defined as the multi-valued map between string representation and the respective path features in Section 5.2.3.

5. Towards Entity Data Fusion

- $\text{fst_triples}(\text{Cluster } 2) = \{ [\text{ex1:TimBL ex1:occ "Web developer"}], [\text{ex2:TimBL ex2:job ex2:WebDev}] \}$
- $\text{fst_triples}(\text{Cluster } 3) = \{ [\text{ex2:TimBL ex2:job ex2:WebDev}] \}$

For the merging we apply the following method: if, in terms of first triples, two clusters have a higher degree of overlap (estimated via Jaccard index, that has a range between 0 and 1) than a threshold ϵ ,⁸ the clusters are merged. In this case, with $\epsilon = 0.5$, Cluster 2 and Cluster 3 are merged:

Cluster 2: $\{ [\text{ex1:TimBL ex1:occ "Web developer"}], [\text{ex2:TimBL ex2:job ex2:WebDev}] \circ \circ [\text{ex2:WebDev rdfs:label "Web developer"}], [\text{ex2:TimBL ex2:job ex2:WebDev}] \circ \circ [\text{ex2:WebDev rdfs:label "Web developer"}] \}$

As another example, clusters that contain path features with $[\text{ex1:TimBL ex1:bp ex1:London}]$ as a first triple would also get merged. While first triples of single-hop path features such as $[\text{ex1:TimBL ex1:occ "Web developer"}]$ can occur only in multiple clusters if there are more labels for the predicate, multi-hop path features can generate a variety of different label-leaf combinations for their string representations and the first triple or—like in the example—the complete path feature can occur in multiple different clusters before the merging step.

5.2.6. Representative Selection

For each cluster, we can select two types of representatives: one general representative and one representative for each source. Both types of representatives are needed for the scenario of Figure 5.2: the general representative to represent the fact in the panel and one representative for each source that support the presented fact. Before we present the details of the representative selection approach, we need to cover how we define the term “source”. For this we tracked the provenance of each triple in the data retrieval step (see Section 5.2.2). For a specific path feature, we take the first triple: the hostname of the delivering IRI (i.e., the IRI that returns data with status 200 in case of redirects) of this triple is considered as the **source** of the path feature. The complete delivering IRI of a source representative may be used for a more detailed output (like in Figure 5.2).

Cluster representative. The cluster representative is selected in accordance to three cases:

1. If the cluster contains only one element, return the first triple of the path feature.

⁸The value of ϵ is flexible and can be adjusted within the range of 0 and 1.

2. If the cluster contains only multi-hop path features or single-hop and multi-hop path features use the first triple of each multi-hop path feature and count its occurrence in the cluster. The first triple that occurs most often in the cluster is returned as the representative.
3. If the cluster contains only single-hop path features, return the triple that has the highest similarity (see Formula 5.1) to all other triples.

In our example, the third case returns any of the two birth-date triples (as they have equal similarity to each other) for Cluster 1 and the second case returns `[ex2:TimBL ex2:job ex2:WebDev]` as a representative for Cluster 2 (the triple occurs twice). The idea of the second case is that links to other resources (multi-hop) are always better than returning a plain string (single-hop) because multi-hop path features can offer further navigation possibilities to the user. However, the single-hop path features in multi-hop clusters support the respective claim as a source. In addition, the second case returns a triple that occurs in most path features and, as such, the linked resource (i.e., `ex2:WebDev` in the example) can provide most information on the fact that is described by the cluster. The third case enables to select the most common representation among multiple candidates. For example, Wikidata provides also `"label"→"Sir Tim Berners-Lee"` for the entity and the according path feature gets clustered together with the path feature represented by `"label"→"Tim Berners-Lee"` from Wikidata⁹, IMDb, Freebase etc. The third case selects the representative that is most similar to all others and chooses the version without “Sir” in this case. For the running example, the output of the representatives would be as follows:

```
[ex1:TimBL ex1:bd "1955-06-08"],
[ex2:TimBL ex2:job ex2:WebDev]
```

For both facts, the two sources `ex1` and `ex2` can be provided as references.

Source representative. Source representatives are selected in the same way as the cluster representative with the following restriction: it is chosen as the most often occurring (2.) or similar (3.) representative from a single source (e.g., `dbpedia.org`) compared to all entries across sources.

5.2.7. Filtering / Ranking

An important aspect that we have not yet addressed is the handling of contradicting information. In general, following the open-world assumption, we consider all made claims of all sources as true. If a fact is missing in one source but occurs in another, it can be true. If, in the case of persons, different sources provide different facts about spouses, employers, and even the birth dates, we consider all of them as true. However, as a general idea, we assume that claims are more likely to be true if they are made by multiple different sources.¹⁰ In fact, the more sources support a claim, the more likely it is to be valid or

⁹Wikidata provides multiple English labels for this entity.

¹⁰Note: In a Web setting, this assumption is not necessarily correct as the sources are often not independent from each other. We discuss this matter in Section 5.4.

5. Towards Entity Data Fusion

important. In contrast, if a fact is stated only by a single source, it is considered less likely or unimportant. In the following, we will exemplify along the use case of a knowledge panel why we consider this probabilistic estimate of “truth” as sufficient.

One of the main scenarios of this work is the scenario of a trustable knowledge panel. In recent times, the lack of (a sufficient amount of) sources and the explanations why certain facts are stated in knowledge panels has led to criticism [Cav16, FG16]. With the presented entity data fusion approach, we can support the identification of additional sources for provided facts. This enables users to verify the individual sources and decide themselves whether they want to trust the claim or not. In addition, in order to enable an automatically produced trustability score, additional measures—such as PageRank [BP98] or knowledge-based trust [DGM⁺15]—can be applied on the sources for each fact.

In a similar way, additional support for the notability of facts can be estimated: the more sources support a fact about an entity, the more it is considered as important. As an example, in different Web shops, the respective data sheets of the same product may cover different aspects. Almost all shops would provide a label for the entity but only few shops may cover its number of USB ports. As such, we can consider the label as more important than the number of USB ports. This is in line with the ideas of [TCC⁺10] that present entities in this manner (ranking facts by the number of sources that support them).

Another important aspect for knowledge panels is objectivity. A single data source is highly vulnerable to cover wrong or outdated information or to miss important aspects about an entity. This can happen without any bad intentions from the operator side. The presented entity data fusion method can enable operators to identify errors or missing information. Even in case source acknowledgements or justifications for notability are undesired by a knowledge panel operator, the presented method can still be implemented as a recommender system for data curators: it would suggest facts from other sources that are not covered by the data source of the knowledge panel provider.

5.3. Experiments

In our experiments we evaluated our fusion method relative to the Sig.ma baseline established in [TCC⁺10]. We compare the coverage and the number of sources with respect to the scenario of a trustable knowledge panel (see Section 5.1). The idea is that we do not want to compare agreement on randomly selected facts but to make sure that the evaluated facts would actually be presented to the user. For this, we use the facts presented in the Google Knowledge Graph (GKG) panels. We match the facts presented by the GKG panels to output clusters of our system and Sig.ma for 80 entities. In this way, we can determine the number of Google facts for which our system or Sig.ma can provide one or more additional references, basically implementing the scenario of Figure 5.2. In the following we report on the setup and our findings.

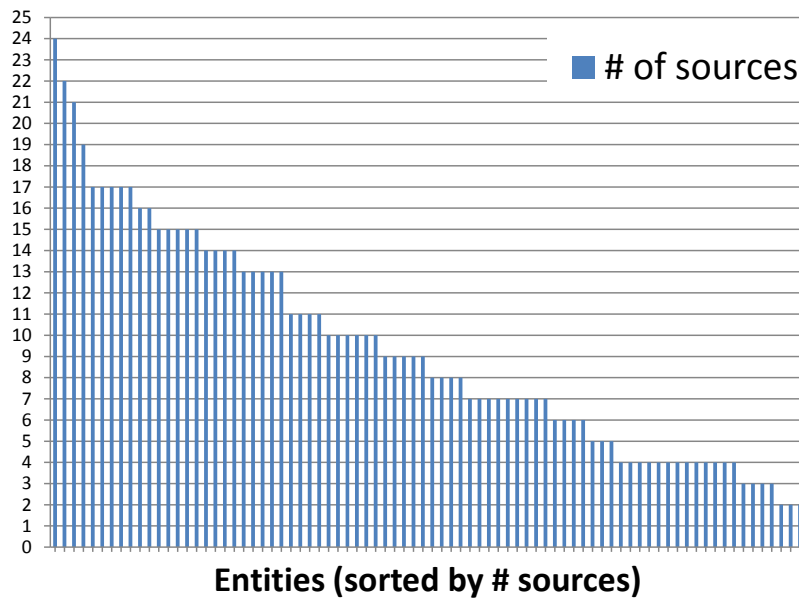


Figure 5.5.: Number of different sources for each entity (the ticks on the x-axes each represent one entity of the TREC dataset).

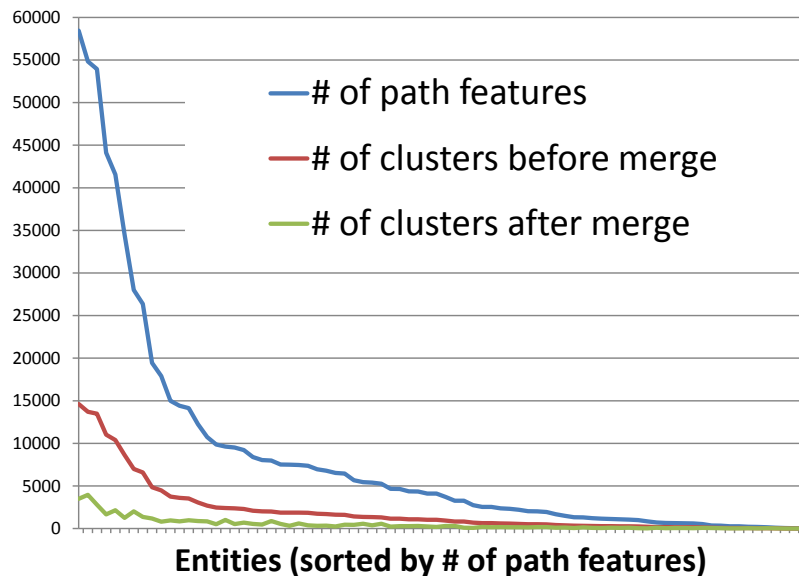


Figure 5.6.: Number of path features and clusters before/after the merging step (the ticks on the x-axes each represent one entity of the TREC dataset).

5.3.1. Dataset

The Text REtrieval Conference (TREC) entity track was last run in 2011.¹¹ We used the provided evaluation data from that year¹² and selected the entity names of both given tasks, the “related entity finding” (REF) task and the “entity list completion” (ELC) task. This

¹¹TREC tracks – <http://trec.nist.gov/tracks.html>, retrieved 2016-07-28.

¹²TREC entity track 2011 – <http://trec.nist.gov/data/entity2011.html>, retrieved 2016-07-28.

5. Towards Entity Data Fusion

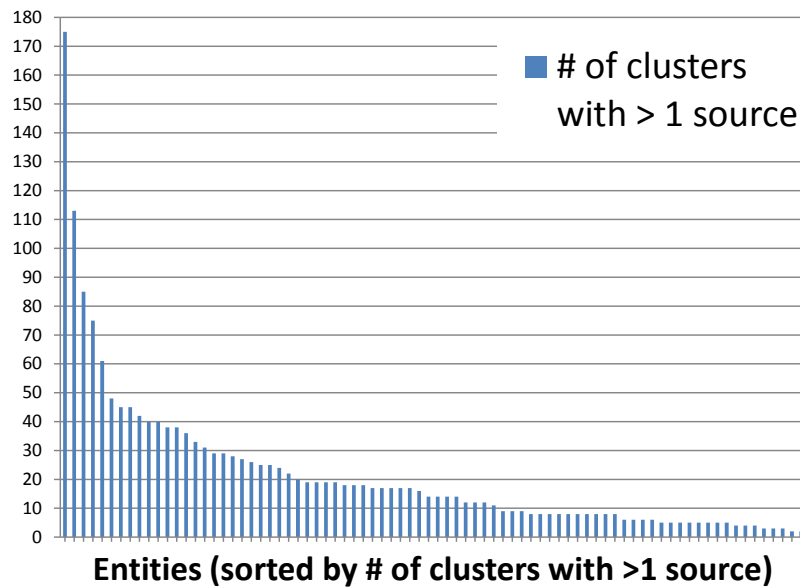


Figure 5.7.: Number of clusters with more than one source (the ticks on the x-axis each represent one entity of the TREC dataset).

produced 100 entities with two duplicates. Afterwards, we tried to identify the DBpedia IRIs for the remaining set of 98 entities. For 18 entities—such as “Landfall Foundation” or “Foundation Morgan horses”—we could not find an according DBpedia identifier (and also Google does not provide a graph panel for these entities). For 80 entities, we retrieved the DBpedia IRI. The service <http://sameAs.org> then enabled the retrieval of the according Freebase identifiers (e.g., `m/027bp7c`) and we could then retrieve Google summaries by adding the GKG API namespace <http://g.co/kg/> to these IDs, for example <http://g.co/kg/m/027bp7c>. We manually retrieved GKG panels by storing the respective HTML to files. In this context, we used <http://google.com> in English language with a clean browser history for each entity. However, we could not fully exclude that the GKG panels were produced with regard to a geo-specific context (in relation to the used IP address).

5.3.2. Baseline: Sig.ma

The Sig.ma system described in [TCC⁺10] provides basic functionality on fact alignment for entities. The approach is mostly based on string modification in order to derive a uniform representation. In particular, the provided IRIs for predicates and the IRIs/literals for objects are analyzed heuristically. The approach can not deal with n-ary statements and can only rudimentary reconcile between 0-hop and 1-hop granularity levels. However, in these cases it can serve as a baseline so we re-implemented the main ideas of Sig.ma by performing the following steps:

1. We use the predicates and objects of triples where an identifier for an entity is involved, for example:
`ex4:occupation "Web developer"@en`

2. For IRIs (in the predicate or object position) we use the last segment of the IRI (e.g., *occupation*). Typical patterns such as *camelCase* and dashes/underscores are split up. Literal values are used without further modification. Ultimately, all strings are transformed to lower case. For the alignment, the Sig.ma approach does not make use of `rdfs:label` triples [TCC⁺10].
3. These basic string representations are then aggregated with an exact match and by attributing their sources: "occupation web developer" (`http://example4.com`, `http://example5.com`)

We omitted several highly customized rules of Sig.ma such as the “[...] manually-compiled list of approximately 50 preferred terms” [TCC⁺10].

5.3.3. System Configuration

We applied the presented fusion method on 80 entities of the TREC entity dataset. We used the `http://sameas.org` service as a record linkage approach with the DBpedia identifiers as an input. Multiple crawls were performed in order to account for temporal unavailabilities of resources. The crawls happened in June 2015. The crawler operated with depth 1 and retrieved RDF data via content negotiation (an RDFa or JSON-LD functionality was not implemented). After the individual crawls were completed the retrieved data was merged. The sources included—in arbitrary order—Freebase (`www.googleapis.com`), YAGO (`yago-knowledge.org`), data from the German National Library (`d-nb.info`), DBpedia (`dbpedia.org`), the British Broadcasting Corporation (`www.bbc.co.uk`), The New York Times (`data.nytimes.com`), Geonames (`sws.geonames.org`), etc. In order to cover Freebase, the crawler used a Google API key and also included a heuristic that partly fixed the incorrect Turtle RDF syntax¹³ provided by the Freebase API. Per entity, there were 2 to 24 different sources while 60 entities included RDF information from least 5 sources. Figure 5.5 provides an overview of the number of sources per entity.

From the crawl, we extracted the path features for each entity. Big entities like “Bozeman, Montana” or “Baltimore” include more than 50.000 path features while only 12 path features could be produced for “National Summer Learning Association”. For 69 entities, the system produced more than 500 path features and the majority of 46 had between 1.000 and 10.000 path features. We then clustered the English string representations of the path features. For predicate labels, the system first tried to retrieve a label in English but also used labels with no language tag if an English label was not available. Literal labels without language tag were also included as candidates. For each entity, we computed the similarity matrix of the English string representations of all path features. For this matrix we produced the linkage and retrieved $n = |L|/4$ clusters for each entity. We merged all clusters at an overlap threshold of $\epsilon = 0.5$ (which provided good results in a set of initial trials). After this step, 55 entities had data grouped in less than 500 clusters, and 70 in less

¹³Documented at multiple sources, for example <https://github.com/RDFLib/rdfliib/issues/415>, retrieved 2016-07-28.

than 1.000 clusters. Only eight entities had less than 30 clusters after the merging step. An example for an output cluster for the entity “Montana State University” of the fusion system is provided in Table 5.1. A general overview of the distribution of the numbers of path features, clusters, and merged clusters is provided in Figure 5.6. All entities had more than two clusters with at least two sources and 47 entities had more than 10 such clusters. An overview of this distribution is provided in Figure 5.7.

5.3.4. Evaluation Setup

The evaluation included two steps, the matching of GKG facts to clusters of the output and the evaluation of the identified matches.

Step 1: Match GKG facts to clusters. For the evaluation of the quality of the results, the Google result pages and the produced output of the system needed to be aligned. The initial idea was to analyze the stored HTML pages, automatically identify Freebase facts covered by the GKG panel, and provide additional sources for the GKG facts by using the output of our approach by retrieving the cluster in which the respective Freebase fact was located. Unfortunately, although the data presented by Google is often found in Freebase, it was not possible to identify a sufficient amount of direct links. On the one hand, this was due to the incorrect Turtle RDF output produced by Freebase. On the other hand, a lot of information covered by Freebase includes n-ary relations that are presented flat in GKG panels (e.g., facts that involve the `fb:people.person.spouse_s` predicate which include temporal information, for example “married since”). This makes the automatic retrieval of Freebase facts from a GKG panel a very difficult task, especially if a variety of domains are covered (as it is the case for the TREC entities).

As a consequence we nominated two human evaluators—both experts on RDF and related technologies—and asked them to provide a manual matching. For all entities, the following was performed: For each fact that was presented in the GKG panel, use the fusion system’s output to identify clusters in which at least one source representative matched the information content of the GKG fact.

The instructions for identifying correct matches were as follows: *labels are considered with the predicate `rdfs:label`; types are considered with the predicate `rdf:type`; synopses are considered with the predicate `rdfs:comment`; compound presentations such as “Born: 1955, Addis Ababa, Ethiopia” are split into two facts (in this case “birth date” and “birth place”; if the GKG panel showed a map for a location, longitude and latitude facts in the output are considered as a match; for presented stock values, output that covers the stock symbol are considered as a match; images are not considered; time-dependent data such as events, weather information, or local time is not considered; recommendations of the type “People also search for...” are not considered.*

Step 2: Evaluation of matches. For all clusters in the output that matched a specific GKG fact, the evaluators were instructed to choose the cluster that had most correct sources (i.e., clusters where most source representatives match the information content of the GKG fact). The number of correct sources of this cluster was then documented.

Table 5.1.: Example: Cluster statistics, cluster representative, and source representatives of a cluster of the entity “Montana State University”.

Cluster statistics: Number of facts: 24 Number of sources: 5 Merged with other clusters: 0	Cluster representative: <http://dbpedia.org/resource/Montana_State_University>, http://www.w3.org/2000/01/rdf-schema#label, "Montana State University-Bozeman" .
Source:	Representative:
sws.geonames.org	<http://sws.geonames.org/5666991/> <http://www.geonames.org/ontology#name> "Montana State University" .
www.dbpedia-lite.org	<http://www.dbpedia-lite.org/things/619269#id> <http://www.w3.org/2000/01/rdf-schema#label>, "Montana State University" .
linkeddata.org	<http://linkeddata.org/triplify/node357958827> <http://www.w3.org/2000/01/rdf-schema#label>, "Montana State University" .
dbpedia.org	<http://dbpedia.org/resource/Montana_State_University> <http://www.w3.org/2000/01/rdf-schema#label>, "Montana State University-Bozeman" .
yago-knowledge.org	<http://yago-knowledge.org/resource/Montana_State_University> <http://www.w3.org/2000/01/rdf-schema#label>, "Montana State University" .

5. Towards Entity Data Fusion

Table 5.2.: Results for our approach and Sig.ma: the number of produced GKG facts, GKG coverage, number of type 1 errors, number of type 2 errors, precision, recall, and f-measure at different thresholds for the number of sources. The # symbol should be read as “number of”.

# sources in output:	≥ 1	≥ 2	≥ 3	≥ 4	≥ 5
Our approach:					
# GKG facts:	414	235	135	76	39
GKG coverage:	55%	31%	18%	10%	5%
# type 1 errors:	81	46	26	17	12
# type 2 errors:	146	81	43	26	16
Precision:	0.84	0.84	0.84	0.82	0.76
Recall:	0.74	0.74	0.76	0.75	0.71
F-measure:	0.78	0.79	0.80	0.78	0.74
Sig.ma:					
# GKG facts:	299	112	70	44	9
GKG coverage:	40%	15%	9%	6%	1%
# type 1 errors:	0	0	0	0	0
# type 2 errors:	304	151	92	57	34
Precision:	1.00	1.00	1.00	1.00	1.00
Recall:	0.50	0.43	0.43	0.44	0.21
F-measure:	0.66	0.60	0.60	0.61	0.35

In the same step the evaluators kept track of the following two types of error:

Type 1 error: Number of source representatives in the best-fit cluster that do not match the GKG fact (false positives).

Type 2 error: Number of source representatives in other clusters, that also match the information content of the GKG fact (false negatives).

Afterwards, together with the authors of this work as mediators, the evaluators consolidated annotation differences that were larger than two. If the differences were smaller than two, in case of true positives, the lower value was automatically selected by default; in case of a type 1 or type 2 error, the higher value was selected by default. As Sig.ma only considers exact matches, type 1 errors do not occur in that system (precision which leads to precision values of 1.0).

5.3.5. Evaluation Results

The evaluators identified 755 facts in the GKG panels of the 80 TREC entities. In average, each GKG panel covered 9.4 facts. Table 5.2 respectively present the main results of our approach and Sig.ma. Our data fusion method produced 414 GKG facts (with a respective coverage of 55%) and, in total, 923 source representatives. The baseline Sig.ma produced 299 GKG facts (with a respective coverage of 40%). In almost all cases our approach

outperforms Sig.ma by $\times 2$ or higher with respect to retrieving multiple sources per GKG fact (GKG coverage at ≥ 2 , ≥ 3 , etc.).

As mentioned earlier, Sig.ma only considers direct 1:1 matches which means that it produces a precision of 1.0 (there are no type 1 errors). As a side effect, this also implies a strongly reduced recall (which stems from the high number of type 2 errors). The recall levels of Sig.ma drop strongly when more than five sources are required. In contrast, our approach produces high precision and recall levels and remains fairly stable when more sources are required (the small increases/decreases are due to the varying proportion of type 1/2 errors with respect to the respective coverage). These scores are also reflected in the respective f-measure scores where our approach outperforms Sig.ma by differences from 0.12 (≥ 1 source) up to 0.39 (≥ 5 sources).

In only 22 cases out of 755, Sig.ma produced more sources than our approach. In these cases, relevant facts ended up in larger clusters that had different representatives chosen.

5.4. Discussion

The results of the experiments demonstrate the effectiveness of the entity data fusion approach. They show, that the recall is significantly improved by considering multiple granularity levels and by the approximate matching via string similarity. As a matter of fact, these factors affect the precision in a negative way, however—as the f-measure scores demonstrate—only to a point where the advantages of the improved recall have a significant overweight. In applications where precision is of ultimate importance, we would suggest an approach that utilizes direct or manually defined mappings. In the presented scenario of a trustable knowledge panel, with the factors trust, notability, and objectivity, we suggest to use the presented entity data fusion approach (which provides a highly improved recall).

Throughout our experiments, we identified a number of findings that deserve further discussion. A number of issues that we encountered deal with structured data on the Web in general: not every IRI is dereferenceable, not every IRI provides RDF data, not all returned RDF data is in (any) correct format, not all RDF data contains information about the retrieved IRI, not all RDF data contains labels, and not all RDF data contains language tags. We still made use of all these features and were able to retrieve RDF data from a number of reference IRIs (up to 24) via content-negotiation and could make sufficient use of the provided data. However, for production environments we would recommend the implementation of a data curation infrastructure that deals with some of the mentioned challenges (e.g., we assume that languages with an own alphabet—such as Korean—can be easily detected).

Another interesting aspect is the directionality of predicates. RDF triples are often used in the subject-predicate-object style but although, technically, the predicate provides a direction every such triple also provides information about the object. Unfortunately, the entity centricity of our approach seems to become a bottleneck at this point, as only few sources (DBpedia is one of them) provide information about an entity when it is in the object position of a triple. One way to circumvent this problem could be to do a full Web

5. Towards Entity Data Fusion

crawl and perform path feature extraction also for triples that use the entity IRI in the object position.

For a variety of parameters of the method, potential extension and optimization with a gold standard is possible. One particular point is literal/object similarity: Many literals are annotated by their type. For example a birth date like "1955-06-08" often has `xsd:date` defined as a data type. For a production environment, for each data type, individual similarity measures could be defined for the most common data types. Ultimately, this could be extended towards media similarity for IRIs that represent an audio file, an image, or a video. Further configuration parameters that could be optimized with a learning-to-rank approach are the following: the depth of the breadth-first search around the entity (see Section 5.2.2), n – the initial number of clusters (see Section 5.2.4), and the overlap threshold ϵ for merging clusters (see Section 5.2.5). According experiments are envisioned with an appropriate training dataset (the production of a ground truth that includes mappings between ordinary triples and n-ary statements is a non-trivial task that can not be easily crowdsourced).

The depth of the data retrieval step deserves particular attention. On the one hand, the number of path features is growing exponentially with each covered layer. On the other hand, the crawling depth = 1 (as we used it) does not retrieve all information covered by n-ary relations. For providing more details on this we revisit the example from Section 5.1:

```
[ex3:TimBL ex3:occ ex3:Work4],  
[ex3:Work4 ex3:work ex3:WebDev],  
[ex3:Work4 ex3:since "1989-03"],  
[ex3:WebDev rdfs:label "Web developer"@en]
```

With depth = 1, we would crawl the nodes `ex3:TimBL` and `ex3:Work4`. Thus, the node `ex3:WebDev` would be left untouched and the label “Web Developer” would not occur in a path feature. This also explains the gap between the number of GKG facts (755) and the number of facts for which we could identify at least one source (414). In many cases the information was present in Freebase, but covered with n-ary relations. We assume that at retrieval depths > 3 the advantage of entity centricity would get weaker as string ambiguities would be more likely to occur: “Apple” at a distance of 4 from `ex:iPhone5` could already mean the fruit (or New York City).

With increased crawling depth, the number of path features grows exponentially. As we compare path features via their string representation, and we have $|L| \cdot (|L| - 1) / 2$ comparisons, this leads to a significant demand for computation time. One solution that we consider in order to mitigate this effect is locality-sensitive hashing [IM98]. This hashing method moves similar strings to similar buckets and strongly reduces the number of candidates for traditional string comparison.

One aspect that is not addressed in this work is the question “how can we verify that the sources gathered their information *independently* from each other?”. Unfortunately, for small information units, such as facts, it is often impossible to gain a deep understanding of provenance if respective information is not explicitly given; especially if the facts

are commonly known and true. However, we assume that, if data was manually or automatically imported from other sources, the data was checked for validity. A related task was addressed in [DBES09] where the authors tackle the problem of copy detection by tracking different datasets and their change over time.

The fusion approach introduced in this chapter is complementary to the entity summarization approaches introduced in Chapter 3. While LinkSUM and UBES rely on single sources for Linked Data, the data fusion approach can mitigate several aspects that are implied by this strategy (mentioned in Section 5.1, that is trust, notability, reliability, objectivity, availability).

5.5. Related Work

Our approach relates most to the alignment and presentation method of Sig.ma by Tumarello et al. [TCC⁺10]. Sig.ma presents a rule-based, entity-centric fact alignment method that is embedded in the greater context of semantic search. As such, further components of Sig.ma include object retrieval via keyword queries, parallel data gathering, live consolidation, and presentation. The presented fact alignment approach is strongly focused on efficiency and relies on meaningful IRIs, a frequently used feature of many vocabularies and datasets. In contrast, in our approach we fully rely on `rdfs:label` and can also deal with multiple languages and opaque identifiers like they are used in Wikidata or `schema.org` (that makes strong use of blank nodes). Further, although n-ary statements are mentioned in [TCC⁺10], they are not addressed by Sig.ma. In contrast, our approach is designed to deal with fact-based information distributed over multiple hops and enables to align sources with different modeling granularities. In a more recent work, Pellissier Tanon et al. provide manually established mappings between Freebase and Wikidata [PTVS⁺16].

In the greater context, our work is related to a number of different fields. In the following we focus on the most important literature in the respective fields.

NLP knowledge base construction: The field of mining facts from natural language has recently attracted much research interest. The most prominent approaches are the Never-Ending Language Learning system [CBK⁺10] and Google’s Knowledge Vault system [DGH⁺14a]. The ambitious goal of such approaches is to learn a view of the world from Web sites such as news articles and blogs. The main difference of our approach to these approaches is that we deal with RDF data—that is already structured to some extent. We regard these approaches as complementary to our method and we could use the extracted facts of such systems as an input for our system. In addition, we do not try to identify or learn one truth but rather present sources that support one claim and other sources for a different or contradicting claim.

Data/Knowledge fusion: In [DGH⁺14b], Dong et al. define knowledge fusion as the problem of constructing a large knowledge base from unstructured data (like Web tables or natural language text) with different extractors from different sources. In contrast, data fusion is defined as the processing of a source-feature matrix for each entity where the

5. Towards Entity Data Fusion

entries mark the actual values. Our work lies between these two extremes as we deal with data for which we do not need extractors but the complexity of the data goes beyond database-like tables as we need to deal with a different identifiers and different modeling approaches. As reported in [DGH⁺14a] only a subset of entity types of human-annotated data is used for Google’s Knowledge Vault via manual mappings from `schema.org` to Freebase. The focus of our work is exactly on this type of human-annotated data—not only from `schema.org`—and its particularities (different identifiers, different schema). In [DGH⁺14b], annotated data is used but it remains unclear how the data was processed. The work on knowledge-based trust by Dong et al. [DGM⁺15] is also very relevant for our work. The authors estimate the trust-worthiness of Web sources by extracting information and verifying its correctness. With this method, a trust value is computed for each Web source. In contrast, we try to identify multiple occurrences of the same or similar fact. However, the methods complement each other and we could use the approach of Dong et al. [DGM⁺15] to compute the trustworthiness of the sources that we provide in our output.

Record linkage: The field of record linkage is of high importance as our entity data fusion approach relies on input from record linkage systems. Herzig et al. [HMBT13] make use of language models in combination with more descriptive features such as `rdfs:label` and `rdfs:comment` in order to integrate Web data on-the-fly from uncooperative environments. An earlier work by Hogan et al. [HHD07] uses features that are unique to a specific entity in order to identify equivalent IRIs. In general, most record linkage approaches focus on identifying features of entities and leverage these features in order to link the records [KSS06].

Schema/Ontology alignment: The field of schema and ontology alignment has been very active in the past decade. Most relevant to our work is the approach by Suchanek et al. [SAS11], that integrates relations, instances, and schemas. The authors use a probabilistic model to integrate each of the mentioned aspects. The approach is tested with the YAGO, DBpedia, and IMDb knowledge bases. In contrast, in our work, we account for different granularities at the modeling level and also match complete facts. Further, we test our approach in a real-world scenario with real data from the Web. The authors explicitly mention n-ary relations as an open topic that they could not address. The authors of [HCZQ11] investigate the problem of the large amount of different vocabularies. They state the question: “How Matchable Are Four Thousand Ontologies on the Semantic Web?” Although we do not explicitly deal with the merging of different vocabularies, our clustering approach could be used to mine complex mapping rules for vocabulary terms: Aligning many different entities with our entity data fusion approach, patterns about the predicates and chains of predicates that occur frequently in the same cluster can be established. These patterns can then be transformed to mapping rules.

5.6. Conclusions

We have introduced a novel entity-centric approach for fusing facts from multiple Web sources. Our approach works without any prior knowledge about the used vocabularies

and just uses core features of the RDF data model. We demonstrated two key features of the approach: the entity-centricity (which enables the application of string similarity measures for clustering) and the robustness of the approach against fine or coarse-grained RDF data modeling. In our experiments, we compared our system to the Sig.ma baseline and demonstrated that our system produces higher coverage, recall, and f-measure scores (with respect to the scenario of a trustable knowledge panel). For 31% of the facts that Google presents in its knowledge panel we can provide at least 2 sources.

Our conclusions can be summarized as follows:

1. Entity centricity is an important aspect for efficient and effective fusion of structured entity data on the Web. It resolves aspects of (string) ambiguity and significantly reduces the search space. The provided fusion approach is agnostic with respect to all involved vocabularies.
2. We presented the first fact alignment approach that is designed to merge Linked Data facts across different modeling granularities. To the best of our knowledge, it is also the first instance-based approach that serves complex alignments of vocabularies.
3. With the results of our experiments, we demonstrated that the fact alignment approach outperforms the Sig.ma [TCC⁺10] baseline in the scenario of a trustable knowledge panel. As such, we consider the approach as a next major step towards fully automatic merging of entities and facts across the Web.

Yet, there are open points that can be addressed in extensions of this work:

- For effective alignment of facts across the Web, an appropriate data pre-processing pipeline is required. This includes detection of missing language tags, literal data types, and (semi-) automatic ways to provide missing labels.
- With increased crawling depths, the number of path feature comparisons needs to be reduced. For this, locality-sensitive hashing [IM98] can be used for initial groupings of similar strings.
- The similarity measures can be fine tuned towards languages and data types. In principle, they can also address further media types such as images or videos.
- Similar to [SAS11], the clustering approach can be augmented by a rule learning system that detects frequent vocabulary alignment patterns and feeds this information back to the similarity measure. Thus, the system could be extended to learn complex alignments in an iterative way.

6. Conclusions

With the rising amount of structured data on the Web and its strong versatility, it becomes more and more important to present summaries of entities in a way that adapts to the entities' individual particularities. With this work, we aimed to address this aspect, known as entity summarization. In particular, the main goals of this thesis were to provide the following contributions:

1. Lightweight entity summarization techniques based on limited background knowledge.
2. Human and machine-readable interfaces for entity summaries.
3. Entity-centric Linked Data integration across knowledge bases.

We answered three research questions with four individual contributions (that we summarized in Figure 1.3).

In the following sections, we first discuss the contributions of this thesis in Section 6.1 and in Section 6.2 we present the overall conclusions. Finally, we provide an overview about open topics in Section 6.3.

6.1. Discussion of Contributions

Research Question 1. *How can we effectively summarize entities with limited background information?*

This research question was motivated by the gap between relevance-oriented summarization approaches that only use a (single) knowledge base (i.e., [CTQ11, SPSS10]) and approaches, that use a unique wealth of background information, in particular Google [Sin12]. Background information that lies in between these extremes—none and an (almost) unreachable wealth—can be of many different shapes. We focused on two particular examples: link structure that is publicly available, and usage data, that is available to any entity summarization service (with users). This led us to two subquestions:

Research Question 1.1. *How can we use link analysis effectively in order to derive summaries of entities?*

Contribution 1. We developed the LinkSUM system which includes different features for link-based entity summarization: PageRank [BP98] and Backlink [WS11] for resource selection and a combination of frequency, exclusivity, and description for relation selection. The approach is based on hyperlinks as background information. We evaluated

6. Conclusions

the approach in comparison to another state-of-the-art entity summarization approach FACES [GTS15]—that has been shown to outperform [CTQ11]—with respect to the SERP scenario and demonstrated significant improvement. In our evaluation, we found that entity summarization systems should primarily focus on the relevance and the strength of the connection of the related resources (as LinkSUM does). The selection of the “best” relation states a second challenge. It is important to mention that redundancies on the resource level should be avoided (this also means relations that can be inferred via reasoning, such as cities that are located in countries). Another important aspect that we addressed with the work on this research question is the general ranking of entities in RDF knowledge bases. We found that the link structure provided by Wikipedia can help to rank entities in common RDF knowledge bases, such as DBpedia, Freebase, and Wikidata.

Answer to Research Question 1.1. Link structure can effectively be used via a combination of the PageRank [BP98] algorithm and the Backlink [WS11] method. Adding link structure as background information produced better results than other state-of-the-art research approaches [CTQ11, GTS15].

Research Question 1.2. *How can we use usage data analysis effectively in order to derive summaries of entities?*

Contribution 2. With our contribution UBES, we introduced a lightweight but effective entity summarization system that builds on the techniques of item-based collaborative filtering [SKKR01] and tf-idf. In the line of Web usage mining [LMN11], we used co-consumption patterns in combination with semantic links in knowledge bases for producing summaries of entities. In addition, we introduced a game-based approach that enables to produce a ground truth for the evaluation of entity summarization. We used the derived game statistics for comparing UBES to summaries of Google Knowledge Graph (GKG) [Sin12] and—with UBES and GKG performing better than the random baseline in the task of fact ranking—provided indications for the assumption, that the scenario of a game-based ground truth could suit the task of entity summarization. However, further experiments are needed to support this indication.

Answer to Research Question 1.2. Usage data can be used for entity summarization by applying the traditional techniques of item-based collaborative filtering [SKKR01] in combination with data from the knowledge base and a tf-idf-based ranking scheme. With the performed experiments we could not directly show efficiency but we provided a promising methodology that can help future systems with the difficult task of evaluation.

Research Question 2. *Is there a minimum set of re-occurring/common features of entity summarization systems that allows us to provide a generic API?*

This research question was motivated by the need of a uniform API for entity summarization. Earlier approaches—in particular Fresnel [PBKL06]—define presentation patterns for knowledge bases. Unfortunately, such patterns are only suitable when they are defined for specific classes. A common API for entity summarization can enable sharing, system-independent presentation, evaluation, and reuse of entity summaries.

Contribution 3. We identified and verified a set of re-occurring/common features of entity summarization systems. From these features, we derived requirements on which basis

we developed an according API definition. Further, we evaluated the API in accordance to the visible output of existing real world systems. In particular, the API is suitable to solve different interoperability issues for the task of evaluation. We demonstrated that quantitative evaluation can benefit from shared scores, qualitative evaluation can benefit from uniform interfaces (that can be combined in a flexible manner), and also A/B testing can be implemented with the developed API. A reference implementation serves as a proof of concept.

Answer to Research Question 2. With the SUMMA API, its empirical evaluation, and its implementation we provided sufficient evidence about the feasibility of a generic API for entity summarization. As such, we provided an entity-centric counterpart to the class-centric Fresnel approach [PBKL06].

Research Question 3. *How can we align duplicate/similar facts about Linked Data entities on the Web?*

This research question was motivated by the need of holistic fact integration with respect to data about entities. This enables entity summarization across knowledge bases. State-of-the-art approaches that focus on parts of this problem are typically ontology-alignment approaches, that only align vocabulary terms (typically in a one-to-one way) [CH13, SAS11], or record linkage approaches [HHD07, HMBT13], that match identifiers (i.e., IRIs). Only Tummarello et al. [TCC⁺10] provided a basic heuristic for fact alignment.

Contribution 4. We developed an approach that enables the efficient and effective fusion of structured entity data on the Web. The entity-centric strategy solves problems of (string) ambiguity and strongly reduces the search space. The approach was designed in a robust manner such that issues of different modeling granularities can be addressed along with the alignment of different vocabularies. We compared the approach to a baseline established by Tummarello et al. [TCC⁺10]. Although the authors envisioned a similar interface, the fusion approach that we developed can be considered as seminal work for large-scale fact alignment for structured entity data on the Web—across modeling granularities. The fact alignment approach can provide solutions that address trust, notability, reliability, objectivity, and availability in the context of entity summarization.

Answer to Research Question 3. With our contribution on entity data fusion, we provided further steps towards answering the research question. In particular, we could identify two main enablers that can make the task feasible and successful: 1) entity-centricity for reducing string ambiguities; 2) the use path features for enabling robustness against modeling granularities.

In summary, we provided answers to all three research questions. In the following subsection, we give an overview on how the individual contributions were integrated with internal and external applications.

6.1.1. Integration of Contributions

This thesis includes four main contributions (see Figure 1.3). Although they were developed independently—all with the aim to contribute to the state of the art in entity summarization—

6. Conclusions

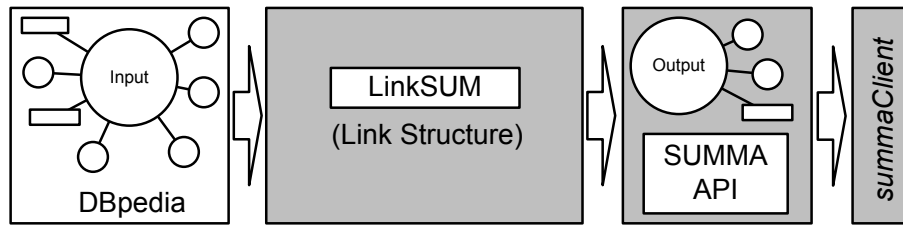


Figure 6.1.: Overview of the internally integrated contributions of this thesis.

we integrated part of the contributions with internal and external applications:

- LinkSUM [TLR16] and the SUMMA API [TS15] are fully integrated: The output of LinkSUM is available to be consumed by any SUMMA client. The reference implementation *summaClient* can make full use of the structured output of LinkSUM (see Figure 6.1).
- The DBpedia PageRank scores [TR16b] are integrated in DBpedia. The scores are available in a separate graph at the official DBpedia SPARQL endpoint.
- The *summaClient* application is integrated with DBpedia Spotlight [DJHM13] via the ELES method [TR16a].
- We implemented a specific version of *summaClient* that includes Wikipedia abstracts and figures via the API of the DuckDuckGo¹ search engine.

A combination of all of the above integration efforts is presented with the ELES demonstrator depicted by Figure 6.2. This constitutes an implementation of the “Annotated Hypertext” scenario of Section 1.1.1.

The UBES approach is not integrated (as of July 2016): In the above setting, the usage data collected by LinkSUM could be directly analyzed with the UBES approach; and the UBES summaries could be fed back to the system via the SUMMA API: an aggregation service (that also implements the SUMMA API as a client as well as a server) would combine the results of LinkSUM and UBES. Ultimately, LinkSUM and UBES would work in symbiosis where LinkSUM, on the one hand, would mitigate UBES’ new item problem [AT05] and UBES, on the other hand, would feed more dynamic data to the system. Unfortunately, as of July 2016, the uptake of LinkSUM is still too small and mostly restricted to research applications that often do not provide meaningful consumption patterns.

As of July 2016, the integration of the entity data fusion approach is still an open task. At this point, we would like to highlight two main enhancements that entity-centric fact alignment can bring: 1) enable trust by presenting “source support” directly to the user (e.g., “this fact is covered by four further sources, please click here to verify”); 2) calibrate and justify notability of facts via the number of sources that provide them. However, due to the described complexity of the task, issues with data quality, and the lack of standardized evaluation techniques, this line of research is still in early stages.

¹DuckDuckGo – <https://duckduckgo.com/>, retrieved 2016-07-31.

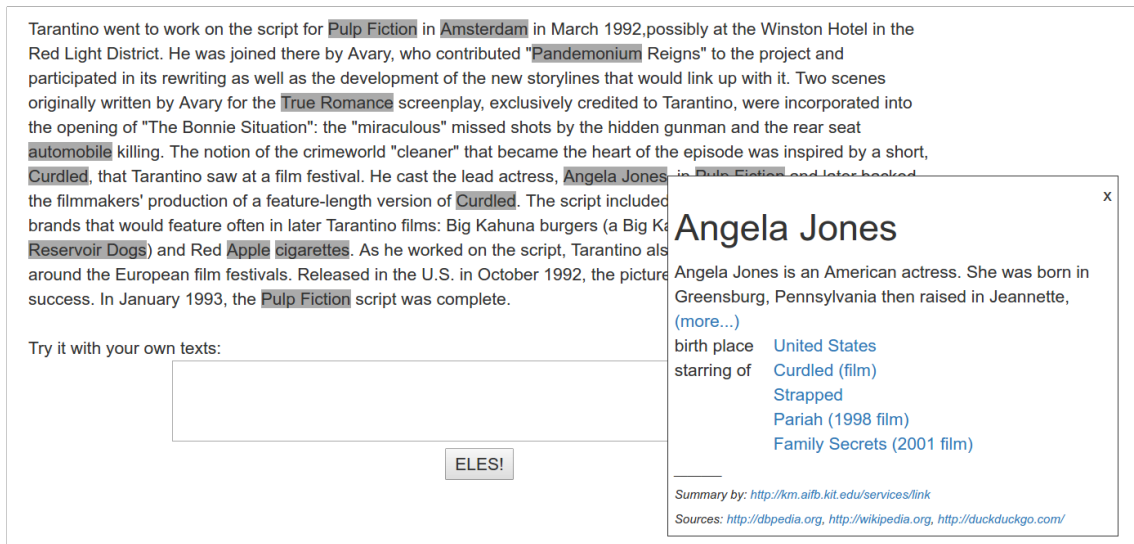


Figure 6.2.: Screenshot of the ELES demonstrator, a combination of internally and externally integrated components. The service is available online at <http://people.aifb.kit.edu/ath/ELES/> (as of July 2016). (Source of the annotated text: https://en.wikipedia.org/w/index.php?title=Pulp_Fiction&oldid=743523099, retrieved 2016-10-13.)

6.2. Overall Conclusions

We motivated this work with Bush’s vision on “memex” [Bus45]. By considering the context of the Web, its structure, and its content, we developed his ideas further towards the vision of an universal knowledge graph and emphasized on its constant growth and its inherent versatility. When an entity is retrieved (via search engines), the growth (and eventually its size) of the knowledge graph is the motivation for the concise presentation of individual entities and its versatility clarifies the need for tailoring the presentations towards each entity by considering its individual particularities. This motivates the need and the importance of the field of entity summarization.

With this work, we aimed to address three main aspects about entity summarization:

First, there is a large space between the type of information a knowledge base provides (typically “only” an RDF graph) and the type of background information that search engines can utilize for entity summarization. With our works on entity summarization, in particular LinkSUM and UBES, we aimed to address this space by demonstrating that the systems produce better results if all available background information is used. Our presented approaches provide a first step towards improving entity summarization with limited background information.

Second, except LinkSUM, all current² publicly available entity summarization systems are available only through their rendered user interfaces. There is currently no way of retrieving structured data from any online entity summarization system. With SUMMA,

²As of August 2016.

6. Conclusions

we broke with this practice and provide a common interface that can be adopted by any entity summarization system. In this way, we set the foundations for open summarization servers and clients that can share, present, evaluate, and remix entity summaries in novel ways.

Third, entities are described in different knowledge bases across the Web. In each of these sources, their data is modeled with different vocabularies and at different levels of granularity. While relying on a single source commonly simplifies the (already complex) task of entity summarization, there are also drawbacks that concur with entity summaries on single knowledge bases: reduced trust, lack of support for notability, quality issues, subjectivity, and a single point of failure. With our work on entity data fusion, we drafted an outline and demonstrated the feasibility of integrating facts about entities from different Web sources. This was done in a way that does not rely on any specific vocabulary or even modeling granularity.

In summary, this thesis can be considered as a next cornerstone in the field of entity summarization. We presented systems that utilize different types of background information, offer their summaries as an open service in order to share, present, evaluate, and remix summaries of entities, and initiated work on automatic integration of knowledge about Linked Data entities on the Web.

6.3. Outlook

For our future work, we plan to apply the introduced technologies to further knowledge bases, in particular Wikidata. In the month of July 2016, the Wikidata knowledge base had more than 6500 users with at least five edits [Zac16]. As a matter of fact, this number has been growing in recent years. We outlined earlier that Wikidata provides a rich source for structured data about entities, covers many languages, and—in case of important events—is updated in near-real time. Especially the high language coverage as well as the possibility for direct integration of new knowledge pose important advantages over DBpedia (with respect to entity summaries). Thus, as an intermediate step, we aim to migrate our integrated entity summarization methods from the DBpedia knowledge base to Wikidata before we move to a solution that is completely based on Web data (that includes data from the Web as well as data from all major RDF knowledge bases). In this setting, the use of RDF allows us to address multiple natural languages. The final goal is to operate a cross-lingual entity data integration and summarization engine available to everyone. This engine can then be used in question answering and Web search settings.

In order to achieve this, we also need to address different limitations of the introduced contributions.

Further integration In the previous section we reported about the status of integration. An open challenge is the full integration of entity data fusion and entity summarization. Entity summarization can highly benefit from integrated knowledge from the Web. However, there are a number of steps that need to be undertaken in order to

fully integrate entity data fusion and entity summarization: 1) implement a large-scale Web crawling infrastructure; 2) implement a rigorous data curation chain; 3) optimize the presented knowledge integration techniques; 4) extend the presented entity summarization approaches for a full Web graph setting; 5) combine the notions of entity-centric fact ranking and notability via source; 6) provide means for presenting facts from multiple sources in an interface for a seamless user experience.

Evaluation The field of entity summarization is currently emerging and, in a world with a constant growth of knowledge, becoming increasingly important: According to comScore, in February 2016, search engines received 16,8 billion “explicit core searches” (searches which were not triggered by previously given context) from desktop home and work devices in the United States (which means about 579 million searches per day).³ Knowing that about 40% of these queries focus on one particular entity [PMZ10] and assuming that a summary can be shown for every entity, roughly 232 million entity summaries are shown on desktop devices every day in the United States (by search engines). In that respect, it is important that summarization systems select the most relevant knowledge for presentation to the end users. However, little is currently known about “what makes good summaries” and which evaluation techniques or standard evaluation datasets are most effective. This is a non-trivial task as summaries, like rankings, are often subjective and can depend on the user’s background knowledge or context.

Literal values Both of the presented entity summarization methods, LinkSUM and UBES, provide summaries with respect to other related resources. This is due to each algorithms’ individual selection approach that does not apply to literal values (in the case of LinkSUM), or only very limited (in the case of UBES). However, as of July 2016, about 30% of all relations in DBpedia mark specific⁴ descriptive literals. Thus, it is important also to include literal values that describe an entity, such as birth dates (for people), release dates (for books/movies), heights/lengths (for buildings), etc. As a matter of fact, an easy way to include literal values is to derive predicate statistics about instances of specific classes (e.g., percentage of persons that have a birth date in DBpedia), and to use thresholds for including this knowledge.⁵ The summaries of LinkSUM and UBES can easily be augmented with such literal patterns.

Personalized/context/task-specific summaries In this work we focused on informative/general summaries of entities. A complementary and relatively open field are personalized/context/task-specific entity summarization systems. In particular, we could not identify any methods that completely focus on personalized entity summarization. Also context-specific summaries have—so far—only been addressed by Google [Bro12]. Task-specific summaries have been addressed by the following

³comScore February 2016 U.S. Desktop Search Engine – <https://www.comscore.com/Insights/Rankings/comScore-Releases-February-2016-US-Desktop-Search-Engine-Rankings>, retrieved 2016-07-13.

⁴Literal values that are not one of the following: `rdfs:label`, `rdfs:comment`, or `dbo:abstract`.

⁵See Section 2.2.2 for more information on this methodology.

6. Conclusions

recent works that focus on the tasks of entity co-reference resolution and human-centered entity linking: [XCQ14, CXQ15a, CXQ15b]. Yet, there are still many open points to be addressed in this field that can easily lead to further subfields of entity summarization. The work of this thesis can serve as a foundation for future research efforts in this domain.

Abstract summaries A completely open—yet interesting—field are abstract summaries [Man01] of entities. In this work we only tackle extract summaries in which systems select the most important aspects of existing data. In a narrow sense, abstract entity summarization aggregates over specific predicates, such as “this movie has 35 actors”. However, this view can be extended and aggregations can be combined with multi-hop aspects, such as “... founded Apple Inc. together with two co-founders, one of which sold his shares eleven days after Apple’s founding date for 800\$” in a summary for the entity “Steve Jobs”. This way of entity summarization can involve reasoning and machine learning techniques together with standard numerical operations (such as addition, subtraction, or averaging) that are defined for different data types, such as date or integer.

The evolution of Linked Data on the Web has shown an increasing demand for more data and more structure. At the same time, more than ever, users are confronted with large amounts of information that they need to understand in short periods of time. Relating to these two factors, we expect the demand for entity summarization systems to rise strongly in the upcoming years.

Bibliography

- [AATC14] Ahmad Assaf, Ghislain A. Atemezing, Raphaël Troncy, and Elena Cabrio. What Are the Important Properties of an Entity? Comparing Users and Knowledge Graph Point of View. In *The Semantic Web: ESWC 2014 Satellite Events*, Lecture Notes in Computer Science, pages 190–194. Springer International Publishing, Cham, 2014.
- [ABK⁺07] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. DBpedia: A Nucleus for a Web of Open Data. In *The Semantic Web*, volume 4825 of *Lecture Notes in Computer Science*, pages 722–735. Springer Berlin Heidelberg, 2007.
- [AGHI09] Rakesh Agrawal, Sreenivas Gollapudi, Alan Halverson, and Samuel Jeong. Diversifying Search Results. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, WSDM '09, pages 5–14. ACM, New York, NY, USA, 2009.
- [AIS93] Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. Mining Association Rules Between Sets of Items in Large Databases. In *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, SIGMOD '93, pages 207–216. ACM, New York, NY, USA, 1993.
- [AL07] Sören Auer and Jens Lehmann. What Have Innsbruck and Leipzig in Common? Extracting Semantics from Wiki Content. In *The Semantic Web: Research and Applications: 4th European Semantic Web Conference, ESWC 2007, Innsbruck, Austria, June 3-7, 2007. Proceedings*, pages 503–517. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.
- [AT05] Gediminas Adomavicius and Alexander Tuzhilin. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749, June 2005.
- [ATM14] Fahad Alahmari, James A. Thom, and Liam Magee. A model for ranking entity attributes using DBpedia. *Aslib Journal of Information Management*, 66(5):473–493, 2014.
- [BBH15] Hannah Bast, Björn Buchhold, and Elmar Haussmann. Relevance Scores for Triples from Type-Like Relations. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '15, pages 243–252. ACM, New York, NY, USA, 2015.

Bibliography

- [BCMT13] Roi Blanco, Berkant Barla Cambazoglu, Peter Mika, and Nicolas Torzec. Entity Recommendations in Web Search. In *The Semantic Web – ISWC 2013: 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part II*, pages 33–48. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.
- [BEP⁺08] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: A Collaboratively Created Graph Database for Structuring Human Knowledge. In *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, SIGMOD '08*, pages 1247–1250. ACM, New York, NY, USA, 2008.
- [BG14] Dan Brickley and Ramanathan V. Guha. RDF Schema 1.1. W3C recommendation – <http://www.w3.org/TR/2014/REC-rdf-schema-20140225/>, W3C, February 2014. Accessed October 28, 2015.
- [BGMD08] Stephan Bloehdorn, Marko Grobelnik, Peter Mika, and Thanh Tran Douc, editors. *Proceedings of the Workshop on Semantic Search (SemSearch 2008) at the 5th European Semantic Web Conference (ESWC 2008)*, volume 334 of *CEUR Workshop Proceedings*. CEUR-WS.org, May 2008.
- [BHBL09] Christian Bizer, Tom Heath, and Tim Berners-Lee. Linked Data - The Story So Far. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 5(3):1–22, 2009.
- [BHH⁺13] Roi Blanco, Harry Halpin, Daniel M. Herzig, Peter Mika, Jeffrey Pound, Henry S. Thompson, and Thanh Tran. Repeatable and reliable semantic search evaluation. *Web Semantics: Science, Services and Agents on the World Wide Web*, 21:14–29, 2013.
- [BHLT06] Tim Bray, Dave Hollander, Andrew Layman, and Richard Tobin. Namespaces in XML 1.1 (Second Edition). W3C recommendation – <https://www.w3.org/TR/2006/REC-xml-names11-20060816/>, W3C, August 2006. Accessed August 1, 2016.
- [BHP04] Andrey Balmin, Vagelis Hristidis, and Yannis Papakonstantinou. Object-Rank: Authority-Based Keyword Search in Databases. In *Proceedings of the Thirtieth International Conference on Very Large Data Bases - Volume 30, VLDB '04*, pages 564–575. VLDB Endowment, 2004.
- [BHS02] Bettina Berendt, Andreas Hotho, and Gerd Stumme. Towards Semantic Web Mining. In *The Semantic Web – ISWC 2002: First International Semantic Web Conference Sardinia, Italy, June 9–12, 2002 Proceedings*, pages 264–278. Springer Berlin Heidelberg, Berlin, Heidelberg, 2002.
- [BL06] Tim Berners-Lee. Linked Data. <https://www.w3.org/DesignIssues/LinkedData.html>, July 2006. Accessed July 24, 2016.

- [BL09] Tim Berners-Lee. Read-Write Linked Data. <http://www.w3.org/DesignIssues/ReadWriteLinkedData.html>, August 2009. Accessed July 24, 2016.
- [BLFM05] Tim Berners-Lee, Roy Fielding, and Larry Masinter. Uniform Resource Identifier (URI): Generic Syntax. Standards track – <https://tools.ietf.org/rfc/rfc2396.txt>, IETF, January 2005. Accessed May 10, 2016.
- [BLHL01] Tim Berners-Lee, James Hendler, and Ora Lassila. The Semantic Web. *Scientific American*, 284(5):29–37, May 2001.
- [BMV11] Roi Blanco, Peter Mika, and Sebastiano Vigna. Effective and Efficient Entity Search in RDF Data. In *The Semantic Web – ISWC 2011: 10th International Semantic Web Conference, Bonn, Germany, October 23-27, 2011, Proceedings, Part I*, pages 83–97. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.
- [BP98] Sergey Brin and Lawrence Page. The Anatomy of a Large-scale Hypertextual Web Search Engine. In *Proceedings of the Seventh International Conference on World Wide Web 7, WWW7*, pages 107–117. Elsevier Science Publishers B. V., Amsterdam, The Netherlands, The Netherlands, 1998.
- [Bro12] Aaron Brown. Get smarter answers from the Knowledge Graph from Portuguese to Japanese to Russian. https://search.googleblog.com/2012/12/get-smarter-answers-from-knowledge_4.html, December 2012. Accessed March 25, 2016.
- [BUNN15] Lorenz Bühmann, Ricardo Usbeck, and Axel-Cyrille Ngonga Ngomo. ASSESS — Automatic Self-Assessment Using Linked Data. In *The Semantic Web - ISWC 2015: 14th International Semantic Web Conference, Bethlehem, PA, USA, October 11-15, 2015, Proceedings, Part II*, pages 76–89. Springer International Publishing, Cham, 2015.
- [Bus45] Vannevar Bush. As We May Think. *The Atlantic*, July 1945. <http://www.theatlantic.com/magazine/archive/1945/07/as-we-may-think/303881/>. Accessed July 13, 2016.
- [BV05] Paolo Boldi and Sebastiano Vigna. MG4J at TREC 2005. In *The Fourteenth Text REtrieval Conference (TREC 2005) Proceedings*, number SP 500-266 in Special Publications. NIST, 2005. <http://mg4j.di.unimi.it/>.
- [BV14] Paolo Boldi and Sebastiano Vigna. Axioms for Centrality. *Internet Mathematics*, 10(3-4):222–262, 2014.
- [BWS16] Tamara Bobić, Jörg Waitelonis, and Harald Sack. FRanCo - A Ground Truth Corpus for Fact Ranking Evaluation. In *Joint Proceedings of the 1st International Workshop on Summarizing and Presenting Entities and Ontologies and the 3rd International Workshop on Human Semantic Web Interfaces (SumPre 2015, HSWI 2015) co-located with the 12th Extended*

- Semantic Web Conference (ESWC 2015), Portoroz, Slovenia, June 1, 2015.*, volume 1556 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2016.
- [BYD04] Ricardo Baeza-Yates and Emilio Davis. Web Page Ranking Using Link Attributes. In *Proceedings of the 13th International World Wide Web Conference on Alternate Track Papers & Posters*, WWW Alt. '04, pages 328–329. ACM, New York, NY, USA, 2004.
- [Cav16] Amy Cavanaugh. You probably haven't even noticed Google's sketchy quest to control the world's knowledge. <https://www.washingtonpost.com/news/the-intersect/wp/2016/05/11/you-probably-havent-even-noticed-googles-sketchy-quest-to-control-the-worlds-knowledge/>, May 2016. Accessed July 29, 2016.
- [CBK⁺10] Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka, and Tom M. Mitchell. Toward an architecture for never-ending language learning. In *AAAI Conference on Artificial Intelligence*. AAAI, 2010.
- [CBK11] Iván Cantador, Peter Brusilovsky, and Tsvi Kuflik. 2nd Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec 2011). In *Proceedings of the 5th ACM Conference on Recommender systems*, RecSys 2011, New York, NY, USA, 2011. ACM.
- [CG98] Jaime Carbonell and Jade Goldstein. The Use of MMR, Diversity-based Reranking for Reordering Documents and Producing Summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, pages 335–336. ACM, New York, NY, USA, 1998.
- [CGT⁺16] Gong Cheng, Kalpa Gunaratna, Andreas Thalhammer, Heiko Paulheim, Martin Voigt, and Roberto García, editors. *Joint Proceedings of the 1st International Workshop on Summarizing and Presenting Entities and Ontologies and the 3rd International Workshop on Human Semantic Web Interfaces (SumPre 2015, HSWI 2015) co-located with the 12th Extended Semantic Web Conference (ESWC 2015), Portoroz, Slovenia, June 1, 2015.*, volume 1556 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2016.
- [CH13] Michelle Cheatham and Pascal Hitzler. String Similarity Metrics for Ontology Alignment. In *The Semantic Web – ISWC 2013: 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part II*, pages 294–309. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.
- [CTQ11] Gong Cheng, Thanh Tran, and Yuzhong Qu. RELIN: Relatedness and Informativeness-Based Centrality for Entity Summarization. In *The Semantic Web – ISWC 2011: 10th International Semantic Web Conference, Bonn, Germany, October 23-27, 2011, Proceedings, Part I*, pages 114–129. Springer Berlin Heidelberg, Berlin, Heidelberg, October 2011.

- [CWL14] Richard Cyganiak, David Wood, and Markus Lanthaler. RDF 1.1 Concepts and Abstract Syntax. W3C recommendation – <http://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/>, W3C, February 2014. Accessed October 20, 2015.
- [CXQ15a] Gong Cheng, Danyun Xu, and Yuzhong Qu. C3D+P: A summarization method for interactive entity resolution. *Web Semantics: Science, Services and Agents on the World Wide Web*, 35, Part 4:203–213, May 2015.
- [CXQ15b] Gong Cheng, Danyun Xu, and Yuzhong Qu. Summarizing Entity Descriptions for Effective and Efficient Human-centered Entity Linking. In *Proceedings of the 24th International Conference on World Wide Web, WWW '15*, pages 184–194. ACM, New York, NY, USA, May 2015.
- [CYC07] Tao Cheng, Xifeng Yan, and Kevin Chen-Chuan Chang. EntityRank: Searching Entities Directly and Holistically. In *Proceedings of the 33rd International Conference on Very Large Data Bases, VLDB '07*, pages 387–398. VLDB Endowment, 2007.
- [DB95] Simon A. Dobson and Victoria A. Burrill. Lightweight databases. *Computer Networks and ISDN Systems*, 27(6):1009–1015, 1995.
- [DBES09] Xin Luna Dong, Laure Berti-Equille, and Divesh Srivastava. Truth Discovery and Copying Detection in a Dynamic World. *Proc. VLDB Endow.*, 2(1):562–573, August 2009.
- [Dem06] Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7:1–30, January 2006.
- [DFDM12] Lorand Dali, Blaž Fortuna, Thanh Tran Duc, and Dunja Mladenić. Query-Independent Learning to Rank for RDF Entity Search. In *The Semantic Web: Research and Applications: 9th Extended Semantic Web Conference, ESWC 2012, Heraklion, Crete, Greece, May 27-31, 2012. Proceedings*, pages 484–498. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
- [DFJ⁺04] Li Ding, Tim Finin, Anupam Joshi, Rong Pan, R. Scott Cost, Yun Peng, Pavan Reddivari, Vishal Doshi, and Joel Sachs. Swoogle: A Search and Metadata Engine for the Semantic Web. In *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management, CIKM '04*, pages 652–659. ACM, New York, NY, USA, 2004.
- [DGH⁺14a] Xin Dong, Evgeniy Gabrilovich, Jeremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmman, Shaohua Sun, and Wei Zhang. Knowledge Vault: A Web-scale Approach to Probabilistic Knowledge Fusion. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14*, pages 601–610. ACM, New York, NY, USA, 2014.
- [DGH⁺14b] Xin Luna Dong, Evgeniy Gabrilovich, Jeremy Heitz, Wilko Horn, Kevin Murphy, Shaohua Sun, and Wei Zhang. From Data Fusion to Knowledge Fusion. *Proc. VLDB Endow.*, 7(10):881–892, June 2014.

Bibliography

- [DGM⁺15] Xin Luna Dong, Evgeniy Gabrilovich, Kevin Murphy, Van Dang, Wilko Horn, Camillo Lugaresi, Shaohua Sun, and Wei Zhang. Knowledge-based Trust: Estimating the Trustworthiness of Web Sources. *Proc. VLDB Endow.*, 8(9):938–949, May 2015.
- [Dij59] Edsger W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1(1):269–271, December 1959.
- [DJHM13] Joachim Daiber, Max Jakob, Chris Hokamp, and Pablo N. Mendes. Improving Efficiency and Accuracy in Multilingual Entity Extraction. In *Proceedings of the 9th International Conference on Semantic Systems, I-SEMANTICS '13*, pages 121–124. ACM, New York, NY, USA, 2013.
- [DNMO⁺12] Tommaso Di Noia, Roberto Mirizzi, Vito Claudio Ostuni, Davide Romito, and Markus Zanker. Linked Open Data to Support Content-based Recommender Systems. In *Proceedings of the 8th International Conference on Semantic Systems, I-SEMANTICS '12*, pages 1–8. ACM, New York, NY, USA, 2012.
- [DPF⁺05] Li Ding, Rong Pan, Tim Finin, Anupam Joshi, Yun Peng, and Pranam Kolari. Finding and Ranking Knowledge on the Semantic Web. In *The Semantic Web – ISWC 2005: 4th International Semantic Web Conference, ISWC 2005, Galway, Ireland, November 6-10, 2005. Proceedings*, pages 156–170. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005.
- [DR11] Aba-Sah Dadzie and Matthew Rowe. Approaches to visualising Linked Data: A survey. *Semantic Web*, 2(2):89–124, 2011.
- [DRF⁺09] Lorand Dali, Delia Rusu, Blaž Fortuna, Dunja Mladenić, and Marko Grobelnik. Question Answering Based on Semantic Graphs. In *Proceedings of the Workshop on Semantic Search (SemSearch 2009) at the 18th International World Wide Web Conference (WWW 2009)*, 2009.
- [DRM⁺08] Debabrata Dash, Jun Rao, Nimrod Megiddo, Anastasia Ailamaki, and Guy Lohman. Dynamic Faceted Search for Discovery-driven Analysis. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management, CIKM '08*, pages 3–12, New York, NY, USA, 2008. ACM.
- [DRPN16] Fariz Darari, Simon Razniewski, Radityo Eko Prasajo, and Werner Nutt. Enabling Fine-Grained RDF Data Completeness Assessment. In *Web Engineering: 16th International Conference, ICWE 2016, Lugano, Switzerland, June 6-9, 2016. Proceedings*, pages 170–187. Springer International Publishing, Cham, 2016.
- [DS05] Martin Duerst and Michel Suignard. Internationalized Resource Identifiers (IRIs). Standards track – <https://www.ietf.org/rfc/rfc3987.txt>, IETF, January 2005. Accessed Mai 10, 2016.

- [DSLS16] Dimitar Dimitrov, Philipp Singer, Florian Lemmerich, and Markus Strohmaier. Visual Positions of Links and Clicks on Wikipedia. In *Proceedings of the 25th International Conference Companion on World Wide Web*, WWW '16 Companion, pages 27–28. International World Wide Web Conferences Steering Committee, 2016.
- [DSW⁺00] Arthur J. Duineveld, Roeland Stoter, Marcel R. Weiden, Bisente Kenepa, and V. Richard Benjamins. WonderTools? A comparative study of ontological engineering tools. *International Journal of Human-Computer Studies*, 52(6):1111–1133, 2000.
- [DTC⁺10] Renaud Delbru, Nickolai Toupikov, Michele Catasta, Giovanni Tummarello, and Stefan Decker. Hierarchical Link Analysis for Ranking Web Data. In *The Semantic Web: Research and Applications: 7th Extended Semantic Web Conference, ESWC 2010, Heraklion, Crete, Greece, May 30 – June 3, 2010, Proceedings, Part II*, pages 225–239. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.
- [Dun93] Ted Dunning. Accurate Methods for the Statistics of Surprise and Coincidence. *Computational Linguistics*, 19(1):61–74, March 1993.
- [DVBV⁺16] Laurens De Vocht, Christian Beecks, Ruben Verborgh, Erik Mannens, Thomas Seidl, and Rik Van de Walle. Effect of Heuristics on Serendipity in Path-Based Storytelling with Linked Data. In *Human Interface and the Management of Information: Information, Design and Interaction: 18th International Conference, HCI International 2016 Toronto, Canada, July 17-22, 2016, Proceedings, Part I*, pages 238–251. Springer International Publishing, Cham, 2016.
- [EAL⁺15] Young-Ho Eom, Pablo Aragón, David Laniado, Andreas Kaltenbrunner, Sebastiano Vigna, and Dima L. Shepelyansky. Interactions of Cultures and Top People of Wikipedia from Ranking of 24 Language Editions. *PLoS ONE*, 10(3):1–27, Marchdo 2015.
- [EGK⁺14] Fredo Erxleben, Michael Günther, Markus Krötzsch, Julian Mendez, and Denny Vrandečić. Introducing Wikidata to the Linked Data Web. In *The Semantic Web – ISWC 2014: 13th International Semantic Web Conference, Riva del Garda, Italy, October 19-23, 2014. Proceedings, Part I*, number 8796 in Lecture Notes in Computer Science, pages 50–65. Springer International Publishing, 2014.
- [EH14] Basil Ell and Andreas Harth. A language-independent method for the extraction of RDF verbalization templates. In *Proceedings of the 8th International Natural Language Generation Conference (INLG)*, pages 26–34. Association for Computational Linguistics, Philadelphia, Pennsylvania, U.S.A., June 2014.
- [Ehr06] Marc Ehrig. *Ontology Alignment – Bridging the Semantic Gap*. PhD thesis, Universität Karlsruhe (TH), Fakultät für Wirtschaftswissenschaften, Karlsruhe, 2006.

Bibliography

- [EHS⁺02] Jennifer English, Marti Hearst, Rashmi Sinha, Kirsten Swearingen, and Ka-Ping Yee. Flexible Search and Navigation using Faceted Metadata. Technical report, University of Berkeley, 2002.
- [Fak08] Georgios John Fakas. Automated generation of object summaries from relational databases: A novel keyword searching paradigm. In *Data Engineering Workshop, 2008. ICDEW 2008. IEEE 24th International Conference on*, pages 564–567. IEEE, April 2008.
- [Fak11] Georgios John Fakas. A novel keyword search paradigm in relational databases: Object summaries. *Data & Knowledge Engineering*, 70(2):208–229, 2011.
- [FBMR17] Michael Färber, Frederic Bartscherer, Carsten Menne, and Achim Rettinger. Linked Data Quality of DBpedia, Freebase, OpenCyc, Wikidata, and YAGO. *Semantic Web*, 2017. to appear.
- [FC09] Georgios John Fakas and Zhi Cai. Ranking of Object Summaries. In *2009 IEEE 25th International Conference on Data Engineering*, pages 1580–1583. IEEE, March 2009.
- [FCM11] Georgios John Fakas, Zhi Cai, and Nikos Mamoulis. Size-*l* Object Summaries for Relational Keyword Search. *Proc. VLDB Endow.*, 5(3):229–240, November 2011.
- [FCM14] Georgios John Fakas, Zhi Cai, and Nikos Mamoulis. Versatile Size-*l* Object Summaries for Relational Keyword Search. *IEEE Transactions on Knowledge and Data Engineering*, 26(4):1026–1038, April 2014.
- [FCM15] Georgios John Fakas, Zhi Cai, and Nikos Mamoulis. Diverse and Proportional Size-*l* Object Summaries for Keyword Search. In *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, SIGMOD '15*, pages 363–375. ACM, New York, NY, USA, 2015.
- [FDES98] Dieter Fensel, Stefan Decker, Michael Erdmann, and Rudi Studer. Ontobroker: The Very High Idea. In *Proceedings of the 11th International Flairs Conference (FLAIRS-98), Sanibal Island, Florida, May 1998.*, pages 131–135. AAAI, 1998.
- [Fel12] Christiane Fellbaum. WordNet. In *The Encyclopedia of Applied Linguistics*. John Wiley & Sons, Inc., November 2012.
- [FG16] Heather Ford and Mark Graham. *Code and the City*, chapter Semantic Cities: Coded Geopolitics and the Rise of the Semantic Web, pages 200–214. Routledge, 2016.
- [FGM⁺99] Roy Fielding, Jim Gettys, Jeff Mogul, Henrik Frystyk, Larry Masinter, Paul Leach, and Tim Berners-Lee. Hypertext Transfer Protocol – HTTP/1.1. Standards track – <https://www.ietf.org/rfc/rfc2616.txt>, IETF, June 1999. Accessed February 04, 2016.

- [FHLW00] Dieter Fensel, Jim Hendler, Henry Lieberman, and Wolfgang Wahlster. Dagstuhl-seminar: Semantics for the WWW. Seminar report – <http://www.dagstuhl.de/fileadmin/files/Reports/00/00121.pdf>, Dagstuhl Seminar Reports, March 2000. Accessed May 13, 2016.
- [Fis87] Douglas H. Fisher. Knowledge Acquisition Via Incremental Conceptual Clustering. *Machine Learning*, 2(2):139–172, September 1987.
- [FMG11] Blaž Fortuna, Dunja Mladenić, and Marko Grobelnik. User Modeling Combining Access Logs, Page Content and Semantics. In *Proceedings of the 1st International Workshop on Usage Analysis and the Web of Data (USEWOD 2011) held in conjunction with the 20th International World Wide Web Conference (WWW2011), Hyderabad, India, March 28th, 2011*, volume abs/1103.5002, 2011.
- [FML⁺13] David Filip, Shaun McCance, Dave Lewis, Christian Lieske, Arle Lommel, Jirka Kosek, Felix Sasaki, and Yves Savourel. Internationalization Tag Set (ITS) Version 2.0. W3C recommendation – <http://www.w3.org/TR/2013/REC-its20-20131029/>, W3C, October 2013. Accessed October 28, 2015.
- [FPSH14] Peter F. Patel-Schneider and Patrick J. Hayes. RDF 1.1 Semantics. W3C recommendation – <http://www.w3.org/TR/2014/REC-rdf11-mt-20140225/>, W3C, February 2014. Accessed October 20, 2015.
- [FR14] Roy Fielding and Julian Reschke. Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content. Standards track – <https://tools.ietf.org/rfc/rfc7231.txt>, IETF, June 2014. Accessed November 19, 2015.
- [FSSS09] Thomas Franz, Antje Schultz, Sergej Sizov, and Steffen Staab. TripleRank: Ranking Semantic Web Data by Tensor Decomposition. In *The Semantic Web - ISWC 2009: 8th International Semantic Web Conference, ISWC 2009, Chantilly, VA, USA, October 25-29, 2009. Proceedings*, pages 213–228. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.
- [GBM15] Ramanathan V. Guha, Dan Brickley, and Steve MacBeth. Schema.Org: Evolution of Structured Data on the Web. *Queue*, 13(9):10:10–10:37, November 2015.
- [GDS14] Anja Gruenheid, Xin Luna Dong, and Divesh Srivastava. Incremental Record Linkage. *Proc. VLDB Endow.*, 7(9):697–708, May 2014.
- [GMDW09] Marko Grobelnik, Peter Mika, Thanh Tran Douc, and Haofen Wang, editors. *Proceedings of the Workshop on Semantic Search (SemSearch 2009) at the 18th International World Wide Web Conference (WWW 2009)*, volume 491 of *CEUR Workshop Proceedings*. CEUR-WS.org, August 2009.

Bibliography

- [GMDW10] Marko Grobelnik, Peter Mika, Thanh Tran Douc, and Haofen Wang, editors. *SEMSEARCH '10: Proceedings of the 3rd International Semantic Search Workshop*, New York, NY, USA, 2010. ACM.
- [GMM03] Ramanathan V. Guha, Rob McCool, and Eric Miller. Semantic Search. In *Proceedings of the 12th International Conference on World Wide Web, WWW '03*, pages 700–709. ACM, New York, NY, USA, 2003.
- [GN14] Daniel Gerber and Axel-Cyrille Ngonga Ngomo. *Towards the Multilingual Semantic Web: Principles, Methods and Applications*, chapter From RDF to Natural Language and Back, pages 193–209. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014.
- [GS14] Fabien Gandon and Guus Schreiber. RDF 1.1 XML Syntax. W3C recommendation – <http://www.w3.org/TR/2014/REC-rdf-syntax-grammar-20140225/>, W3C, February 2014. Accessed October 20, 2015.
- [GTS15] Kalpa Gunaratna, Krishnaparasad Thirunarayan, and Amit Sheth. FACES: Diversity-Aware Entity Summarization Using Incremental Hierarchical Conceptual Clustering. In *AAAI Conference on Artificial Intelligence*. AAAI, February 2015.
- [GTSC16] Kalpa Gunaratna, Krishnaprasad Thirunarayan, Amit Sheth, and Gong Cheng. Gleaning Types for Literals in RDF Triples with Application to Entity Summarization. In *The Semantic Web. Latest Advances and New Domains: 13th International Conference, ESWC 2016, Heraklion, Crete, Greece, May 29 – June 2, 2016, Proceedings*, pages 85–100. Springer International Publishing, Cham, 2016.
- [Har10] Andreas Harth. VisiNav: A system for visual search and navigation on web data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 8(4):348–354, 2010.
- [HB15] Silviu Homoceanu and Wolf-Tilo Balke. A Chip Off the Old Block - Extracting Typical Attributes for Entities Based on Family Resemblance. In *Database Systems for Advanced Applications: 20th International Conference, DASFAA 2015, Hanoi, Vietnam, April 20-23, 2015, Proceedings, Part I*, pages 493–509. Springer International Publishing, Cham, 2015.
- [HCZQ11] Wei Hu, Jianfeng Chen, Hang Zhang, and Yuzhong Qu. How Matchable Are Four Thousand Ontologies on the Semantic Web. In *The Semantic Web: Research and Applications: 8th Extended Semantic Web Conference, ESWC 2011, Heraklion, Crete, Greece, May 29-June 2, 2011, Proceedings, Part I*, pages 290–304. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.
- [HEE⁺02] Marti Hearst, Ame Elliott, Jennifer English, Rashmi Sinha, Kirsten Swearingen, and Ka-Ping Yee. Finding the Flow in Web Site Search. *Communications of the ACM*, 45(9):42–49, September 2002.

- [HG12] Rakebul Hasan and Fabien Gandon. Explanation in the Semantic Web: a survey of the state of the art. Research Report RR-7974, Inria, 2012.
- [HHD06] Aidan Hogan, Andreas Harth, and Stefan Decker. ReConRank: A Scalable Ranking Method for Semantic Web Data with Context. In *Proceedings of 2nd International Workshop on Scalable Semantic Web Knowledge Base Systems (SSWS 2006) held in conjunction with the 5th International Semantic Web Conference 2006 (ISWC 2006), Athens, GA, USA, November 5th, 2006*, 2006.
- [HHD07] Aidan Hogan, Andreas Harth, and Stefan Decker. Performing Object Consolidation on the Semantic Web Data Graph. In *Proceedings of 1st I3: Identity, Identifiers, Identification Workshop co-located with the 16th International World Wide Web Conference (WWW2007), Banff, Alberta, Canada, 2007*.
- [HHK15] Daniel Hernández, Aidan Hogan, and Markus Krötzsch. Reifying RDF: What Works Well With Wikidata? In *Proceedings of the 11th International Workshop on Scalable Semantic Web Knowledge Base Systems*, volume 1457 of *CEUR Workshop Proceedings*, pages 32–47. CEUR-WS.org, 2015.
- [HK09] Andreas Harth and Sheila Kinsella. TOPDIS: Tensor-based Ranking for Data Search and Navigation. Technical report, DERI, 2009.
- [HK15] F. Maxwell Harper and Joseph A. Konstan. The MovieLens Datasets: History and Context. *ACM Trans. Interact. Intell. Syst.*, 5(4):19:1–19:19, December 2015.
- [HKB⁺13] Jörn Hees, Mohamed Khamis, Ralf Biedert, Slim Abdennadher, and Andreas Dengel. Collecting links between entities ranked by human association strengths. In *The Semantic Web: Semantics and Big Data: 10th International Conference, ESWC 2013, Montpellier, France, May 26-30, 2013. Proceedings*, pages 517–531. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.
- [HKD09] Andreas Harth, Sheila Kinsella, and Stefan Decker. Using Naming Authority to Rank Data and Ontologies for Web Search. In *The Semantic Web - ISWC 2009: 8th International Semantic Web Conference, ISWC 2009, Chantilly, VA, USA, October 25-29, 2009. Proceedings*, pages 277–292. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.
- [HLS10] Philipp Heim, Steffen Lohmann, and Timo Stegemann. Interactive Relationship Discovery via the Semantic Web. In *The Semantic Web: Research and Applications: 7th Extended Semantic Web Conference, ESWC 2010, Heraklion, Crete, Greece, May 30 – June 3, 2010, Proceedings, Part I*, pages 303–317. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.
- [HMBT13] Daniel M. Herzig, Peter Mika, Roi Blanco, and Thanh Tran. Federated Entity Search Using On-the-Fly Consolidation. In *The Semantic Web – ISWC*

2013: *12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part I*, pages 167–183. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.

- [HMS⁺05] Eero Hyvönen, Eetu Mäkelä, Mirva Salminen, Arttu Valo, Kim Viljanen, Samppa Saarela, Miikka Junnila, and Suvi Kettula. MuseumFinland – Finnish museums on the semantic web. *Web Semantics: Science, Services and Agents on the World Wide Web*, 3(2–3):224–241, 2005.
- [Hox14] Julia Hoxha. *Cross-domain Recommendations based on semantically-enhanced User Web Behavior*. PhD thesis, Karlsruher Institut für Technologie, Fakultät für Wirtschaftswissenschaften, Karlsruhe, 2014.
- [HRBB⁺12] Jörn Hees, Thomas Roth-Berghofer, Ralf Biedert, Benjamin Adrian, and Andreas Dengel. *Search Computing*, chapter BetterRelations: Collecting Association Strengths for Linked Data Triples with a Game, pages 223–239. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
- [HS13] Steve Harris and Andy Seaborne. SPARQL 1.1 Query Language. W3C recommendation – <http://www.w3.org/TR/2013/REC-sparql11-query-20130321/>, W3C, March 2013. Accessed November 09, 2015.
- [HUH⁺12] Aidan Hogan, Jürgen Umbrich, Andreas Harth, Richard Cyganiak, Axel Polleres, and Stefan Decker. An empirical survey of Linked Data conformance. *Web Semantics: Science, Services and Agents on the World Wide Web*, 14:14–44, 2012.
- [IM98] Piotr Indyk and Rajeev Motwani. Approximate nearest neighbors: Towards removing the curse of dimensionality. In *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing, STOC '98*, pages 604–613. ACM, New York, NY, USA, 1998.
- [JHD13] Amit Krishna Joshi, Pascal Hitzler, and Guozhu Dong. Logical Linked Data Compression. In *The Semantic Web: Semantics and Big Data*, volume 7882, pages 170–184. Springer Berlin Heidelberg, 2013.
- [JW04] Ian Jacobs and Norman Walsh. Architecture of the World Wide Web, Volume One. W3C recommendation – <https://www.w3.org/TR/webarch/>, W3C, December 2004. Accessed February 2, 2016.
- [KCN06] Christian Kohlschütter, Paul-Alexandru Chirita, and Wolfgang Nejdl. Efficient Parallel Computation of PageRank. In *Advances in Information Retrieval: 28th European Conference on IR Research, ECIR 2006, London, UK, April 10-12, 2006.*, pages 241–252. Springer, 2006.
- [Ken38] Maurice G. Kendall. A New Measure of Rank Correlation. *Biometrika*, 30(1/2):81–93, 1938.
- [KH14] Tobias Käfer and Andreas Harth. Billion Triples Challenge data set. <http://km.aifb.kit.edu/projects/btc-2014/>, August 2014. Accessed July 21, 2016.

- [KHS12] Magnus Knuth, Johannes Hercher, and Harald Sack. Collaboratively Patching Linked Data. In *Proceedings of the 2nd International Workshop on Usage Analysis and the Web of Data (USEWOD 2012) held in conjunction with the 21st International World Wide Web Conference (WWW 2012), Lyon, France, April 17th, 2012*, volume abs/1204.2715, 2012.
- [KKL11] Markus Kirchberg, Ryan K. L. Ko, and Bu-Sung Lee. From Linked Data to Relevant Data – Time is the Essence. In *Proceedings of the 1st International Workshop on Usage Analysis and the Web of Data (USEWOD 2011) held in conjunction with the 20th International World Wide Web Conference (WWW2011), Hyderabad, India, March 28th, 2011*, volume abs/1103.5046, 2011.
- [KSI06] Georgia Koutrika, Alkis Simitsis, and Yannis Ioannidis. Précis: The Essence of a Query Answer. In *22nd International Conference on Data Engineering (ICDE'06)*, pages 69–69, April 2006.
- [KSS06] Nick Koudas, Sunita Sarawagi, and Divesh Srivastava. Record Linkage: Similarity Measures and Algorithms. In *Proceedings of the 2006 ACM SIGMOD International Conference on Management of Data, SIGMOD '06*, pages 802–803. ACM, New York, NY, USA, 2006.
- [Kul15] Swapna Kulkarni. A Recommendation Engine Using Apache Spark. Master's thesis, San José State University, 2015.
- [KVV⁺07] Markus Krötzsch, Denny Vrandečić, Max Völkel, Heiko Haller, and Rudi Studer. Semantic Wikipedia. *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(4):251–261, 2007.
- [Las98] Ora Lassila. Web metadata: a matter of semantics. *IEEE Internet Computing*, 2(4):30–37, July 1998.
- [LGMF04] Jure Leskovec, Marko Grobelnik, and Natasa Milic-Frayling. Learning Substructures of Document Semantic Graphs for Document Summarization. In *Workshop on Link Analysis and Group Detection (LinkKDD) at KDD 2004, August 2004, Seattle, WA, USA., 2004*.
- [Liu11] Bing Liu. *Web Data Mining*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.
- [LM04] Vanessa Lopez and Enrico Motta. Ontology-Driven Question Answering in AquaLog. In *Natural Language Processing and Information Systems: 9th International Conference on Applications of Natural Language to Information Systems, NLDB 2004, Salford, UK, June 23-25, 2004. Proceedings*, pages 89–102. Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.
- [LMFG05] Jure Leskovec, Natasa Milic-Frayling, and Marko Grobelnik. Impact of Linguistic Analysis on the Semantic Graph Coverage and Learning of Document Extracts. In *AAAI Conference on Artificial Intelligence*. AAAI, 2005.

Bibliography

- [LMN11] Bing Liu, Bamshad Mobasher, and Olfa Nasraoui. *Web Data Mining*, chapter Web Usage Mining, pages 527–603. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.
- [LMU06] Vanessa Lopez, Enrico Motta, and Victoria Uren. PowerAqua: Fishing the Semantic Web. In *The Semantic Web: Research and Applications: 3rd European Semantic Web Conference, ESWC 2006 Budva, Montenegro, June 11-14, 2006 Proceedings*, pages 393–410. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.
- [LP11] Elena Lloret and Manuel Palomar. Text summarisation in progress: a literature review. *Artificial Intelligence Review*, 37(1):1–41, 2011.
- [LPS16] José Lages, Antoine Patt, and L. Dima Shepelyansky. Wikipedia ranking of world universities. *The European Physical Journal B*, 89(3):1–12, 2016.
- [LRABH15] Markus Luczak-Roesch, Saud Aljaloud, Bettina Berendt, and Laura Hollink. USEWOD 2016 Research Dataset. <http://eprints.soton.ac.uk/385344/>, December 2015.
- [LS58] Linné, Carl von and Salvius, Lars. *Caroli Linnaei...Systema naturae per regna tria naturae :secundum classes, ordines, genera, species, cum characteribus, differentiis, synonymis, locis.*, volume 1. Holmiae :Impensis Direct. Laurentii Salvii, 1758.
- [LSR96] Sean Luke, Lee Spector, and David Rager. Ontology-Based Knowledge Discovery on the World-Wide Web. In *Working Notes of the Workshop on Internet-Based Information Systems at the 13th National Conference on Artificial Intelligence (AAAI96)*, 1996.
- [LUCM13] Vanessa Lopez, Christina Unger, Philipp Cimiano, and Enrico Motta. Evaluating question answering over linked data. *Web Semantics Science Services And Agents On The World Wide Web*, 21:3–13, 2013.
- [Luh58] Hans Peter Luhn. The Automatic Creation of Literature Abstracts. *IBM Journal of Research and Development*, 2(2):159–165, April 1958.
- [LUMP07] Vanessa Lopez, Victoria Uren, Enrico Motta, and Michele Pasin. AquaLog: An ontology-driven question answering system for organizational semantic intranets. *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(2):72–105, 2007.
- [LWWwH13] Taesung Lee, Zhongyuan Wang, Haixun Wang, and Seung won Hwang. Attribute extraction and scoring: A probabilistic approach. In *Data Engineering (ICDE), 2013 IEEE 29th International Conference on*, pages 194–205. IEEE, April 2013.
- [Man01] Inderjeet Mani. *Automatic Summarization*, volume 3. John Benjamins Publishing, 2001.

- [MB03] Gary Marchionini and Ben Brunk. Towards a General Relation Browser: A GUI for Information Architects. *Journal of Digital Information*, 4(1), 2003.
- [MG14] Nicolas Marie and Fabien Gandon. Survey of Linked Data Based Exploration Systems. In *Proceedings of the 3rd International Conference on Intelligent Exploration of Semantic Data - Volume 1279*, volume 1279 of *IESD'14*, pages 66–77, Aachen, Germany, Germany, 2014. CEUR-WS.org.
- [MGH⁺09] Boris Motik, Bernardo Cuenca Grau, Ian Horrocks, Zhe Wu, Achille Fokoue, and Carsten Lutz. OWL 2 Web Ontology Language Profiles. W3C recommendation – <https://www.w3.org/TR/2009/REC-owl2-profiles-20091027/>, W3C, October 2009. Accessed July 03, 2016.
- [MHC⁺10] Knud Möller, Michael Hausenblas, Richard Cyganiak, Gunnar Grimnes, and Siegfried Handschuh. Learning from Linked Open Data Usage: Patterns & Metrics. In *Proceedings of the WebSci10: Extending the Frontiers of Society On-Line, April 26-27th, 2010, Raleigh, NC, USA*. Web Science Overlay Journal, 2010.
- [Mil67] Stanley Milgram. The Small-World Problem. *Psychology Today*, 1(1):61–67, May 1967.
- [MPB14] Robert Meusel, Petar Petrovski, and Christian Bizer. The WebDataCommons Microdata, RDFa and Microformat Dataset Series. In *The Semantic Web – ISWC 2014: 13th International Semantic Web Conference, Riva del Garda, Italy, October 19-23, 2014. Proceedings, Part I*, pages 277–292. Springer International Publishing, Cham, 2014.
- [NBS14] Vinh Nguyen, Olivier Bodenreider, and Amit Sheth. Don't Like RDF Reification?: Making Statements About Statements Using Singleton Property. In *Proceedings of the 23rd International Conference on World Wide Web, WWW '14*, pages 759–770. ACM, New York, NY, USA, 2014.
- [NFM00] Natalya F. Noy, Ray W. Ferguson, and Mark A. Musen. The Knowledge Model of Protégé-2000: Combining Interoperability and Flexibility. In *Knowledge Engineering and Knowledge Management Methods, Models, and Tools*, volume 1937 of *Lecture Notes in Computer Science*, pages 17–32. Springer Berlin Heidelberg, Berlin, Heidelberg, 2000.
- [NM01] Natalya F. Noy and Deborah L. McGuinness. Ontology Development 101: A Guide to Creating Your First Ontology. Stanford Medical Informatics Technical Report SMI-2001-0880, Stanford University, March 2001.
- [Nor06] Peter Norvig. Peter Norvig forecasts the future. *New Scientist*, November 2006. <https://www.newscientist.com/article/mg19225780-098-peter-norvig-forecasts-the-future/>. Accessed July 13, 2016.

Bibliography

- [NPG⁺17] Andrea Giovanni Nuzzolese, Valentina Presutti, Aldo Gangemi, Silvio Peroni, and Paolo Ciancarini. Aemoo: Linked Data Exploration based on Knowledge Patterns. *Semantic Web*, 8(1):87–112, 2017.
- [NZWM05] Zaiqing Nie, Yuanzhi Zhang, Ji-Rong Wen, and Wei-Ying Ma. Object-level Ranking: Bringing Order to Web Objects. In *Proceedings of the 14th International Conference on World Wide Web, WWW '05*, pages 567–574. ACM, New York, NY, USA, 2005.
- [OBHG03] Daniel Oberle, Bettina Berendt, Andreas Hotho, and Jorge Gonzalez. Conceptual User Tracking. In *Advances in Web Intelligence: First International AtlanticWeb Intelligence Conference, AWIC 2003, Madrid, Spain, May 5–6, 2003. Proceedings*, pages 155–164. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003.
- [ODD06] Eyal Oren, Renaud Delbru, and Stefan Decker. Extending Faceted Navigation for RDF Data. In *The Semantic Web - ISWC 2006: 5th International Semantic Web Conference, ISWC 2006, Athens, GA, USA, November 5-9, 2006. Proceedings*, pages 559–572. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.
- [OGD07] Eyal Oren, Sebastian Gerke, and Stefan Decker. Simple Algorithms for Predicate Suggestions Using Similarity and Co-occurrence. In *The Semantic Web: Research and Applications: 4th European Semantic Web Conference, ESWC 2007, Innsbruck, Austria, June 3-7, 2007. Proceedings*, pages 160–174. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.
- [Pal15] Gurpreet Pall. Bing announces availability of the knowledge and action graph API. <https://blogs.bing.com/search/2015/08/20/bing-announces-availability-of-the-knowledge-and-action-graph-api-for-developers/>, August 2015. Accessed July 18, 2016.
- [Pas10] Alexandre Passant. dbrec — Music Recommendations Using DBpedia. In *The Semantic Web – ISWC 2010: 9th International Semantic Web Conference, ISWC 2010, Shanghai, China, November 7-11, 2010, Revised Selected Papers, Part II*, pages 209–224. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.
- [PB13] Heiko Paulheim and Christian Bizer. Type Inference on Noisy RDF Data. In *The Semantic Web – ISWC 2013: 12th International Semantic Web Conference, Sydney, NSW, Australia, October 21-25, 2013, Proceedings, Part I*, pages 510–525. Springer Berlin Heidelberg, Berlin, Heidelberg, 2013.
- [PBA13] Eric Prud'hommeaux and Carlos Buil-Aranda. SPARQL 1.1 Federated Query. W3C recommendation – <https://www.w3.org/TR/2013/REC-sparql11-federated-query-20130321/>, W3C, March 2013. Accessed July 25, 2016.

- [PBKL06] Emmanuel Pietriga, Christian Bizer, David Karger, and Ryan Lee. Fresnel: A Browser-Independent Presentation Vocabulary for RDF. In *The Semantic Web - ISWC 2006: 5th International Semantic Web Conference, ISWC 2006, Athens, GA, USA, November 5-9, 2006. Proceedings*, pages 158–171. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.
- [PC14] Eric Prud’hommeaux and Gavin Carothers. RDF 1.1 Turtle. W3C recommendation – <http://www.w3.org/TR/2014/REC-turtle-20140225/>, W3C, February 2014. Accessed October 20, 2015.
- [PD09] Addison Phillips and Mark Davis. Tags for Identifying Languages. Best current practice – <http://tools.ietf.org/html/bcp47>, IETF, September 2009. Accessed October 29, 2015.
- [PGM⁺12] David Peterson, Shudi Gao, Ashok Malhotra, C. M. Sperberg-McQueen, and Henry S. Thompson. W3C XML Schema Definition Language (XSD) 1.1 Part 2: Datatypes. W3C recommendation – <http://www.w3.org/TR/2012/REC-xmlschema11-2-20120405/>, W3C, April 2012. Accessed October 29, 2015.
- [Pis07] David Pisinger. The quadratic knapsack problem – a survey. *Discrete Applied Mathematics*, 155(5):623–648, 2007.
- [PMd08] Silvio Peroni, Enrico Motta, and Mathieu d’Aquin. Identifying Key Concepts in an Ontology, through the Integration of Cognitive Principles with Statistical and Topological Measures. In *The Semantic Web: 3rd Asian Semantic Web Conference, ASWC 2008, Bangkok, Thailand, December 8-11, 2008. Proceedings.*, pages 242–256. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.
- [PMZ10] Jeffrey Pound, Peter Mika, and Hugo Zaragoza. Ad-hoc Object Retrieval in the Web of Data. In *Proceedings of the 19th International Conference on World Wide Web, WWW ’10*, pages 771–780. ACM, New York, NY, USA, 2010.
- [PTVS⁺16] Thomas Pellissier Tanon, Denny Vrandečić, Sebastian Schaffert, Thomas Steiner, and Lydia Pintscher. From Freebase to Wikidata: The Great Migration. In *Proceedings of the 25th International Conference on World Wide Web, WWW ’16*, pages 1419–1428. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland, 2016.
- [Pun12] Golan Pundak. “People also search for” – now with explanations! https://search.googleblog.com/2012/10/people-also-search-for-now-with_9451.html, October 2012. Accessed March 25, 2016.
- [PWTY08] Thomas Penin, Haofen Wang, Thanh Tran, and Yong Yu. Snippet Generation for Semantic Web Search Engines. In *The Semantic Web: 3rd Asian*

Bibliography

- Semantic Web Conference, ASWC 2008, Bangkok, Thailand, December 8-11, 2008. Proceedings.*, pages 493–507. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.
- [Qia13] Richard Qian. Understand Your World with Bing. <http://blogs.bing.com/search/2013/03/21/understand-your-world-with-bing/>, March 2013. Accessed July 25, 2016.
- [RFGM08] Delia Rusu, Blaž Fortuna, Marko Grobelnik, and Dunja Mladenić. Semantic graphs derived from triplets with application in document summarization. In *Proceedings of the Conference on Data Mining and Data Warehouses (SiKDD 2008) held in conjunction with the 11th International Multi-conference on Information Society (IS-2008)*, 2008.
- [RR07] Leonard Richardson and Sam Ruby. *RESTful Web Services*. O’Reilly Media, 2007.
- [RSP15] Petar Ristoski, Michael Schuhmacher, and Heiko Paulheim. Using Graph Metrics for Linked Open Data Enabled Recommender Systems. In *E-Commerce and Web Technologies: 16th International Conference on Electronic Commerce and Web Technologies, EC-Web 2015, Valencia, Spain, September 2015, Revised Selected Papers*, pages 30–41. Springer International Publishing, Cham, 2015.
- [RVS14] Antonio J. Roa-Valverde and Miguel-Angel Sicilia. A survey of approaches for ranking on the web of data. *Information Retrieval*, 17(4):295–325, 2014.
- [RVTTS12] Antonio J. Roa-Valverde, Andreas Thalhammer, Ioan Toma, and Miguel-Angel Sicilia. Towards a formal model for sharing and reusing ranking computations. In *Proceedings of the 6th International Workshop on Ranking in Databases (DBRank 2012) held in conjunction with the 38th Conference on Very Large Databases (VLDB 2012)*, 2012.
- [RZ09] Stephen Robertson and Hugo Zaragoza. The Probabilistic Relevance Framework: BM25 and Beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389, 2009.
- [SAS11] Fabian M. Suchanek, Serge Abiteboul, and Pierre Senellart. PARIS: Probabilistic Alignment of Relations, Instances, and Schema. *Proc. VLDB Endow.*, 5(3):157–168, November 2011.
- [SBF98] Rudi Studer, V. Richard Benjamins, and Dieter Fensel. Knowledge engineering: Principles and methods. *Data & Knowledge Engineering*, 25(1):161–197, 1998.
- [SGPD⁺04] York Sure, Asunción Gómez-Pérez, Walter Daelemans, Marie-Laure Reinberger, Nicola Guarino, and Natalya F. Noy. Why evaluate ontology technologies? Because it works! *IEEE Intelligent Systems*, 19(4):74–81, July 2004.

- [SH08] Katharina Siorpaes and Martin Hepp. Games with a Purpose for the Semantic Web. *IEEE Intelligent Systems*, 23(3):50–60, 2008.
- [SH11] Sebastian Speiser and Andreas Harth. Integrating Linked Data and Services with Linked Data Services. In *The Semantic Web: Research and Applications: 8th Extended Semantic Web Conference, ESWC 2011, Heraklion, Crete, Greece, May 29-June 2, 2011, Proceedings, Part I*, pages 170–184. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.
- [SHHS15] Philipp Singer, Denis Helic, Andreas Hotho, and Markus Strohmaier. Hyp-Trails: A Bayesian Approach for Comparing Hypotheses About Human Trails on the Web. In *Proceedings of the 24th International Conference on World Wide Web, WWW '15*, pages 1003–1013. ACM, New York, NY, USA, 2015.
- [Sin12] Amit Singhal. Introducing the Knowledge Graph: things, not strings. <http://googleblog.blogspot.com/2012/05/introducing-knowledge-graph-things-not.html>, May 2012. Accessed February 15, 2016.
- [SKKR01] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based Collaborative Filtering Recommendation Algorithms. In *Proceedings of the 10th International Conference on World Wide Web, WWW '01*, pages 285–295. ACM, New York, NY, USA, 2001.
- [SKL14] Manu Sporny, Gregg Kellogg, and Markus Lanthaler. JSON-LD 1.0. W3C recommendation – <http://www.w3.org/TR/2014/REC-json-ld-20140116/>, W3C, January 2014. Accessed November 24, 2015.
- [SPS10] Marcin Sydow, Mariusz Piękna, and Ralf Schenkel. DIVERSUM: Towards diversified summarisation of entities in knowledge graphs. In *2010 IEEE 26th International Conference on Data Engineering Workshops (ICDEW)*, pages 221–226, March 2010.
- [SPS11] Marcin Sydow, Mariusz Piękna, and Ralf Schenkel. To Diversify or Not to Diversify Entity Summaries on RDF Knowledge Graphs? In *Foundations of Intelligent Systems: 19th International Symposium, ISMIS 2011, Warsaw, Poland, June 28-30, 2011. Proceedings*, pages 490–500. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.
- [SPS13] Marcin Sydow, Mariusz Piękna, and Ralf Schenkel. The notion of diversity in graphical entity summarisation on semantic knowledge graphs. *Journal of Intelligent Information Systems*, 41(2):109–149, 2013.
- [SPSS10] Marcin Sydow, Mariusz Piękna, Ralf Schenkel, and Adam Siemion. Entity summarisation with limited edge budget on knowledge graphs. In *Proceedings of the 2010 International Multiconference on Computer Science and Information Technology (IMCSIT)*, pages 513–516, October 2010.

Bibliography

- [SR13] Tom Stocky and Lars Rasmussen. Introducing Graph Search Beta. <http://newsroom.fb.com/news/2013/01/introducing-graph-search-beta/>, January 2013. Accessed July 14, 2016.
- [SRP15] Benjamin Schäfer, Petar Ristoski, and Heiko Paulheim. What is Special about Bethlehem, Pennsylvania? Identifying Unusual Facts about DBpedia Entities. In *Proceedings of the ISWC 2015 Posters & Demonstrations Track*, volume 1486 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2015.
- [SSHS13] Steffen Stadtmüller, Sebastian Speiser, Andreas Harth, and Rudi Studer. Data-Fu: A Language and an Interpreter for Interaction with Read/Write Linked Data. In *Proceedings of the 22nd International Conference on World Wide Web, WWW '13*, pages 1225–1236. ACM, New York, NY, USA, 2013.
- [TCC⁺10] Giovanni Tummarello, Richard Cyganiak, Michele Catasta, Szymon Danielczyk, Renaud Delbru, and Stefan Decker. Sig.ma: Live views on the Web of Data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 8(4):355–364, 2010.
- [TCG16] Andreas Thalhammer, Gong Cheng, and Kalpa Gunaratna, editors. *Proceedings of the 2nd International Workshop on Summarizing and Presenting Entities and Ontologies (SumPre 2016) co-located with the 13th Extended Semantic Web Conference (ESWC 2016), Anissaras, Greece, May 30, 2016.*, volume 1605 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2016.
- [THS09] Thanh Tran, Peter Haase, and Rudi Studer. Semantic Search – Using Graph-Structured Semantic Models for Supporting the Search Process. In *Conceptual Structures: Leveraging Semantic Technologies: 17th International Conference on Conceptual Structures, ICCS 2009, Moscow, Russia, July 26-31, 2009. Proceedings*, pages 48–65. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009.
- [TKDP15] Georgia Troullinou, Haridimos Kondylakis, Evangelia Daskalaki, and Dimitris Plexousakis. RDF Digest: Efficient Summarization of RDF/S KBs. In *The Semantic Web. Latest Advances and New Domains: 12th European Semantic Web Conference, ESWC 2015, Portoroz, Slovenia, May 31 – June 4, 2015. Proceedings*, pages 119–134. Springer International Publishing, Cham, 2015.
- [TKS12] Andreas Thalhammer, Magnus Knuth, and Harald Sack. Evaluating Entity Summarization Using a Game-Based Ground Truth. In *The Semantic Web – ISWC 2012*, volume 7650 of *Lecture Notes in Computer Science*, pages 350–361. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
- [TLR16] Andreas Thalhammer, Nelia Lasierra, and Achim Rettinger. LinkSUM: Using Link Analysis to Summarize Entity Data. In *Web Engineering: 16th International Conference, ICWE 2016, Lugano, Switzerland, June 6-9, 2016. Proceedings*, volume 9671 of *Lecture Notes in Computer Science*, pages 244–261. Springer International Publishing, Cham, 2016.

- [TM15] Nava Tintarev and Judith Masthoff. *Recommender Systems Handbook*, chapter Explaining Recommendations: Design and Evaluation, pages 353–382. Springer US, Boston, MA, 2015.
- [TMWG11] Thanh Tran, Peter Mika, Haofen Wang, and Marko Grobelnik. Sem-Search’11: The 4th Semantic Search Workshop. In *Proceedings of the 20th International Conference Companion on World Wide Web, WWW ’11*, pages 315–316. ACM, New York, NY, USA, 2011.
- [Tor14] Nicolas Torzec. Yahoo’s Knowledge Graph. <http://semtechbizsj2014.semanticweb.com/sessionPop.cfm?confid=82&proposalid=6452>, August 2014. Accessed July 25, 2016.
- [TR14] Andreas Thalhammer and Achim Rettinger. Browsing DBpedia Entities with Summaries. In *The Semantic Web: ESWC 2014 Satellite Events*, Lecture Notes in Computer Science, pages 511–515. Springer International Publishing, Cham, 2014.
- [TR16a] Andreas Thalhammer and Achim Rettinger. ELES: Combining Entity Linking and Entity Summarization. In *Web Engineering: 16th International Conference, ICWE 2016, Lugano, Switzerland, June 6-9, 2016. Proceedings*, volume 9671 of *Lecture Notes in Computer Science*, pages 547–550. Springer International Publishing, Cham, 2016.
- [TR16b] Andreas Thalhammer and Achim Rettinger. PageRank on Wikipedia: Towards General Importance Scores for Entities. In *The Semantic Web: ESWC 2016 Satellite Events, Heraklion, Crete, Greece, May 29 – June 2, 2016, Revised Selected Papers*, pages 227–240. Springer International Publishing, Cham, October 2016.
- [TS15] Andreas Thalhammer and Steffen Stadtmüller. SUMMA: A Common API for Linked Data Entity Summaries. In *Engineering the Web in the Big Data Era*, volume 9114 of *Lecture Notes in Computer Science*, pages 430–446. Springer International Publishing, Cham, 2015.
- [TSW11] Tomasz Tylenda, Mauro Sozio, and Gerhard Weikum. Einstein: Physicist or Vegetarian? Summarizing Semantic Type Graphs for Knowledge Discovery. In *Proceedings of the 20th International Conference Companion on World Wide Web, WWW ’11*, pages 273–276. ACM, New York, NY, USA, 2011.
- [TTRVF12] Andreas Thalhammer, Ioan Toma, Antonio J. Roa-Valverde, and Dieter Fensel. Leveraging Usage Data for Linked Data Movie Entity Summarization. In *Proceedings of the 2nd International Workshop on Usage Analysis and the Web of Data (USEWOD 2012) held in conjunction with the 21st International World Wide Web Conference (WWW 2012), Lyon, France, April 17th, 2012*, volume abs/1204.2718, 2012.

Bibliography

- [vEMP⁺16] Marieke van Erp, Pablo Mendes, Heiko Paulheim, Filip Ilievski, Julien Plu, Giuseppe Rizzo, and Jörg Waitelonis. Evaluating Entity Linking: An Analysis of Current Benchmark Datasets and a Roadmap for Doing a Better Job. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. European Language Resources Association (ELRA), Paris, France, May 2016.
- [VK14] Denny Vrandečić and Markus Krötzsch. Wikidata: A Free Collaborative Knowledgebase. *Communications of the ACM*, 57(10):78–85, September 2014.
- [VSD⁺11] Ruben Verborgh, Thomas Steiner, Davy Van Deursen, Rik Van de Walle, and Joaquim Gabarró Vallés. Efficient runtime service discovery and consumption with hyperlinked RESTdesc. In *Next Generation Web Services Practices (NWeSP), 2011 7th International Conference on*, pages 373–379. IEEE, October 2011.
- [VVC⁺12] Miel Vander Sande, Ruben Verborgh, Sam Coppens, Tom De Nies, Pedro Debevere, Laurens De Vocht, Pieterjan De Potter, Davy Van Deursen, Erik Mannens, and Rik Van de Walle. Everything is Connected: Using Linked Data for Multimedia Narration of Connections between Concepts. In *Proceedings of the ISWC 2012 Posters & Demonstrations Track*, volume 914 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2012.
- [VVMD03] Maria Vargas-Vera, Enrico Motta, and John Domingue. AQUA: An Ontology-Driven Question Answering System. In *AAAI Spring Symposium, New Directions in Question Answering 24-26 March 2003, Stanford University, CA, USA*. AAAI, 2003.
- [vZGMS10] Roelof van Zwol, Lluís Garcia Pueyo, Mridul Muralidharan, and Borkur Sigurbjornsson. Ranking Entity Facets Based on User Click Feedback. In *Semantic Computing (ICSC), 2010 IEEE Fourth International Conference on*, pages 192–199. IEEE, September 2010.
- [W3C12] W3C OWL Working Group. OWL 2 Web Ontology Language Document Overview (Second Edition). W3C recommendation – <http://www.w3.org/TR/2012/REC-owl2-overview-20121211/>, W3C, December 2012. Accessed July 21, 2016.
- [W3C13] The W3C SPARQL Working Group. SPARQL 1.1 Overview. W3C recommendation – <http://www.w3.org/TR/2013/REC-sparql11-overview-20130321/>, W3C, March 2013. Accessed November 09, 2015.
- [WKOS11] Lina Wolf, Magnus Knuth, Johannes Osterhoff, and Harald Sack. RISQ! Renowned Individuals Semantic Quiz: A Jeopardy Like Quiz Game for Ranking Facts. In *Proceedings of the 7th International Conference on Semantic Systems, I-Semantics '11*, pages 71–78. ACM, New York, NY, USA, 2011.

- [WL12] Robert West and Jure Leskovec. Human Wayfinding in Information Networks. In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12, pages 619–628. ACM, New York, NY, USA, 2012.
- [WLKS11] Jörg Waitelonis, Nadine Ludwig, Magnus Knuth, and Harald Sack. Who-Knows? - Evaluating Linked Data Heuristics with a Quiz that Cleans Up DBpedia. *International Journal of Interactive Technology and Smart Education*, 8(3):236–248, 2011.
- [WLT11] Andreas Wagner, Günter Ladwig, and Thanh Tran. Browsing-Oriented Semantic Faceted Search. In *Database and Expert Systems Applications: 22nd International Conference, DEXA 2011, Toulouse, France, August 29 - September 2, 2011. Proceedings, Part I*, pages 303–319. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.
- [WS11] Jörg Waitelonis and Harald Sack. Towards exploratory video search using linked data. *Multimedia Tools and Applications*, 59(2):645–672, 2011.
- [XCQ14] Danyun Xu, Gong Cheng, and Yuzhong Qu. Facilitating Human Intervention in Coreference Resolution with Comparative Entity Summaries. In *The Semantic Web: Trends and Challenges: 11th International Conference, ESWC 2014, Anissaras, Crete, Greece, May 25-29, 2014. Proceedings*, pages 535–549. Springer International Publishing, Cham, 2014.
- [YSLH03] Ka-Ping Yee, Kirsten Swearingen, Kevin Li, and Marti Hearst. Faceted Metadata for Image Search and Browsing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '03, pages 401–408. ACM, New York, NY, USA, 2003.
- [YWC13] Zhen Yang, Guoqing Wang, and Feng Chu. An effective GRASP and tabu search for the 0–1 quadratic knapsack problem. *Computers & Operations Research*, 40(5):1176–1185, 2013.
- [Zac16] Erik Zachte. Other Projects Statistics Wikidata. <https://stats.wikimedia.org/wikispecial/EN/TablesWikipediaWIKIDATA.htm>, July 2016. Accessed August 1, 2016.
- [ZCQ07] Xiang Zhang, Gong Cheng, and Yuzhong Qu. Ontology Summarization Based on RDF Sentence Graph. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, pages 707–716. ACM, New York, NY, USA, 2007.

A. Appendix

A.1. UBES

Listing A.1: Used property chain rules.

```
<http://example.com/hasActor>
<http://www.w3.org/2002/07/owl#propertyChainAxiom> (
  <http://rdf.freebase.com/ns/film.film.starring>
  <http://rdf.freebase.com/ns/film.performance.actor>
) .

<http://example.com/hasBudget>
<http://www.w3.org/2002/07/owl#propertyChainAxiom> (
  <http://rdf.freebase.com/ns/film.film.estimated_budget>
  <http://rdf.freebase.com/ns/measurement_unit.dated_money_value.
  amount>
) .

<http://example.com/hasRunningTime>
<http://www.w3.org/2002/07/owl#propertyChainAxiom> (
  <http://rdf.freebase.com/ns/film.film.film.runtime>
  <http://rdf.freebase.com/ns/film.film_cut.runtime>
) .

<http://example.com/hasAward>
<http://www.w3.org/2002/07/owl#propertyChainAxiom> (
  <http://rdf.freebase.com/ns/award.award_winning_work.awards_won>
  <http://rdf.freebase.com/ns/award.award_honor.award>
) .
```

Table A.1.: White list of covered Freebase predicates by *WhoKnows?Movies!*.

fb:base.parody.parodied_subject.parodies
fb:fictional_universe.work_of_fiction.events
fb:film.film.cinematography
fb:film.film.directed_by
fb:film.film.edited_by
fb:film.film.featured_song
fb:film.film.film_casting_director
fb:film.film.film_festivals
fb:film.film.film_series
fb:film.film.genre
fb:film.film.initial_release_date
fb:film.film.music
fb:film.film.prequel
fb:film.film.production_companies
fb:film.film.rating
fb:film.film.sequel
fb:film.film.story_by
fb:film.film.subjects
fb:film.film.written_by
fb:media_common.adaptation.adapted_from
ex:hasActor
ex:hasAward
ex:hasBudget
ex:hasRunningTime
fb:film.film.featured_film_locations