LINKS IN LARGE INTEGRATED KNOWLEDGE GRAPHS: ANALYSIS, REFINEMENT, AND DOMAIN APPLICATIONS

SHUAI WANG

Vrije Universiteit Amsterdam

This research is a part of the MaestroGraph project (project ID: 612001552) financially supported by the Dutch Research Council (Nederlandse Organisatie voor Wetenschappelijk Onderzoek; NWO).

SIKS Dissertation Series No. TODO

The cover was designed by Shuai Wang. The front cover shows a painting of his titled "Inheritance", which illustrates an abstract representation of the relations between masters in the history of art and science, and how their breakthroughs influence each other. An instantiation of this idea is illustrated in Figure 1. The painting is now preserved by Joe Raad.

ISBN: xxxxx
An electronic version of this thesis is available at https://www.shuai.ai/research/.

Vrije Universiteit Amsterdam

Links in Large Integrated Knowledge Graphs: Analysis, Refinement, and Domain Applications

ACADEMISCH PROEFSCHRIFT

ter verkrijging van de graad van doctor aan de Vrije Universiteit Amsterdam, op gezag van de rector magnificus prof.dr. Stefan Schlobach, in het openbaar te verdedigen ten overstaan van de promotiecommissie van de Faculteit der Wetenschappen op TBD X X 2025 om X uur in de aula van de universiteit, De Boelelaan 1105

door

Shuai Wang

geboren in Binnen-Mongolië, China

promotoren:	prof.dr. Frank van Harmelen
co-promotoren:	dr. Peter Bloem
	dr. Joe Raad
promotiecommissie:	prof. dr. Stefan Schlobach
	prof. dr. Aidan Hogan
	prof. dr. Harald Sack
	dr. Ilaria Tiddi
	dr. Wouter Beek

To Prof. Igor Potapov,
Prof. Roger Penrose,
Marcos Vinicius,
family, and friends.

Contents

LIST OF ACRONYMS AND ABBREVIATIONS

AI/A.I. Artificial Intelligence

CC Connected Componnet

DAG Directed Acyclic Graph

DID Decentralized Identifiers

DOI Digital Object Identifier

FAIR Findability, Accessibility, Interoperability, Reusability.

FAS Feedback Arc Set

GRC Global Reaching Centrality

GSSO Gender, Sex, and Sex Orientation Ontology

IRI Internationalized Resource Identifiers

KG Knowledge Graph

LCSH Library of Congress Subject Headings

LOD Linked Open Data

LGBTQ+ LGBTQ stands for lesbian, gay, bisexual, transgender, and queer. The '+' holds space for the expanding and new understanding of different parts of the very diverse gender and sexual identities.

MAXSAT The Maximum Satisfiability problem

MWFAS Minimum Weighted Feedback Arc Set

OWL Web Ontology Language

QLIT Queer Literature Indexing Thesaurus

RDF Resource Description Framework

RDFS RDF Schema

RQ Research Question

SAT Boolean satisfiability problem

SMT Satisfiability Modulo Theory

SCC Stronly Connected Component

SHACL Shapes Constraint Language

UI User Interface

UNA Unique Name Assumption. Its various definitions in this thesis include nUNA (naive UNA), iUNA (internal UNA), and qUNA (quasi UNA).

URI Uniform Resource Identifier

WCC Weakly Connected Component

SAMENVATTING (NEDERLANDS)

Dit werk richt zich op een specifiek formaat voor kennisrepresentatie, namelijk *kennisgrafen*, waarbij knopen entiteiten voorstellen en verbindingen relaties aanduiden. Het integreren van meerdere kennisgrafen kan rijkere informatiebronnen opleveren, maar ook leiden tot ongewenste structuren en zelfs logische inconsistenties. Daarom zijn verfijningsmethoden die dergelijke problemen opsporen en corrigeren essentieel. Schaal is ook van belang. Problemen die eenvoudig zijn bij kleine kennisgrafen worden aanzienlijk complexer op grotere schaal. Het aanpakken van deze uitdagingen vereist data-analyse, algoritme-ontwikkeling en rigoureuze evaluatie. Dit proefschrift onderzoekt kernproblemen in grote, geïntegreerde kennisgrafen—zoals identiteit, oorzaken van fouten en kennisevolutie. De gebruikte tools voor analyse en verfijning maken gebruik van grafentheorie, geautomatiseerd redeneren en meer.

Transitieve relaties zijn alomtegenwoordig in kennisgrafen—voorbeelden zijn klassesubsumptie, deel-geheel-hiërarchieën en afstamming. Echter, transitiviteit kan kleine fouten tijdens integratie ver buiten hun lokale context verspreiden. We breiden ons onderzoek uit naar relaties die bedoeld zijn om zowel transitief als antisymmetrisch te gebruiken, zelfs als dit niet formeel is vastgelegd. We noemen deze *pseudo-transitieve relaties*. Hoofdstuk 2 introduceert een algoritme en de bijbehorende benchmark, bestaande uit verschillende grafen met transitieve en pseudo-transitieve relaties, compleet met handmatig gelabelde gouden standaarden en referentiemethoden. We stellen nieuwe analysemethoden voor en introduceren een algoritme voor het verfijnen van kennisgrafen met dergelijke relaties. Ons algoritme onderzoekt de grafstructuur. Traditioneel worden herhaalde uitspraken als logisch equivalent beschouwd en tijdens integratie genegeerd. In geïntegreerde kennisgrafen is het echter mogelijk bij te houden hoeveel brongrafen elke uitspraak bevestigen, geïnterpreteerd als *gewichten*. Uitgaande van de intuïtie dat uitspraken die door meer bronnen worden ondersteund een hogere kans hebben om correct zijn, breiden we ons algoritme uit met een wegingsschema dat heuristisch verdachte verbindingen identificeert en verwijdert om acycliciteit te herstellen.

Een speciaal geval van transitieve relaties is de *identiteitsrelatie*, die stelt dat twee entiteiten naar hetzelfde concept verwijzen. De subgraaf gevormd door deze verbindingen staat bekend als de *identiteitsgraaf*. Hoofdstuk 3 richt zich op het verfijnen van zulke grafen. Het bepalen van de juiste representatie van een concept—vooral wanneer dit wordt gemodelleerd als een cluster van onderling verbonden entiteiten—

kan lastig zijn. Fouten kunnen leiden tot het onterecht samenvoegen van niet-gerelateerde entiteiten. Meestal nemen we aan dat elk gegevensbestand elk concept met één entiteit representeert—dit staat bekend als de *Unique Name Assumption* (*UNA*). In de praktijk faalt deze aanname vaak. Identiteitsuitspraken gaan vaak over entiteiten die verschillende versies, talen of coderingen representeren. Om hiermee rekening te houden in grote geïntegreerde kennisgrafen, definiëren we een versoepelde aanname genaamd *internal UNA* (*iUNA*). Op basis van dit concept ontwikkelen we nieuwe algoritmen voor het detecteren en elimineren van foutieve identiteitsuitspraken.

In Hoofdstuk 4 bestuderen we de evolutie en dynamiek van kennisgrafen door het analyseren van doorverwijzingen tussen entiteiten en de ketens die ze vormen. We classificeren verschillende doorverwijsscenario' s en schatten het aandeel van doorverwijzingen dat geïnterpreteerd kan worden als identiteitsrelaties. Daarnaast analyseren we de statistische en grafentheoretische eigenschappen van doorverwijsgrafen.

Terwijl de voorgaande hoofdstukken zich richten op het analyseren en verfijnen van bestaande grootschalige geïntegreerde kennisgrafen, richt Hoofdstuk 5 zich op een domeinspecifieke toepassing. We selecteren en integreren meerdere kennisgrafen uit de domeinen economie, financiën en het bankwezen. Via statistische en grafentheoretische analyse tonen we aan hoe integratie leidt tot entiteiten met rijkere en completere informatie. De kwaliteit van de geïntegreerde graaf wordt geëvalueerd door subgrafen te analyseren gevormd door identiteits- en (pseudo-)transitieve relaties. We bestuderen ook de oorzaken van fouten en stellen methoden voor hun verfijning voor, waarbij we de voordelen van onze integratieaanpak benadrukken.

Hoofdstuk 6 verkent een andere domeinspecifieke toepassing, gericht op LGBTQ+-entiteiten en -relaties. We construeren een kennisgraaf over identiteitsgerelateerde relaties binnen dit domein en analyseren de eigenschappen ervan. We laten zien hoe uitdagingen zoals meertaligheid, conceptuele verschuiving en taalkundige ambiguïteit de complexiteit aanzienlijk verhogen en eerder waargenomen problemen versterken.

Al met al stelt dit proefschrift een benadering voor die logisch redeneren combineert met grafentheoretische analyse om grootschalige geïntegreerde kennisgrafen te bestuderen en te verfijnen. Hoewel schaalbaarheid een blijvende uitdaging vormt, tonen we de toepasbaarheid en effectiviteit van onze methodologie aan in verschillende domeinen. Bovendien hebben we herbruikbare hulpmiddelen gecreëerd en hun nut aangetoond, waarmee we bijdragen aan de basis voor toekomstig onderzoek.

Dit proefschrift is het resultaat van het promotieonderzoek van Shuai Wang, gefinancierd door de NWO TOP-subsidie.

SUMMARY (ENGLISH)

This work focuses on a specific knowledge representation format known as *knowledge graphs*, where nodes represent entities and edges denote relations. Integrating knowledge graphs can result in richer resources but also lead to undesirable structures and even logical inconsistencies. Therefore, refinement methods that detect and correct such issues are essential. Scale matters. Problems that are easy for small knowledge graphs can become significantly more challenging at scale. Addressing these challenges requires data analysis, algorithm development, and rigorous evaluation. This thesis investigates key issues in large, integrated knowledge graphs—such as identity, error sources, and knowledge evolution. Tools used for analysis and refinement take advantage of graph theory, automated reasoning, and more.

Transitive relations are ubiquitous in knowledge graphs—examples include class subsumption, part-whole hierarchies, and concept specification. However, transitivity can propagate small errors far beyond their local contexts as a result of integration. We extend our investigation to relations that are intended to be both transitive and antisymmetric, even if not formally declared. We refer to these as *pseudo-transitive relations*. Chapter 2 introduces an algorithm and the corresponding benchmark comprising several graphs of transitive and pseudo-transitive relations, complete with hand-labeled gold standards and baseline methods. We propose new analytical measures and introduce an algorithm for refining knowledge graphs with such relations. Our algorithm inspects the graph structure. Traditionally, repeated statements are treated as logically equivalent and are discarded during integration. However, it is possible to track how many source graphs assert each statement, interpreted as *weights*. Building on the intuition that statements supported by more sources are more likely to be correct, we extend our algorithm with a weighting scheme that heuristically identifies and removes suspect edges to restore acyclicity.

A special case of transitive relations is the *identity relation*, which asserts that two entities refer to the same concept. The subgraph formed by these assertions is known as the *identity graph*. Chapter 3 focuses on refining such graphs. Determining the correct representation of a concept—especially when modeled as a cluster of interlinked entities—can be challenging. Errors here may result in falsely merging clusters of unrelated entities. Typically, we assume that each dataset represents each concept with a single entity—this is known as the *Unique Name Assumption* (*UNA*). In practice, however, this assumption often fails. Identity assertions frequently involve entities representing different versions, languages, or encodings. To account for this for

large integrated knowledge graphs, we define a relaxed assumption called *internal UNA* (*iUNA*). Based on this notion, we develop a new algorithm for detecting and eliminating erroneous identity statements.

In Chapter 4, we study the evolution and dynamics of knowledge graphs by analyzing entity redirections and the chains they form. We classify different redirection scenarios and estimate the proportion of redirects that can be interpreted as identity links. Additionally, we analyze the statistical and graph-theoretic properties of redirection graphs.

While the previous chapters focus on analyzing and refining existing large-scale integrated knowledge graphs, Chapter 5 turns to a domain-specific application. We select and integrate multiple knowledge graphs from the domains of economics, finance, and banking. Through statistical and graph-theoretic analysis, we demonstrate how integration yields entities with richer, more complete information. The quality of the integrated graph is evaluated by analyzing subgraphs formed by identity and (pseudo-)transitive relations. We also study the sources of errors and propose methods for their refinement, highlighting the benefits of our integration approach.

Chapter 6 explores another domain-specific application, focusing on LGBTQ+ entities and relations. We construct a knowledge graph about identity-related relations in this domain and analyze its properties. We show how challenges such as multilingualism, conceptual drift, and linguistic ambiguity significantly increase complexity, amplifying issues previously observed. We demonstrate how our knowledge graph can be used to address these problems.

Overall, this thesis proposes an approach that combines logical reasoning with graph-theoretic analysis to study and refine large-scale integrated knowledge graphs. Although scalability remains a challenge, we demonstrate the applicability and effectiveness of our methodology using two domain applications. Furthermore, we have created reusable resources and showcased their utility, contributing to the foundation for future research.

This thesis is the result of Shuai Wang's PhD research, funded by the NWO TOP grant.

ACKNOWLEDGEMENTS

> Excellence is never an accident. It is always the result of high intention, sincere effort, and intelligent execution; it represents the wise choice of many alternatives –choice, not chance, determines your destiny.
>
> Aristotle

I still vividly remember the night when I was among the audience in the Ordos Concert Hall, uncertain about my future while preparing for the National College Entrance Examination. On stage was Marcos Vinicius, a classical guitarist. He recounted how he first encountered a guitar at the age of five —a love at first sight that would shape his life. Since that moment, he has performed with unwavering passion, pouring his heart into every note, no matter the place or audience. I felt the same spark of passion while reading about Artificial Intelligence in Roger Penrose's book *The Emperor's New Mind*[1] [54]. The idea of becoming a scientist had begun to take shape in my mind, but it remained somewhat vague until I met my tutor, Igor Potapov. Igor, I am extremely fortunate to have had you as my tutor at the beginning of this journey. You have not only provided invaluable guidance for my studies and career but also stood by me during some of the darkest days of my life —especially when I came out as gay to my family, in anger and tears. Your support meant the world to me. I am deeply grateful for everything you have done for me, which was the light that led me to this Ph.D.

For this thesis, I wish to express my deepest gratitude to my supervisors, Frank van Harmelen, Peter Bloem, and Joe Raad. Frank, I feel incredibly privileged to have you as my promoter and to have been supported by the prestigious NWO TOP grant. Your mentorship has been transformative; you taught me the value of collaboration, empathy, and embracing mistakes as opportunities to grow. You have been a true role model, constantly learning, adapting, and guiding the team towards paths of greatest potential. Your contagious enthusiasm and limitless energy greatly influenced me. You have shown me not only how to craft sentences that are clear and

1 https://en.wikipedia.org/wiki/The_Emperor%27s_New_Mind

meaningful but also how to make them inspiring. Above all, I am profoundly grateful for your unwavering support during my struggles with depression. Your belief in me, even when I doubted myself, has made all the difference. Thank you for your guidance, encouragement, and faith in my abilities.

Peter, you have been an incredible source of support throughout this entire journey. I cannot thank you enough for everything you have done. There have been many moments when I felt drained or uncertain, but your encouragement has always helped me regain my energy and motivation. You not only offered guidance in refining my research, but also pushed me to improve my communication skills, which has been invaluable in making my work clearer and more impactful. You patiently highlighted the mistakes I made in experimental design, helped improve the organization of my paragraphs, and thoroughly reviewed my writing. Your close guidance and mentorship have been essential in shaping this journey. I truly cannot imagine going through this process without your constant support and thoughtful advice. Thank you for being such a dedicated and supportive supervisor!

Joe, you are such a great researcher with skills and qualities that sometimes make me jealous. You showed me how to design projects that have an impact. I often find myself leaving meetings full of inspiration and an urge to get things done. I will never forget the nights we chased deadlines and modified our rebuttals. I am very lucky to have you as a supervisor, a co-author, a mentor, and a friend!

I find the colleagues in the Knowledge Representation and Reasoning (KR&R) group very supportive. A special thanks to Wouter Beek, whose data and research results play a crucial role in this thesis. I am thankful to my colleagues: Ting Liu, Albert Meroño-Peñuela, Ali Khalili, Márk Adamik, Majid Mohammadi, Andreas Sauter, Romana Pernisch, Lise Stork, Ilaria Tiddi, Jan-Christoph Kalo, Loan Ho, Taewoon (Tae) Kim, Taraneh Younesian, Daniel Daza, Xu Wang, Selene Baez Santamaria, Dimitris Alivanistos, to name a few. It has been a great joy to work and hang out with such wonderful colleagues. A T-break without Erman Acar is like a break without the tea; a cigarette without Nikos (Nikolaos Kondylidis) is like smoking without the Nicotine. I will try to become more skilled at cello so that I can jam with Benno Kruit in the future!

I am sincerely grateful for the support from Jacco van Ossenbruggen and Ronald Siebes. It has been a pleasure working with you over the past two years. I am especially grateful for your encouragement and the motivating conversations that kept me focused and inspired throughout this PhD journey. Equivalently, thanks to all the colleagues in the User-centric Data Science (UCDS) group. I appreciate the advice from Victor de Boer, Emma Beauxis-Aussalet, Elena Beretta, Richard Zijdeman, Margherita Martorana, Roderick van der Weerdt, Go Sugimoto, Xander Wilcke, Andrei Nesterov, Sarah Binta Alam Shoilee, and colleagues. Xueli Pan, I cannot thank you enough for all the support you have given me, especially during this final mile to-

ward completing my PhD. I sincerely thank Angelica Manari, Tobias Kuhn, and André Valdestilhas for their invaluable collaboration on the FAIR Expertise Hub Project. Finally, I would like to thank Mojca Lovrenčak, our secretary, for all the help over the past few years!

Over the past year and a half, I have been working as the department data steward. I would like to take this opportunity to thank a few colleagues that I have worked closely with in the department and the faculty: Kees Verstoep, Jaap Heringa, and Brett Olivier. My work has been kindly supported by the RDM team of the university library, especially Lena Karvovskaya, Tycho Hofstra, Demet Yazilitaş, Mark Bruyneel, Stephanie van de Sandt, Tim Veken, Meron Vermaas, Elisa Rodenburg, Marcel Ras, and Abeer Pervaiz. Many thanks to everyone!

Apart from the supervisors and co-authors, some papers in this thesis have been proofread by colleagues and experts in the field: Siska Humlesjö, Clair Kronk, Jacco van Ossenbruggen, Xueli Pan, Majid Mohammadi, Xu Wang, and Jan-Christoph Kalo. I appreciate the help and advice of Annette ten Teije, Margherita Martorana, Ronald Siebes, Victor de Boer, Stefan Schlobach, Emma Beauxis-Aussalet, Michael Simpson, Stefan Schlobach, Jan Wielemaker, Michael Cochez, Luigi Asprino, Wouter Beek, and Jacopo Urbani. Reviews from anonymous reviewers, especially those of ESWC and EKAW, are greatly appreciated. I have taken into account your constructive suggestions and helpful comments. Thank you all!

During my PhD, I have served as a teaching assistant for several courses. Interacting with students from diverse backgrounds and explaining concepts to them was something I found enjoyable in academia. I have gained considerable insight into teaching through Jakub Tomczak, Michael Cochez, Peter Bloem, Frank van Harmelen, Annette ten Teije, and Stefan Schlobach. I am very lucky to have been the supervisor of many excellent students (for their theses and group projects), including Robin Stöhr, Hidde Makimei, Jeroen Klaver, Elif Ayten, Manar Attar, Daniel Vlantis, Navroop K. Singh, Maria Adamidou, Tianyang Lu (Angela), Khaled Rabata, Ata Turan Oguz, Idries Nasim, Tico van der Laan, Stein de Bever, Anna-Maja Kazarian, Ikrame Zizar, Lucas de Vries, Guilherme Arashiro, and most recently Mateusz Grzegorz Kędzia. Through co-supervising these students, I collaborated with many internal and external researchers: Zhisheng Huang (VU), Ronald Siebes (VU), Erwin Flomer (TU Twente & Kadaster), Alexandra Rowland (Kadaster), Tibor Bosse (RU Nijmegen), Gongjin Lan (VU), Jadran Sirotkovic (Accenture), Daniel Formolo (VU), Iva Gornishka (Amsterdam Intelligence), Eirik Kultorp (Triply), Wilfred van Buuren (IHLIA), Jack van der Wel (IHLIA/Homosaurus), Willem van Peursen (ETCBC, VU), Siska Humlesjö (QLIT, Gothenburg University Library), Olov Kriström (QLIT, Gothenburg University Library), Xander Wilcke (VU), Tycho Hofstra (University Library, VU), Mark Bruyneel (University Library, VU), Stephanie van de Sandt (University Library, VU), Hjalmar Snoep (Snoep Animation), and most recently Jiancheng

Weng (Beijing University of Technology). It is incredibly exciting to see our results being published at prestigious international conferences, including the ACM International Conference on Advances in Geographic Information Systems (ACM SIGSPATIAL), the International Conference on Computational Linguistics (COLING), the International Conference on Agents and Artificial Intelligence (ICAART), the International Conference on Metadata and Semantics Research (MTSR), and the International Conference on Multilingual Digital Terminology Today (MDTT). The publications underscore the importance of our findings and the fruitful partnership with the aforementioned collaborators. Additionally, I am honored to mention that the thesis project of Navroop won the prestigious Open Science Community Amsterdam (OSCA) Award.

I consider myself extremely fortunate to have had my psychologist, Jennifer Stock, who not only helped me overcome my depression but also rediscovered my talent in art. I am deeply grateful for the understanding and support that I have received from all of my friends, especially Shiya Zhou, Junye Qu, Wenjia Liu (Elix), Fan Li, and Tom Clements. Your presence in this journey has made a significant impact on me. I have never lost hope completely during the depression or the pandemic because of you. I have learned so much from my friends about the importance of loving oneself and showing love to others. They are Łukas Koterba, Geoffrey Quak, Yngve M. Hereide, Franco Egidi, Idris Hakkar, Francisco Braccini, Vincent Meijer, Boy Kouwenberg, and Patrick Honnebier. I also appreciate the support from friends of the GvC church: Willem and Jane Nieuwboer, Allen Foster and Evangeline Gnanaraj (Eve), Jeremiah and Natasja Niles, Joris van Hijfte and Gisèle van Hijfte-Robert, Deanne De Vries, Kimberly Fuqua (Kim), Louis Tung and Chih Lin, Femke Oosterbaan, Timothy van Hijfte, Aloïs Rosset, and many more.

Throughout this PhD, the Chinese community has been incredibly supportive. The list would be too long if I include every name in this acknowledgment section, but I would like to mention a few individuals who are of great significance: Yunzhen Kuang, Guolai Li, Jiaxu Zhang, Jingxin Xu, Mingxuan Chen, Yancheng Jiang, Tingting Chen, Kaixin Hu, Shuaidong Yu, Nuo, Chaohui Guo, Gezi Fu, Echo Chen, and Yanchen Jiang. I will never forget the kind words from Master Miaoyi in the Fo Guang Shan He Hua Temple about the power of forgiveness. Looking forward to meeting you again, Kok Kuen Chan, Ethan Tan, Zhiguang Zhao, Yujie Xing, Xiulin Bai, Zhongya Meng, Yin Liao, Wenlan An, Kuo-Jen Mao, and Zeng Ming!

I wish to express my gratitude to a few friends who have played important roles in various aspects of my life in the past few years: Aaron Merino Clark, Shao (Ionuț Balan), Majd Alhasan, Duco Ottevanger, José Diogo da Cruz Rodrigues (Diogo), Jose-Carlos, Adityajit Singh Kang (Adi), John C. Lokman (Can), Ido Holkamp and Andrea Rose, Rutger Schaafsma, Jan-Jelle de Meijer, and Niko Chan. I would like to extend my thanks to my flatmates and neighbors for their kindness during the pan-

demic, especially Gaurav Sehrawat, Anke Eweg, Caio Sampaio, and Igor Zen. Thank you, Twan Hopstaken, for training with me in the gym and keeping me motivated.

I feel fortunate to have numerous friends who are artists, providing me with abundant inspiration. Please forgive me for limiting myself to the following due to the page limit: Antonis Pratsinakis (my cello teacher), Caroline van Bavel (my vocal coach), Thalia Giardini, Quintus van Amstel (a.k.a. Quintus de paarse goochelaar; a.k.a. Juffrouw Mina), Hjalmar Snoep, Zhipeng Guo (Zippo), Shiyuan Liang, Minhong Yu, Aaron Wan, Jia Zhao, Cristian Mercado, Jacques van Paassen (Jack), Javier Torras Casas, Kuo-Jen Mao, Costanza Spadafora, Andi Georgescu (a.k.a. George G. Leotta), Roy Seerden, Tenzing Woing (Tenna), Denzel Maple, etc.

Aside from the academic challenges that come with pursuing a PhD, this journey has been marked by numerous personal and external obstacles. As mentioned above, I have faced a long and difficult period of depression, which at times made it difficult to see the light at the end of the tunnel. On top of that, the global pandemic disrupted both my personal and academic life in ways I could never have anticipated. The isolation, uncertainty, and sudden shifts in how we work and interact were incredibly challenging. These, along with many other unforeseen struggles, have tested my resilience, but they have also shaped and strengthened me in ways I never imagined when I first began this journey. I greatly appreciate the love from my family, my dog Soleil, and my cat Tensor. I would like to thank again all my supervisors, colleagues, and friends. I believe that completing this thesis marks the beginning of an extraordinary journey ahead. The best part is that I won't have to face it alone, thanks to all of you.

Finally, I would like to thank in advance all those who will be contributing to the defense, especially the two paranymphs: Lucian Paul-Trifu and Martin van Harmelen. Thanks, Lucian, for traveling from Romania for this defense. Your kind words during my depression mean a lot to me. Martin, I appreciate your support very much! Concerning the thesis evaluation, I want to express my appreciation to the members of my examination committee, Stefan Schlobach, Ilaria Tiddi, Aidan Hogan, Harald Sack, and Wouter Beek, for devoting their time to examine my thesis and attending the defense.

Shuai Wang
Maastricht, July 22, 2025

1 INTRODUCTION

朝闻道，夕死可矣 (Know the way at dawn; die without regret at dusk).

Confucius

This chapter provides a general introduction to knowledge and its representation in Section 1.1. To help readers better understand the thesis, preliminaries are included in Section 1.2. Finally, the structure of the thesis and the research questions are in Section 1.3.

1.1 Knowledge and its Representation

Imagine assembling Bertrand Russell, Sigmund Freud, Confucius, Erwin Olaf, Lo Ta-yu, Liu Cixin, and Frank van Harmelen around the same table for a spirited exchange of ideas. What would their collective knowledge look like? Despite their intellectual stature, the conversation might get off to a rocky start: language barriers would pose immediate challenges, and even once those are bridged, disagreements would likely emerge —not due to a lack of intelligence, but because each participant holds distinct beliefs shaped by their backgrounds, cultures, and disciplines. Their respective "knowledge bases" —rich but fallible —could include contradictions, gaps, or outdated assumptions. As if that were not enough, their views would continue to evolve with new experiences. To further complicate things, even shared vocabulary might mask divergent meanings: a "queen" in one context is a monarch, in another a

chess piece, and elsewhere a pop icon; scientific terms like "autism" shift its scope as research progresses; and cultural concepts such as "love" challenge universal understanding. In short, creating a coherent collective knowledge from such a gathering is no simple task. It is precisely this kind of challenge that motivates this research.

A means of capturing their knowledge is to use *knowledge graphs* (KGs). A knowledge graph consists of *entities* and their *relations*. Figure 1 is an example based on information retrieved from the Mathematics Genealogy Project[1], a large knowledge base where the supervisory relationship between scientists are captured. In this case, scientists are the entities in the KG. An arrow in the graph represents a supervisory relationship, e.g. Bertrand A.W. Russell was a supervisor of Ludwig Wittgenstein. To avoid confusion in cases like J. Bernoulli, we associate a unique identifier with each entity. In practice, we use the Uniform Resource Identifier (URI), or more generally, the IRI (Internationalized Resource Identifier), which extends URI by allowing for the use of Unicode characters that include those outside the ASCII character set. In the ambiguous case of "J. Bernoulli", Jacob Bernoulli[2] was assigned the `http://dbpedia.org/resource/Jacob_Bernoulli` and his brother and student Johann Bernoulli[3] was assigned `http://dbpedia.org/resource/Johann_Bernoulli`. Here, we have `http://dbpedia.org/resource/` as the *namespace*. To avoid using such long URIs everywhere in the text, *prefix* was introduced. The prefix `dbr` corresponds to `http://dbpedia.org/resource/`. Thus, we have `dbr:Johann_Bernoulli` and `dbr:Jacob_Bernoulli`, respectively. The list of prefix used in this thesis is summarized in Appendix A. Using the prefix `genealogy` for the namespace `https://example.org/genealogy/`, we can introduce an URI for the supervisory relationship, e.g. `genealogy:is_a_supervisor_of`. We can write the relationship between them in the format of a *triple* such as (`dbr:Jacob_Bernoulli`, `genealogy:is_a_supervisor_of`, `dbr:Johann_Bernoulli`).

This KG is not always free from errors and missing information during its *construction*. For many knowledge graphs, operations were applied after the initial construction to identify erroneous links and add missing information [52]. Such post-processing operations are called *refinement* [52]. For example, in Figure 1, we observed a cycle between Euler, Lagrange, and Laplace. Although such cycles are possible, it is not quite plausible in reality. Structural insights can assist in identifying potential error locations, though determining which edge to eliminate still poses a challenge. Moreover, it is unlikely that Gottfried W. Leibniz is the supervisor of Friedrich Leibniz, his father (according to Wikipedia[4]). Thus, we remove the green edge. Finally, we observe that Michel Chasles appears twice in the graph. We could

1 `https://www.genealogy.math.ndsu.nodak.edu/`
2 `https://en.wikipedia.org/wiki/Jacob_Bernoulli`
3 `https://en.wikipedia.org/wiki/Johann_Bernoulli`
4 https://en.wikipedia.org/wiki/Friedrich_Leibniz

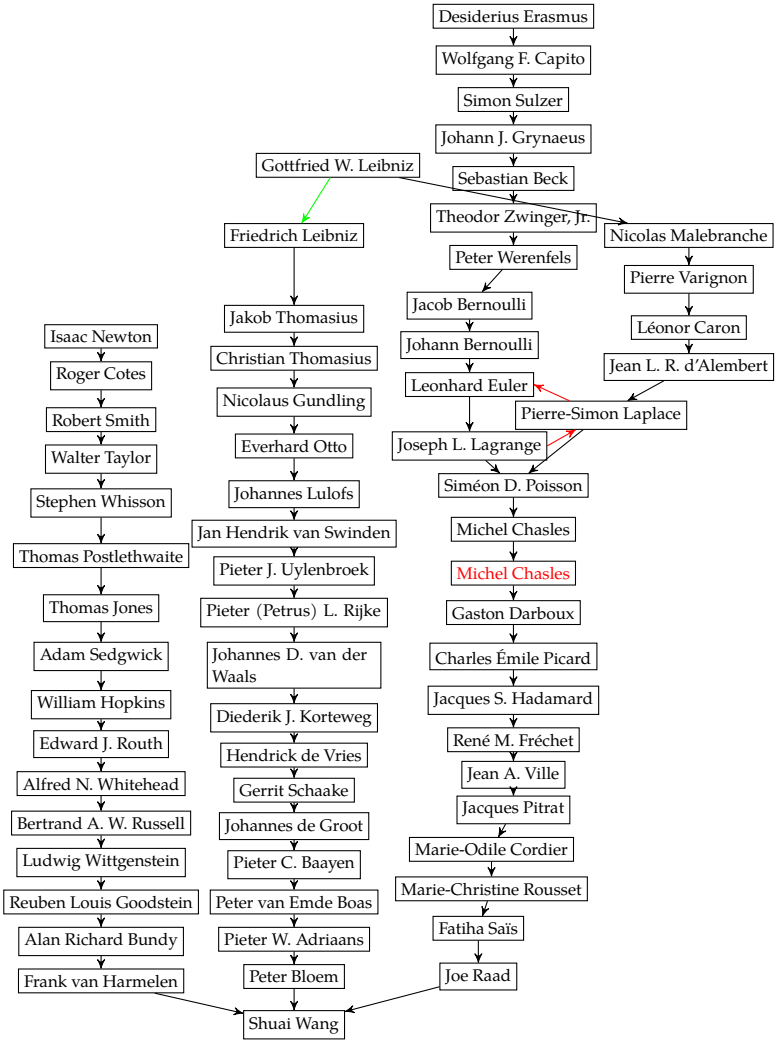


Figure 1: A sample KG with links representing "is a supervisor of" extracted from the Mathematics Genealogy Project. The green and red links are mistaken links introduced by the author to demonstrate the process of refinement. Text in red indicates redundancy, which is an error in this case. The cover of this thesis is an abstract representation of the repeated patterns of "inheritance" and "innovation".

remove one and finish the refinement process. The process of adding information (for example, adding new labels, multilingual information, new relations) is often called *enrichment*. Although enriching KGs is a significant subject, the primary focus of this thesis is on analysis and refinement.

More specifically, this thesis focuses on the analysis and refinement of large integrated knowledge graphs constructed from multiple sources of the *linked open data cloud*, a.k.a. LOD cloud. There are links about identity relations, concept subsumption relationships, as well as identification and provenance information. These errors will not only be harmful to the quality of KGs but also have some negative impact on their applications. However, maintaining high quality for such large KGs can be difficult. As we shall see in this thesis, problems that are not seriously harmful on a small scale can become very difficult to handle on a much larger scale as the complexity accumulates.

1.2 Preliminaries

In this section, we provide the preliminaries. Section 1.2.1 introduces the basic concepts and definitions of linked data and the semantic web. Section 1.2.2 provides the basics of graph theory, graph algorithms, their complexity, and related tools.

1.2.1 Knowledge Representation, Linked Data, and Semantic Web

The study of formal representation of knowledge in computer science is arguably as old as the subject itself. *Knowledge bases* are commonly used to refer to information organized in a structured way with the ability to integrate new facts. Knowledge bases that take advantage of semantic technologies and references to concepts in each other become *linked data*. As more and more such data are published on the web as open source, they are often referred to as the *Linked Open Data cloud*, a.k.a. LOD cloud. Linked data, together with other means of semantic representation and related techniques, extend the World Wide Web to form the *semantic web*. In this thesis, all the KGs we study are constructed from linked open data. KGs differ from data graphs due to their adoption of semantics, with key concepts such as classes, properties, subclass subsumption relations, subproperty relations, domain and range restrictions [4]. Such semantics rely on the use of RDF, RDFS, as well as OWL and other languages. OWL is a richer language than RDF/RDFS [4].

Uniform Resource Identifiers (URIs) are unique identifiers for the representation of abstract or physical resources, with their representation limited to ASCII. Internationalized Resource Identifiers (IRIs) extend URIs by permitting a wider range of Unicode characters [49, 76]. Relations are represented as triples in the form of a construct (s, r, o), where s is the *subject*, $r \in R$ is the *property*, and o is the *object*. The subject can be an IRI or a blank node. The predicate should be an IRI of the relation. The object is an IRI, blank node, or *literal*. A literal can be a number, a string, a date,

etc. Here, the "resources" should be understood as not only "things" like persons, animal species, mathematical symbols, historical events, cities, but also abstract entities, including the societal attitudes towards abortion[5], a template of webpages about other entities, a generic class of abstract concepts, etc. DBpedia has its foundation based on data extracted from Wikipedia. An example is an abstract entity[6] representing the disambiguation of "dreams" in different contexts: the novel by Ivan Bunin published in 1904, an India Hindi film in 2005, a song by The Cranberries in 1992, etc. There are disambiguation links to the corresponding entities.

The example of supervisory relations in the previous section is not enough to provide all the information about each scientist (e.g., the birthplace of the scientists and their publications). Integrating multiple KGs can result in richer information. However, such integration raises challenges regarding how the same entity and its relations are represented across different KGs. To address this, *identity links* are introduced, which assert that two entities in different KGs refer to the same real-world entity. By linking entities through identity links, it becomes possible to retain and manage entities locally within their original graphs while sharing and reusing their descriptive information globally – these links can be used for retrieving related information about entities from different resources and perspectives. The set of such identity relations is called a *linkset*. In the context of ontology, the term often employed is *alignment* (and can also include other relations such as `skos:broader`). In cases without ambiguity, a more general term, *mapping*, is utilized. Despite their utility, these mappings are prone to errors. Moreover, there can be multiple entities representing the same concept, so duplicates are common in large integrated graphs.

The Unique Name Assumption (UNA) [87] supposes that two terms in the same knowledge base with distinct names (IRIs in the context of the thesis) do not refer to the same real-world entity. However, in practice, the UNA does not always hold due to redundant IRIs that capture various encodings, languages, namespaces, versions, letter cases, etc. The identity relations between such IRIs are often captured by constructs of `owl:sameAs`. For instance, `(v1:dog, owl:sameAs, v2:Dog)`. Despite that the UNA does not hold for integrated knowledge graphs, some previous work showed that it can be used for refinement purposes with some adaption [69].

Maintaining multiple large KGs can be difficult. Knowledge engineers need to handle multilingual issues, understand mismatches between different versions and encodings, resolve inconsistency, remove duplicates, etc. These very large KGs display modern phenomena that are not explained by standard model-theoretic semantics [70]. A significant part of this thesis focuses on VERY large (integrated) KGs and their subgraphs. Indicating the order of scale of data used in this thesis,

5 https://dbpedia.org/page/Societal_attitudes_towards_abortion.

6 The DBpedia entity `dbr:Dreams_(disambiguation)` that was constructed using the Wikipedia page https://en.wikipedia.org/wiki/Dreams_(disambiguation).

the LOD-a-lot is a KG merged from 650K datasets crawled from the LOD Cloud in 2015 [26]. It contains over 28 billion triples, making it one of the largest publicly available KGs. Additionally, we examine its subgraphs, such as the identity graph `sameAs.cc`, which encompasses 558 million unique `owl:sameAs` statements [7]. This makes it impossible to assess each and every link manually while keeping track of their sources.

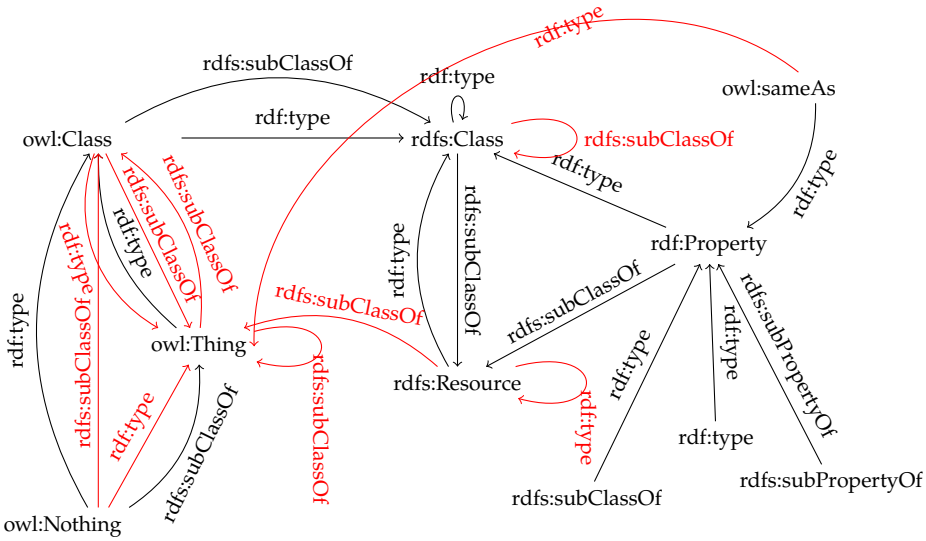


Figure 2: The "logic kernel" of LOD-a-lot. Red edges are additional edges not included in the original design (indicated in black) of RDF, RDFS, or OWL.

Next, we show some key classes and their relations as well as their representation and relations in the LOD-a-lot. Figure 2 is the (integrated) "logic kernel" of LOD-a-lot. The black edges are from the definitions of RDF[7], RDFS[8], and OWL[9]. As illustrated, there are additional edges (in red)[10] that have been included in the LOD cloud (and thus collected and integrated to the LOD-a-lot) which "hacked" the original design of logical foundation of RDF, RDFS, and OWL. These additional edges

7 https://www.w3.org/1999/02/22-rdf-syntax-ns#.
8 https://www.w3.org/2000/01/rdf-schema#.
9 https://www.w3.org/2002/07/owl#.
10 These additional edges were first generated using the Python script available on GitHub at https://github.com/shuaiwangvu/Logical_Inconsistency_LOD. The author then manually revised them and selected representative edges in this example. Many more edges were reported but not included in the figure.

can lead to confusion and errors, particularly when considered in logical inference. The LOD-a-lot is not only erroneous but can be also difficult to refine, especially considering its scale. This is because many mistakes were inherited in the integration. This analysis shows that it requires specially designed algorithms, which are to be presented in the remaining chapters of this thesis.

1.2.2 Knowledge Graphs, Properties, and Refinement

To explain more straightforwardly, a knowledge graph can be treated as a directed and labelled graph $G = \langle V, E, L, l \rangle$, where V is the set of vertices (nodes), $E \subseteq V \times V$ the set of relations (edges), and L is the set of edge labels [76]. $l : E \rightarrow 2^L$ is a function that assigns to each edge in E a set of labels belonging to L. The nodes V can be IRIs (denoted I), literals, or blank nodes. For some specific relations $R \subseteq L$, we denote $G_R = \langle V_R, E_R \rangle$ the edge-induced subgraph that only includes those edges whose labels are in R, with $V_R \subseteq V$ and $E_R \subseteq E$. When the relations in R are about identity, the corresponding subgraph can be called the identity graph.

An integrated KG [76] $\mathbf{G} = \langle \mathbf{V}, \mathbf{E}, \mathbf{L}, \mathbf{l} \rangle$ is a combination of a set of N knowledge graphs $\{G_1, \ldots, G_N\}$ where $\mathbf{V} = V_1 \cup \ldots \cup V_N$, $\mathbf{E} = E_1 \cup \ldots \cup E_N$, and $\mathbf{L} = L_1 \cup \ldots \cup L_N$. A function $\mathbf{l} : \mathbf{E} \rightarrow 2^{\mathbf{l}}$ assigns to each edge a set of labels, which is the union of the labels: $\mathbf{l}(e) = l_1(e) \cup \ldots \cup l_N(e)$. This definition takes advantage of the set union operation and does not consider how many times the edge appears in each graph in integration. This accords with the semantics: repeated statements can not introduce additional information and can thus be omitted. Alternatively, the resulting integrated KG can be viewed as a directed multigraph (with self-loops). A weight function can be introduced, e.g., in Chapter 3, to capture the number of edges with the same label. The weight represents the number of graphs integrated with a specified edge.

A Strongly Connected Component (SCC) of a directed graph is a set of vertices where any two of its vertices are connected by a path (i.e. the corresponding subgraph is strongly connected) and is maximal for this property: no additional edges or vertices can be included in the subgraph without breaking strong connectivity. Weakly Connected Components (WCCs) of a directed graph are defined in a similar way, except that the direction of edges is ignored. The process of removing edges from SCCs to make them acyclic is called *resolving cycles*, and the resulting graph without cycles is a Directed Acyclic Graph (DAG) [86]. In the case where all relations are symmetric, the (sub)graph can be considered an undirected graph. For an undirected graph, a Connected Component (CC) is a group of vertices where there is a path between any two vertices in the group. When restricted to a specific symmetric relation, a weakly connected component of the corresponding directed graph

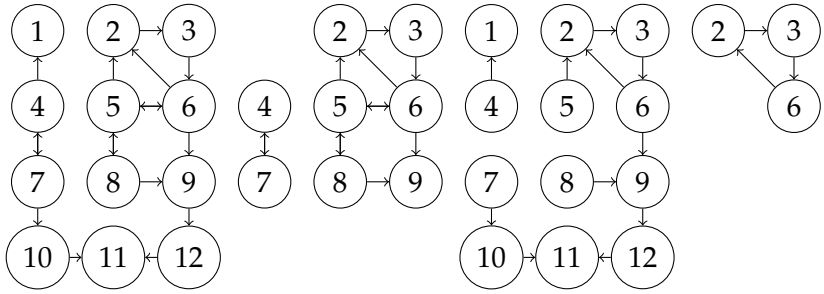


Figure 3: An example graph and its variants (from left: G, G^{SCC}, G', G'^{SCC}). G' is obtained by removing cycles with two entities from G.

is the same set of vertices as that of the connected component of the corresponding undirected graph. In practice, these components can also refer to the corresponding subgraphs. In an ideal case, logical properties should be reflected in the graph structures, and vice versa. For instance, an asymmetric relation's two-node cycle implies inconsistency. Likewise, self-loops are invalid for irreflexive relations. In contexts where there is no confusion, the definitions of components above could also refer to the subgraph about the selected vertices.

In an ideal case, logical properties should be reflected in the graph structures, and vice versa. For instance, in a graph of asymmetric relations, a two-node cycle implies inconsistency. Likewise, self-loops are invalid for irreflexive relations. When studying class subsumption relations where p is restricted to `rdfs:subClassOf`, the subgraph is solely about the subsumption relations on classes, which is a directed graph with self-loops. Cyclic class relations can be in the form $A \sqsubseteq A$ (a reflexive relation; self-loop), or $A \sqsubseteq B$ and $B \sqsubseteq A$ (size two cycle), or more generally $A_1 \sqsubseteq A_2 \sqsubseteq ... \sqsubseteq A_N \sqsubseteq A_1$ (size N).

Figure 3 presents an example of a graph G with its introduced variants: G^{SCC}, G', and G'^{SCC}. Note that cycles of size two are not necessarily SCCs of size two as they can be nested into other cycles and form a bigger SCC (e.g., size-two cycle between node 5 and 6). G' is a graph with size-two cycles removed[11] and G'^{SCC} is the corresponding graph of SCCs (containing only one SCC).

There are efficient algorithms for computing the SCCs of a graph, such as Tarjan [36], which take linear time $O(|V| + |E|)$ assuming constant time to retrieve edges. It is useful to know that cycles in a graph G can never span across multiple SCCs (since if there were any such cycle, the SCCs involved would form a bigger SCC, which contradicts its maximality w.r.t. strong connectivity). Therefore, since the

11 Given that the number of nodes involved in such size-two cycles can be significantly fewer than that in SCCs, one can take them for manual assessment when the quantity of such pairs is significantly restricted.

cycles in G are always contained inside a single SCC, and since the collection of all SCCs of G forms a partition of the vertices of G, we can safely divide-and-conquer the process of resolving cycles in G across all SCCs of G. This allows us to focus the cycle resolution locally in comparison with inefficiently and exhaustively listing all simple cycles as in [82].

A pilot case study on the refinement of a graph of class subsumption (i.e., a subgraph restricting the relation to `rdfs:subClassOf`) is included in Appendix B. As the pilot study shows, this thesis takes a **hybrid approach**: we take into consideration not only logical properties (class subsumption in this case study), knowledge representation, and automated reasoning, but also graph properties (cycles and strongly connected components in this case study) and related tools. Next, we present the research questions in this thesis and its structure.

1.3 Research Questions and Thesis Structure

Debugging a small KG can be done manually by conducting a detailed examination of its structure and reviewing triples where errors are present. Validation of larger KGs can be done using the Shapes Constraint Language[12] (SHACL), performing correctness checks based on patterns [20], or using inference engines for the detection of inconsistency in KGs [52]. As the scale of KGs increases and when integration is taken into consideration, new scalable algorithms may need to be designed that take into account different information beyond the explicit representation of triples. Due to the lack of gold standards and ground truth, measuring the performance of such algorithms can be challenging. This leads to the main research question of this thesis.

How can we take advantage of the graph structure of large integrated knowledge graphs for analysis and refinement?

12 https://www.w3.org/TR/shacl/

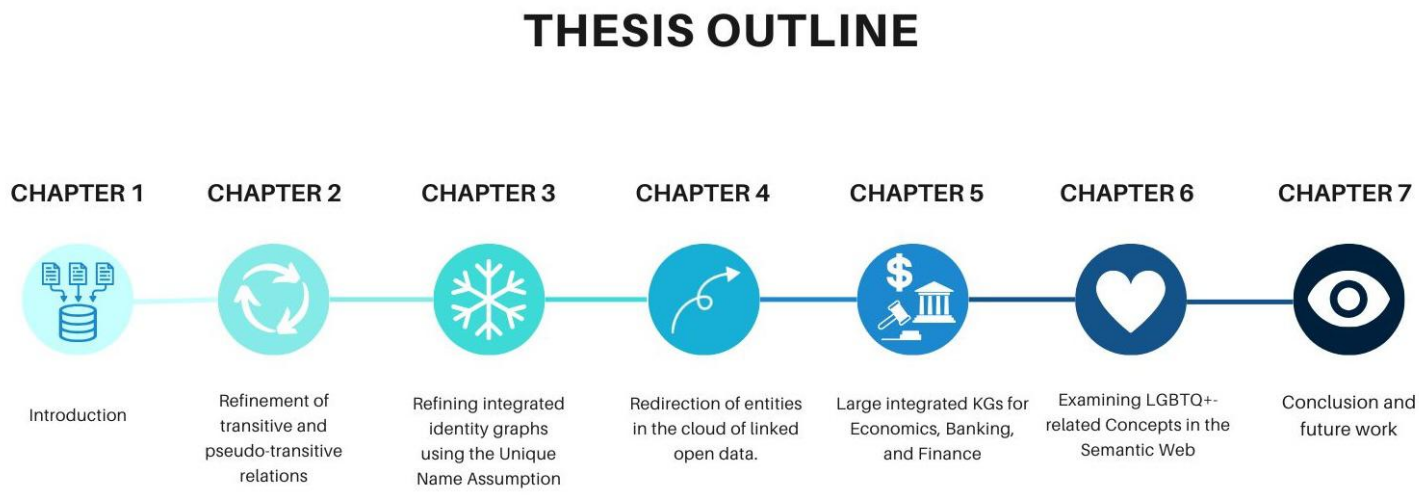


Figure 4: Thesis outline

This main research question has been grounded in research projects that have been captured in the following chapters as illustrated in Figure 4. Chapter 2 focuses on the refinement of graphs of relations that suggest transitivity. Chapter 3 takes advantage of the UNA for the refinement of the identity graphs. We provide a study of KG evolution by understanding the problems of redirection of URIs for the entities in the identity graph in Chapter 4. This thesis also includes some domain applications. An examination of KGs and their integration in domains of Economics, Law, and Finance was carried out in Chapter 5. An in-depth study of LGBTQ+ concepts in the semantic web and their relations in Chapter 6. Finally, we end the thesis with conclusions and some future work in Chapter 7. Next, we provide details of our research questions in each chapter with a summary of their methodology.

Chapter 2 studies the refinement of subgraphs corresponding to relations that suggest transitivity, such as `dbo:isPartOf`, `skos:broader`, and `dbo:subsequentWork`. More specifically, we use *pseudo-transitivity* for cases where the transitivity is formally asserted (see Chapter 2 for the definition). Our first research question of the thesis is below.

RQ1.1: How can we design algorithms to make knowledge graphs acyclic with respect to specific transitive or pseudo-transitive relations, while preserving as much original information as possible?

Chapter 2 presents an algorithm with an SMT solver employed to refine transitive and pseudo-transitive relations for large integrated KGs by removing as few edges as possible to obtain acyclic graphs. Our approach is independent of domain and language, with better precision than general-purpose graph-theoretical methods. We also discuss its scalability and efficiency. Following the intuition that the more graphs contain certain statements, the more certain they are, we extend our algorithm by allowing the use of weighting schemes to heuristically determine which possibly erroneous edges should be removed to make the graph cycle-free. The research corresponding to Chapter 2 has been published in the following papers.

- Shuai Wang, Joe Raad, Peter Bloem, and Frank van Harmelen. Refining transitive and pseudo-transitive relations at web scale. In Ruben Verborgh et al., editors, *The Semantic Web - 18th International Conference, ESWC 2021, Virtual Event, June 6-10, 2021, Proceedings*, volume 12731 of *Lecture Notes in Computer Science*, pages 249–264. Springer International Publishing, 2021.
- Shuai Wang, Peter Bloem, Joe Raad, and Frank van Harmelen. Submassive: Resolving subclass cycles in very large knowledge graphs, 2020. Workshop on Large Scale RDF Analytics, DOI: 10.48550/arXiv.2412.15829

Next, we study the refinement of identity graphs using an adapted definition of the Unique Name Assumption tailored to large integrated KGs in Chapter 3. It focuses on the following two research questions [87].

RQ2.1: How can we formally define and validate a Unique Name Assumption (UNA) for large integrated knowledge graphs to support identity graph refinement?

RQ2.2: Can the UNA be used for the design of an algorithm to detect erroneous identity links in practice reliably?

We develop new algorithms for the removal of mistaken identity statements. We take a similar approach as Chapter 2 by employing an SMT solver. We also provide some indicators and a manually annotated gold standard for evaluation. The results have been published in the following paper [87].

- Shuai Wang, Joe Raad, Peter Bloem, and Frank van Harmelen. Refining large integrated identity graphs using the Unique Name Assumption. In Catia Pesquita et al., editors, *The Semantic Web - 18th International Conference, ESWC 2023, Hersonissou, Greece, May 28 - June 1, 2023, Proceedings*. Springer Nature, 2023.

The semantic web is dynamic, so are the KGs relying on it. In Chapter 4, we study the evolution of entities in large KGs by examining IRI redirection, a phenomenon that is widely observed. Such redirection may occur due to an update of the namespace, a different encoding scheme, or other reasons. We perform quantitative analysis and study the semantics of redirection and investigate whether redirection reflects the evolution of entities in the LOD cloud. Furthermore, we describe characteristics of the graphs created by redirection links. More specifically, in Chapter 4, we study the following two research questions [49].

RQ3.1: How can we interpret and model the implicit semantics of IRI redirection in integrated identity graphs?

RQ3.2: What are the properties and structure of the redirection graphs?

Different from the approach of previous research questions, for RQ3.1, we obtain and examine redirection chains. We classify the scenarios of redirection and estimate the proportion of redirection that can be interpreted as identity links. As for RQ3.2, we study the redirection graphs by performing some statistical analysis and examining their graph-theoretical properties. The research in this chapter is based on the following paper published in 2022:

- Idries Nasim, Shuai Wang, Joe Raad, Peter Bloem, and Frank van Harmelen. What does it mean when your uris are redirected? Examining identity and

redirection in the LOD cloud. In Damien Graux et al., editors, *Proceedings of the 8th Workshop on Managing the Evolution and Preservation of the Data Web (MEPDaW) co-located with the 21st International Semantic Web Conference (ISWC 2022), Virtual event, October 23rd, 2022*, volume 3339 of *CEUR Workshop Proceedings*, pages 36–45. CEUR-WS.org, 2022.

The research in this thesis ends with two domain applications. The first one is included in Chapter 5. We studied several aspects mentioned above in domain-specific integrated KGs. More specifically, we selected and integrated some KGs in economics, finance, and banking [76]. We demonstrate by statistical and graph-theoretical analysis how integration results in more entities with richer information [76]. We showed, when scaling down and restricting to limited number of domains, how some of the previously addressed issues can be manually resolved with no problem-specific algorithms required (in contrast to the KGs in Chapter 2 and Chapter 3). Finally, we study the sources of error, their refinement, and discuss the benefits of this integration. The research in this chapter has been published in the following paper.

- Shuai Wang. On the analysis of large integrated knowledge graphs for economics, banking and finance. In Maya Ramanath and Themis Palpanas, editors, *Proceedings of the Workshops of the EDBT/ICDT 2022 Joint Conference, Edinburgh, UK, March 29, 2022*, volume 3135 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2022.

We study the following two research questions. In comparison with the original paper, the latter has been made explicit in this thesis.

RQ4.1: How can the integration of domain-specific KGs in finance and economics enhance entity descriptions and contribute to identifying errors?

RQ4.2: How do refinement challenges differ between domain-specific and general-purpose knowledge graphs?

The second domain application concerns a more complicated case where concepts face ambiguity and drift in their semantics: LGBTQ+-related concepts. The research in this chapter is based on the following spotlight paper [80].

- Shuai Wang and Maria Adamidou. Examining lgbtq+-related concepts in the semantic web: Link discovery, concept drift, ambiguity, and multilingual information reuse. In Mehwish Alam et al., editors, *Knowledge Engineering and Knowledge Management*, pages 1–17, Cham, 2025. Springer Nature Switzerland

RQ5.1: How can we construct a knowledge graph that captures identity-related information regarding LGBTQ+ concepts and their relations in representative conceptual models?

RQ5.2: How can the constructed knowledge graph be used to examine, enrich, and maintain evolving LGBTQ+ representations, including link discovery, change tracking, and multilingual enrichment?

As for RQ5.1, we construct a KG about LGBTQ+-related concepts by extracting entities and identity-like relations from published ontologies, structured vocabularies, and other conceptual models. We study RQ5.2 by demonstrating its use in three cases: a) we use the KG obtained to find missing links between conceptual models, b) we use the KG to help with identifying the change of concept, and c) we evaluate how much multilingual information can be reused to enrich entities.

Finally, we conclude the thesis and present ideas for future research in Chapter 7. The list of prefixes and namespaces used in this thesis, the index for tables, figures, and a pilot study are in the appendix. The pilot study is based on the following paper.

- Shuai Wang, Peter Bloem, Joe Raad, and Frank van Harmelen. Submassive: Resolving subclass cycles in very large knowledge graphs, 2020. Workshop on Large Scale RDF Analytics, DOI: 10.48550/arXiv.2412.15829

This thesis presents research studies on consistency, identity, source of error, evolution, and many other aspects of such large integrated forms of knowledge. Figure 5 outlines the tasks of this thesis. As illustrated, the research studies in the chapters share similar tasks. Together with my coauthors, I performed data analysis (Chapter 2, 3, 4, 5, 6), constructed evaluation datasets through manual annotation (Chapter 2, 3, 4, 6), developed new algorithms, and benchmarked their performance (Chapter 2, 3). Moreover, we attempted to explore the discovery of links between entities (Chapter 4, 6) and the reuse of information from other datasets for the enrichment of multilingual information (Chapter 6). In addition, we study the dynamics of KGs with a focus on the drift of concepts and redirection of URIs (Chapter 6). A summary of the scientific contribution is in the Appendix C.

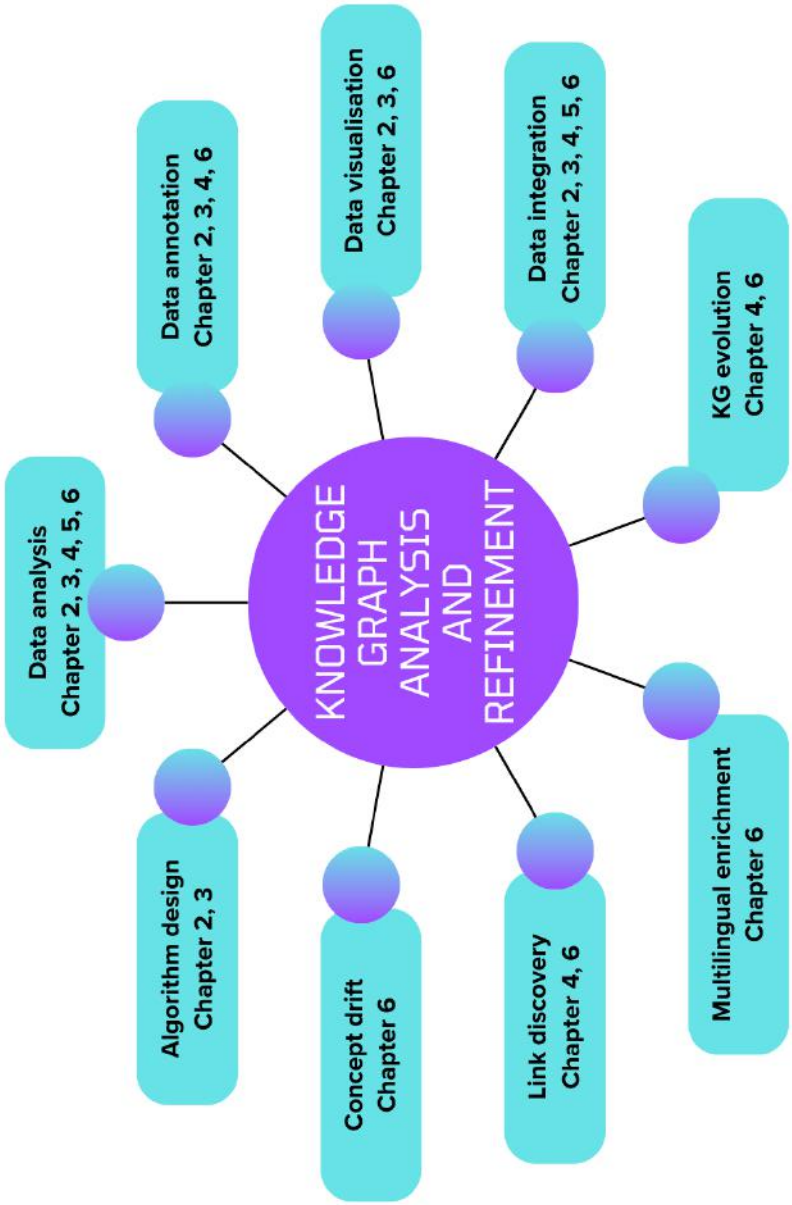


Figure 5: An overview of the main tasks covered by this thesis

2 REFINING KGS OF TRANSITIVE AND PSEUDO-TRANSITIVE RELATIONS

> I do not know what I may appear to the world, but to myself I seem to have been only like a boy playing on the sea-shore, and diverting myself in now and then finding a smoother pebble or a prettier shell than ordinary, whilst the great ocean of truth lay all undiscovered before me.
>
> Isaac Newton

Combining knowledge graphs whose basis are linked data can result in undesirable graph structures and logical inconsistencies. Refinement methods that can detect and repair such undesirable graph structures are therefore of crucial importance. In this chapter, we provide an in-depth exploration of a challenge that encompasses multiple datasets. These datasets are about both transitive relations and those we refer to as "pseudo-transitive" relations [86]. We introduce an algorithm designed for knowledge graph refinement which examines the graph's structure and enables the extension with weighting schemes [86]. These schemes heuristically assess which potentially inaccurate edges should be eliminated to ensure the graph is cycle-free. Before the end of this chapter, we provide manually labeled gold standards, as well as benchmarks using them. This chapter is based on two papers [82, 86]. The text of this chapter is based on [86]. For completeness, we include a summary of an early attempt in Appendix B.

2.1 Introduction

Integrating multiple knowledge graphs of linked data can result in logical inconsistencies or undesirable graph structures. For transitive relations, this can result in chains of relation instances forming complex nested cycles involving many entities across datasets in the corresponding graph. In practice, even logically valid cycles

may have negative consequences. For example, a cycle of `rdfs:subClassOf` triples in an intended hierarchy enforces equality of all classes in the cycle, which may prevent algorithms such as query expansion from termination. To ensure data quality, refinement methods have been developed [52]. However, these methods often depend on domain-specific functionalities [32], or limited to a specific relation (e.g. owl:sameAs) [57,82] or suffer from limited scalability [82]. Such limitations call for the development of scalable and domain-independent algorithms.

This chapter presents a new approach for detecting undesirable cycles in transitive relations. It uses graph structural characteristics and a heuristic notion of reliability of triples, without the need for any domain-dependent information such as labels, comments and other textual information in context [32]. Our approach (i) is independent from domain and language, (ii) has a better precision than general-purpose graph-theoretical methods and (iii) maintains good scalability and efficiency.

As mentioned in Chapter 1, in an ideal scenario, the graph structure reflects logical properties and vice versa. For example, when a relation is asymmetric, any cycle of size two in its graph violates consistency. Similarly, for irreflexive relations any self-loop is invalid. This suggests the use of graph-theoretic algorithms to detect logical inconsistencies. In OWL, the transitivity of a relation is typically specified directly through *owl:TransitiveProperty*. In this work, we extend to what we call *pseudo-transitive* relations: that of a sub-property or the inverse of a (pseudo-)transitive relation and those whose intended semantics is assumed to be both transitive and anti-symmetric, even if not formally asserted. In this chapter, we exclude equivalence relations (e.g. `owl:sameAs`) and those whose (pseudo-)transitivity are mistakenly asserted or implied on the LOD Cloud (e.g. `foaf:knows`).

Besides the graph structure, another feature to be used for the refinement of knowledge graphs that is independent from domain and language is the *reliability* of triples. While there can be different heuristics, we measure reliability of an edge by counting the number of occurrences of this edge across datasets of the web-scale integrated graph. For small self-sufficient datasets, this feature is not of great value because the logical foundations of knowledge graphs dictate that repeated statements in datasets are redundant. However, such a feature has been shown to be useful for the ranking of documents and entities [31] and the identification of erroneous assertions and improvement of data quality [14] when the sources of data are present. Figure 6 is an example subgraph of `skos:broader` with such weights extracted from the LOD Laundromat 2015 crawl [8]. It is more likely that `dbc:Numbers` is a broader than `dbc:Integers` (weighting 72), while it is unlikely that `dbc:Integers` is a broader concept for `dbc:Numeral_systems` (weighting 1), showing that weights can indicate the reliability of edges. This example also shows that the relation between some entities can be ambiguous, making it difficult to construct a perfect gold standard. For example, some may believe that numbers are parts of numerical systems while oth-

ers may think the study of numbers includes the study of numerical systems. Finally, it also indicates that the weights of edges in the neighborhood can have an impact on the reliability of edges.

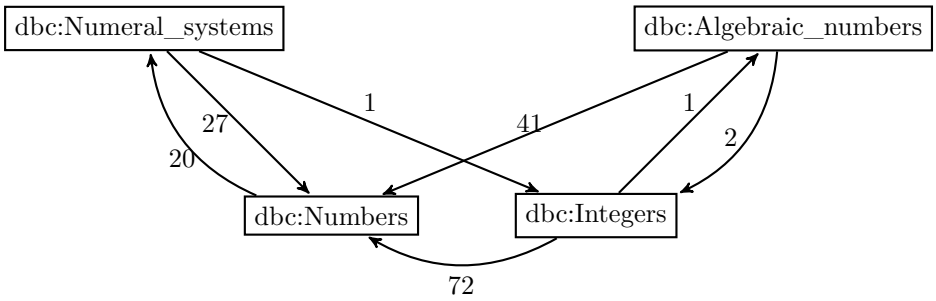


Figure 6: An example subgraph of `skos:broader` with weights.

We study the research question below.

RQ1.1: How can we design algorithms to make knowledge graphs acyclic with respect to specific transitive or pseudo-transitive relations, while preserving as much original information as possible?

The hypotheses that we pursue in this chapter are as follows:

H1: By taking graph structural properties (how edges are involved in complex nested cycles) into account, we can make knowledge graphs acyclic while removing fewer edges than graph theoretical methods.

H2: Taking the reliability of triples into account improves the accuracy for identifying erroneous edges.

This chapter presents an algorithm to refine transitive and pseudo-transitive relations for large integrated knowledge graphs at web scale by removing as few edges as possible to obtain acyclic graphs. More specifically, the chapter makes the following contributions.

1. a new metric for the hardness of resolving cycles based on strongly connected components.
2. a generic scalable approach for the refinement of (pseudo-)transitive relations using an SMT solver by exploiting Strongly Connected Components.
3. an evaluation that shows how taking into account the reliability of triples can improve the precision of the graph refinement algorithm.
4. a dataset of several widely used (pseudo-)transitive relations with their reliability weights.

5. a new gold standard of thousands of manually annotated triples to be used in the evaluation and comparison of graph refinement algorithms.

This chapter is structured as follows: Section 2.2 discusses related work. Section 2.3 describes the dataset and analyses its complexity. Section 2.4 presents our approach for refining (pseudo-)transitive relations. Section 2.5 presents the implementation details, our gold standard, and the conducted evaluation. Section 2.6 discusses the results and concludes the chapter.

2.2 Related Work

2.2.1 Knowledge Graph Refinement Methods

According to Paulheim's survey [52], there are two main goals for knowledge graph refinement methods: *completing* the knowledge graph with missing knowledge, and *correcting* asserted information. This work falls into the latter category of approaches, as we aim at refining transitive and pseudo-transitive relations by removing edges that lead to unwanted cycles and are potentially erroneous. The closest predecessor of our work is the approach by [82], introduced for refining edges of `rdfs:subClassOf` by exhaustively listing simple cycles[1] and removing minimal edges so that the resulting graph is cycle free. However, this approach faces a combinatorial explosion when listing all simple cycles of large nested clusters, and therefore cannot be applied on some relations we study in this work. Sun et al. [66] propose similar strategies for removing edges causing cycles in a graph. However, this approach requires inferring a graph hierarchy (e.g., using a Bayesian skill rating system), and has been tested only on synthetic datasets and the Wikipedia category graph. Another recent approach that targets the refinement of categorical and list information is introduced by [32]. To our best understanding, this graph-based refinement approach relies on external information of hypernyms, and applies only to English DBpedia categories and lists. Moreover, and similarly to the work presented in [27], these approaches assume the existence of a hierarchy and takes advantage of pre-defined roots. In this work, we show that such hierarchies are frequently violated in the Web, therefore the applicability of such approaches becomes limited in such context. Finally, the graph-based approach presented in this work is similar to other approaches that have also exploited the graph structure for detecting and removing different types erroneous edges at the scale of the Web, such as type [53] and identity links [57].

1 A simple cycle is a cycle in which the only repeated vertices are the first and last vertices.

2.2.2 General MWFAS Algorithms

In this section we discuss general-purpose graph algorithms for making graphs cycle-free. When restricting to a single relation, the problem of resolving cycles is identical to finding the *Maximum Weighted Directed Acyclic Subgraph* (MWDAS).[2] Historically, the removed edges are also called *arcs* and form a *feedback arc set* (FAS). Therefore, the problem is equivalent to *Minimum Weighted Feedback Arc Set* (MWFAS), and we will use these names for the rest of the chapter.

Existing algorithms either rely on an order using heuristics or remove all incoming or outgoing edges of each node to guarantee freedom from cycles. Other FAS/MWFAS methods include a modified version of a depth-first search algorithm, ant-inspired Monte Carlo algorithm [43], etc. These algorithms are not designed to be extended to capture logical or other additional properties. The MWFAS problem belongs to Richard. M. Karp' s famous list of 21 NP–complete problems and is APX-hard. Despite that there is a hard limit on its approximation quality, there are polynomial-time approximation algorithms.

The following is a summary of some algorithms that scales to at least tens of millions of edges according to the paper [64], where more details are presented.

KwikSort(KS). The algorithm was inspired by the Quicksort algorithm. The underlying idea is that the vertices can be sorted based on the number of back arcs induced. Given a starting linear arrangement, the algorithm recursively perform a sorting process by comparing against the pivot elements. The algorithm runs at $O(n \log n)$ when assuming the arc membership can be tested in constant time. We use the implementation with optimisation by using $O(n \log n)$ additional space. Since it takes a random initial linear arrangement, it makes sense to take the best result out of several attempts within time limit. The original paper used the best result of 200 runs.

Greedy(GRD). This algorithm greedily move all "sink-like" vertices and append to a sequence s and the "source-like" vertices and insert to the front of s. More specifically, the existing implementation [64] follows the idea of bins [93] for the selection of vertices in each iteration. The bins are defined as follows:

$$V_{n-2} = \{u \in V | d^-(u) = 0; d^+(u) > 0\} \quad (1)$$

$$V_{2-n} = \{u \in V | d^+(u) = 0\} \quad (2)$$

$$V_d = \{u \in V | d = \sigma(u); d^+(u) > 0; d^-(u) > 0\} \quad (3)$$

2 Note that the resulting graph may not be a spanning tree but a set of directed acyclic graphs (DAGs). Therefore, this problem cannot be solved by minimum spanning tree algorithms.

with $d \in [-n+3, n-3]$ in (3). Every vertex $u \in V$ falls into exactly one of these $2n-3$ bins. By using s as a linear arrangement and picking all the feedback arcs, it minimize the number of arcs with different orientation. The optimised *ddl* version takes advantage of a data structure named double-linked lists, while the *array* version uses three flat arrays that mimic the behavior of the lists. Both versions run in time $O(m+n)$ and uses $O(m+n)$ space. It has a guarantee of $\frac{1}{2}|E| - \frac{1}{6}|V|$ but experiments from [64] observed that the size of FAS is drastically smaller than the worst-case bound.

BergerShor (BS). For a graph $G = (V, E)$, this algorithm begins with a random permutation over the vertices V. It then processes each vertex in order by comparing its in-degree and out-degree. If a vertex has more incoming arcs than outgoing ones, the incoming ones are removed and added to a set E' while the outgoing arcs are removed and discarded. The collected arcs E' form an acyclic graph $G' = (V, E')$ while its counterpart are the arcs removed. The algorithm runs in time $O(m+n)$ and produces an acyclic subgraph containing at least $(1/2 + \Omega(1/\sqrt{d_{max}}))|E|$ edges where d_{max} is the maximum vertex degree. Experiments from [64] show that the algorithm far outperforms the worst-case bound.

Depth-first traversal algorithm (DFS). In addition, we can adapt the depth-first traversal algorithm and remove all arcs that form a cycle during the search to ensure that the resulting graph is acyclic. Its runtime complexity is $O(m+n)$. The algorithm does not make any intelligent decision nor minimize the resulting size of FAS.

Graph structure reflects logical properties and vice versa. For example, when a relation is asymmetric, any cycle of size two in its graph violates consistency. Similarly, for irreflexive relations, any self-loop is invalid. This suggests the use of graph-theoretic algorithms to detect logical inconsistencies. In OWL, the transitivity of a relation is typically specified directly through *owl:TransitiveProperty*. In this work, we extend to what we call *pseudo-transitive* relations: that of a sub-property or the inverse of a (pseudo-)transitive relation and those whose intended semantics is assumed to be both transitive and anti-symmetric, even if not formally asserted. In this chapter, we exclude equivalence relations (e.g. `owl:sameAs`) and those whose (pseudo-)transitivity are mistakenly asserted or implied on the LOD Cloud (e.g. `foaf:knows`).

2.3 (Pseudo-)Transitive Relations in the LOD Cloud

2.3.1 Dataset

In this work, we use the LOD-a-lot dataset [25] as a representative copy of the LOD Cloud. This compressed data file of 28 billion unique triples is the result of the integration of over 650K datasets that are crawled and cleaned by the LOD Laundromat in 2015 [8]. In the LOD-a-lot dataset, there are 2,486 relations explicitly stated as `owl:TransitiveProperty`, used in more than 776 million triples (2.7% of all triples). When the semantics of `rdfs:subPropertyOf` and `owl:inverseOf` are exploited, the number of (pseudo-)transitive relations increases to 8,804 relations, used in around 5.5 billion unique triples (19.5% of the triples). Our manual examination shows that a number of these properties are incorrectly asserted or inferred, such as the widely used `foaf:knows` relation.

For transitive relations, graph characteristics can reflect the logical properties, and vice versa. For example, irreflexive and antisymmetric relations (e.g. `iwwem:dependsOn`) allow for no cycle anywhere in the graph. We consider `skos:broader` a pseudo-transitive relation, as it was not designed to be a transitive property despite being a subproperty of `skos:broaderTransitive`, which is typed `owl:TransitiveProperty` [47]. Unless otherwise specified, we assume that the graph of relations such as `rdfs:subClassOf` and `geo:parentFeature` should be cycle-free despite the logical validity of cycles. Our maual examination also found that many relations are defined together with their inverse (e.g. `skos:broader` and `skos:narrower`). There can also be a relation like that of equivalence (e.g. `owl:sameAs, rdfs:equivalentClassOf`). This chapter examines a selection of 10 relations (see e.g. Figure 8 and Table 2). These are popular relations, all of them directly typed as `owl:TransitiveProperty` with over 100K triples. We exclude the few that actually represent equivalence relations, or whose biggest SCC has less than 10 vertices unless its inverse is to be studied.

2.3.2 Strongly Connected Components Analysis

To get a sense of how difficult it is to make graphs cycle-free, we introduce in this section a number of metrics. We may turn to standard metrics for the degree of transitivity of a graph, such as the transitivity index T (the number of actual triangles in a graphs as a fraction of the number of all possible triangles), the average clustering index C (the average over the local clustering coefficients of all vertices, where a local clustering coefficient of a vertex is the actual number of edges in the

direct neighbourhood of the vertex divided by the possible number of such edges), or the global reaching centrality (GRC) [48]. These measures can be useful for the understanding of graph-theoretical properties. Details of the definitions below can be found in Section 1.2.2.

1. Average Clustering:

$$C = \frac{1}{n} \sum_{u \in V} \frac{2T(u)}{\sigma(u)(\sigma(u) - 1)}$$

2. Transitivity:

$$T = 3\frac{\#triangles}{\#triads}$$

3. Global Reaching Centrality:

$$GRC = \frac{\sum_{i \in V}[C_R^{max} - C_R(i)]}{N - 1}$$

Our analysis shows that there is no obvious relation between these measures and #SCC and maxSCC. More specifically, Table 1 shows that none of T, C or GRC manages to capture the size of SCCs or the hardness of cycle-resolution. Thus they cannot be used as a measure for cycle resolving. We therefore introduce new quantitative measures based on SCCs.

Relation	#Triples	C	T	GRC	#SCC	maxSCC
dbo:predecessor	358,244	0.011	0.039	0.034	4,299	2,408
sioc:parent_of	101,219	0.049	0.044	0.360	334	1,173
dbo:parent	105,868	0.017	0.137	0.047	921	979
rdfs:subClassOf	4,461,717	0.009	0.006	0.133	196	301

Table 1: Examples of Graph-theoretical Measures

When examining the SCCs of the LOD-a-lot knowledge graph regarding popular relations, we observe two facts: 1) cycles of size two are very common across the graphs. When not nested into other cycles, they are SCCs with two nodes (SCCs of size two), which is the most common type of SCC. This suggests the ambiguity in definition and semantics of the relation; 2) there often exist a very big SCC that covers a majority of nodes involved in the SCCs. This is very different from synthetic models typically used in the evaluation of MWFAS algorithms. The following are measures on how much the SCCs are due to size-two cycles, and other complex nested cycles.

Alpha measure. Let α be the number of edges in cycles of size two divided by the number of all edges in its SCCs $\alpha = f_\alpha(G) = |E_{C2}|/|E^{SCC}|$. By definition, $f_\alpha(G) = f_\alpha(G^{SCC})$. This gives the fraction of edges that can be determined locally if given additional information (e.g. the reliability on each edge).

Beta measure. Remove all the cycles of size two from G and obtain $G' = G \backslash E_{C2}$. The corresponding SCCs of G' form a graph G'^{SCC}. Let β be the number of edges in G'^{SCC} divided by the number of all edges in G^{SCC}: $\beta = |E'^{SCC}|/|E^{SCC}|$. Similarly, we have $f_\beta(G) = f_\beta(G^{SCC})$. This measures the proportion of edges to make decisions on if all edges in size-two cycles are not involved in any SCC. In other words, it gives a measure of the fraction of edges in more complex nested cases.

In our datasets, there is often one SCC that is significantly larger (with the most vertices and edges) than the others among all the SCCs of a graph. We refer to it as G^B when discussing its properties. To give a better intuition, figure 7 presents an outer pie chart represent the proportion of edges in the components of the SCCs corresponding to four relations: yellow for that corresponds to α, light blue for that of β, and grey for the rest. The outer pie chart is that of the entire graph. The diameter corresponds to the size of the graph (proportional to the square root of the total number of edges). The inner pie chart is for that of the biggest SCC: the brown and dark blue parts are the proportions of the edges corresponding to their α and β, respectively. The darker grey corresponds to the remaining. The radius of each slice represents the percentage in each category. From the chart, we can tell that the biggest SCC is where the majority of edges are for `skos:broader` while it is not the case for the rest. The proportion of edges that corresponds to β in `skos:broader` is significantly higher than that of `dbo:parent`. This visualisation gives an intuition of the hardness of the problem.

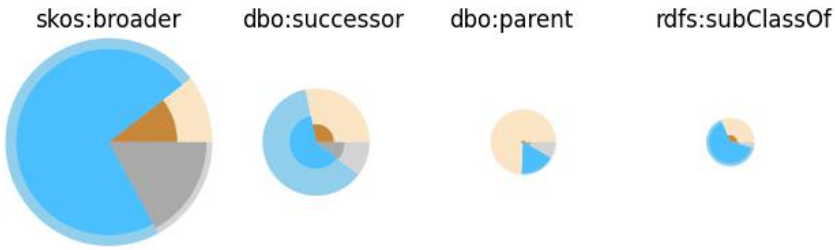


Figure 7: The Alpha-Beta measures of four representative relations

For the graph G in Figure 3, $\alpha = 0.5$ and $\beta = 0.25$. As for its biggest SCC, $\alpha = 0.4$ and $\beta = 0.3$. Figure 8 reports on the α and β values for the 10 selected relations in Table 2. The figure on the left illustrates the alpha-beta measure. In general, the greater α is, the more size-two cycles there are. The smaller β is, the more likely it is to resolve the cycles by simply making decisions on the edges of cycles of size two (e.g. `skos:narrower` has $\beta = 0$). On the other hand, `skos:broader` and `dbo:previousWork` are examples with more complex cycles nesting into each other.

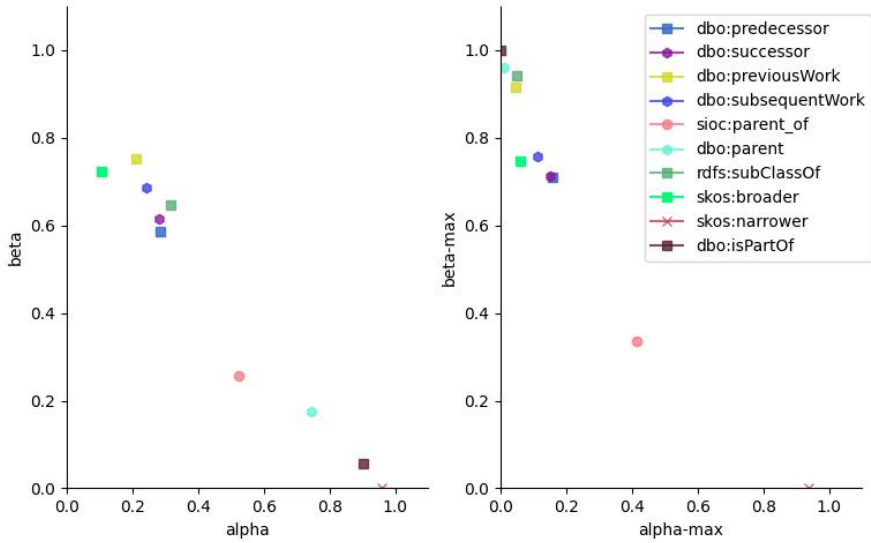


Figure 8: The Alpha-Beta measures of representative relations

An observation is that the tangent of a line crossing the origin and each point can indicate the hardness. This inspires the following definition.

Gamma measure: For an SCC G, the minimum fraction of decisions to be made to make G cycle-free can be captured by $\alpha + \beta$. Note that an SCC G gets harder when its β is greater, or α is smaller. This can be captured by β/α or $\beta - \alpha$. To avoid cases where $\alpha = 0$ and to make γ a term between 0 and 1, we use the latter and define $\gamma = f_\gamma(G) = (\alpha + \beta)(1 - \alpha + \beta)/2$. This gives a measure of the hardness to make an SCC cycle-free.

Delta measure: Note that using the Gamma measure, we can then estimate the effort required to make each of a graph's SCCs cycle-free. For a graph G in general, $\delta = f_\delta(G) = \sum_{s \in \kappa(G)} f_\gamma(s) * |E|$. It is a sum over the γ value multiplied by the number of edges of each of its SCCs.

Table 2 presents key information of the graphs of the 10 selected relations, together with the values of our metrics. For instance, among the 11.8M skos:broader edges, 356.9K of them are included in a total of 6.7K SCCs. The biggest SCC, G_R^B, captures 227K edges, amounting to 63.6% of the edges in at least one cycle. Due to its complexity and scale, G_R^B has a big δ. In correspondence, that of G_R is proportionally big. In contrast, the problem of its inverse, `skos:narrower`, is a much easier problem with only 48 edges involved in cycles with a very small δ value. Thus, it is possible to resolve the cycles by manually annotating each edge in its E_R^{SCC}. Comparing the entries of `skos:narrower` with `dbo:successor`, we can see that the δ measures can

reflect cases where some big graphs are not necessarily harder to resolve, which is not captured by graph-theoretical measures. These new measures provide a quantitative evaluation on the hardness of cycle resolving, help to study the nature of its complexity, and serve as references for the design of algorithms, choice of parameters as well as the sampling of data.

Table 2: Popular transitive and pseudo-transitive relations and their measures

Relation (R)	G_R		SCCs of G_R				G_R^B (the biggest SCC)			
	$\lvert E_R\rvert$	$\lvert V_R\rvert$	$\lvert E_R^{SCC}\rvert$	$\lvert V_R^{SCC}\rvert$	$\lvert\kappa(G_R)\rvert$	δ	$\lvert E_R^B\rvert$	$\lvert V_R^B\rvert$	γ	δ
skos:broader	11.8M	5.7M	356.9K	82.0K	6.7K	238.4K	277.0K	43.7K	0.6	188.1K
rdfs:subClassOf	4.4M	3.6M	1.4K	837	196	961.05	780	301	0.9	730.4
dbo:isPartOf	1.0M	408.3K	4.7K	3.8K	1.5K	312.8	60	29	1.0	60.0
skos:narrower	817.1K	737.3K	48	24	7	0.9	16	5	0.0	0.4
dbo:previousWork	551.2K	550.1K	10.6K	8.4K	1.5K	8.0K	710	469	0.9	639.2
dbo:subsequentWork	511.0K	527.5K	15.7K	11.9K	1.8K	10.8K	2.2K	1.5K	0.7	1.6K
dbo:successor	440.7K	417.3K	60.2K	38.0K	5.8K	36.2K	12.5K	5.9K	0.6	8.4K
dbo:predecessor	358.2K	348.1K	40.0K	25.8K	4.2K	22.9K	4.8K	2.4K	0.6	3.2K
dbo:parent	105.8K	97.0K	9.7K	4.3K	921	1.9K	1.5K	979	0.9	1.4K
sioc:parent_of	101.2K	46.6K	6.3K	2.1K	334	1.7K	4.3K	1.1K	0.3	1.5K

2.4 Algorithms

2.4.1 Algorithms for Cycle Resolving

We aim to design an algorithm that deals with knowledge graphs of (pseudo-) transitive relations that (i) does not rely on a ranking of nodes; (ii) captures the transitivity of relations; (iii) is capable of handling the complex structure resulted from large amount of nested cycles (graphs with high γ values); (iv) is as conservative as possible in removing edges when resolving the cycles; and (v) is extendable to capture other logical and graph properties. The following section presents our algorithm with an evaluation.

We present Algorithm 1 as a general purpose cycle-resolving method for refinement. The algorithm exploits off-the-shelve technology for SMT solvers (Satisfiability Modulo Theories) [12]. The algorithm does not deal with reflexive edges as they can be processed trivially and in linear time. There are three main steps in the algorithm. We first compute the SCCs of the input graph (line 3). Then, we perform partitioning over big SCCs to a given bound b_1 (line 4). This is due to the limit of SMT solvers' capability to handle large amount of clauses. Finally, we sample some cycles and repeatedly call an SMT solver to identify edges to be removed (line 5-17). In the following, we discuss the strategies adopted to deploy it at web scale.

Strategies for Graph Partition.

The graph partition problem is well-studied in graph theory. The minimum k-cut problem requires finding a set of edges whose removal partitions the graph to at least k connected components. There exist efficient algorithms and open-source implementations. However, our experiments show that breaking an SCC s into k partitions directly results in a significant amount of edges being removed, whereas our goal is to be as conservative as possible in our repair strategy. For reducing the amount of edges to be removed, our **Strategy P1** partitions the graph into two subgraphs and then computes the SCCs. This process is repeated until each of the resulting SCCs are within the size bound b_1. For weighted graphs (see the next section for how weights are computed), we can adopt a **Strategy P2** by first removing the edges with the lower weights in size-two cycles, and then use Strategy P1.

Strategies for Cycle Sampling.

The bottleneck for the earlier work in [82] was the combinatorial explosion when exhaustively listing all cycles of a graph. We therefore focus on sampling an amount of cycles in each iteration that balances the tradeoff between representative capacity

Algorithm 1: General-purpose algorithm for cycle resolving

Input: a graph G with no reflexive edges, its weight function f_w (optional), a bound b_1 for the number of maximum nodes for each SCC, and a bound b_2 for the number of hard clauses.
Result: a cycle-free graph H and a set of edges removed A.
Initiate A as an empty set;
Compute the set of SCCs as S;
Follow a graph partitioning strategy and reduce the size of S to bound b_1 as S' with removed edges collected and added to A;
while S' *is not empty* **do**
    Initiate S'' as an empty set;
    foreach $s \in S'$ **do**
        Follow a sampling strategy, and obtain cycles C from s with $|C| < b_2$;
        Initiate an SMT solver o;
        Introduce to o a set P of propositional variable p_e for each edge e of s;
        Encode cycles in C as hard constraints in o;
        Add to o a clause of each variable $p_e \in P$ as a soft constraint with weight $f_w(e)$ if f_w is present, otherwise 1;
        Run the solver o for optimal solution and decode the output model m;
        From the model m, collect the edges E to remove and let $A := A \cup E$;
        Obtain a graph s' from s with E removed;
        Compute the SCCs N of s' and update $S'' := S'' \cup N$;
    $S' := S''$;
Remove edges A from G and obtain H.

and redundancy. **Strategy S1** focuses on the edges: choose a random edge (s, t) in an SCC, then compute the shortest path from (t, s). This forms a cycle. In total we collect b_2 such cycles. As an alternative strategy, we can adopt the **Strategy S2** that selects two nodes randomly and computes the shortest path from one to the other, and back.

Resolving Cycles with an SMT solver.

This section gives details of the interaction with the SMT solver (line 9 and 13 in the algorithm). The SMT solver is used for two purposes: to satisfy all hard constraints and to satisfy the maximal amount of soft constraints.[3] The use of an SMT solver makes it possible to easily extend the current algorithm to weighted cases. In each iteration, for every $s \in S'$, we introduce a propositional variable p_e for each edge e. When there is a cycle $v_1, \ldots . v_k$, we add a hard clause $[\neg p_{(v_1,v_2)} \vee \ldots \vee \neg p_{(v_{k-1},v_k)} \vee \neg p_{(v_k,v_1)}]$ to the SMT solver (accumulated in conjunction). The clause is satisfied when at least one of the $p_{i,j}$ is assigned False in the returned model of the solver, which indicates the removal of the edge (i, j). To keep the maximal amount of edges, we add a soft clause $[p_e]$ for each edge e. The SMT solver performs a constrained optimisation process within a bounded time. The result of this is a near-optimal solution with the least amount of propositional variables set to False. From the model, we can retrieve edges to be removed to resolve all the encoded cycles. We repeat this process until all the SCCs are resolved and return the DAG and the removed edges. This approach can be easily extended to weighted cases, with the weight for each soft clause as the reliability for each edge.

2.4.2 Weights

Due to the logical foundation of knowledge graphs, repetition of statements is ignored because of the idempotency of the conjuction operator: $(\phi \wedge \phi) \leftrightarrow \phi$. Nevertheless, we believe that there is an important signal to be gained: the occurrence of the same triple in multiple knowledge graphs is an informal signal that multiple information providers have expressed support for. Thus, the chance that a statement is erroneous decreases with the number of knowledge graphs including this statement. Our algorithm takes the two kinds of weights for soft constraints (Algorithm 1, line 12). **Counted Weights**: the simplest way to obtain the weight of a triple is to count the number of occurrences across the graphs. The LOD-a-lot file consists of 650K datasets (graphs), making it feasible to compute such weights for popular relations; **Inferred Weights**: inspired by the observation and analysis in Sec-

3 A sub-optimal result is returned when an SMT solver reaches timeout.

tion 2.3.1, we take advantage of the logical redundancy between implied properties to compute weights. If a triple (A `rdfs:subClassOf` B) is present in the integrated graph, and there is also an equivalence relation `(A owl:equivalentClass B)` or `(B owl:equivalentClass A)`, then we make its weight 2 (i.e. we give more credence to (A `rdfs:subClassOf` B), otherwise 1). For `skos:broader`, we can take advantage of its inverse relation `skos:narrower`. If together with the triple `(A skos:broader B)` the triple `(B skos:narrower A)` exists in the dataset, then we assign weight 2 to the triple `(A skos:broader B)`, otherwise 1. While counted weights always exist, inferred weights are more restricted and less common and requires some manual examination. Still, we experiment different weighting scheme for the sake of comparison in evaluation.

2.5 Experiments and Evaluation

2.5.1 Implementation and Parameter Settings

We implemented our algorithm[4] in Python. We adopt the Python binding of METIS[5], a graph partitioning package based on the multilevel partitioning paradigm providing quick and high-quality partitioning. We use Z3[6] as SMT solver [12], and the `networkx` package[7] for the handling of graphs and SCCs.

Based on some trial-and-error experience, in the following experiments we set $b_1 = 15,000$ (i.e. maximum size of an SCC before requiring graph partitioning), and apply Strategy P1 with $k = 2$. To balance the trade-off between efficiency against accuracy, we obtain $b_2 = 3,000$ clauses at most and set the time limit for the SMT solver to 10 seconds for each SCC.

All experiments were conducted on a 2.2 GHz Quad-Core i7 laptop with a 16GB memory running Mac OS. All reflexive edges were eliminated in preprocessing.

2.5.2 Gold Standard

For the evaluation of hypothesis H2, we annotated a number of statements from the two most frequent (pseudo-)transitive relations (`rdfs:subClassOf` and `skos:broader`, according to Table 2). For each relation, we have two gold standards. In the first gold standard **G1**, we randomly pick 500 edges from E_R^{SCC}. The second gold standard separates SCCs of two nodes (**G2-a**, 200 edges) from the rest (**G2-b**, 500 edges). When

4 https://github.com/shuaiwangvu/Refining-Transitive-Relations
5 https://github.com/inducer/pymetis
6 https://github.com/Z3Prover/z3
7 https://networkx.github.io

sampling **G2-b**, we first assign a number on each SCC according to their δ-value and then sample the amount of edges assigned to each SCC randomly, thus providing an evaluation set for edges in complex nested cases. There are 1,199 unique edges in the gold standard of `skos:broader` with a total of 632 (52%) annotated 'remain' in contrast to 401 (33%) 'remove'. This analysis suggests that its under-specified definition caused confusion and subsequently resulted in a complex faulty graph structure. The great proportion of unknown entries for `rdfs:subClassOf` is discussed in Section 2.6.2.

The annotation process was conducted using the platform ANNit[8]. These gold standard datasets are online[9] together with detailed criteria, analysis and limitations. Its consistency was validated manually and by a Python script.

2.5.3 Efficiency Evaluation

In this section, we compare our refinement algorithm against other MWFAS algorithms. Table 3 presents the results of the number of edges removed for ten subgraphs of LOD-a-lot, both overall and within the SCCs. The highlighted cells show that our approach removes fewer edges during refinement. The result supports our Hypothesis `H1`. Both approaches are fast: general-purpose MWFAS algorithms take 2-12 seconds except KS, which may take up to 1 minute. Our algorithm takes 8-115 seconds except for `skos:broader`, which can take up to 8 minutes. Details of benchmarks are included in the repository of gold standard. The results in Table 3 are the best records of three runs. Finally, the results are validated to be free from SCCs except singletons.

8 https://github.com/shuaiwangvu/ANNit
9 https://zenodo.org/record/4610000

Table 3: The number of removed edges (both P1S1 and P1S2 are unweighted; the best results are underscored)

Relation \ Approach	BS		GRD		KS		DFS	P1S1	P1S2
	Overall	SCCs	Overall	SCCs	Overall	SCCs			
skos:broader	1.1M	327.0K	493.1K	356.9K	5.8M	177.1K	125.6K	114.8K	144.0K
rdfs:subClassOf	4.3M	1.1K	25.3K	430	219.2K	716	529	330	360
dbo:isPartOf	18.8K	3.2K	2,175	2,153	359.3K	2.3K	2.2K	2,143	2,169
skos:narrower	57.7K	33	3.4K	20	405.9K	21	21	21	22
dbo:previousWork	113.4K	7.5K	11.9K	2.5K	267.6K	5.3K	2.3K	1.9K	2.1K
dbo:subsequentWork	107.1K	11.1K	12.4K	3.7K	253.5K	7.9K	3.5K	2.9K	3.2K
dbo:successor	85.5K	43.7K	24.8K	17.6K	218.1K	29.9K	17.0K	13.3K	20.1K
dbo:predecessor	67.8K	28.9K	17.2K	11.8K	176.4K	19.9K	11.4K	9.0K	10.1K
dbo:parent	16.9K	7.2K	5.2K	3.9K	52.4K	4.8K	3,943	3,988	4.1K
sioc:parent_of	6.5K	6.0K	1.8K	1.6K	46.9K	2.5K	1.9K	1.2K	2.6K

Table 4: Number of removed edges $|A|$, precision p, and recall r for refinement

Method	skos:broader							rdfs:subclass						
	$\|A\|$	G1		G2-a		G2-b		$\|A\|$	G1		G2-a		G2-b	
		p	r	p	r	p	r		p	r	p	r	p	r
BS	1.1M	0.32	**0.85**	0.68	**0.72**	0.31	**0.91**	4.3M	0.40	**0.74**	0.40	**0.67**	0.54	**0.79**
GRD	493.1K	0.42	0.22	0.71	0.50	**0.40**	0.26	25.3K	0.42	0.40	0.35	0.45	0.57	0.21
KS	5.9M	0.33	0.52	0.74	0.53	0.28	0.46	2.1M	0.38	0.43	0.43	0.55	0.54	0.53
DFS	125.6K	0.35	0.37	0.68	0.49	0.34	0.34	529	0.43	0.42	**0.49**	0.63	0.55	0.29
P1S1-unweighted	114.8K	0.32	0.26	0.73	0.52	0.30	0.28	**330**	0.50	0.51	0.45	0.57	0.40	0.11
P1S2-unweighted	142.6K	0.32	0.35	0.73	0.52	0.31	0.37	350	0.49	0.44	0.45	0.57	0.58	0.15
P1S1-inferred	115.0K	0.31	0.25	0.73	0.52	0.30	0.28	**330**	0.50	0.51	0.45	0.57	0.40	0.11
P1S2-inferred	143.8K	0.33	0.38	0.73	0.52	0.30	0.36	354	0.49	0.46	0.45	0.57	0.60	0.14
P2S1-inferred	114.8K	0.31	0.25	0.73	0.50	0.30	0.29	**330**	0.50	0.51	0.45	0.57	0.40	0.11
P2S2-inferred	142.7K	0.33	0.35	0.73	0.52	0.31	0.37	356	0.50	0.47	0.45	0.57	0.58	0.15
P1S1-counted	95.4K	0.40	0.33	**0.78**	0.55	0.34	0.26	335	**0.53**	0.49	0.45	0.57	0.67	0.16
P1S2-counted	98.3K	0.42	0.38	**0.78**	0.55	0.34	0.28	354	0.51	0.45	0.45	0.57	**0.70**	0.20
P2S1-counted	**93.4K**	0.43	0.32	**0.78**	0.55	0.34	0.26	335	**0.53**	0.49	0.45	0.57	0.67	0.16
P2S2-counted	94.6K	**0.44**	0.35	**0.78**	0.55	0.32	0.24	357	0.50	0.45	0.45	0.57	0.66	0.17

2.5.4 Accuracy Evaluation

As for Hypothesis H2, we evaluate our algorithm's unweighted version against the two weighted versions (counted and inferred weights), as well as the MWFAS algorithms. Table 4 presents the precision (p) and recall (r) as well as the number of removed edges ($|A|$) for `skos:broader` and `rdfs:subClassOf`. Each entry represents the average of three runs. Taking weights into account (especially counted weights) has a positive impact on precision while maintaining a similar recall. Our approach achieves the best precision among all methods while removing the least amount of edges. As for `rdfs:subClassOf`, the impact on precision can be positive but unstable due to the limits to be discussed in the next section. When weights are not provided, P1S1 removes the least amount of edges. Otherwise, P2S1 is more optimal for our approach considering the balance between efficiency and accuracy. Overall, this evaluation gives positive support to our Hypothesis H2 and further enhances our conclusion for Hypothesis H1.

2.6 Discussion and Future Work

2.6.1 Summary

This chapter presented a new algorithm for the refinement of transitive relations and pseudo-transitive relations in very large knowledge graphs. We employed an SMT solver in implementation and evaluated on 10 datasets and validated our Hypothesis H1. As a proof-of-concept, we extended our work to weighted knowledge graphs and evaluated on our gold standard. The results provided positive support for our Hypothesis H2 and we also showed that taking weights into account during refinement has a good potential.

2.6.2 Discussion

The graph of `rdfs:subClassOf` has 4.4 million triples, of which 1.4K are in SCCs. Only 17 triples have inferred weights greater than 1, while 292 triples have such counted weights. The `skos:broader` graph has 11.8 million triples, of which 265.9K are among SCCs. There are only 39 triples with inferred weights of 2 compared to 284.6K for counted cases. It is clear that far fewer triples are assigned inferred weights than counted weights, making it a less general weighting scheme. Table 4 shows that inferred weights have no significant impact on the results due to their small number. The following focuses on counted weights.

Figure 9 plots the frequency distribution of counted weights for both datasets. It shows a power law distribution for the weights of `skos:broader`, implying that some relation instances have been stated repeatedly across the web. This justifies the use of frequency of triples as a heuristic for reliability. In comparison, `rdfs:subClassOf` is less popular and its frequency distribution is less clear and thus less reliable for decision making.

Finally, the unstable result of `rdfs:subClassOf` is mostly due to the biggest SCC which has 780 edges, amounting to 52% of all the edges in SCCs. All these edges come from a single big faulty dataset, and are all annotated 'unknown'. This explains the big variance for precision and recall as in Table 4.

2.6.3 Limitations and Future Work

The algorithm essentially grounds relations to propositional logic, thus making it possible to combine it with additional logical constraints with optimisation in the future.

For symmetric relations, we can map the vertices in size-two cycles to one in a new graph while keeping track of the correspondence between the new graph and the original graph. If an edge in the new graph is removed, we remove the corresponding edges of the original graph. Graph partition is an imprecise step in the algorithm. For example, among the 121.2K edges removed in the case of `skos:broader`, around 99.6K were identified during the graph partitioning step, amounting to 82.2%. Future work may optimise the parameters to balance the trade-off between accuracy and efficiency.

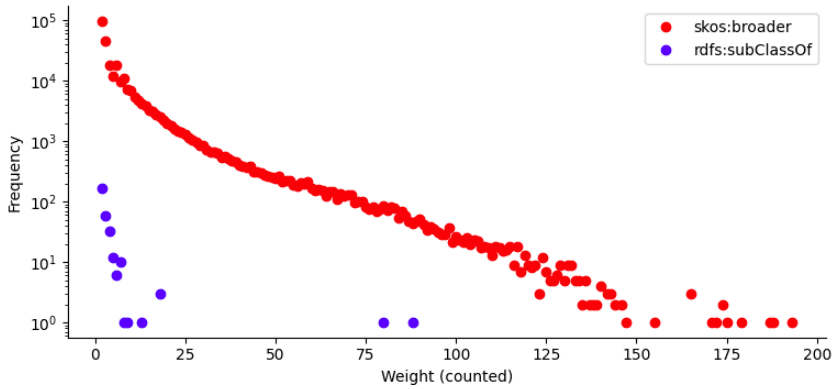


Figure 9: The frequency distribution of counted weights in SCCs

Figure 9 shows that the frequency distribution of `skos:broader` follows a power-law distribution. It can mislead the algorithm when an edge with a great weight is

actually erroneous. Different weighting scheme can be explored such as that in [33]. Another possible way to improve the accuracy of weights and reduce the number of ties of weights in size-two cycles is by taking the reliability or centrality of sources into account as in [14], for example. General-purpose MWFAS algorithms can be adapted to their weighted cases for future evaluation.

Our recall is limited due to the small amount of edges removed. In Section 2.5.4, we restricted to these two relations due to their popularity and the great effort required for manual annotation. We plan to extend the gold standard to relations with different alpha-beta measures.

3 REFINING INTEGRATED IDENTITY GRAPHS WITH THE UNA

> To become what one is, one must not have the faintest idea what one is.
>
> Friedrich Nietzsche

The Unique Name Assumption (UNA) [62] supposes that two terms with distinct identifiers from the same knowledge base do not refer to the same real-world entity. The UNA can be used to detect errors in large integrated knowledge bases [69]. Certain identity links can become inaccurate when they form part of a path connecting two entities that, despite being part of the same knowledge base, refer to distinct real-world objects. However, UNA is not always applicable due to the presence of redundant IRIs, which arise from different encodings, languages, namespaces, versioning, case formats, and other variations. Nevertheless, the UNA can be leveraged effectively to identify inaccurate links if properly adapted to accommodate such exceptions. To address this, we introduce a well-defined version of the UNA that tolerates multiple exceptions, termed the internal UNA (iUNA) [87]. In our research, we put forward an algorithm designed for refinement purposes, allowing for a comparative analysis of iUNA and other UNA variants. Analogous to the preceding chapter, this algorithm makes use of a SMT solver, capitalizing on its capability to efficiently process equality reasoning. Our evaluation focuses on detecting erroneous links within an identity graph (a subgraph consisting only of the `owl:sameAs` relation) comprising half a billion triples sourced from the LOD Cloud. We then evaluate the efficacy of our methodology in comparison with community detection algorithms, specifically the Louvain and Leiden algorithms.

3.1 Introduction

The question "*What is an entity?*" and the related question "*When are two entities equal?*" are not only longstanding philosophical questions[1] but are also longstanding technical issues in information systems [16]. The Semantic Web, and in its wake, Linked Open Data, have operationalised the notion of an "entity" as an Internationalized Resource Identifier (IRI): each is represented as an IRI, and using the same IRI implies referring to the same entity. Entities are connected by the identity links (e.g. `owl:sameAs`) to form identity graphs. Many existing approaches for detecting errors in identity graphs require information such as vocabulary alignments, textual descriptions [19, 60] or the presence of a large number of ontology axioms and alignment of the vocabularies [35, 51]. However, such information is often restricted to certain languages or simply not always available [19, 60], thus not appropriate for refinement tasks at web scale. Identity graphs on the web exhibit special properties which must be considered: they are integrated from multiple sources, sources can be multilingual, many suffer from a lack of maintenance and some have multiple encoding schemes.

Since `owl:sameAs` is a symmetric relation, we reduce the directed graph to a simple, undirected graph. In an undirected graph G, a *Connected Component* (CC) is a maximal subgraph with any two vertices connected by a path (Figure 10a). A *gold standard* is the ground truth that maps each node (IRI) to the real-world entity, which can be used for evaluation (Figure 10b). An *equivalence class* (EC) is a set of vertices corresponding to the same real-world entity (may or may not be connected by a path). In an identity graph, a CC is an EC if and only if all its nodes refer to the same real-world entity[2].

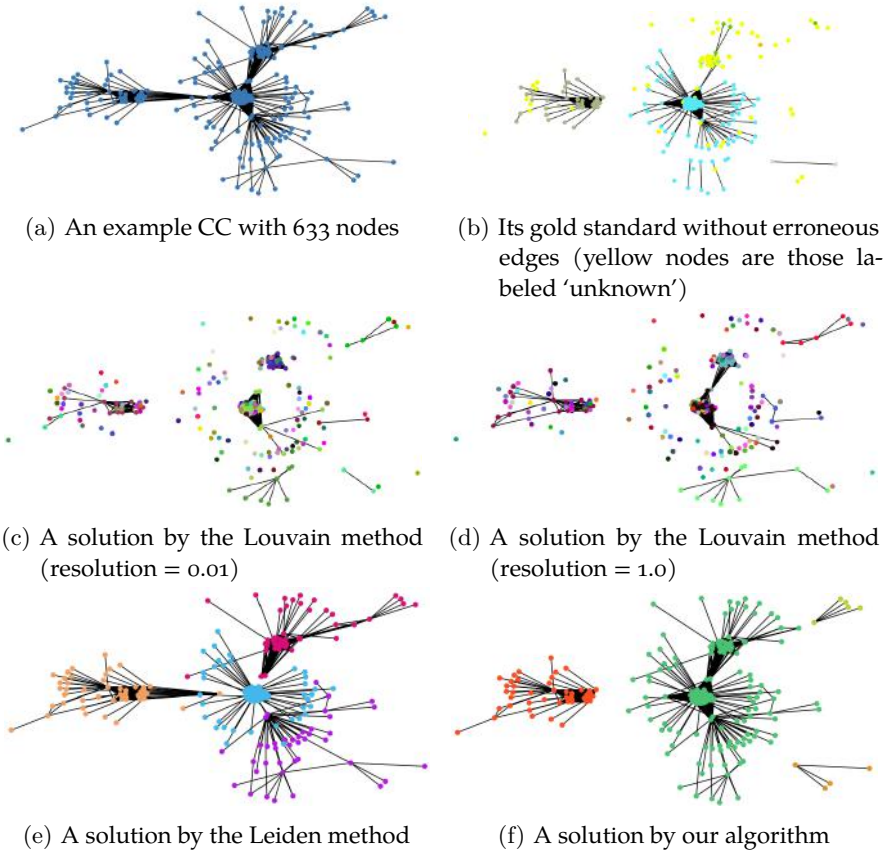
The Unique Name Assumption (UNA) supposes that two terms with distinct IRIs do not refer to the same real-world entity. Although the UNA does not always hold due to redundant IRIs that capture various encodings, languages, namespaces, versions, letter cases, the UNA can still be useful for identifying erroneous links. We design a refinement algorithm that removes a minimal number of edges with good precision (Figure 10f). We compare the results against the Louvain algorithm (Figure 10c and 10d) and the Leiden algorithm (Figure 10e).

This paper focuses on the following research questions.

RQ2.1: How can we formally define and validate a Unique Name Assumption (UNA) for large integrated knowledge graphs to support identity graph refinement?

1 https://plato.stanford.edu/entries/object/

2 However, when constructing the gold standard by annotating IRIs extracted from the Web, some may be annotated 'unknown' if the subject cannot be established.



(a) An example CC with 633 nodes

(b) Its gold standard without erroneous edges (yellow nodes are those labeled 'unknown')

(c) A solution by the Louvain method (resolution = 0.01)

(d) A solution by the Louvain method (resolution = 1.0)

(e) A solution by the Leiden method

(f) A solution by our algorithm

Figure 10: An example of a connected component (No. 4170), its gold standard, and solutions by the Louvain algorithm, the Leiden algorithm, and our algorithm.

RQ2.2: Can the UNA be used for the design of an algorithm to detect erroneous identity links in practice reliably?

We present existing definitions of the UNA and related work in Section 3.2. In Section 3.3, we propose a new definition of the UNA and we test the different UNA definitions and examine their reliability for error detection in Section 3.4, by validating them over data of the LOD cloud. In Section 3.5, we present our refinement algorithm. Evaluation results of the algorithm together with its improvement are included in Section 3.6. Finally, discussion and future work are presented in Section 3.7.

Our main contributions[3] are as follows:

3 The data is published on Zenodo (https://zenodo.org/record/7765113) with DOI 10.5281/zenodo.7765113.

1. We propose a new definition of the UNA, namely the iUNA and check it against a large integrated knowledge graph together with other definitions.
2. We design an inconsistency-based refinement algorithm that evaluates definitions of the UNA by employing an SMT solver.
3. We publish a gold standard of over 8K manually annotated entities (200K owl:sameAs links) together with some additional information such as disambiguation, weights, redirection, and equivalence under different encoding schemes.
4. We introduce new evaluation metrics and provide a benchmark using our gold standard and algorithm.

3.2 Related Work

Estimates of the proportion of erroneous identity links in the semantic web range from around 3% [35, 58] to 20% [29]. Existing approaches for detecting errors in identity graphs fall into three categories [60]. *Content-based* approaches to exploit the descriptions associated with each resource for evaluating the correctness of an identity link. They typically rely on additional information such as vocabulary alignments and textual descriptions for each entity. However, such information is not always available [19, 60] on open Web datasets, and in practice, such algorithms often do not scale to the size of the LOD Cloud. The *network-based* approaches [28, 59] take advantage of graph-theoretical algorithms for the detection of erroneous links. For instance, [59] rely on the Louvain community detection algorithm for assigning an error degree for each identity link. This error degree is based on the density of the community in which an identity link occurs in, and the weight of the `owl:sameAs` (i.e. reciprocally asserted owl:sameAs have a lower error degree, hence a higher chance of correctness). These error degrees are published online as part of the MetaLink dataset [6]. However, the accuracy of these methods is limited due to a lack of understanding of the underlying semantics. Finally, the *inconsistency-based* approaches [35, 51] hypothesize that `owl:sameAs` links that lead to logical inconsistencies have a higher chance of being incorrect. They typically require the presence of a large number of ontology axioms and alignment of the vocabularies.

The use of the UNA to detect errors in identity graphs is an inconsistency-based approach. This idea has been explored in [21,69]. Despite that UNA is a well-defined definition in relational database theory (a.k.a. Unique Name Axiom) [62], the lack of an agreed-upon definition of UNA in semantic web leads to different conclusions. The primitive adaption of UNA in the semantic web postulates that any two ground terms with distinct names are non-identical [21]. In the scope of integrated knowl-

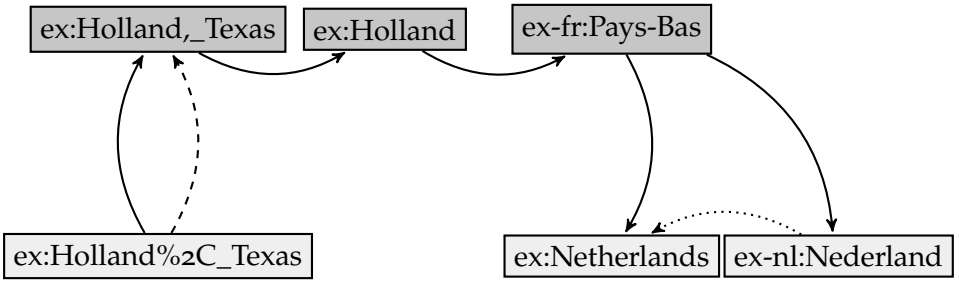


Figure 11: An example CC with links expressing identity (thick black arrows), redirection (the dotted arrow), and encoding equivalence (the dashed arrow) (see also Section 3.3).

edge graphs, Valdestilhas et al. [69] formalize this as any two URIs in the same knowledge base cannot refer to the same thing in the real world. We name this definition *naive UNA*, or *nUNA* for short. In practice, an integrated knowledge graph violates the nUNA if at least one of its connected components (from the identity graph) has two entities from the same source.

Figure 11 is a fictional example of six entities from two knowledge bases (corresponding to nodes in light grey and dark grey, respectively). The six entities connected by the black edges form a connected component. The two equivalence classes are about the Netherlands (the three nodes on the right), and a city in Texas named Holland (the two nodes on the left). The node `ex:Holland` can be confusing (could be annotated as "unknown"). The blue arrow is an example how encoding schemes can lead to redundancy. Due to transitivity, the mistake between `ex:Holland,_Texas`, `ex:Holland` and `ex-fr:Pays-Bas` was carried over to other entities such as `ex-nl:Nederland`. This example shows how entities in various languages can be confusing. This connected component violates the nUNA: for the knowledge base of light grey, there are three entities in the connected components. This helps the detection of spurious links. Note that removing the links between `ex:Holland,_Texas` and `ex:Holland` and `ex-fr:Pays-Bas` results in three connected components, which are correct but still violate the nUNA.

De Melo [21] points out that the Semantic Web is very different from traditional closed scenarios because multiple parties can publish data about the same entity using different identifiers. Thus, they propose to use a quasi-unique name constraint (*quasi UNA*, or *qUNA*) for entities: they use the namespace of an IRI as its source of provenance, with a focus on 6 major hubs including DBLP, DBpedia, FreeBase, GeoNames, MusicBrainz, and UniProt. This definition also takes into account some exceptions: two DBpedia entities from the same dataset/source do not violate the UNA if one redirects to the other, or either is a dead node (those that can no longer be resolved).

These definitions have several drawbacks in practice. First, both the nUNA and the qUNA lack a clear definition of *provenance*, i.e. the source of entities. The algorithm using the nUNA relies on LinkLion[4] for computing the provenance of entities [69]. That of the qUNA takes an entities' namespace as the source by default. As for DBpedia, the paper studied only the namespace `http://dbpedia.org/resource/` for violation and redirection. The algorithm developed based on nUNA outputs only partitions of the identity graph rather than the edges to remove [69]. Despite that the paper proposed to handle cases of DBpedia with exception, qUNA is restricted to awareness of redirect within DBpedia [21]. In fact, recent work estimates that between 45% and 83% of redirection links can be taken as identity link[5] [49]. Furthermore, the work in [21] does not specify how redirection and dead nodes were obtained. In addition, we believe that there are other forms of exceptions that must be considered. For example, the IRIs `wikidata.dbpedia.org/resource/Q6453410`, `www.wikidata.org/entity/Q6453410` and `wikidata.org/entity/Q6453410` are about the same entity but in different versions of Wikidata. Despite issues with the definition, the refinement algorithm using these two UNA definitions takes violations as hard constraints: entities are considered different as long as the UNA is violated. Due to the lack of a gold standard, neither definition was validated on real-world data, or compared with other existing baselines. In this work, we propose a new definition of the UNA that is suited for large integrated graphs on the Web and compare it with the existing UNA variations previously proposed by [21, 69].

3.3 The iUNA

When examining the data in the LOD Cloud, we note that identity links are often used to connect the same entity in different language, versions or encodings. Therefore, we propose our own definition of the UNA, which we call the internal UNA (iUNA), to take these differences into account. Our iUNA definition assumes that two different IRIs $e1$ and $e2$ within the same namespace should refer to distinct real-world entities only when: a) they are in the same knowledge base according to a certain provenance information, b) they don't satisfy any of the following exceptions:[6]

1. if $e1$ can be percent encoded/decoded into e_2 by one or more steps,[7]
2. if e_1 redirects to e_2 (or vice versa), or both redirect to the same location,

4 LinkLion (`https://www.linklion.org/`) is no longer available.
5 The uncertainty is due to the presence of a large number of 'unknown' entities
6 These exceptions are based on our manual examination of the entities in the linksets.
7 For example, `ex:Bandon_(Oreg%C3%B3n)` and `ex:Bandon_(Oregón)` can be equivalent.

3. if at least one of e_1 and e_2 is a dead node, not found, unresolvable, redirects until reaching some error or has a timeout error while resolving.

To check whether two entities violate the iUNA, condition (a) requires us to check whether they are from the same knowledge base. This requires some form of provenance to determine where an entity is defined. The nUNA relies on the provenance information of LinkLion, which consists of multiple linksets. It is questionable if linksets can in fact be taken as the knowledge base where the entities are defined, not to mention that LinkLion is no longer available. As for the qUNA, it takes the namespace of an entity to define its knowledge base (regardless of the actual knowledge bases where the corresponding identity links are). This can be problematic for popular namespaces: an entity in DBpedia can be defined in one knowledge base but used in other knowledge bases. Authors can specify where an entity is defined using `rdfs:isDefinedBy`, but an ad-hoc examination shows that this information is rare. We therefore propose two additional means for the estimation of the provenance of an entity e. Table 5 provides a comparison of the three UNA definitions.

Table 5: Comparing the definition of the UNA

	nUNA	**qUNA**	**iUNA**
Definition	Two URIs in the same KB cannot refer to the same thing	Refinement of nUNA, considering exceptions of DBpedia	Refinement of nUNA by considering multiple exceptions and provenance estimations
Provenance	Rely on LinkLion	Namespace (in 6 major hubs)	Three means of provenance
Exceptions	None	Redir. between some DBpedia entities	Encoding variants, redirection, dead nodes
Algorithm (see sections below)	Violation as hard constraint; returns partitions that are contradiction free	Violation as hard constraint; remove links that violate qUNA	Violations as hard and soft constraints; remove fewer identity links
Limitations (see sections below)	No tolerance towards exceptions; relies on an external server for provenance	Not enough exceptions taken into consideration; restricted definition of provenance; violations taken as hard constraints	Not every exception is included or handled explicitly. Can be relaxed by taking violations as soft constraints.

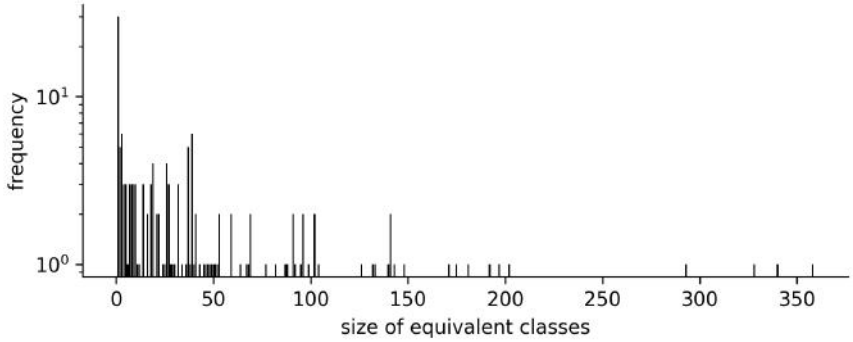


Figure 12: Size distribution of the equivalence classes in the gold standard.

Explicit sources: an explicit source of e is the object in any triple with subject e and predicate `rdfs:isDefinedBy` (or any equivalent or sub-properties).

Implicit label-like sources: an implicit label-like source of e is the RDF file containing triples where e is the subject and `rdfs:label` (or any of its equivalent or sub-properties) is the predicate.

Implicit comment-like sources: an implicit comment-like source of e is the RDF file containing triples where e is the subject and `rdfs:comment` (or any of its equivalent or sub-properties) is the predicate.

3.4 Testing the UNA

3.4.1 Dataset & Gold standard

We use the `http://sameas.cc` dataset [7], which provides the transitive closure of 558 million distinct `owl:sameAs` statements. These identity statements were extracted from the 2015 LOD Laundromat crawl [8] that provides more than 38 billion triples from over 650K RDF files. The identity links are distributed over 49 million connected components (CCs), with each CC being associated with a unique ID. We manually annotated all IRIs from 28 CCs with fewer than 1K nodes each. Our gold standard consists of 8,394 manually annotated entities covering a total of 232,311 `owl:sameAs` links. There are 987 entities (11.75%) annotated as 'unknown'. A total of 209,160 edges (90.02%) are between nodes with the same annotation while 3,678 edges (1.58%) link entities with different manual annotations. The remaining edges involve at least one node annotated as 'unknown'. Based on this manual examination, we estimate the error rate to be between 1.58% and 9.98%. We divide our gold standard randomly into two parts of 14 files each for training and evaluation,

respectively. To better understand the gold standard, we show their size ECs and their distribution in Figure 12. The plot shows that redundancy is common in the LOD cloud. The majority of ECs contain fewer than 200 nodes, while there could be as many as 358 identifiers referring to the same real-world entity at the right end of the spectrum. This gives a reference for the setting of parameters in our algorithms in Section 3.5.

3.4.2 Validating the UNA

Using the gold standard, we validate our definitions. For this, we use the sources of entities in our gold standard retrieved also from LOD Laundromat. Our examination shows that only 0.71% of the entities have an explicit source. In contrast, 61.97% of the entities have at least one implicit label-like source and 40.71% have a comment-like source. This indicates that explicit sources are too rare and thus we only use two variants of iUNA in this work: *iUNA-label* and *iUNA-comment* corresponding to label-like sources and comment-like sources respectively.

For each source, we analyze the number of entities in each EC. Although the original work that examines qUNA was restricted to only 6 major hubs' namespace as provenance, it can be easily adapted to any namespace. Thus, we generalize its definition of provenance in the experiments below. Considering that the nUNA lacks a proper definition of provenance, we use the label-/comment-like source defined for iUNA for the sake of comparison. Table 6 provides the proportion of sources with the number of entities in each implicit label-/comment-like source in the equivalence classes. A source follows the UNA if there is only one unique entity in the EC. An estimate of 1,351 out of 1,737 label-like sources follows the nUNA. On the other hand, 14.40% of the sources violate the nUNA by having two entities in at least one equivalence class in the gold standard, and an additional 7.82% of the sources violates the nUNA by having more than two entities. Table 6 shows that the iUNA is better than the nUNA and the qUNA in terms of capturing how the community is implementing the UNA in their knowledge bases. This also shows that taking encoding equivalence and redirection can indeed align the UNA with its use in practice. Thus, the algorithm should not remove all edges that violate the UNA when refining the identity graphs.

3.4.3 Detecting Errors Using UNA

In this section, we focus on how we can use the UNA to give a reliable indication of identity errors in practice. Our analysis shows that the errors can be classified as two types. The first type are erroneous edges between entities that refer to two real-

Table 6: Analysis of sources of the gold standard that follow the UNA

		nUNA	qUNA	iUNA
one unique entity	label-like	1,351 (77.78%)	204 (59.48%)	**1,566 (90.15%)**
	comment-like	519 (68.56%)		**670 (88.51%)**
up to two entities	label-like	250 (14.40%)	69 (20.11%)	119 (6.85%)
	comment-like	153 (20.21%)		57 (7.53%)
more than two entities	label-like	136 (7.82%)	70 (20.41%)	52 (2.99%)
	comment-like	85 (11.23%)		30 (3.96%)

Table 7: Percentage of pairs violating different definitions of the UNA with the lower/upper bound of their error rates using different sources

		Violation (%)	**Lower bound (%)**	**Upper bound(%)**
random		47.0	68.1	
nUNA	label	61.9	33.4	49.8
	comment	42.5	32.6	46.2
iUNA	label	0.3	8.5	75.9
	comment	0.1	11.7	35.0
qUNA		1.4	16.1	61.3

world entities. The others are edges involving nodes annotated as 'unknown'. Thus, we provide upper and lower bound of error rate depending on how these edges are treated. First, we study how two random entities in a connected component are identical. For this, in each connected component G in the gold standard, we sample $|V|$ (i.e. the number of nodes) different pairs of entities at random. The estimated error (proportion of non-identical pairs) is between 47.0% and 68.1%, depending on the interpretation of the nodes labeled "unknown" in the gold standard. We use this as our baseline for the analysis below (see the first row of Table 7).

For these same sampled pairs, we test the error rate and the UNA violation percentage for the three UNA definitions. The second row in Table 7 shows that when using label-like sources, 61.9% of the sampled pairs violate the nUNA, the estimated error is between 33.4% and 49.8%. In contrast, only 0.3% sampled pairs violates iUNA, with an error rate between 8.5% and 75.9%. Recall that 11.75% nodes were annotated "unknown". This analysis also indicates that such nodes are heavily involved in pairs violating the UNA. More pairs violate the UNA when using label-like sources than when using comment-like sources. In all cases, the lower bounds of error reduce when compared against that of randomly sampled pairs. Using iUNA with comment-like sources reaches the lowest error rate for the lower bound. These selected pairs are then used in the algorithm to identify erroneous edges in the paths that connect them.

Next, we study the impact of redirection. There are in total 13,922 nodes in the graphs that capture redirect relations[8]. We find that 3,072 out of 8,394 entities were redirected. Among them, 5,528 correspond to new IRIs that are in the extended graph but not in the original graphs. There are in total 6,991 edges in the redirect graphs. Among them, 546 are between entities in the original graph with 504 correct ones and 8 erroneous ones. That is, the error rate is between 1.47% and 7.69%. In addition, we have 12,531 pairs of entities that redirect to the same entity in the extended graph. The error rate is between 4.29% and 6.32%.

Moreover, we study the equivalent entities suffering from different encodings (recall the example given in Figure **??**). We have 1,818 pairs of entities in the gold standard.[9] Among them, there are edges between 1,130 pairs in the original identity graphs with an error rate between 2.21% and 8.50%. We discovered 688 new pairs that differ only by encoding with an error rate between 1.16% and 14.83%. Finally, there is a pair of entities whose IRIs in alternative encoding are the same but they actually refer to different real-world entities. We conclude that though the exception do not always hold, they are often useful.

3.5 Algorithm Design

We limit the scope of refinement algorithms in this paper to removing erroneous identity links and forego identifying erroneous entities or adjoining additional links. The intuition is that for two inter-connected clusters, if there is more force pushing them apart than holding them together, then some edge(s) should be removed to split the clusters apart. The "force" that pushes the clusters apart are between pairs of entities violating the UNA. These pairs might not be directly connected, but they can be connected through multiple paths. The removed edges as the output of the algorithm is a *cut* for the graph. Computing an optimal cut whose removal makes the graph consistent within each CC is APX-hard (i.e. where there are polynomial-time approximation algorithms) [21]. We can encode this problem (as soft and hard clauses) to an optimization problem and employing an SMT solver [12]. The goal is to maximise the sum of weights over all soft clauses while satisfying all the hard clauses. We choose this approach because it enables fast reasoning over weighted constraints of relations of equality and inequality and it returns a sub-optimal answer in case of timeout.

8 Redirection was tested with the *requests* Python package using the *get* function with a max timeout of 5 seconds for connection and 25 seconds for reading.

9 We used the *parse* function in the *rfc3987* and *urllib* Python library.

Algorithm 2: partition

1 **Input:** an identity graph G, a weighting scheme w, a graph of redirect G^R, a graph of equivalence under various encodings G^E
Result: status s, a set of edges removed A, the graph of partitions G_P
2 initiate A as an empty set (to store removed edges);
3 initiate H_{ccs} as a set of the connected components of G;
4 **while** $|A|$ *is increasing* (*no new edge to remove*) *and* H_{ccs} *is not empty* **do**
5 **foreach** $H_{cc} \in H_{ccs}$ **do**
6 (optional: obtain the corresponding subgraphs H^R_{cc}, H^E_{cc} from G^R, G^E);
7 $(N_{ccs}, A') =$ partition_iter$(H_{cc}, w, H^R_{cc}, H^E_{cc})$;
8 $A := A \cup A'$;
9 remove H_{cc} from H_{ccs};
10 add new graphs N_{ccs} that are not singleton to H_{ccs}.
11 remove A from G to get G_P;
12 return (A, G_P).

3.5.1 Algorithm using UNA

Since the iUNA/nUNA requires the same parameters, we present the algorithm using the iUNA. That of qUNA can be derived simply by removing the parameters of redirect graphs and that of encoding equivalence. Algorithm 2 takes as input a graph G, the corresponding redirect graph G^R, the graph of equivalence under various encodings G^E, and a weighting scheme w. As a first step, we load H_{css} with the connected components of G. We obtain the corresponding subgraphs H^R_{cc}, H^E_{cc} from G^R, G^E respectively. G_{ccs}, together with G^R_{cc}, G^E_{cc} and the weighting scheme is then taken as the input of Algorithm 3. The removed edges are collected in A. The algorithm stops when no more edges can be removed.

In the while-loop of Algorithm 2, there is a repeated call to Algorithm 3 that examines each graph of a connected component in H_{ccs} (line 7). Algorithm 3 takes advantage of an SMT solver's power of reasoning over weighted relations of equality and returns a solution within a given time bound. We first randomly sample some pairs of nodes. We keep those that violate the iUNA, denoted P (line 2). If there is at most one pair in graph G_{cc} that violates the iUNA, we keep the graph as it is (line 4). Otherwise, we initiate an SMT solver (line 5). For each node, we introduce a integer

Algorithm 3: partition_iter

Input: a graph of connected component G_{cc}, a weighting scheme w, a graph of redirect G_{cc}^{R}, a graph of equivalence under various encodings G_{cc}^{E}
Result: a set of graphs of connected components N_{ccs}, edges removed A_{cc}
obtain random pairs of nodes, select only those that violates the iUNA, as P;
if $|P| \leq 1$ **then**
    return $(G_{cc}, \emptyset)$.
initiate an SMT solver o;
foreach *entity* e *in* G_{cc} **do**
    introduce an integer variable I_e in the SMT solver;
    assert **hard clauses** $(0 \leq I_e)$ and $(I_e \leq M)$ in o.
foreach *pair* (s, t) *in* P **do**
    assert in o a **soft clause** NOT$(I_s == I_t)$ with weight according to w.
let F be the minimum spanning forest of G_{cc};
sample a small amount of additional edges from G_{cc} as B;
foreach *pair* (s, t) *in* $F \cup B$ **do**
    assert in o a **soft clause** $(I_s == I_t)$ with weight according to w.
obtain $G_{cc}^{\prime R}$ the undirected graph of the (directed) graph G_{cc}^{R};
foreach *pair* (s, t) *in* $G_{cc}^{\prime}$ **do**
    if *there is a path between* s *and* t *in* $G_{cc}^{\prime R}$ **then**
        initiate/update the weight of a **soft clause** c_r in o according to w.
foreach *pair* (s, t) *in* G_{cc}^{E} **do**
    initiate/update the weight of a **soft clause** $(I_s == I_t)$ in o according to w.
let m be the model of o after solving;
extract the removed edges A_{cc} from m;
remove A_{cc} from G_{cc};
compute N_{ccs} as the connected components without singletons;
return (N_{ccs}, A_{cc}).

variable. We encode two hard clauses to ensure the values to be between 0 and M in the model m. These integer variables will eventually be assigned an integer value in the model m after solving.

Next we explain how the soft clauses are generated. For each pair (s, t) in P, we obtain a clause NOT$(I_s = I_t)$ and associate it with a weight according to the weighting scheme w (line 10). Instead of taking all the edges of G_{cc}, we take the edges of its minimum spanning forest and a small sample of the edges to reduce the load on the SMT solver. In line 11, we obtain the minimum spanning forest F. For efficiency, we keep a set of edges in B (line 12) for the back propagation process of SMT's internal algorithm design. The edges of $F \cup B$ forms the set of edges in G_{cc} to examine this round (line 11-14). Recall that in Section 3.4.3, our analysis showed that it provides relatively reliable information when considering redirection and equivalence under different encoding. We therefore encode the edges of the redirection (line 15-18) as soft clauses. The undirected graph is used for the checking of convergence of redirection of two entities (line 15, 17).

While not every soft clause is true in the model, all the hard clauses must be satisfied. The goal is to maximise the sum of weights over all soft clauses while satisfying all the hard clauses. Note that if an SMT solver fails to get an optimal solution within the timeout, it will return the best sub-optimal solution (line 21). The edge (s, t) remains if and only if I_s equals I_t in the model m (line 22).

The weighting scheme w consists of a series of functions that map clauses to weights: $w = (f_G, f_R, f_E, f_P)$. We used the training dataset to fine-tune the weighting scheme. For a soft clause c_e corresponding to an edge e, the weight is $f_G(c_e) + f_R(c_e) + f_E(c_e) + f_P(c_e)$. The first weighting scheme w_1 consists of four functions: f_G assigns the clause of each edge in the $F \cup B$ a weight of 5, the rest 0; Similarly, f_P assigns the clauses corresponding to pairs in P a weight of 2. f_R and f_E both increase the weight by 1 for that of G'^{R}_{cc} and $G^{E}_{c}c$ respectively. After some manual tuning, we provide an alternative weighting scheme w_2 with the corresponding values being 31, 16, 5, and 5, respectively. Other parameters and hyper parameters were set according to Section 3.4.1 and fine-tuned. The upper bound M was set to $2+|G_{cc}|/50$. A random selection of 12% of the edges from the original graph were kept in B. Finally, based on our experience with Z3, the timeout bound for SMT solving was set to $(|G_{cc}|/100 + 0.5)$ second.

3.6 Evaluation

3.6.1 Implementation

We used the *networkx* Python package[10] for the computation of the connected components and the minimum spanning forests. For the manual annotation of the entities, we used ANNit[11]. We used the implementation of the Leiden algorithm and the Louvain algorithm in CDlib[12]. As for SMT solver, we employed Z3[13] and used its Python binding [12]. We published all the code as an open source project[14]. All our experiments were conducted on the LOD Labs machine. It has 32 64-bit Intel Xeon CPUs (E5-2630 v3 @ 2.40GHz) with a RAM of 264GB.

3.6.2 Evaluation Metrics

While precision and recall are commonly used in evaluation metrics [60], the presence of 'unknown' annotations makes them less suitable for this task since no edge involving an entity of 'unknown' counts toward precision or recall. Thus, precision and recall do not adequately capture the qualities. Moreover, we noticed that 11 graphs in our gold standard have no erroneous edges except those with nodes labeled "unknown". Therefore, we provide an additional metric. In its design, we focus on two properties that the equivalence classes should possess within the CCs resulting from refinement: (a) the equivalence class should not be separated over multiple CCs; (b) two equivalence classes should not share the same CC. This leads to the following metric for the graph G' that results from applying a refinement algorithm to G:

$$\Omega(G') = \sum_{C \in G'_{ccs}} \sum_{Q_e \in E(C)} \frac{|Q_e|}{|V|} \frac{|Q_e|}{|O_e|} \frac{|Q_e|}{|C|}.$$

Here, C iterates over all connected components in G', and $E(C)$ is a partitioning of the nodes in C by equivalence class, so that Q always represents the set of nodes within a given C that refers to the same real-world entity e. V represents the total number of vertices, and O_e is the set of all entities in G' referring to e.

10 https://networkx.github.io

11 ANNit is a user-friendly interface for fast annotation of entities and triples. See https://github.com/shuaiwangvu/ANNit for details.

12 Community Discovery Library is a meta-library for community discovery in complex networks: https://pypi.org/project/cdlib/.

13 https://github.com/Z3Prover/z3

14 The code and implementation details are at https://github.com/shuaiwangvu/sameAs-iUNA together with the results of several parametric settings.

Within the summation, there are three factors. The first, $|Q_e|/|V|$ is the proportion of the current set of vertices to the total. This turns $\Omega(G')$ into a weighted sum over all subsets $|Q|$, with the weights summing to the total proportion of nodes not annotated "unknown". The second, $|Q_e|/|O_e|$, is 1 if all references to e are in C, and lower if there are more references in other connected components. This penalizes deviating from (a). The third, $|Q_e|/|C|$, is 1 if all nodes in C refer to e and lower if the connected component is shared with nodes referring to other entities. This penalizes deviating from (b). Note that if the graph contains no "unknown" nodes, the max. of Ω is 1.

3.6.3 Evaluation Results

We compare our algorithm using two variants of sources (implicit label-like and comment-like sources) with two weighting schemes (w_1 and w_2, as defined in Section 3.5) against the Louvain algorithm [13], the Leiden algorithm [2], as well as the result of MetaLink with two threshold values [6, 59]. Table 8 presents the results of the average of 5 runs for each method with best results highlighted. The Louvain algorithm removes the most amount of edges. It has the highest recall but relatively low precision. Recall the example in Figure **??**, the results of Louvain can be smaller isolated components. This problem also exhibits in our evaluation, due to the significant amount of edges removed, its Ω values are low despite varying its resolution parameter from 0.01 to 1.0. Compared with Louvain, the result of the Leiden algorithm shows obvious improvements. There are fewer edges removed while the precision and Ω have improved for both the training set and the evaluation set. As for Metalink, we run the algorithm with two thresholds: 0.9 and 0.99 (only links with an error degree higher than the threshold are considered erroneous). There are fewer edges removed in both cases, with higher Ω values compared against that of Leiden and Louvain.

In almost all cases, using comment-like sources results in better precision values while having fewer edges removed. The difference of Ω between using label-like sources and comment-like sources is minor. In general, fewer links were removed when using the UNA and Metalink for refinement. Comparing the nUNA with the iUNA, we can see that using the nUNA results in more edges removed with a lower precision. When comparing the qUNA with the iUNA, we find as well that the qUNA removes a larger amount of edges, which leads to a slightly higher recall. In almost all settings, using the iUNA results in higher precision, which could be the benefit of better modeling using exceptions. The best Ω values in both sets are obtained using the qUNA, while using the iUNA results in better precision with similar Ω values. Compared with Metalink, our algorithm shows higher precision and better

Table 8: Evaluation of the Louvain algorithm with two resolution values, the Leiden algorithm, MetaLink with two threshold values, and our algorithm using different UNA and settings.

		Training set				**Evaluation set**			
		precision	recall	Ω	$\lvert A \rvert$	precision	recall	Ω	$\lvert A \rvert$
Louvain	res=0.01	0.020	**0.803**	0.091	39,471.4	0.042	**0.727**	0.087	42,424.2
	res=1.0	0.020	0.778	0.087	39,226.2	0.042	0.660	0.084	43,610.0
Leiden		0.249	0.198	0.377	3,398.4	0.068	0.323	0.439	2,782.6
MetaLink	t=0.9	0.076	0.029	0.522	241	0.086	0.032	0.524	337
	t=0.99	0.036	0.004	0.591	58	0.013	0.001	0.635	99
nUNA	label, w1	0.126	0.150	0.590	406.2	0.042	0.063	0.597	684.6
	label, w2	0.153	0.181	0.591	529.0	0.061	0.075	0.580	697.4
	comment, w1	0.201	0.146	0.595	263.0	0.098	0.040	0.618	356.4
	comment, w2	0.209	0.178	0.597	360.2	0.063	0.036	0.606	431.2
qUNA	w_1	0.258	0.152	**0.641**	492.0	0.058	0.036	0.662	706.4
	w_2	0.227	0.174	0.640	566.6	0.101	0.054	**0.671**	634.2
iUNA	label, w1	**0.333**	0.127	0.606	78.0	0.122	0.013	0.652	236.8
	label, w2	0.204	0.118	0.616	125.8	**0.136**	0.028	0.647	235.0
	comment, w1	0.267	0.090	0.598	63.8	0.097	0.002	0.636	141.2
	comment, w2	0.258	0.117	0.607	133.2	0.117	0.003	0.638	173.8

Ω values. Overall, our evaluation indicates that different algorithms have different advantages, but using the UNA shows clear benefits.

As for time efficiency, the Louvain and Leiden algorithm completes processing both the training and evaluation sets within 40 seconds. For the algorithm using the UNA, it takes around 8 minutes to process the training set in contrast to up to 27 minutes for the evaluation set. In addition, we note that up to three graphs in the evaluation set can suffer from timeout using our algorithm[15]. When there is a timeout, the SMT solver returns a sub-optimal solution. Our manual examination shows that some "harder" and larger graphs were distributed to the evaluation set when constructing the two sets.

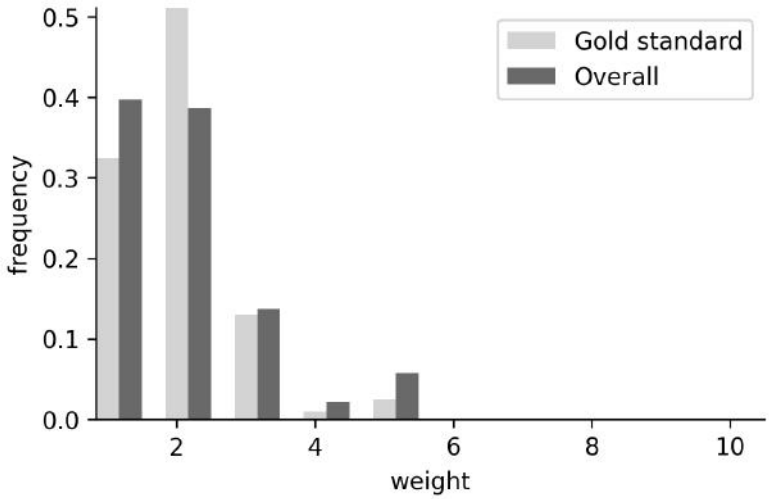


Figure 13: Weight distribution of the `owl:sameAs` links in the LOD Laundromat.

3.6.4 Improving the Results

Next, we study if we can use additional information to improve results. We can define the weight of an edge as the number of datasets in which the corresponding triple can be found (not to be confused with the weight of a soft clause in the algorithm). Out of the 650K available LOD Laundromat files, the `owl:sameAs` links are distributed over 7,024 files. Thus, the weight of an edge can vary between 1 and 7,024. Figure 13 compares the weight distribution in the gold standard and across

15 These are connected components with the IDs 14872, 4635725, and 37544.

the entire (undirected) identity graph. The bar chart shows that most of the triples are associated with weights less than 5.

In addition, we find a significant number of entities that correspond to disambiguation pages in Wikipedia. Of the 3,678 edges in the gold standard that are identified as erroneous, 1,395 edges (38%) involve at least one entity about disambiguation[16]. This can not only make the results less stable but also confirm our finding that the precision-recall is not suitable for this problem. For all links in the gold standard, the error rate is between 27.27% and 71.81% when disambiguation entities are involved. This is significantly higher than the average error rate in the gold standard. In addition, we noticed that after removing 501 disambiguation entities, the largest connected component is reduced from 177,794 to 82,685 entities (a reduction of 53.4%).

As a primitive experiment, we extend the weighting scheme by increasing the weight of the corresponding soft clause by 2 when the weight of the edge is ≥ 2 (denoted w_S). We take also such disambiguation into account. If a clause is about an edge involving at least one disambiguation entity, its weight is reduced by 5 (denoted w_D). When both conditions apply, we denote w_{SD}. Table 9 shows that taking disambiguation entities into account can improve both precision and recall noticeably in most cases. However, there is little improvement in Ω. This addresses the importance of taking semantics into consideration in the refinement of the identity graphs. However, using solely the weights of the edges or both conditions results in no significant improvement.

3.7 Discussion and Future Work

In this paper, we studied three definitions of UNA and proposed a UNA-based identity refinement approach. RQ2.1 was answered by defining the iUNA that considers certain exceptions that are common in large integrated graphs. Furthermore, we created a gold standard and compared the reliability of iUNA against the qUNA and the nUNA, and illustrated its use for refinement. For RQ2.2, we proposed an identity refinement algorithm and evaluated its performance on different definitions of UNA. Finally, we explored the use of additional information to improve the results.

Strictly speaking, our gold standard is not large enough for an accurate estimate of the error rate of the entire identity graph. Using our sample, we found that among the 3,678 erroneous edges, only 5 entities have multiple label-like or comment-like

16 The disambiguation entities were identified where there is a triple with the relation `dbp:wikiPageUsesTemplate` and the object `dbr:Template:Disambiguation` or about a select of 17 multilingual relations with similar meaning.

Table 9: Evaluation results of the algorithm using additional information with extended weighting schemes.

		Training set				**Evaluation set**			
		precision	recall	Ω	$\lvert A \rvert$	precision	recall	Ω	$\lvert A \rvert$
qUNA	w_1	0.258	0.152	0.641	492.0	0.058	0.036	0.662	706.4
	w_1^S	0.231	0.192	0.642	446.0	0.052	0.046	0.658	683.4
	w_1^D	0.302	**0.306**	0.642	637.6	0.044	0.051	0.683	738.4
	w_1^{SD}	0.275	0.218	0.640	523.2	0.039	0.068	0.662	682.2
	w_2	0.227	0.174	0.640	566.6	0.101	0.054	0.671	634.2
	w_2^S	0.236	0.209	0.644	638.0	0.042	0.034	0.668	645.6
	w_2^D	0.244	0.211	**0.645**	601.6	0.107	**0.077**	**0.675**	658.8
	w_2^{SD}	0.218	0.193	0.642	658.6	0.060	0.064	0.666	694.2
iUNA	label, w_1	0.333	0.127	0.606	78.0	0.122	0.013	0.652	236.8
	label, w_1^S	0.216	0.111	0.601	89.4	0.095	0.020	0.639	251.8
	label, w_1^D	**0.404**	0.269	0.606	271.8	0.106	0.057	0.661	242.4
	label, w_1^{SD}	0.327	0.122	0.608	150.4	0.070	0.092	0.661	262.2
	label, w_2	0.204	0.118	0.616	125.8	0.136	0.028	0.647	235.0
	label, w_2^S	0.141	0.094	0.607	133.6	0.120	0.026	0.649	228.4
	label, w_2^D	0.278	0.138	0.617	163.0	**0.143**	0.035	0.661	200.6
	label, w_2^{SD}	0.218	0.114	0.610	150.4	0.117	0.070	0.664	295.6
	comment, w_1	0.267	0.090	0.598	63.8	0.097	0.002	0.636	141.2
	comment, w_1^S	0.162	0.068	0.584	67.6	0.106	0.011	0.626	126.2
	comment, w_1^D	0.351	0.135	0.593	211.4	0.123	0.046	0.639	193.0
	comment, w_1^{SD}	0.336	0.083	0.598	134.2	0.120	0.054	0.631	134.8
	comment, w_2	0.258	0.117	0.607	133.2	0.117	0.003	0.639	173.8
	comment, w_2^S	0.248	0.088	0.596	99.8	0.086	0.014	0.634	192.2
	comment, w_2^D	0.261	0.110	0.611	120.4	0.127	0.033	0.640	166.0
	comment, w_2^{SD}	0.187	0.091	0.593	117.0	0.109	0.057	0.637	191.2

sources. This indicates that redundancy is not the direct cause of the error. This contradicts the conclusion of [21] (see type 2 error: consistency and conciseness error). However, based on this gold standard, it is possible to generate synthetic data for the evaluation of future methods that can handle larger connected components. It can also be used to trace the source of error, which may inspire future algorithms. The gold standard can be used for the evaluation of other identity relations.[17]

The performance of our algorithm is sensitive to the parameters and hyper-parameters. For example, the upper bound for each integer value M can significantly influence the results if too small. Future work includes studying how our algorithm scales with different time limits, automatic tuning of the parameters, and extending the gold standard. The results of some other parametric settings are included in the supplementary material in the repository.

The performance of MetaLink is comparable with the best outcome of our algorithms. However, our analysis shows that no more than 10% edges removed are shared between Metalink and our algorithms in various settings. It could be promising to explore a hybrid approach in future work. Since our evaluation confirms the superiority of the communities detected using the Leiden algorithm compared to Louvain, it is also reasonable to quest how far the results can be improved if MetaLink uses Leiden's outputs for calculating its error degree.

The identity graph we study contains a large number of connected components of size two, as well as two very large connected components. The biggest CC in this dataset has 177,794 entities and 2,849,426 edges (No. 4073). The second biggest has 21,191 entities and 101,269 edges (No. 142063). The rest are significantly smaller with no more than 5076 nodes. Some past attempts using SMT solvers have also discovered the bottleneck in scalability [**?**, 82]. In future work, we plan to design scalable algorithms following a divide-and-conquer approach for the handling of large connected components using pairs of entities that violate the UNA as heuristics. More specifically, by removing 501 nodes identified captured by the first two relations about disambiguation mentioned above, we manage to break the largest component (No. 4073) into a sequence of smaller connected component with 89,215 entities, 2,258 entities, and some smaller components. We are interested in how the

17 The gold standard can also be used for the evaluation of other identity relations such as `skos:closeMatch`, `skos:exactMatch`, `rdfs:seeAlso`, etc. Similarly, it is also possible to test that of `owl:differentFrom`. We retrieve from the LOD-a-lot knowledge graph the edges corresponding to the entities in the gold standard. Our examination shows that there is no overlapping edge in the gold standard with other relations except `rdfs:seeAlso`. For the corresponding graph, there are 6,185 edges and 3,178 nodes. Further analysis using the gold standard shows that the error rate is between 2.70% and 9.73%, which is not significantly different from that of `owl:sameAs` (between 1.58% and 9.98%).

removal of such entities reduce the size of connected components. Following that, we are interested in designing algorithms that can scale to the entire identity graph.

Our analysis shows that 211,348 (90.98%) out of 232,311 edges in the gold standard are about DBpedia entities between different languages[18]. Moreover, among 3,678 erroneous edges, 3,029 (82.35%) involve multilingual DBpedia entities. These links could be automatic generated using transitive closure or inherited as DBpedia enriches. Our analysis shows that these edges have not only made the identity graph more complex, but also less usable as the errors propagate through the graph due to the transitivity of `owl:sameAs`. Lessen such edges can benefit the correctness of the identity graph and improve the efficiency of refinement algorithms.

In our work, we noticed that a significant amount of entities are dead nodes, some are equivalent under various encodings, and some are redirect of each other. In future work, it worth exploring how the identity graph and the results of refinement would change after removing dead nodes and merging some entities.

In contrast to graph-based methods, our algorithm takes the logical properties into account. In future work, we would like to take advantage of the unsat core of the SMT solver and provide some essential explanation for the removal of each edge.

Our analysis shows that a significant amount of 11.75% entities were annotated 'unknown' in the gold standard. Our manual assessment shows that 526 (53.29%) of them are leaf nodes. Among the remaining 461, 179 (38.83%) of them have neighbours with more than one annotations other than 'unknown'. This could be a reason for the wide range of error rate in Section 3.4.3. 2 of them have only neighbours annotated 'unknown'. The remaining 280 (60.74%) of them have exactly one annotation other than 'unknown'.

Fig 10 shows that some resulting graphs consist of some singletons. Some additional evaluation was performed. Table 10 below shows that the methods in evaluation result in some singletons after the removal of erroneous edges. We examine these singletons in three measures: the proportion of singletons not connected to major CCs (PS); the proportion of singletons annotated 'unknown' (PU); the proportion of singletons with unique annotation only for themselves (i.e. no other entity of the same annotation , PT). Generally speaking, the more edges removed, the more singletons there are. The numbers in the column of PS indicate that adding identity links that connect singletons with their corresponding major connected components could be beneficial for the result. In fact, this would improve the Ω value. A hypothesis is that adding some edges can be a beneficial additional step (e.g., using string matching) in the refinement algorithm. However, more research needs to be done to validate this hypothesis.

18 We identify the language of an entity by its namespace. For example, those using the namespace `http://ru.dbpedia.org/resource/` are assumed to be in Russian.

Table 10: Singletons and their semantics.

		#singletons	PS (%)	PU (%)	PT (%)
Louvain	res=0.01	4947.0	84.93	13.03	2.05
	res=1.0	4499.0	81.62	11.22	71.57
MetaLink	t=0.9	127	41.73	15.75	42.51
	t=0.99	57	17.54	12.28	70.17
nUNA	label, w1	486.2	79.94	15.35	4.71
	label, w2	473.2	80.40	17.09	2.51
	comment, w1	112.2	95.74	3.97	0.29
	comment, w2	103.2	94.27	4.10	1.64
qUNA	w1	226.4	54.73	43.07	2.20
	w2	202.6	43.44	54.62	1.93
iUNA	label, w1	116.4	41.27	54.24	4.50
	label, w2	111.6	32.60	62.92	4.49
	comment, w1	33.2	82.96	1.75	15.29
	comment, w2	32.0	87.92	5.39	6.69

We noticed that given a longer processing time, the returned result of the SMT solver can be improved with fewer cases of timeout. While our algorithm uses the SMT solver as a standalone tool, it is possible to take full advantage of the SMT solver by calling its internal functions to test if the performance can be further improved.

All the algorithms except the Louvain algorithm presented in this paper suffer from low recall (see Table 8). This is partially due to the optimization criteria that limit the number of edges removed. Table 9 demonstrates the potential to use additional information to improve the results further. The Ω value indicates that some correct edges to remove could be adjacent to the edges removed by the algorithms. Future work includes taking multilingual labels into account to further refine the result by removing alternative adjacent edges instead.

These experiments and analysis of the corresponding evaluation results show the necessity of developing accurate and scalable hybrid refinement algorithms for integrated identity graphs. The problem boils down to reasoning over equality with optimization, which could be a use case of future algorithms in graph theory as well as applications that use multiple sources of information. Our analysis also addresses the importance of verifying identity links before consuming the corresponding linked open data in real-world scenarios in future work.

4 UNDERSTANDING REDIRECTION IN INTEGRATED IDENTITY GRAPHS

> We know very little, and yet it is astonishing that we know so much, and still more astonishing that so little knowledge can give us so much power.
>
> Bertrand Russell

IRI redirection is a common practice in the LOD cloud, forming part of best practice guidelines for addressing the "curation problem" on the semantic web—essentially, resolving errors. When dereferencing, an IRI is redirected to an alternative IRI, potentially due to namespace updates, encoding changes, or other factors. We investigate entities from `sameAs.cc` [7], an identity graph previously used in chapter 3 [49]. Our analysis covers edges and redirection chains and includes statistical insights into the redirection patterns of sampled entities [49].

4.1 Introduction

The semantic web is a decentralized, worldwide information space for sharing machine-readable data about entities and their relations. This information space contains a vast and rapidly increasing quantity of scientific, corporate, government, and crowd-sourced data openly published on the Web. Open data plays an important role in the way structured information is exploited on a large scale. In this space, resources are identified by global identifiers called Uniform Resource Identifiers (URIs), or more generally, the Internationalized Resource Identifiers (IRIs), extending the character set allowed in URIs to include Unicode characters, making them more suitable for internationalized web addressing. A traditional view of digitally preserving these resources is by "pickling and locking them away" for future use, like groceries, but this conflicts with their evolution. Instead, when resources change or become outdated, a common (and even recommended) solution to the "curation problem" (i.e., repairing data imperfections) is to redirect the user or agent to a new location. We

investigate how such redirections can indicate the evolution of entities in the cloud of linked open data.

Semantic web resources can be divided into two main categories[1]: information resources whose essential characteristics can be conveyed in a message (e.g., web pages, documents), and non-information resources that are outside the information space of the Web (e.g., Amsterdam, Tim Berners-Lee, the concept of color). When dereferencing an outdated IRI of a non-information resource such as the city of Amsterdam (e.g. `https://dbpedia.org/resource/Amsterdam`), it is best practice [37] to redirect the user or agent to the information resource about this city (e.g. `https://dbpedia.org/page/Amsterdam`) using the HTTP response code 303 known as 'see other'.

In practice, redirections through 3XX response codes are not limited to such cases, and are also used to prevent information loss when an IRI can no longer be dereferenced. Precisely, redirecting between two information resources (e.g. in case of a website's update) or between two non-information resources (e.g. for preserving backwards compatibility when an RDF dataset is updated). As the semantic web develops, such redirection links capture the information evolution between IRIs. In fact, when dereferencing an IRI there can be multiple intermediate IRIs involved in the redirection. For instance, Figure 14 illustrates different scenarios that occur in practice when dereferencing IRIs [2]. It shows five entities of an RDF graph: e_0, e_3, e_6, e_8, and e_9 that are connected by any object property, represented in this figure with the black edges (in this chapter, we will restrict to `owl:sameAs` identity links). Red edges represent HTTP redirection links, showing for instance a redirection from e_3 to e_5 with an intermediate redirection to e_4. The links from e_0 are an example of redirections that ultimately lead to an error (e.g., because of a 4XX response code when dereferencing e_2), illustrated as a cross-out node. Finally, this figure shows another case where two resources (e_6 and e_9) are redirected to the same IRI (e_7), before reaching e_{10}, which faces a timeout error (denoted with a question mark) when attempting to resolve the IRI it redirects to.

Although these redirection mechanisms are an integral part of the architecture of the web, and are part of the best practice guidelines for linked data, the semantics of such redirection is unclear. It is tempting to identify a redirection with an implicit statement of identity: the source of the redirection is semantically equivalent to the target of the redirection; it is only the location of the resource that is different. In this chapter, we set out to clarify the semantic intent of redirections as they are being used in practice.

1 See `https://www.w3.org/2001/tag/doc/httpRange-14/2007-05-31/HttpRange-14` for more details about IRIs, dereferencing, redirection, (non-)information resources, and their relations.

2 See Section 4.3.2 and Table 11 for our annotation of different scenarios.

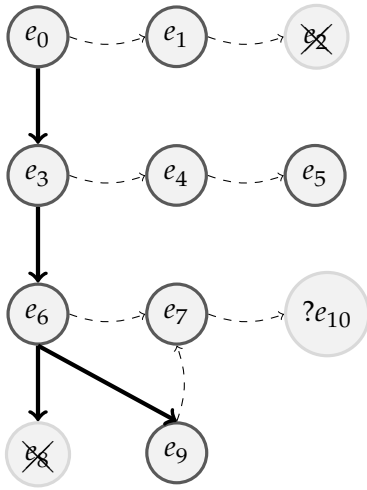


Figure 14: An illustration of HTTP GET request of IRIs (black thick arrows for `owl:sameAs` and dashed arrows for redirection)

Practically, our examination focuses on the redirection of entities in subgraphs that are restricted to identity links between entities (such as `owl:sameAs`). When considering the IRIs identical to entities in these subgraphs, there can be ambiguity and unwanted consequences due to the semantics of identity links. We discuss how redirection can indicate the evolution of entities in the cloud of linked open data (i.e. the LOD cloud). We study the following two research questions.

RQ3.1: How can we interpret and model the implicit semantics of IRI redirection in integrated identity graphs?

For this question, we examine sampled edges and chains of redirection. We classify the scenarios of redirection and estimate the proportion of redirection that can be interpreted as identity links.

RQ3.2: What are the properties and structure of the redirection graphs?

To answer this question, we study the redirection graphs by performing a statistical analysis and examining their graph-theoretical properties, followed by a discussion about the impact of redirection on the LOD cloud.

Our main contributions are as follows:[3]

1. four redirection graphs corresponding to different sampling methods using the `sameas.cc` identity graph;
2. 4,000 semi-automatically annotated edges (as pairs of IRIs) in the uniformly sampled redirection graph;

3 The source code can be found at https://github.com/shuaiwangvu/redirection. The datasets were published online at https://doi.org/10.5281/zenodo.7225383 with DOI 10.5281/zenodo.7225383.

3. a qualitative study of the semantics of redirection in the identity graphs;
4. a quantitative study of properties of the redirection graphs.

The paper is organised as follows. In Section 4.2, we present related work on redirection and identity graphs. Section 4.3 introduces the new redirection graphs, based on which we sample data for analysis. Section 4.4 studies the semantics of redirection. The analysis of the redirection graphs is discussed in Section 4.5 followed by conclusions and future work in Section 4.6.

4.2 Related Work

As a domain with a strong focus on unambiguous identifiers and meaning, semantic web research has been suffering from an ill-defined sense of identity [72]. This crisis becomes even worse when taking into account the impact of the evolution of datasets on the identity links. The identity crisis was already studied by Halpin, et al. [30] in 2009. They propose to study how an HTTP resource responds to a GET request, including how they redirect to new IRIs (both the hash convention and the HTTP 303 redirection). However, this work did not retrieve or study any data from the web, nor performed any quantitative assessment on the reliability of interpreting redirection as identity relations.

The evolution of datasets can result in missing IRIs. De Melo presented an initial analysis in 2013 and revealed that, for the BTC2011 sameAs triples, 205,231 out of 1,055,626 unique DBpedia IRIs did not exist in the DBpedia 3.7 dataset [21]. This analysis shows that around 19.4% of entities no longer existed after only two years since their first publication. The paper also investigated the reasons for this. For example, IRIs with incorrectly escaped titles, i.e. using a different encoding scheme than DBpedia itself, resulted in IRIs that do not exist in DBpedia. Secondly, since Wikipedia is a living resource, articles may be deleted, merged, or renamed. Thus, many IRIs no longer exist in DBpedia.

Regino et al. [61] studied semantically broken links. These are newly added links between the new IRIs of the subjects or objects that may have evolved. When the evolved IRIs refer to different real-world entities, the change of semantics would result in errors (thus the name "semantically broken links"). For example, a link between e_3 and e_4 in Figure 14 could be such an example if e_4 refers to a different real-world entity than e_3. They studied the links between Wikidata and GeoNames and two versions of DBpedia. While their analysis found some semantically broken links, their approach cannot be scaled to the web since they only studied English entities and rely on WordNet and BabelNet as background knowledge for the determination

of similarity by analyzing on their labels. Moreover, tracking every version of entities in each dataset is not practically feasible.

To the best of our knowledge, the latest web scale examination of the identity graphs dates back to the 2015 crawl of the web[4]. It consists of 558.9M `owl:sameAs` links between about 179.7M entities [7]. However, this graph is now outdated, and as far as the authors are aware, there is no quality assessment of its entities, in comparison to the presence of multiple assessment of its links. In contrast, the current paper aims at addressing the importance of dynamics in identity graphs.

4.3 Data Preparation

In this chapter, we extract our entities from the `sameas.cc` dataset [7]. This identity graph represents a subgraph restricted to `owl:sameAs` links of the 2015 LOD Laundromat dataset [8] that covers more than 650K datasets. We refer to this identity graph as G. Section 4.3.1 provides details of sampling. Based on the sampled entities, we construct the redirection graphs in Section 4.3.2. Finally, in Section 4.3.3 we sample 4,000 edges and 100 chains of redirection in the redirection graph based on uniformly sampled entities. These datasets will be analyzed in Section 4.4 and 4.5 to answer our research questions.

4.3.1 Sampling from identity graphs

For this study, four samples were created. The first sample E^U is created by randomly choosing 100K entities from G. The remaining three samples contain 20K entities each, to study the presence of a correlation between the size of the connected components of G and the semantics of redirection. In G, the set of entities in a CC refers to an equivalence class (i.e. set of entities that refer to the same real-world entity). These entities were sampled equally from CCs containing only 2 entities, those containing 3 to 10 entities, and CCs with more than 10 entities. We refer to these samples as $E^{CC(2)}$, $E^{CC(3-10)}$ and $E^{CC(>10)}$, respectively. We use a disk-based

4 The resulting identity graph and its related research results are hosted at `https://sameas.cc`.

key-value method[5] to obtain a merger of the components until reaching its maximal status while still being space efficient as described in [7].

4.3.2 Constructing the redirection graphs

We analyse the IRI of the sampled entities by sending an HTTP GET request. If the response status code is HTTP 200, we label it as **OK**. If it is a 400+ HTTP error indicating a client error, we label it as 'Not Found' (**NF**). Otherwise, if the entity is a literal or the request fails, we label it with 'Error' (**ER**). We use the label 'Timeout' (**TO**) if the request times out. In practice, some IRIs take longer to connect or read. Hence, we increase the timeout threshold in three steps. We first set the connection timeout to 0.01 second and read timeout parameters to 0.05 second. We collect all IRIs with a timeout for processing in the next step and add labels to the rest. We then use the parameters 0.5 and 2.5 seconds and again collect those that faced a timeout. Finally, our last attempt uses 5 and 25 seconds as parameters. As for cases with redirection we used the history in the response to check if redirection happens. Thus, we include also HTTP 300 (redirection with multiple choice), 301 (moved permanently), 307 (temporal redirect), 308 (permanent redirect), etc. We label the remaining as 'Redirect Until Timeout' (**RUT**). Similar as above, we label IRIs that redirect as either 'Redirect Until Not Found' (**RUNF**), 'Redirect Until Error' (**RUE**), or 'Redirect Until Found' (**RUF**). We create an edge in the redirection graph for each redirection. Similarly to the uniform sampling, we name this graph R^U for G, and similarly we name the three redirection graphs $R^{CC(2)}$, $R^{CC(3-10)}$, and $R^{CC(>10)}$ corresponding to the sampled entities $E^{CC(2)}$, $E^{CC(3-10)}$, and $E^{CC(>10)}$, respectively.

All the scripts were written in Python[6]. We performed all the HTTP GET requests on a computer on August 23, 2022. The computer has 32 CPUs of Intel Xeon E5-2630 v3 (2.40GHz) with 256GB of memory running Ubuntu 18.04.6. Its downloading

5 We create two key-value databases by using the key-value store RocksDB with Python bindings (`https://github.com/NightTsarina/python-rocksdb`). *set_id* maps each entity to a unique integer (i.e. the id) while *identity_set* maps each unique id as a key with an identity set as its corresponding value. Thus, the composition of these two mappings $identity\_set(set\_id(e))$ are all the entities in the corresponding component of e. To obtain these key-value databases, we iterate through each triple of the input identity graph. We check if the subject and object have corresponding ids. If neither is the case, we assign a new id to both. If one has an id, the other adopts its id. Finally, if their ids are different, we obtain the entities of the one with greater id as mentioned above and update the ids of these entities with the smaller id.

6 All the code and scripts are open source in the repository at `https://github.com/shuaiwangvu/redirection`.

speed is 871.56 MB/s. The construction of the redirection graphs took 33.5 hours in total.

4.3.3 Sampling edges and chains for manual analysis

To understand what these redirections are about, we sampled 4,000 edges from R^U. These edges are stored in a file as pairs of IRIs. Moreover, we track the redirection behavior of 100 entities whose number of hops of redirection is greater than two. These chains will then be manually analyzed in the next section.

4.4 Implicit Semantics of Redirection

Next, we estimate the implicit semantics of each of these redirections (RQ3.1). In this section, we perform a qualitative analysis of redirection in the identity graphs. More specifically, Section 4.4.1 studies pairs of redirection and Section 4.4.2 provides details of our manual assessment of chains of redirection.

4.4.1 Analysing pairs of redirection

In this section, we study the nature of redirection links. For this, we sample 4,000 redirection links from R^U for semi-automatic analysis. Figure 15 illustrates the proportion of different cases. We found that 39.1% of IRIs in R^U redirect to their https equivalent. A further 4.7% of IRIs only differ from their redirect by encoding, while another 1.3% in R^U redirects to IRIs only differ in upper/lower case. Together, this amounts to 45.1% of redirects that are mainly concerned with engineering technicalities. A second very common case are the updates of namespaces in the same domain (33.0%). For example, `https://www.worldcat.org/oclc/67950327` redirects to `https://www.worldcat.org/title/pro-patria/oclc/67950327`. We surmise that these are the result of dataset evolution. A specific case of intra-namespace redirections are the 12.1% that redirects from a DBpedia resource to a DBpedia page (but not the inverse). For example, `https://dbpedia.org/resource/Rimula_californiana` to `http://dbpedia.org/page/Rimula_californiana`. These are redirects from a representation to a description[7]. In addition, we found some cases where some suffixes are added to the original IRIs (12.7%), including '.json' (4.0%) and '.rdf' (0.1%). Another 0.6% is about automatic truncation of fragment of hash IRIs (i.e. the hash convention). Finally, various other cases make up the remaining 8.6%, with new

7 In the sense of `https://www.w3.org/TR/coolIRIs`.

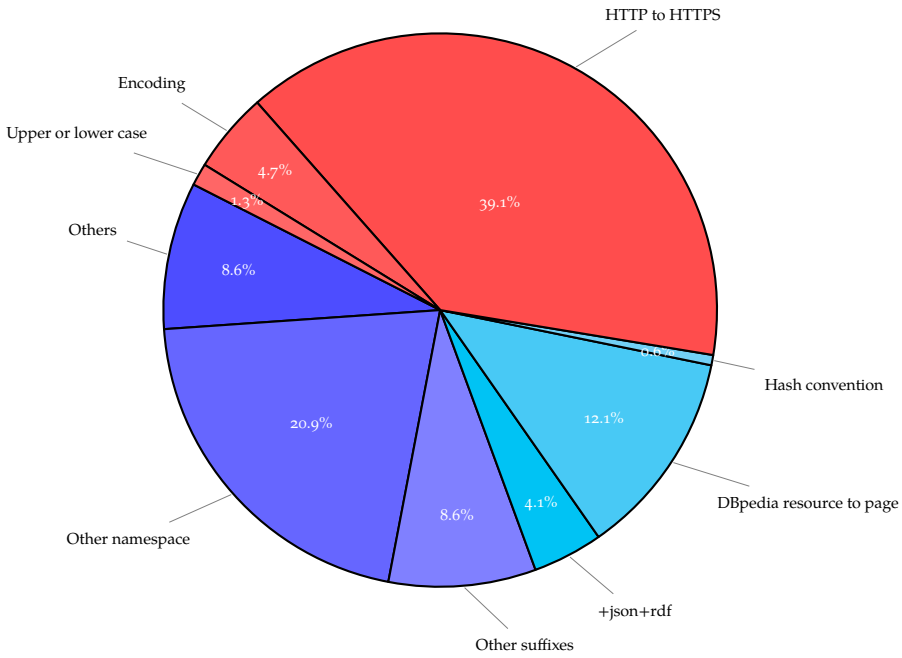


Figure 15: Proportion of redirection behavior among sampled entities

IRIs updated with ids and names, embedded queries, mistaken encoding, or other complex cases.

Our analysis shows that the HTTPS protocol has been widely adopted over the past years. It is likely that the semantics are preserved if they only differ by the choice of protocol. Similarly, if two IRIs only differ by encoding or upper/lower case in their names, they are also likely to refer to the same real-life entities. This sums up to 45.1% (colored red, indicating identity preserving). As for redirection from non-information resources to information resources, only less than 1% concerns hash convention. We also observed at least 4.1% redirects to its corresponding files (with suffix of .json or .rdf) or from DBpedia resources to the corresponding pages (12.1%). This sums up to 16.8% (colored cyan, indicating non-identity preserving). Given all the results, our best approximation is that between 45.1% and 83.2% (100%-16.8%) of redirection links can indeed be taken as identity links.

This primitive analysis shows that the semantics of redirection is rich in practice and requires further investigation with more detailed semi-automatic analysis. Given our analysis that a sizeable share of redirects (up to over half of them) cannot be reliably assumed to signify an identity link, we conclude that redirection should not be used to update outdated mappings without further refinement or manual assessment.

4.4.2 Analysing chains of redirection

Next, we perform an analysis of chains of redirection in R^U. On average, redirection chains have 1.7 hops. More precisely, entities redirected before timeout (RUT) take on average 1.7 hops to reach. Those redirected until not found (RUNF) take 1.6 hops. Those redirected until found (RUF) take 1.8 hops on average. Finally, there are only few redirected until error (RUE) with an average of 1.5 hops. Given the little difference we observed between each category, we uniformly sample 100 chains of redirection across these categories.

We extract 100 chains of redirection with at least 2 hops. Our manual examination shows that the individual redirections in these chains are rarely restricted to a specific type (from Section 4.4.1) but rather mix multiple types. This makes it very difficult to classify these chains. We also observe that these redirections mostly happen within a domain (85%). Among these chains, redirects within the domain `wikidata.org` is most common (28%). Redirection between DBpedia's resources, pages, and their various encodings are also very common (26%). Moreover, these chains are among the longest in our sample with an average number of hops of 3.2. Other domains that occur frequently in these chains are `bibsonomy.org` (5%) and `viaf.org` (1%).

4.5 Analyzing the Redirection Graphs

Table 11 shows an analysis of the behavior of HTTP GET request when applying our redirection typology to G (see Section 4.3.2 for the name of each column). When sampled uniformly, only 33.7% of the IRIs are valid entities: information of the IRI can be found (HTTP 200) with or without redirection (i.e. the sum of the 'OK' and 'RUF' column).[8] Surprisingly, this result implies that only around 1% of the IRIs return meaningful information directly. A comparison of the column 'OK' with 'RUF' shows that redirection is a well adapted means to provide updated information for outdated IRIs. In contrast, a disappointing 66.3% of entities are invalid: IRIs that led to an error, could not be found, or resulted in a timeout (even after a few hops of redirection). When examining sampling w.r.t. connected components (CCs) of different sizes, we observe that the proportion of valid IRIs decreases as the size of the CC increases. Correspondingly, the opposite trend shows for columns labelled 'NF' (not found), 'TO' (time out), 'RUNF' (redirect until not found), or 'RUE' (error). This would suggest that large connected components are a signature of poorly maintained subsets of IRIs. This could be associated to the greater proportion of in-

Table 11: Behavior of HTTP GET request of entities

Graph	RUF	OK	Valid[1]	ER	TO	RUT	RUNF	RUE	NF	Invalid[2]
R^{U}	32.6%	1.1%	33.7%	23.9%	8.2%	8.1%	12.8%	0.01%	13.3%	66.3%
$R^{CC(2)}$	37.1%	0.7%	37.8%	39.5%	12.3%	0.9%	5.5%	0.0%	4.0%	62.2%
$R^{CC(3-10)}$	30.4%	0.3%	30.7%	43.4%	5.8%	0.9%	5.8%	5.0%	8.4%	69.3%
$R^{CC(>10)}$	26.0%	0.8%	26.8%	26.5%	23.2%	2.3%	10.1%	0.1%	11.0%	73.2%

The valid entities include RUF (redirected until found), OK (found with HTTP 200)
The rest are invalid entities, including ER (error), TO (timeout), RUT (redirected until timeout), RUNF (redirected until not found), RUE (redirected until error), and NF (not found).

Table 12: Properties of the redirection graph

Graph	#Entities	#Entities Redirected	#Nodes	#Edges	Avg #Hops	Max #Hops
R^{U}	100K	53,487 (53.49%)	169,021	116,031	1.71	8
$R^{CC(2)}$	20K	8,693 (43.46%)	30,091	21,602	1.64	8
$R^{CC(3-10)}$	20K	8,412 (42.06%)	29,697	21,490	-	-
$R^{CC(>10)}$	20K	7,704 (38.52%)	24,914	18,102	2.05	8

valid entities as the size of CC increases. This might provide a useful heIRIstic for LOD maintenance.

Table 12 presents an analysis on how entities in E^U, $E^{CC(2)}$, $E^{CC(3-10)}$, and $E^{CC(>10)}$ are redirected. Over half of the entities are involved in redirection when sampled uniformly. The average hops of redirect is around 1.71. We observed that $R^{CC(3-10)}$ has a cycle of two entities redirecting to each other. The longest paths can be as many as 8 hops. Our manual examination shows that they are all about redirections between IRIs involving DBpedia resources and pages.

4.6 Conclusion

In this paper, we investigated different scenarios when IRIs are redirected. We studied the semantics by examining edges and chains of redirection. The intuition behind redirects in the LOD cloud is that they preserve identity. Our analysis in section 4.4.1 shows that this is indeed the case for a large proportion of redirects sampled from the `sameas.cc` dataset, with 45% being almost certainly identity preserving, possibly up to 83%. In short, the answer to our first research question is that identity is indeed a plausible estimate of the semantics of redirects. However, given that for somewhere between 17-55% of redirects it is unclear whether they are identity preserving, we suggest that redirection should not be used to update outdated dataset mappings without further refinement or manual assessment.

In answer to our RQ3.2, concerning the properties and structure of the redirection graphs, we found that without any redirects, only 1% of all sampled IRIs return meaningful information directly, rising to 33% after redirection. This means that a disappointing 66% of all IRIs end in error, failure, or timeout at the end of their redirection chain. Furthermore, such failure cases are more frequent in larger connected components, suggesting that such large connected identity components are indicative of poor maintenance, which may serve as a useful heuristic for LOD repair.

Prior work has documented that 19% of unique IRIs in identity graphs do not exist after only two years since publication [21]. Our analysis shows that information of only around 1% of entities is still maintained at their original location, while some 33% of entities can still provide valid information when taking redirection into account, showing that redirection plays a crucial part in the maintenance of the LOD cloud.

Section 4.4.1 presented an analysis of sampled redirection links. In future work, we would like to compare this distribution against existing identity links and study

8 As with the `sameas.cc` graph, we discovered a small number of literals. They were included as exceptions in the 'ER' column.

how similar they are. This could provide further evidence how we can take certain redirection links as identity links. Moreover, the identity graph we used is now considerably outdated. We could create a new updated identity graph and study redirects of sampled entities.

This chapter restricted the analysis to entities in the identity graph. In future work, we would like to remove this restriction and compare against the redirection of IRIs in the LOD cloud. Finally, it could be interesting to examine how redirection can help update existing mappings.

5 ANALYSIS OF LARGE INTEGRATED KGS FOR ECONOMICS, BANKING, AND FINANCE

> I have never let schooling
> interfere with my education.
>
> Mark Twain

Building on earlier chapters' lessons and methods, we apply them to a domain-specific use case, focusing on knowledge graphs in economics, banking, and finance [76]. Through statistical and graph-theoretical analysis, we show that integrating these graphs enriches entity information. We assess quality by examining identity link subgraphs and (pseudo-)transitive relations. Finally, we address error sources, their refinement, and discuss the potential use of our integrated graph.

5.1 Introduction

The 2008 financial crisis urged early detection of systemic risk to national and world economies in derivatives markets. The relative size of these markets is a fundamental risk to geopolitical as well as economic security [45]. As a tool, knowledge graphs (KGs) show great potential in use as they can represent companies structured in complex shareholdings, as well as information about investment, acquisition, bankruptcy, etc. Shao et al. used knowledge graphs of real financial data where nodes are customer, merchant, building, etc [63]. The edges can be transactions between customers, residential information about customers, etc. As a benefit of its graphical structure, their knowledge graph captures interrelations and interactions across tremendous types of entities more effectively than traditional methods. They performed extensive experiments and demonstrated the usage of knowledge graphs in the consumer banking sector [63]. Bellomarini et al. addressed the impact of the COVID-19 outbreak on the network of Italian companies using knowledge graphs of millions of nodes [9]. Such projects require multiple types of domain knowledge, from company ownership to public health policy, from bankruptcy to social resilience. The essence of such knowledge becomes clear for strategy formation and policy-making based on the dynamics of complex interconnected systems. Unfor-

tunately, many sources of knowledge were developed independently of each other. Fusing these independent KGs could lead to a significantly richer source of knowledge, which could improve the performance of existing applications. In this chapter, we study properties of the integration of knowledge graphs by analyzing the statistical and graph-theoretical properties. **RQ4.1:** How can the integration of domain-specific KGs in finance and economics enhance entity descriptions and contribute to identifying errors? **RQ4.2:** How do refinement challenges differ between domain-specific and general-purpose knowledge graphs? More specifically, we study properties of integrated knowledge graphs by combining existing knowledge graphs in the domains of economics, banking, and finance.

Finance The Financial Industry Business Ontology (FIBO) [10] includes formal models that are intended to define unambiguous shared meaning for financial industry concepts. Another popular ontology is the Financial Regulation Ontology (FRO), which has been used as a higher level, core ontology for ontologies such as the Insurance Regulation Ontology[1] (IRO), the Fund Ontology[2], etc.

Economics The STW (Standard Thesaurus Wirtschaft) Thesaurus for Economics was developed by the German National Library of Economics (ZBW) and gained popularity in scientific institutes, libraries and documentation centers, as well as business information providers. The JEL classification system was initially developed for use in the Journal of Economic Literature (JEL) [17] and is now a standard method of classifying scholarly literature in the field of economics.

Banking Knowledge graphs have attracted increasing attention in the banking industry over the past decade. The WBG Taxonomy[3] includes 3,882 concepts. It serves as a small classification schema which represents the concepts used to describe the World Bank Group ' s topical knowledge domains and areas of expertise, providing an enterprise-wide, application-independent framework. In comparison, the Bank Regulation Ontology (BRO) is much bigger and uses two industrial standards, namely FIBO and LKIF [34], as its upper ontology. It was built on top of the FRO ontology, as mentioned above. Unfortunately, many knowledge graphs are developed by banks and are not open source.

In this chapter, we study several properties of integrated knowledge graphs in the domain of economics, banking, and finance. Our contributions include the following. a) We integrate some knowledge graphs in the domain of economics, banking, and finance and present the integrated knowledge graph consisting of over 610K nodes and 1.7 million edges[4]. b) We study how the integration can enrich the infor-

1 https://insuranceontology.com/
2 https://fundontology.com/
3 https://vocabulary.worldbank.org/PoolParty/wiki/taxonomy
4 The data and Python scripts are available at https://github.com/shuaiwangvu/EcoFin-integrated.

mation of entities with some statistical and graph-theoretical analysis. c) We discuss its refinement of the integrated knowledge graph. d) We further compare it against the large knowledge graphs and discuss how refinement challenges change.

The paper is organized as follows. Section 5.2 presents the knowledge graphs and their integration. Section 5.3 presents some detailed analysis of the integrated knowledge graph. Section 5.4 includes an analysis of the source of error, followed by a discussion of its use in the study of interoperability. Finally, we conclude our studies and suggest future work in Section 5.5.

5.2 Integrating Knowledge Graphs

By integrating knowledge graphs from various domains, we expect to see more entities and richer information about these entities. In this work, we limit ourselves to knowledge graphs produced by integrating ontologies (which can themselves be taken as knowledge graphs). Oftentimes, such integration requires the process of determining correspondences between entities in ontologies. Such a process is called *ontology alignment*. The set of correspondences is called a *mapping* or an *alignment*. The following is a list of 11 knowledge graphs we collected from 9 projects in the domains of economics, banking, and finance.

1. the Financial Industry Business Ontology (we collected the FIBO ontology using OWL and FIBO vocabulary using SKOS)[5]
2. the Financial Regulation Ontology (FRO)[6]
3. the Hedge Fund Regulation (HFR) ontology[7]
4. the Legal Knowledge Interchange Format (LKIF) ontology[8]
5. the Bank Regulation Ontology (BRO)[9]
6. the Financial Instrument Global Identifier (FIGI)[10]

5 The product version retrieved from https://edmconnect.edmcouncil.org/fibointerestgroup/fibo-products/fibo-owl (147 files in Turtle format) and https://edmconnect.edmcouncil.org/fibointerestgroup/fibo-products/fibo-voc (1 file in Turtle format) respectively on 14th January, 2022.

6 32 Turtle files were retrieved from https://finregont.com/ontology-directory-files-prefixes/ on 14th Janurary, 2022.

7 12 Turtle files were retrieved from https://hedgefundontology.com/ontology-files/ on 14th January, 2022

8 Retrieved from http://www.estrellaproject.org/lkif-core/#download on 30th January, 2022.

9 16 Turtle files were retrieved from https://bankontology.com/ontology-directory-files-prefixes/ on 30th January, 2022.

10 4 RDF files were retrieved from https://www.omg.org/spec/FIGI/ on 22nd December, 2021.

7. the STW Thesaurus for Economics (and its mappings)[11]
8. the Journal of Economic Literature (JEL) classification system[12]
9. the Fund Ontology[13]

Not all knowledge graphs are available: some are not open source (e.g., the Italian Ownership Graph [9]), some others are commercial (e.g., the enterprise knowledge graphs by Agnos.ai[14]) and a few are not maintained anymore (e.g., the OntoBacen project [56]).

Table 13: Alignment of knowledge graphs

	FIBO-vD	FIBO-OWL	LKIF	FIGI	STW	JEL	Fund
FIBO-vD	-	599	1	147	12	204	11
FIBO-OWL	-	-	24	516	5	57	70
LKIF	-	-	-	1	0	0	23
FIGI	-	-	-	-	0	34	2
STW	-	-	-	-	-	2	0
JEL	-	-	-	-	-	-	1
Fund	-	-	-	-	-	-	-

Table 14: General statistics of knowledge graphs

Name	|V|	|I|	|E|	**Size**
FIBO-vD	17,547	8,173	28,128	3.1MB
FIBO-OWL	103,288	40,780	249,992	16MB
FRO	94,215	68,355	283,976	16MB
HFR	14,235	8,720	34,771	2.6MB
LKIF	1,005	675	2,363	141KB
BRO	259,074	150,241	838,007	43MB
FIGI	12,180	4,003	16,434	822KB
STW	51,128	8,022	113,276	3.4MB
JEL	12,109	1,009	177,57	1.1MB
Fund	10,119	6,269	35,005	3.2MB
STW-mappings	78,398	39,846	177,603	11MB
alignment	2,327	2327	1,698	255KB
integrated	610,866	324,817	1,778,755	93MB

11 The paper used STW v9.12 based on the SKOS ontology. The ontology and its 9 mappings files were retrieved from https://zbw.eu/stw/version/latest/download/about.en.html on 30th Janurary, 2022.

12 The Turtle file was retrieved from https://zbw.eu/beta/external_identifiers/jel/about on 30th January, 2021.

13 The paper used 8 Turtle files retrieved from https://fundontology.com/ontology-files/ on 28th December, 2021.

14 https://agnos.ai/services

We used LogMap[15] for the alignment between knowledge graphs [40]. It is a highly scalable ontology matching system with 'built-in' reasoning and inconsistency repair capabilities. It can efficiently match semantically rich ontologies containing tens (and even hundreds) of thousands of classes. Considering the size of our files, we used the version with mapping repair but not the aid of any reasoner. Unfortunately, FRO, BRO, and HFR failed to load due to parsing errors in some files they import. Table 13 summarizes the number of pairs of entities generated by LogMap. Overall, 1,698 unique identity links of `skos:exactMatch` were added to the integrated graph.

All the selected knowledge graphs were first converted to Turtle format and then integrated with duplicated triples removed using RDFpro[16] [18]. RDFpro is an open-source stream-oriented toolkit for the processing of RDF triples. The integration took 23 seconds on a 2.2 GHz Quad-Core i7 laptop with a 16GB memory running Mac OS. All files were then converted to their HDT format for further experiments. The integrated knowledge graph consists of 1,778,755 unique triples (edges) and 610,866 nodes. It has 93MB and 22MB in its Turtle and HDT format respectively. Table 14 summarize the statistics of the number of nodes, edges and the size of their Turtle files. For the sake of speed, when studying properties of these knowledge graphs, we use files in HDT format.

5.3 Analyzing the Integrated KG

In this section, we first study how the information of entities can be enriched with some statistical analysis of graph structure (Section 5.3.1). We then examine identity links (e.g. `skos:exactMatch`) in the integrated graph **G** and their corresponding subgraphs (Section 5.3.2). We study also transitive and pseudo-transitive relations such as concept generalisation in Section 5.3.3.

5.3.1 Statistical and Graph-theoretical analysis

We study how the information of entities can be enriched when combining different resources. When an entity is described in different domains, its in- and out-degree are expected to increase. Figure 16 illustrates the in-/out-degree of the knowledge graphs and the integrated knowledge graph. Both the in- and out-degrees of the integrated graph show a power-law distribution. Moreover, the figures show that the integration increases both the number of degrees in general and the number of

15 http://krrwebtools.cs.ox.ac.uk/logmap/

16 http://rdfpro.fbk.eu/. We used RDFpro (version 0.6) without smushing.

nodes with high degrees, which demonstrates how this integration can enrich the information of entities. For example, `lkif-norm:allowed_by` has an out-degree of 7 in the integrated graph but the three graphs that contain information about it has out-degrees of 2, 5, and 1 respectively[17].

Table 15 provides the in-/out-degree of main hubs, i.e., entities with the highest in-/out-degrees (excluding literals)[18]. While entities with the highest in-degrees vary from terms in the XBRL ontology to `owl:Class` and `skos:Concept`, entities with the highest out-degrees are mostly from the BRO ontology. Thus, we expect the integrated graph exhibit a scale-free network structure.

Table 15: Entities with high in-/out-degree

Entity	In-degree
`sxml:TextNode`	57,737
`fro-xbrl:linkbase.ttl#loc`	24,767
`owl:NamedIndividual`	23,960
`owl:Class`	22,731
`fro-xbrl:instance.ttl#Item`	15,375
`skos:Concept`	9,822
Entity	**Out-degree**
`bro:Call_Report_v129_ec_mess.ttl#r-2`	18,355
`bro:Call_Report_v129_ref.ttl#r-1`	13,551
`bro:Call_Report_v129_ec.ttl#r-2`	11,015
`bro:Call_Report_v129_pres.ttl#r-2`	9,026
`bro:Call_Report_v129_cap.ttl#r-2`	6,755

Table 16 summarizes the graph-theoretical statistics. Let maxSCC and maxWCC represent the number of nodes in the largest strongly connected component (SCC) and weakly connected component (WCC), respectively. In addition, we compute the fraction of nodes in the biggest SCC and WCC, denoted p_S and p_W respectively. The high values of p_W in the table show that the graphs are mostly connected. More specifically, $p_W = 99.98\%$ for the integrated graph, which is due to the overlapping domains of the knowledge graphs and the mappings. The low values of p_S indicate that the underlying structure of these graphs is mostly hierarchical, especially that of JEL, BRO, and FIBO-vD.

17 The prefix lkif-norm corresponds to the namespace `http://www.estrellaproject.org/lkif-core/norm.owl#`.

18 The prefix `bro` corresponds to the namespace `http://bankontology.com/br/form/`. The prefix `sxml` corresponds to `http://topbraid.org/sxml#`. The prefix `fro-xbrl` corresponds to `http://finregont.com/fro/xbrl/`.

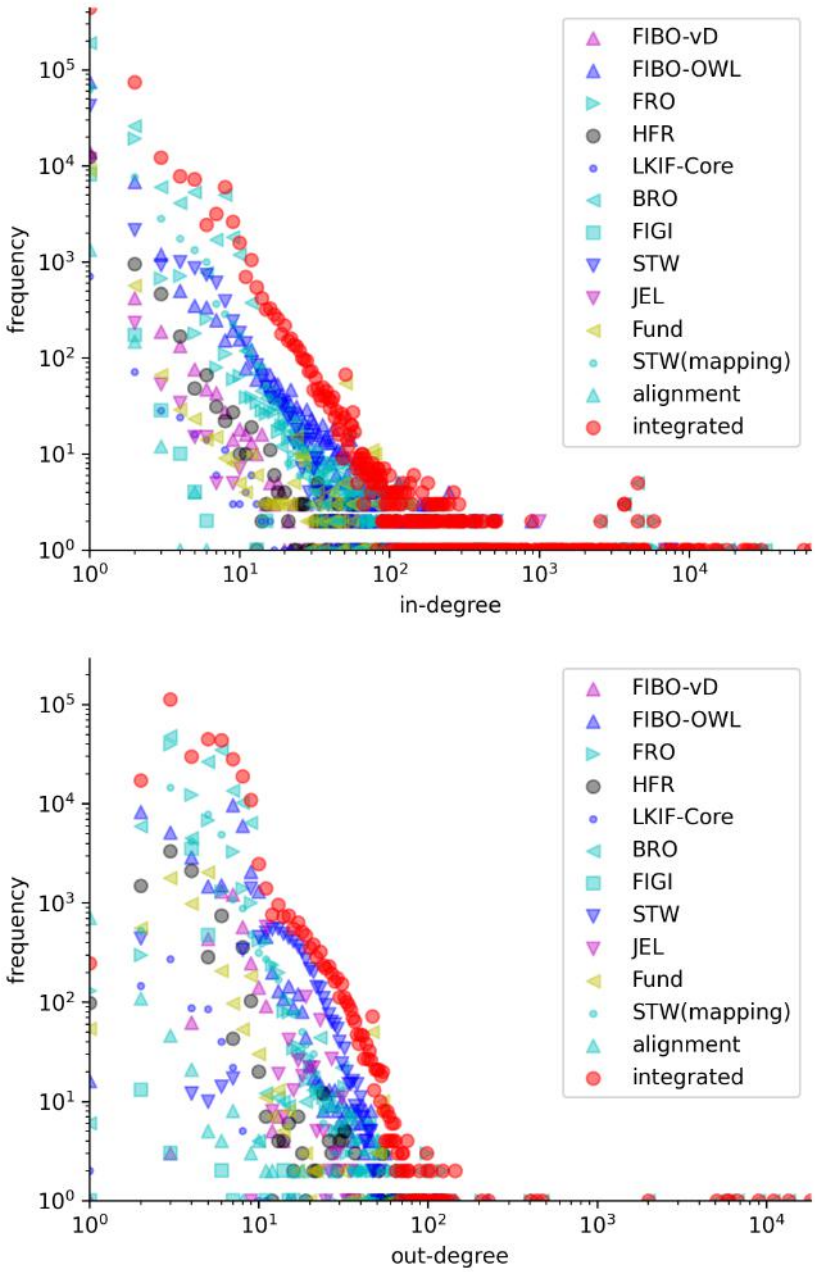


Figure 16: Distribution of in-/out-degree of nodes in knowledge graphs

Table 16: Graph-theoretical statistics of knowledge graphs

Name	maxSCC	p_S(%)	maxWCC	p_W(%)
FIBO-vD	1	0.01	17,535	99.93
FIBO-OWL	297	0.29	103,208	100
FRO	17	0.02	94,015	99.79
HFR	849	5.96	14,230	99.96
LKIF	88	8.76	963	95.82
BRO	13	0.01	258,982	99.96
FIGI	13	0.11	12,180	100
STW	6777	13.25	51,128	100
JEL	1	0.01	12,099	99.92
Fund	109	1.08	10,111	99.92
STW-mappings	617	0.79	78,398	100
alignment	3	0.13	119	5.11
integrated	36,853	6.03	610,792	99.98

5.3.2 Analysis of identity links and redundancy

Integrating multiple datasets results in redundant elements of the same concept. While from a logical point of view, such entities are harmless, they make concept definitions unnecessarily hard to construct and maintain. In addition, redundant elements might lead to content-related problems when concepts drift [22]. While it is beyond the scope of the paper to provide a manually annotated gold standard, we can study the properties of these entities by performing analysis of the corresponding identity graphs, the subgraphs that correspond solely to some identity link. Identity links are relations between entities that are considered identical and intended to refer to the same real-world entities. Typical identity links use relations such as `owl:sameAs` and `skos:exactMatch`. We first study identity links in **G** and their corresponding subgraphs. In contrast to the statistics reported by Raad et al., where `owl:sameAs` is much more popular than `skos:exactMatch` [60], our analysis shows that only 5,253 triples about `owl:sameAs` are in **G** against 31,254 triples about `skos:exactMatch`. In addition, there are 8,172 triples about `skos:relatedMatch`, and 6,418 triples about `skos:closeMatch`. Figure 17 shows the frequency distribution of the weakly connected components in their corresponding subgraphs.

The largest connected components are summarized as follows:

1. a connected component of 15 entities about the telephone system, telecommunications engineering, telecommunications, etc.

2. a connected component of 15 entities about the social insurance, the social law, the social legislation, etc.

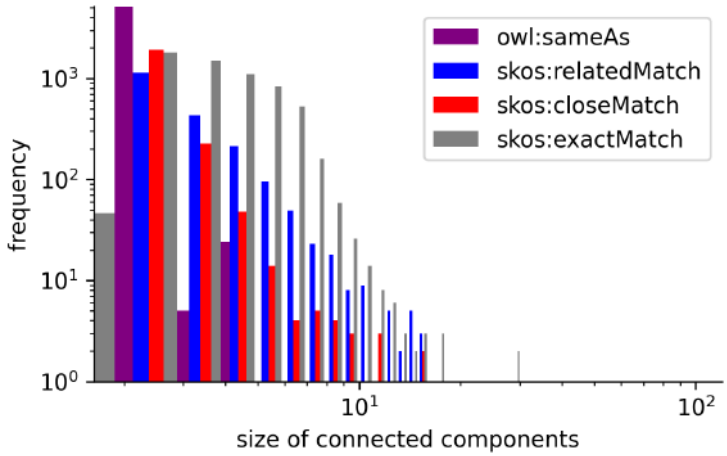


Figure 17: Frequency distribution of connected components in the integrated graph

3. a connected component of 15 entities about "Niue", "Samoa", "Tonga", "Polynesien", etc.

4. a connected component of 14 entities about waste management, waste disposal, garbage removal, etc.

5. a connected component of 14 entities concerning the general education system, primary school, middle school, the general education school, the secondary school, and so on.

The largest two connected components of the subgraph of `owl:sameAs` are with 8 and 6 entities each. In contrast, the largest two connected components of `skos:exactMatch` are much bigger, with 119 and 45 entities respectively. For `skos:relatedMatch`, the largest weakly connected component consists of 21 entities. That of `skos:closeMatch` consists of 52 entities. A manual examination below shows that there are errors in these large connected components. The mis-use of these SKOS mapping properties can have less implications than the `owl:sameAs` since `skos:exactMatch` indicates only "a high degree of confidence that the concepts can be used interchangeably across a wide range of applications" [60]. Moreover, `lkif-core:mereology.owl#strictly_equivalent` is an equivalence relation, but no corresponding triple[19] was found.

19 The prefix `lkif-core` corresponds to the namespace `http://www.estrellaproject.org/lkif-core/`.

5.3.3 Analysis of transitive and pseudo-transitive relations

There are in total 20 relations typed `owl:TransitiveProperty` in **G**. We also study the pseudo-transitive relations: those not typed `owl:TransitiveProperty` but show transitivity in their intended semantics [**?**]. In this study, we focus on two pairs of such relations: `skos:broader` and its inverse `skos:narrower`, `skos:broaderMatch` as well as its inverse relation `skos:narrowerMatch`. This section excludes relations of identity links such as `skos:exactMatch`, which was discussed in Section 5.3.2.

Take `skos:broadMatch` for example. A manual analysis of the three largest SCCs shows that there are edges that could be erroneous. These SCCs are: a component with four entities about plebiscite, referendum, and popular initiative; a component with three entities about insurance and private insurance; a component with three distinct entities about the CARICOM countries, Caribbean countries, and the Caribbean Community.

Let $\mathbf{B} = \{$`skos:broader`, `skos:broaderMatch`$\}$ and $\mathbf{G_B}$ be the subgraph of the integrated graph **G** and $\mathbf{G_N}$ for $\mathbf{N} = \{$`skos:narrower`, `skos:narrowerMatch`$\}$. Next, we combine the $\mathbf{G_B}$ with the graph $\mathbf{G'_N}$, where $\mathbf{G'_N}$ is a graph with each edge of **G** reversed in direction. After performing the same analysis, we discover a new strongly connected component with four entities about adjustable peg, fixed exchange rate, exchange rate regime, and internationales Währungssystem, respectively. Moreover, the resulting graph has 44 connected components of two entities, which is more than that of the subgraphs corresponding to each individual relation. This indicates that such integration can result in more complex errors that do not exhibit in stand-alone graphs.

Our analysis shows that `rdfs:subClassOf` is a popular relation with 47,597 triples. However, there is no SCC with more than one component, which implies that the underlying class hierarchy is a directed acyclic graph. In addition, `lkif-core:component`, `fro:divides`[20], and its inverse `fro:divided_by` are also popular transitive relations. None of them has strongly connected components of size greater or equal to two.

5.3.4 Comparing Refinement Challenges with Large KGs

Next, we compare the experience in Section 5.3.2 and 5.3.3 with that from previous chapters. Our analysis shows that, when restricting to KGs of specific domains, there can be a difference in the popularity of relations. In the case of LOD-a-lot, `owl:sameAs` is the most popular identity relation, while our study shows that `skos:exactMatch` is more popular. The corresponding identity graphs are much smaller. A detailed

20 The prefix `fro` corresponds to the namespace `http://finregont.com/fro/ref/LegalReference.ttl#`.

examination shows that the CCs are small enough for manual refinement. In the future, systematic manual refinement could be conducted to justify the ambiguity of concepts and refine the links between KGs. Similarly, regarding the refinement of transitive and pseudo-transtive relations, we showed that the largest connected components are small enough for manual refinement. Errors in transitive relations and pesuo-transitve relations have become much easier to refine. In particular, the graph on class subsumption is a DAG after construction. Thus, no domain-specific or graph-specific algorithm needs to be developed.

5.4 Discussion

When tracing back to the sources of each edge, we found that `skos:broader` and `skos:narrower` are mostly from three sources: STW, JEL, and FIBO-vD. When combined with the subgraph of `skos:broadMatch` and `skos:narrowMatch`, there are in total 44 SCCs of two entities, two SCCs of three entities, and two SCCs of four entities. It is feasible that some domain experts manually examine all these small SCCs without employing any refinement algorithm.

Our analysis also shows that the identity links come solely from two sources: the `owl:sameAs` triples are from the FIBO-OWL knowledge graph, the triples about `skos:exactMatch`, `skos:closeMatch`, and `skos:relatedMatch` are from STW-mappings and our alignment. Mapping files about the STW subject categories were created by the alignment tool Amalgame[21] [71]. Our manual examination shows that these identity links are closely related concepts and require knowledge from experts for refinement.

Next, we discuss how our findings above would help with the study of interoperability. Interoperability is defined as the ability of data or tools from non-cooperating resources to integrate or work together with minimal effort [92]. Given that all the files we study in this chapter can be easily converted to the Turtle format, we skip the discussion on syntactic interoperability and focus on semantic interoperability – whether the terms in different ontologies refer to the same real-world entity. Figure 18 illustrates the ontological dependency. Despite LKIF having no use of the Dublin Core (DC) and the SKOS ontology, all the other knowledge graphs directly or indirectly use all three of OWL, DC, and SKOS. This analysis shows that FIBO and LKIF play a critical role in the study of interoperability. For example, how the concepts captured by them overlap with JEL, STW, and FIGI play an important role. Table 13 shows that the greatest amount of identity links were discovered between FIBO and

21 https://github.com/jrvosse/amalgame

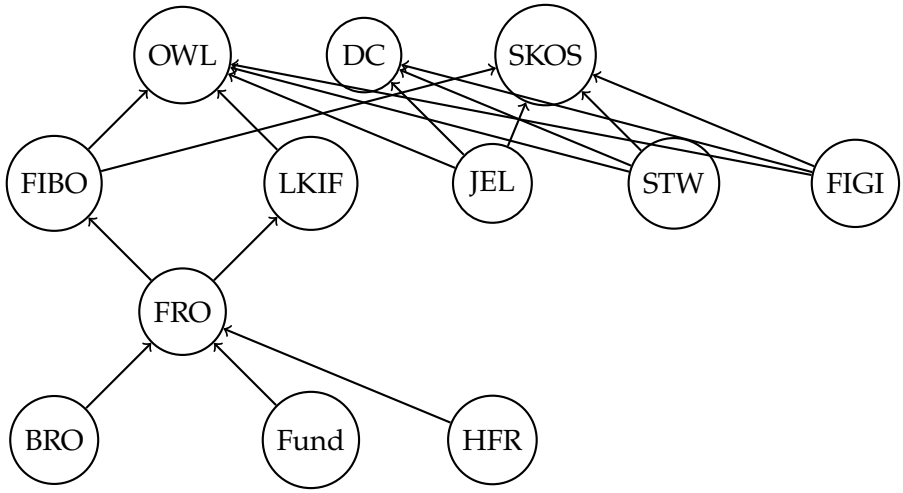


Figure 18: Ontological dependency

FIGI. This may be due to the lack of direct dependencies between them or differences in domain-specific concepts.

5.5 Conclusion and Future Work

In this chapter, we present an integrated knowledge graph in the selected domains of Economics, Finance, and Banking. We discussed the properties of the integrated knowledge graph and demonstrated how the integrated graph has richer information for the entities. Following the methodology of previous chapters, we studied subgraphs of (pseudo-)transitive and identity relations as well as their refinement. Given that the integrated knowledge graph has some minor errors that have been created due to incorrect identity links and (pseudo-)transitive relations, we showed that its quality could be improved after manual refinement. This is different from that of larger integrated knowledge graphs, which require the development of algorithms and manual annotation of datasets for evaluation. Future work includes enriching the properties of entities and some research on whether using entities with enriched information could result in better accuracy in applications.

Our approach and the integrated knowledge graph could be adopted in future work to study how such overlap has an impact on interoperability in practice, especially when retrieving entities' different properties from multiple resources of knowledge graphs. Our approach could also be adopted to research at the metadata level, which could be specified differently between infrastructures. The corresponding integrated knowledge graph may be used to help find the corresponding terms in the

right knowledge graph by following identity links for the conversion and enrichment of metadata. This shows the potential for use in sharing data across infrastructures and may reduce manual work.

6 EXAMINING LGBTQ+-RELATED CONCEPTS

> Call me by your name and I'll call you by mine.
>
> André Aciman

In recent years, there has been a notable increase in the use of systematically structured LGBTQ+ vocabularies, thesauri, and ontologies in library systems, digital archives, online databases, and cultural heritages. Many were released as linked data within the semantic web. However, there is limited reporting on the concepts these encompass and their relations. This chapter seeks to gain some insights by conducting a preliminary analysis using a newly created knowledge graph on LGBTQ+-related entities and their identity-related relations [80]. We illustrate this graph's utility in identifying missing identity links and analyzing conceptual changes and shifts [80]. Additionally, we examine how multilingual resources can be reused for knowledge graph enrichment [80].

6.1 Introduction

While the LGBTQ+ community has achieved enhanced visibility and acceptance over the past decades, a significant gap remains between the complexities of their lived experiences and the knowledge representation efforts. In recent years, there has been a growing demand for LGBTQ+ glossaries and structured vocabularies as well as their multilingual forms, aiming at enhancing the findability of multilingual resources and improving accessibility for individuals who may not be proficient in English but seek to utilize LGBTQ+ resources. Capturing LGBTQ+ terminology and accurately aligning these concepts across various languages can be challenging, owing to the ambiguity of terms, shifts in concept meanings over time, historical, ethical, and political factors, multilingual correspondences, and the incorporation of reclaimed terms, abbreviations, slang, and slurs, among other complexities. Moreover, as some projects become outdated, these issues can lead to consequences such as interoperability challenges when different versions are used in projects of differ-

ent needs, necessitating updates to ensure compatibility with external projects and resources. These aforementioned shortcomings create disparities when attempting to align concepts across these diverse initiatives and demand a comprehensive examination of the status of the concepts captured. This paper presents a primitive study on the analysis of these concepts and their representations in the semantic web.

We observe that LGBTQ+ concepts and their relations are captured in the semantic web in various representations. For the sake of clarity, we use the umbrella term *conceptual models* to refer to thesauri, structured vocabularies, subject headings, ontologies, and knowledge bases, with a particular focus on those published as linked data in the semantic web. We choose to avoid utilizing the alternative term, 'knowledge organization systems' (KOS), as our emphasis lies on examining alterations in the semantics of concepts and their interconnections, rather than on the system itself. Concepts are represented as entities associated with (multilingual) *labels* with *links* between them. The prefixes of entities are included in Appendix A. A *mapping* is a set of links between entities from two different conceptual models, typically referring to the equivalence of classes (owl:equivalentClass) and entities (owl:sameAs), as well as subclass relations between classes (rdfs:subClassOf) and broader/narrower relations of entities (skos:broader/narrower). Due to the diversity in the representation of changes of identity, in this paper, we use a broader definition and include different types of identity-related links between entities of selected conceptual models in our study: replaces, equivalence, identifier, derivation (`prov:wasDerivedFrom`), as well as other variants of identity-related links tailored for various scenarios. Moreover, we include redirection (to be introduced in Section 6.3). In this paper, by *the community*, we specifically refer to the linked data experts, linguistics experts, knowledge/data engineers, data stewards, and data managers who are engaged in the creation and development of the selected conceptual models to be studied below. For simplicity, we refer to the individuals who are closer to the tasks of data engineering and maintenance as *developers* and those who work closer to data annotation as *experts*. Oftentimes, these two groups are essentially the same individuals. Therefore, interpretation should be contextualized.

One of the most used conceptual models is the Homosaurus [68],[1], which was initiated by librarians in the IHLIA LGBTI Heritage[2] and has become a popular conceptual model in LGBTQ+ libraries and archives. Homosaurus has been used at IHLIA as a complementary material of the Library of Congress Subject Headings (LCSH) [55] with mappings between overlapped concepts. We observed that certain terms have undergone semantic evolution (captured in different versions) or have been erroneously linked to reclaimed terms. For instance, the term *wolves* has

1 https://homosaurus.org/
2 https://ihlia.nl/en/collection/homosaurus/

been reclaimed as a slang term (`h3:homoit0001508`) referring to "masculine gay men who are often characterized as having hairy bodies and facial hair". Nevertheless, it is directly linked through `skos:exactmatch` to a term (`lcsh:sh85147257`) in the LCSH, which is about the animal species wolf. Furthermore, given that these entities and links are published in the semantic web, there is the potential for ambiguity, mistakes, subtle changes, and differences in different languages to accumulate into complex errors and concept drift which demand careful examination from multiple parties. In this paper, we attempt to study how entities in selected conceptual models have changed as well as ambiguity and other issues introduced due to reclaimed terms, accumulated changes, and errors. Recently, there has been a rising demand for LGBTQ+ conceptual models with multilingual labels, such as Spanish [50] and Chinese [38]. Given the suboptimal efficacy of machine translation [41], experts are required to put considerable effort into manually translating terms and associated information (e.g., preferred labels, alternative labels, abbreviations, comments, and scope notes). In addition, experts may overlook some alternative labels for an entity when translating into the target language, or some terms may lack accurate or culturally/ethically appropriate translations for either their preferred or alternative labels. Therefore, we investigate how multilingual information from one conceptual model can be used for the enrichment of the other.

We first construct a knowledge graph by studying **RQ5.1**: How can we construct a knowledge graph that captures identity-related information regarding LGBTQ+ concepts and their relations in representative conceptual models? Subsequently, we illustrate how this knowledge graph can help with tasks that are difficult to perform manually by the community. **RQ5.2:** How can the constructed knowledge graph be used to examine, enrich, and maintain evolving LGBTQ+ representations, including link discovery, change tracking, and multilingual enrichment? More specifically, we ground this research question on three challenging scenarios that communities face.

Broadly speaking, the community of LGBTQ+ conceptual models is made up of researchers, librarians, and volunteers who develop or maintain LGBTQ+ conceptual models of all kinds. In this chapter, we take a narrower definition and restrict it to those who are directly involved in the development and maintenance of LGBTQ+ conceptual models. The community relies heavily on manual work for development and maintenance: the gathering of (new) LGBTQ+ terms, discussion on the definition and updates, removing bias and discrimination in the (historical) terms, removing use of outdated terms, maintaining links between different conceptual models, translating terms and scope notes, etc. The community has not fully adopted computational approaches in its pipelines, making analysis and refinement on scale laborious. It was noticed that the links between entities in different conceptual models are often incomplete or outdated. Finding such missing links manually can be tedious. Thus, we attempt to suggest missing links automatically and evaluate them

manually. **Research Scenario A:** How can we use the knowledge graph to refine the conceptual models by discovering missing and outdated links? Another challenge that the community faces is about maintaining conceptual models: the lack of (semi-)automated means to analyze outdated links and their implications, outdated URLs that cannot be resolved or were redirected but not documented, drifting of meaning in concepts, as well as some ambiguity and potentially implied errors. Our preliminary analysis indicates that entities can engage in intricate situations that cannot be attributed to a single cause. We demonstrate how we can utilize the knowledge graph we constructed for the task to analyze LGBTQ+-related concepts' changes and ambiguity. The third research scenario is **Research Scenario B:** How can we use the knowledge graph to illustrate ambiguity and the change in concept? Recently, there has been a growing demand for multilingual lGBTQ+ vocabularies. It can be time-consuming to develop a linked open vocabulary totally from scratch. A less time-consuming approach is to translate existing terms. For example, the majority of labels in QLIT are simply a translation of Homosaurus' English labels (skos:prefLabel). Given that a concept could have multiple labels, and thus multiple (related/similar) translations, it can be hard to obtain all the translated labels. In the semantic web, we observed multilingual information that could be used to enrich the entities of other conceptual models (or help with initiating translation). This leads to our last research scenario, **Research Scenario C:** How much multilingual information can be reused for enriching entities? Our goal is not to precisely assess the number of accurate multilingual labels due to the necessity of specialized knowledge and the considerable time investment.

We provide all the code regarding the construction of the knowledge graph and data that could be released while respecting the licensing of each conceptual models[3]. Although the main purpose of this paper is not to discuss bias, sensitivity, ethics, and political issues and are beyond the expertise of the authors, these issues are unavoidable in some examples. We provide some discussion to the best of our capacity and experience.

The chapter is organized as follows. Section 6.2 introduces the conceptual models and presents the related work. In Section 6.3, we present details of data engineering.

3 The Python scripts, SPARQL queries, detailed explanation of experiments, annotated links by Swedish-speaking experts from QLIT, and the analytical results are in the supplementary material on Zenodo with DOI: `10.5281/zenodo.12684869`. Wikidata and QLIT are licensed under CC0. Thus, only datasets extracted from them are provided. However, due to the strict CC-BY-NC-ND license of GSSO and Homosaurus, the remaining data files and their associated intermediate results are only accessible upon request from IHLIA, the developers of GSSO and QLIT, and the authors. To reproduce the results or extend this work, the instructions can be found in the GitHub repository: `https://github.com/Multilingual-LGBTQIA-Vocabularies/Examing_LGBTQ_Concepts`.

We demonstrate the use of the integrated graph in use in three research scenarios in Section 6.4. Finally, we discuss the results in Section 6.5 followed by the conclusion in Section 6.6.

6.2 LGBTQ+ Conceptual Models and Related Work

As far as the authors are aware, there is no prior work directly related to the analysis of concept drift, and multilingual information of LGBTQ+-related concepts in the semantic web. Therefore, in this section, we summarize the updates of terms in the release notes of conceptual models and present some related research, but not all are exclusively about LGBTQ+ concepts.

Homosaurus is a linked data vocabulary focusing on LGBTQ+ terminology, aimed at enriching general subject term vocabularies, and it undergoes updates every six months. It was intended as a companion to LCSH [68]. Captured concepts are instances of `skos:Concept` and are related to each other using `skos:broader`. In recent years, three versions of Homosaurus have been released with updates every half a year. It contains English terms along with their corresponding translations in Dutch, offering a valuable bilingual dimension to its utility. Serving as a robust and state-of-the-art conceptual model widely used in libraries and heritages, Homosaurus significantly improves the findability of LGBTQ+ resources and information. Furthermore, Homosaurus offers a SPARQL endpoint[4] for accessing its data. At each release, information on updates of "labels" and newly added terms are provided on the website.[5] Despite some statistical analysis on terms in Homosaursus and how they overlap with others such as LCSH [23], to the authors' knowledge, there is no systematic examination or literature on the evolution of terms in Homosaurus and how its links to other conceptual models change.

The QLIT (Queer Literature Indexing Thesaurus) [46] is a recent Swedish thesaurus dedicated to indexing literature with LGBTQI themes. It was mainly used in Queerlit[6] [11], a bibliographic database on Swedish fiction. More than half of the terms in QLIT were translated from the English terms of Homosaurus (v3.3). Terms[7] without mapping to Homosaurus include terms related to symbolism (e.g.

4 https://data.ihlia.nl/PoolParty/sparql/homosaurus. Note that this endpoint may be delayed compared to the latest release on the Homosaurus website.

5 See for example https://homosaurus.org/releases/show/3. This latest release in January 2024 added 255 new terms and changed the 24 terms. One term has been replaced.

6 https://queerlit.dh.gu.se/

7 These examples were provided by Siska Humlesjö, from QLIT.

qlit:lc03jf41 with label "Regnbågssymbolik", where rainbows symbolize a LGBTQI theme in the story), terms of historical events for LGBTQI people in Sweden (e.g. qlit:nd56bj55 about "Ockupationen av Socialstyrelsen 1979", i.e. the occupation of the National Board of Health and Welfare in 1979), and historical Swedish terms for LGBTQI people (e.g. qlit:ox94in52 about Sapfister) [46]. QLIT has mappings to the two main Swedish library thesauri: Svenska ämnesord (SAO) and Barnämnesord (Barn) [46].

The Gender, Sex, and Sexual Orientation (GSSO)[8] ontology was designed to facilitate communication in gender, sex, and sexual orientation research and assist knowledge discovery in literature [42]. Its second version includes 10,060 entries, an increase from 6,250 in its first version. Its application ranges from clinical studies [44] to archives [67].

The three conceptual models mentioned above have links to LCSH (Library of Congress Subject Headings) [55]. Despite the fact that it includes some LGBTQ+-related terms, it was reported to have flaws and can be influenced by politics [91], which is beyond the authors' expertise. We study it from a semantic web point of view. Wikidata [73] contains identifiers of GSSO, QLIT, and Homosaurus. However, they have not been analyzed from this perspective to the best of the authors' knowledge.

These conceptual models serve distinct purposes, were revised at various times, are managed by teams with different expertise, and have not always been developed with full awareness of the changes of each other. Therefore, a perfectly unified representation of concepts and their relations is not possible. Braquet [15] briefly examined the provision of support for LGBTQ+ patrons within library settings, offering insights through various library-based scenarios. Dobreski et al. compared the overlap of the Homosaurus, LCSH, and Library of Congress Demographic Group Terms (LCDGT) [23]. They examined an old version of Homosaurus with 1,754 terms and found 618 terms related to identity. They reported 153 matches in the LCSH (exact matches and closest matches). Similarly, they found 176 matches in LCDGT, including faceted matches. Furthermore, it has been reported that there are outdated terms in LCSH, which leads to problems with the mapping of terms between Homosaurus and LCSH [23, 68]. However, to the authors' knowledge, there is no systematic report on the quality and reliability of these links and what kind of consequences would there be following erroneous links or involve ambiguous entities. A comprehensive comparison of all the entities released and their relations in these conceptual models is missing.

We observed redirection when resolving URIs of Homosaurus. Previous examinations of entities indicate frequent redirections among entities in identity graphs, with

8 https://github.com/Superraptor/GSSO

Table 17: A summary of the version updates of Homosaurus

Version	Release Date	#Terms newly added	#Terms removed	#Terms with labels changed	Available	Comment
v2.1	Jun 2020	99	0	0	No	no information found for earlier versions
v2.2	Dec 2020	23	0	0	No	
v2.3	Jul 2021	69	2	3	Yes	newly added terms include 45 pronouns-related terms
v3.0	Sep 2021		-	-	No	a new release
v3.1	Dec 2021	77	3	276	No	'LGBTQ' changed to 'LGBTQ+' in all terms. All terms formatted as [Term](LGBTQ) changed to LGBTQ+ [term]
v3.2	Jun 2022	263	0	148	No	8 terms were redirected
v3.3	Dec 2022	308	0	32	Yes	25 terms replaced some older terms
v3.4	Jun 2023	530	0	1	Yes	
v3.5	Jan 2024	255	0	24	Yes	the latest release

an estimate of 45% to 83% that maintains the semantics of identity [49]. However, no research has been done to study how many URIs have been redirected among those corresponding to LGBTQ+-related concepts as far as the authors are aware.

Concept drift refers to the phenomenon where the meanings or nuances of terms, concepts, or language evolve over time [74]. In the context of LGBTQ+ vocabularies, concept drift occurs as societal attitudes, understandings, and discussions about gender identity, sexual orientation, and related topics change and progress. An example is a term like *queer* which has changed in meaning over time. *Queer* was used as a slang for homosexuals as well as a term of homophobic abuse, but it has been reclaimed as an umbrella term for a coalition of culturally marginal sexual self-identifications in recent years [39]. The term *homosexual* is now considered somewhat *clinical* [15]. When it comes to the analysis of concept drift at scale, a method for large knowledge bases with instances of classes was proposed by Wang et al. [74], but it does not apply to our data due to the lack of instances. As far as the authors are aware, there is no systematic report on the concept drift and change in the field.

6.3 Data Engineering

Next, we answer our RQ5.1 by constructing a knowledge graph. In this section, we present details of selected conceptual models and links extracted for our analysis in Section 6.3.1. Section 6.3.2 includes details of multilingual labels extracted for the study of information reuse. Moreover, it was observed that some outdated URIs were redirected to new URIs, but not explicitly captured in the conceptual models. Thus, we present how these redirection relations were obtained in Section 6.3.3. Finally, we integrate all the links in Section 6.3.4.

6.3.1 Dataset selection and data preprocessing

Since Homosaurus' version 2.1 in June 2020[9], Homosaurus experienced 8 updates (on average twice per year). In this study, we focus on the last release of version 2 (v2.3) and the latest release of version 3 (January 2024, v3.5), which captures 3,149 terms. We noticed that some URIs[10]. were no longer maintained in version 2 and they were therefore replaced. Although replacement does not necessarily imply

9 Version 1 is no longer available on the official website. Some later versions (v2.1, v2.2, v3.0, v3.1, and v3.2) are no longer available on their website.

10 In this chapter, we use the prefix `h2` for the namespace `http://homosaurus.org/v2/` and `h3` for the namespace `https://homosaurus.org/v3/`. See more details of our use of prefixes of namespaces in the Appendix A.

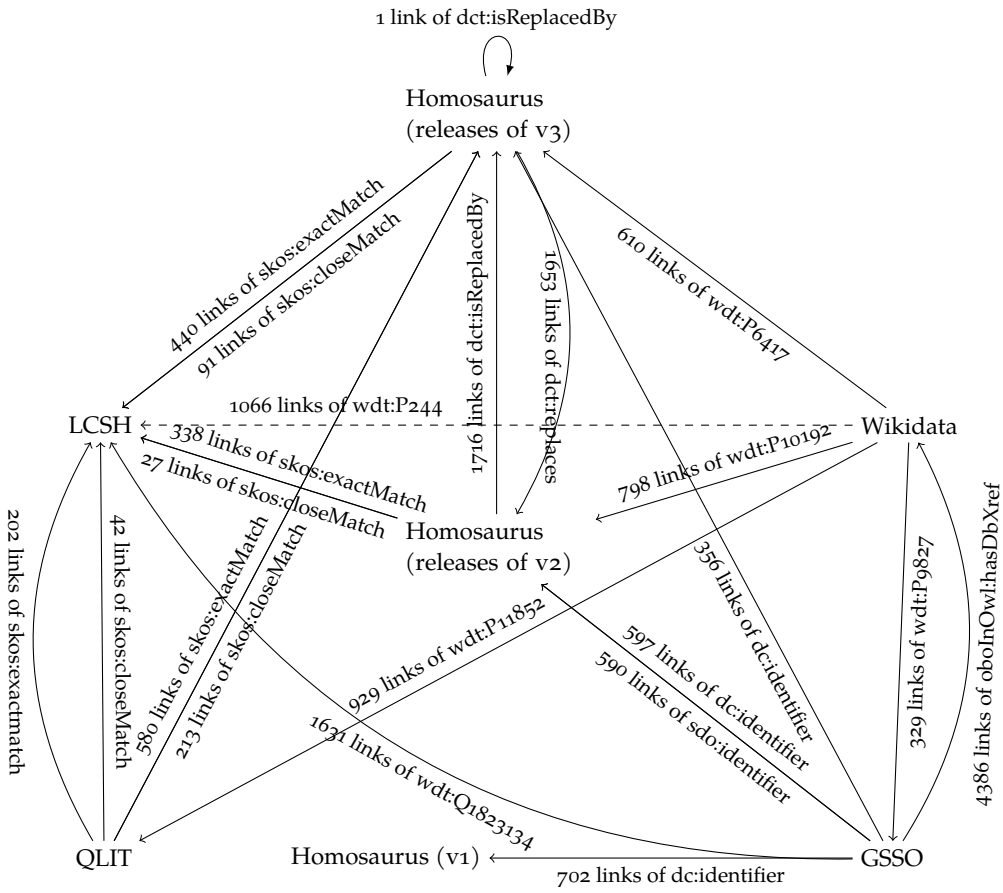


Figure 19: Conceptual models and their extracted links. The dashed edge indicates that only edges about LCSH entities that appear in the rest of the selected concept models were chosen in this study for further integration and analysis.

equivalence, we include this type of relation in our study to capture the evolution of conceptual models. Replacement was captured in Homosaurus using the relation `dct:isReplacedBy` and its inverse `dct:replaces`. Each entity is accompanied by an identifier (e.g. `homoit0002950`), a preferred label using `skos:prefLabel` and some using `skos:altLabel` as well as a scope note using `rdfs:comment`. Additionally, we noticed that some outdated URIs were redirected to newer ones. However, this information was not explicitly stated in the latest version. In Section 6.3.3, we extract these relations by resolving the URIs and capture this redirection relation for

further analysis. Homosaurus has links to LCSH[11]. They are being used together in libraries and heritages. Our examination shows that none of the entities in LCSH have outdated URI.

Recall that QLIT[12] was developed mostly based on translating terms in Homosaurus. Our examination shows that among QLIT's 914 terms, 774 exhibit either an exact match (`skos:exactMatch`) or close match (`skos:closeMatch`) to terms in Homosaurus, with an additional 140 terms not mapped to terms in Homosaurus, some of which are exclusive to QLIT. Only 244 links were found when examined against LCSH. Missing links will be discussed in Section 6.4.1. Moreover, its scope is limited to LGBTQI instead of LGBTQ+, thus the translation misses '+' in its Swedish labels.

Next, we extract links between GSSO and Homosaurus. It was noticed that it uses both `sdo:identifier` and `dc:identifier`. Since its publication in September 2022, the links from GSSO to Homosaurus version 2.2 and version 3.1 have not been updated. GSSO has 597 links of `dc:identifier` and 590 `sdo:identifier` to version 2 of Homosaurus. In comparison, there are only 356 `dc:identifier` links to version 3. Moreover, these links are using version 2.2 and version 3.1, which are outdated. This is because GSSO has not be updated since September 2022. Moreover, GSSO has 1,830 links to LCSH. The links of Wikidata and the two versions of Homosaurus are mostly asserted and maintained by experts and members of the Wikidata community. This demands significant human labor.

Links from Wikidata[13] to Homosaurus were provided using specified relations: `wdt:P10192` for entities in version 3 and `wdt:P6417` for entities in version 2. There are 610 and 798 links between Wikidata and versions 2 and 3 of Homosaurus, respectively. Similarly, for links from Wikidata to GSSO, a specific relation `wdt:P9827` was used. Only 329 links were found. As far as the authors know, links from Wikidata to GSSO and Homosaurus are maintained by hand by members of the Wikidata community without any use of automation. In total, 929 links were found for entities in QLIT, corresponding to the Wikidata property `wdt:P11852`. A total of 55,980 links were found between Wikidata and LCSH. Given that we study only LGBTQ+-related concepts. We restrict the entities to only those that appear in the links between conceptual models. Thus, only 1,066 links from Wikidata to LCSH were to be integrated

11 The N-Triple file of LCSH was obtained on 9th May, 2024 from `https://id.loc.gov/authorities/subjects.html`. For fast analysis of its entities, it was converted to its HDT format. Both were included in the supplementary material. We use the prefix `lcsh` for the namespace `http://id.loc.gov/authorities/subjects/`.

12 We use `qlit` for the namespace `https://queerlit.dh.gu.se/qlit/v1/`.

13 We use the prefix `wdt` for the namespace `http://www.wikidata.org/prop/direct/` and `wd` for the namespace `http://www.wikidata.org/entity/`.

and studied in the next steps. We obtain the complete URIs for the entities GSSO, Homosaurus v2 and v3, LCSH, and QLIT.[14]

6.3.2 Multilingual Information Extraction

GSSO consists of labels of 77 languages, while entities of Wikidata in the integrated graph (see Section 6.3.4) are associated with 507 languages. For the study of reuse of multilingual information to be presented in Section 6.4.3, we extract also multilingual information in GSSO[15] regarding labels (`rdfs:label`), paradigmatic synonyms (`oboinowl:hasSynonym`, `oboinowl:hasExactSynonym`, and `oboinowl:hasRelatedSynonym`), short names (`wdt:P1813`, about "short name"), (`wdt:P5191`, about "derived from lexeme"), replaces (`dct:replaces`), `sdo:alternateName`, and `owl:annotatedTarget` in all its languages. Similarly, 595,167 multilingual labels using `rdfs:label` and `skos:altLabel` about the entities in the integrated graph were extracted from Wikidata.

6.3.3 Redirection

Our analysis showed that Homosaurus switched its protocol from HTTP to HTTPS. Thus, 1,738 URIs were redirected to their HTTPS equivalent in version 2. No redirection was found from version 2 to 3. Another 63[16] redirections were found in version 3.[17] None were covered by existing replace relations (`dct:replaces` or `dct:isReplacedBy`). Among them, 61 Homosaurus entities could be from outdated release(s) of Homosaurus and were redirected to entities in the latest release (v3). Only 2 redirections were found between entities in the latest Homosaurus (v3). Moreover, we noticed that some URIs in version 2 cannot be resolved anymore, such as `h2:aromantic`.

14 Using the Wikidata SPARQL endpoint (`https://query.wikidata.org/sparql`), we can obtain the corresponding identifiers of Homosaurus, QLIT, and GSSO. To obtain the full URI, we process these identifiers using the "frommatter" as specified on their pages. Take GSSO for example, 002171 is the identifier. Using the formmater `http://purl.obolibrary.org/obo/GSSO_$1`, we can replace the place-holder and get the full URI: `http://purl.obolibrary.org/obo/GSSO_002171`. In this chapter, we use the prefix `obo` for the namespace `http://purl.obolibrary.org/obo/`. The preparation of Wikidata and links was done between 5th and 8th May, 2024.

15 We use the prefix `oboinowl` for the namespace `http://www.geneontology.org/formats/oboInOwl#` and `sdo` for the namespace `https://schema.org/`

16 We include the entities that no longer exist in the latest version of Homosaurus but were still referenced in GSSO.

17 All redirect relations were obtained using the *webdriver* of the *selenium* Python package (`https://selenium-python.readthedocs.io/`) between 6PM and 8PM on 30th April, 2024. We used the LOD server for this job.

Table 18: Extracted relations from sources and the number of triples

Source	Relations	#Triples	Comments
Homosaurus	`dct:isReplacedBy` and `dct:replaces`	3,370	Mostly links about replacing between version 2 and version 3.
	`skos:exactMatch` and `skos:closeMatch`	896	Links to entities in LCSH extracted from Homosaurus v2 and v3.
	`meta:redirecsTo`	63	Links representing redirection between entities in Homosaurus v3. Redirects for v2 were not included.
GSSO	`wd:Q1823134`	1,827	Links from entities in GSSO to subject headers in LCSH. It is mistaken to use `wd:Q1823134`. It was replaced by `wdt:P244` in the integrated graph.
	`oboInOwl: hasDbXref`	4,643	Links from entities in GSSO to entities in Wikidata
	`dc:identifier` and `sdo:identifier`	2,245	Links from entities in GSSO to entities in Homosaurus (all three versions)
QLIT	`skos:exactMatch` and `skos:closeMatch`	793	There are only links to Homosaurus v3.
	`skos:exactMatch` and `skos:closeMatch`	244	Links from QLIT to LCSH
Wikidata	`wdt:P244`	1,066	Selected links from Wikidata to LCSH
	`wdt:P6417` and `wdt:P10192`	1,408	Links from Wikidata to Homosaurus 2 and 3
	`wdt:P11852`	929	links from Wikidata to QLIT
	`wdt:P9827`	328	links from Wikidata to GSSO
Overall		17,812	The integrated graph involves 19,200 entities.

We use the redirection relation https://krr.triply.cc/krr/metalink/def/redirectedTo (meta:redirectedTo in short) as in Chapter 4 [88].

6.3.4 Integrating Extracted Links

Table 18 presents the components of the integrated graph. We noticed a mistake that, for links from GSSO to LCSH, wd:Q1823134 (representing the LCSH controlled vocabulary, rather than a property about links to their identifiers) was mistakenly used as a property. For consistency with the representation elsewhere, we change it to wdt:P244. When examining its links to Wikidata, it was also observed that GSSO has the URLs of webpages rather than the entities in Wikidata.[18] For example, http://www.wikidata.org/entity/Q190845 should not have been used as https://www.wikidata.org/wiki/Q190845 in its published data. We have corrected this in the integrated dataset. We further double-checked that all the 2,085 LCSH entities in the integrated file are in the latest version of LCSH except one due to a suffix of '.html' from GSSO, which was corrected in the integrated file. The integrated graph involves 19,200 entities with 17,812 links. Its N-Triple file is 2.4MB. Considering only entities with the above-mentioned links are in the integrated graph, no singleton is present.

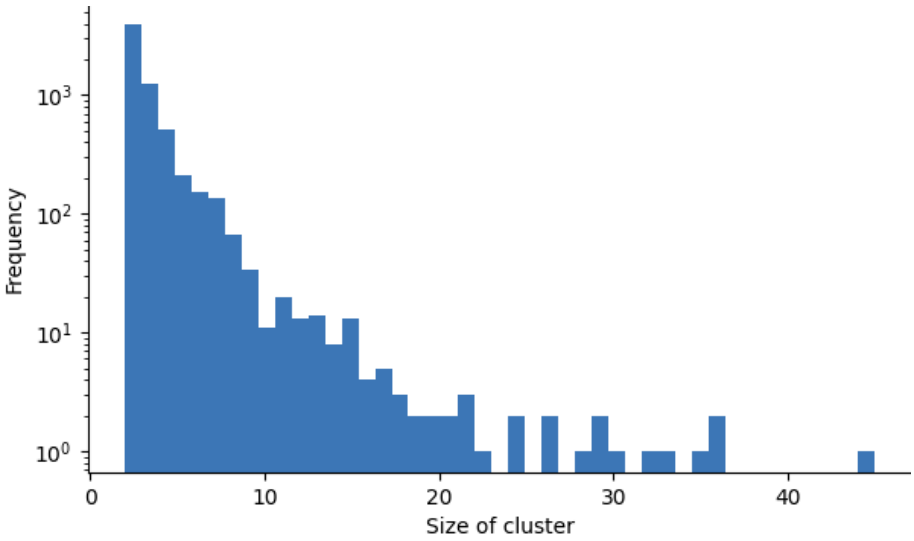


Figure 20: Frequency histogram of the size of clusters

18 See https://www.wikidata.org/wiki/Wikidata:Identifiers for details.

6.3.5 Clustering

We compute the *Weakly Connected Components* (WCCs) for our integrated directed graph. Recall the definition in Section 1.2, a WCC of a graph is a subgraph with maximal entities where there is a path between any of its entities, regardless of the direction of edges. In this chapter, WCCs are clusters of entities that mostly share a similar or related meaning. For our integrated graph, there are 6,406 WCCs. Figure 20 shows that the largest four WCCs consist of 45, 36, 36, and 35 entities, respectively. The largest one consists of 12 entities from Wikidata, 6 from GSSO, 5 from QLIT, 7 from Homosaurus v2, 6 from Homosaurus v3, 4 from LCSH, etc. More specifically, it involves related concepts about *human sexuality* (e.g. `wd:Q154136`), *Sexual intercourse* (e.g. `h3:homoit0000662` and `lcsh:sh85120739`), *Sex* (*Act*) (e.g. `h3:homoit0001267`), *fucking* (e.g. `h2:fucking`), *gender*(`h2:gender`), and *sexuality* (e.g `wd:Q3043188`). This example shows how the ambiguity of these entities accumulates into bigger clusters. This is in line with the intuition that larger weakly connected components may have more potential errors, which has been shown in Chapter 3. The size of these clusters is within the capability of manual revision, but the number of these WCCs remains significant.

6.4 Research Scenarios

In addressing RQ5.2, we study three research scenarios guided by community requirements to assess the data's usefulness in facilitating tasks that could ease the development and maintenance of the conceptual models and their links. In the first research scenario, we demonstrate how our data can be used to ease the maintenance of links by automatically detecting missing links (Section 6.4.1). In Research Scenario B, we explore how we can illustrate ambiguity and concept change (Section 6.4.2). Finally, in Research Scenario C, we evaluate how much multilingual information could be reused (Section 6.4.3).

6.4.1 Research Scenario A: Identity-related Link Discovery

As observed in Figure 19, there are missing links and outdated links. In this subsection, we study the discovery of missing links. There are multiple reasons for overlooking links during manual maintenance. A possibility is that the experts failed to keep track of all the original terms from all sources during the selection of terms to include or translate. To complicate the issue further, as far as the authors are aware, there is no method developed to address these missing links before release in the

community. Another possibility is that new terms are independently added to conceptual models, and during a new release, experts overlook linking these new terms to those in other updated conceptual models.

Next, we study Scenario-A using the knowledge graph we constructed. The intuition is that if two entities from different conceptual models are in the same WCC and are unique in their corresponding conceptual models in the WCC, they are likely to refer to identical or closely related things. We may infer that there is probably minimal confusion or concept change, which could result in complexity between them. A link could be added after manual examination. A counter example is when two entities, from the same conceptual model and of the same version, are in the same WCC, connected by some (undirected) path of directed links. If there is an entity from another conceptual model in the same WCC but not directly linked to any of the two above-mentioned entities, it is not feasible by the algorithm to determine which entity to link to.

In the next paragraphs, we only study the discovery of links between conceptual models of different projects. We do not study adding links between different versions of conceptual models. Instead, we consider different releases as different conceptual models. We take into consideration redirection and replacement, and only study the entities from the latest version. Moreover, for the case of Homosaurus, we exclude entities no longer maintained. In other words, we take advantage of the WCCs for the discovery of links between two conceptual models of different projects. We do not claim that these newly found links are for sure identity links. They need to be manually examined by experts before being added. Next, we report the links discovered for Homosaurus (v3) and QLIT respectively.

Regarding Homosaurus, only 531 links were observed between Homosaurus v3 and LCSH. However, 2,085 LCSH entities were found in the integrated graph.[19] Using the method mentioned above, 25 links were found. These discovered links have been submitted to the experts in Homosaurus for consideration before the next release.

Similarly, QLIT has only 244 links to LCSH. Using the same method, we found 105 potential missing links between QLIT and LCSH, which require further manual revision by Swedish-speaking experts. Given that some entities were redirected in Homosaurus v3, we also found one outdated link and its corresponding entity in the latest Homosaurus v3 (see `qlit:oj77yj15` in Figure 21). Further review by Swedish-speaking experts from the QLIT team shows that 78 (72.38%) suggested links should be included: 38 (36.19%) can be included using `skos:exactMatch` and equivalently another 38 (36.19%) using `skos:closeMatch`. 28 (26.67%) suggested links are incor-

19 This little overlap of concepts has been considered evidence by many that Homosaurus can be used as a complementary conceptual model of LCSH.

rect. Moreover, one suggested link is uncertain for experts. Finally, given that some entities were redirected in Homosaurus v3, we also found an outdated link in the latest Homosaurus (see `qlit:oj77yj15` in Figure 21).

Another kind of links that require maintenance are the outdated links, which are common as shown in Figure 19. When updating these links, a crucial question arises: do the new terms match the old ones exactly? If not identical, removing the old link and adding a link to the new term could result in a reduction in accuracy. This will be further examined in the next research scenario.

6.4.2 Research Scenario B: Ambiguity and Concept Change

As addressed in Section 6.2, there are reclaimed terms whose use can be ambiguous. During data processing, we also noticed that the change in some concepts is reflected between different versions of Homosaurus. These issues can be particularly complicated in a multilingual setting. These changes in concepts were captured by different conceptual models in different versions at various levels, reflecting complex coevolution. As shown before, experts' knowledge is required for manual refinement of entities and their links in this domain. Within the context of this study, our objective is not to resolve these problems or update the entities along with their links. Instead, we demonstrate how the WCCs can be used to assist in manual examination of various scenarios and how they illustrate the challenges mentioned above. In the following paragraphs, we provide a detailed examination of three representative scenarios, each highlighting an aspect: concept convergence, ambiguity, and changes in scope. Furthermore, we illustrate how various challenges are intricately intertwined, thereby complicating the task of designing a completely automated algorithm.

First, we use an example in Figure 21 to demonstrate how missing identity links, redirection relations, evolving concepts, as well as no longer maintained URIs can result in ambiguity and difficulty in identifying erroneous links. It was observed that `h3:homoit00442` replaced `h2:fetishism` and is the target of a few redirected URIs, including `h3:homoit0000102`, which was linked by many. It could be that `h3:homoit00442` is a merge of the concept of 'BDSM' and 'fetishism'. As a result, two clusters of entities about BDSM and fetish/kinks are in the same connected component. Further examination shows that it is the case that 'BDSM' is now an alternative label (`skos:altLabel`) for `h3:homoit00442` in the latest version. It was also noticed that a few URIs (highlighted in red) no longer appear in the latest version of Homosaurus, which leads to missing label information and relations. This example shows how multiple parties can have different views on concepts as conceptual models develop and the consequences of such changes.

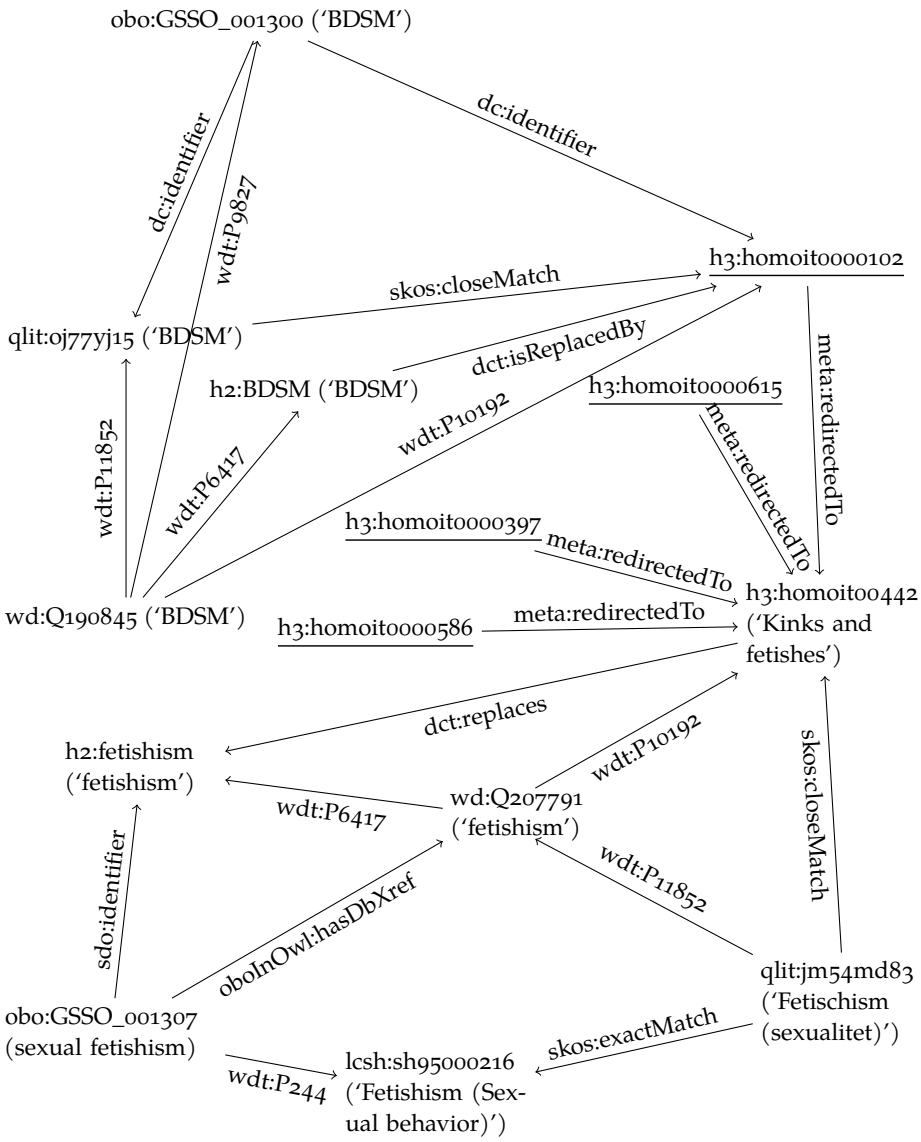


Figure 21: A subgraph of the seventh largest weakly connected component with 30 entities and 44 edges including concepts related to BDSM and sexual fetish. Labels that can be found are included. Some links and entities were omitted for clear visualisation. Entities with underlines are no longer in the latest version of Homosaurus.

In practice, concept drift frequently accompanies version changes, errors, and ambiguities, creating a complex network that necessitates thorough manual assess-

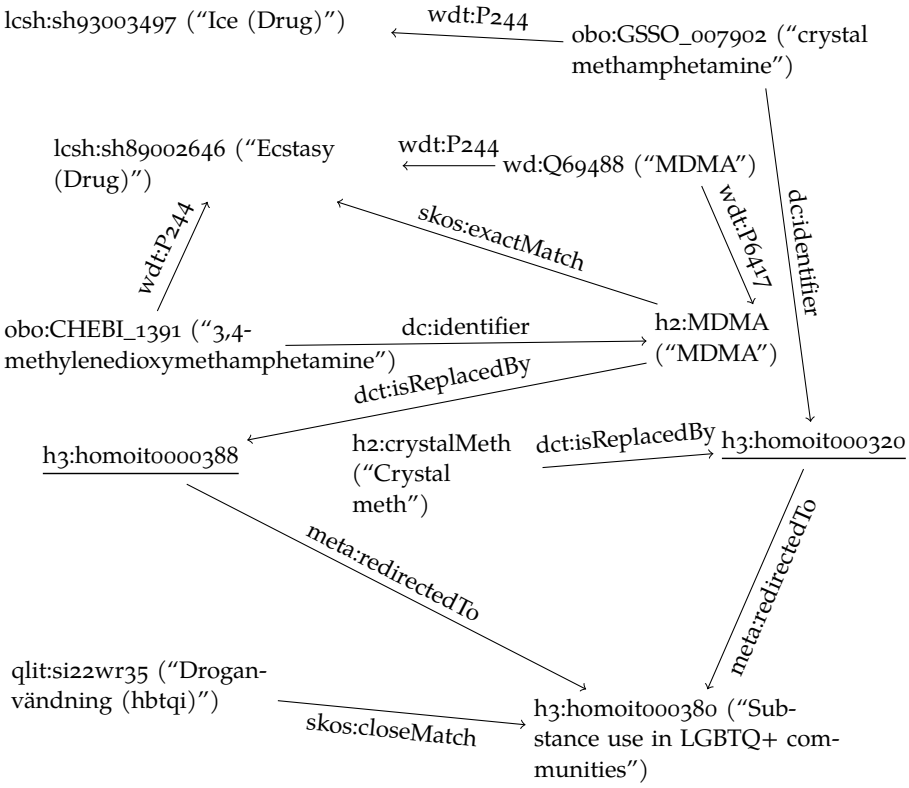


Figure 22: An example of concept drift and change involving MDMA, Crystal Meth, Ecstasy, Ice, Substance use in LGBTQ+ communities, etc. Some entities and links are not included for clear visualization. Highlighted with underlines are two entities in Homosaurus but not in v3.

ment. Next, we demonstrate the complexity by focusing on entities in GSSO and Homosaurus. Figure 22 is an example of concept drift involving the GSSO term "3,4-methylenedioxymethamphetamine" with the URI `obo:CHEBI_1391`, linking to the Homosaurus version 2 term "MDMA" with the URI `h2:MDMA`, which is replaced by the URI `h3:homoit0000388` in Homosaurus version 3. However, the URI no longer exists in the latest version, and was automatically redirected to `h3:homoit0000380`, which corresponds to the term "Substance use in LGBTQ+ communities" with a few other labels (`skos:altLabel`) including "Drug use (LGBTQ)", "Alcohol use in LGBTQ+ communities", etc. The term "3,4-methylenedioxymetha-mphetamine" from GSSO is the same cluster as the term "Substance use in LGBTQ+ communities" in Homosaurus version 3, which can be inaccurate and misleading. MDMA is an abbreviated form of methylenedioxymethamphetamine and is the main component of the popular party drug ecstasy. Crystal methamphetamine (a.k.a. Ice) is a different drug. Figure 22 illustrates the complexity and ambiguity when considering your identity by taking into account all related entities and their interconnected links. Moreover, when updating outdated links in conceptual models relying on translated Homosaurus terms, if such information were used, the subsequent conceptual model would inherit this problem. In the case above, `h3:homoit0000380` has a `skos:closeMatch` link to the QLIT term "Droganvändning (HBTQI)" ("Drug use (LGBTQ+)" in English) with the identifier `qlit:si22wr35`. Should a connection be established between this Swedish term in QLIT and GSSO through Homosaurus, the established link would be problematic and inaccurate, and following these links would result in confusion and incorrect labels.

Next, we show an example of the change in the scope of concepts using Figure 23. In GSSO, the term "being in love" with identifier `obo:GSSO_007692` has note "The state in which a person is when they are in love.". Its scope is not limited to the LGBTQ+ community. It was linked to `h2:beingInLove` ("Being in love"), which is a broader (`skos:broader`) term than "LGBTQ Love". Moreover, this entity in GSSO was linked to `h3:homoit0000107`, which was redirected to `h3:homoit0000894` ("LGBTQ+ love"). Meanwhile, we observed a change of scope in the release of Homosaurus v3.1 from 'LGBTQ' to 'LGBTQ+', which was an intentional scope change. Thus, `h2:LGBTQLove` was replaced by `h3:homoit0000894`. The above exemplifies the challenge that experts face when seeking for an update of the links of one conceptual model to the latest version of another (GSSO to Homosaurus, in this case). It shows how small changes can accumulate, leading to multiple steps that demands experts' close manual examination. In this case, simply adding a link from `obo:GSSO_007692` to `h3:homoit0000894` with the relation `sdo:identifier` is not correct.

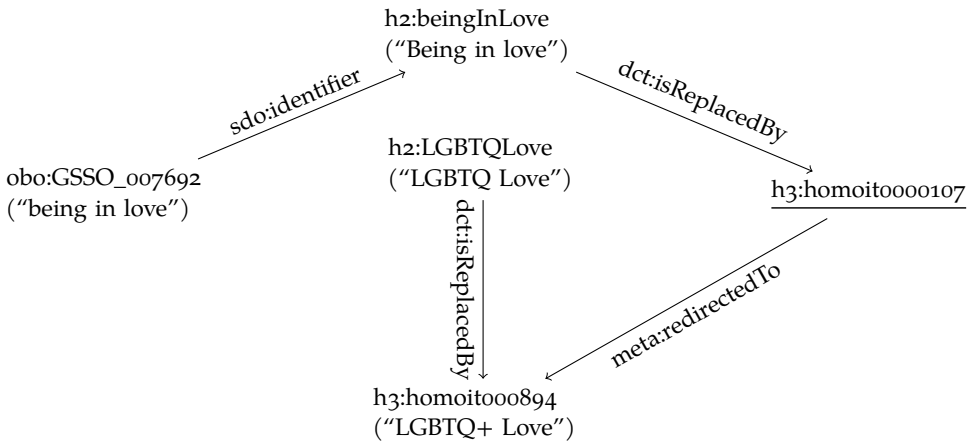


Figure 23: Convergence of "Being in love" and "LGBTQ love" to "LGBTQ+ love". Following the links could lead to a change in scope. Highlighted with an underline is an entity no longer maintained in Homosaurus v3. To simplify the depiction, this illustration does not encompass every single entity and link.

6.4.3 Research Scenario C: Multilingual Information Enrichment

As introduced in Section 6.2, the demand for multilingual conceptual models is increasing over the past years. For example, Homosaurus is widely used in libraries and archives, but despite being offered only in English and Dutch, there is still a significant need for reliable multilingual applications. As far as the author is aware, adding Spanish labels, scope notes, and comments have been entirely done by experts manually. Similarly, the development of QLIT started with a set of selected terms mostly from Homosaurus and was carried out by manually translating the labels (all the prefLabels and some altLabels when necessary). Constructing a complete conceptual model requires not only experts' awareness of most, if not all, of the alternative labels but also their subtle differences. We propose to take advantage of the labels that have been captured by other conceptual models. However, the quantity and quality have not been studied. In this research scenario, we perform some quantitative assessment on how many multilingual labels could be used as suggestions for experts and developers of Homosaurus. For comparison, we test our approach on QLIT. As illustrated above, entities in large WCCs involving multiple entities from the same source can face problems such as concept drift and ambiguity. Thus, the numbers reported below are to their upper bounds (i.e. not all of them could be correct or useful).

Here, we study the reuse of multilingual labels for entities not involved in WCCs where there is a "one-to-one mapping" to entities of Homosaurus v3. More specifically, for GSSO, we study how much information can be reused regarding a total of nine relations on labels, synonyms, and alternative labels (see Section 6.3.2). We compare it with Wikidata where multilingual information is being provided by `skos:altLabel` and `rdfs:label`.

There are 1,779 GSSO entities in the integrated graph. 1,681 are the unique entity of GSSO in the WCC. When further restricting to exactly one entity of Homosaurus v3 in the WCC (with consideration of redirection and replacement), only 48 entities remain. As illustrated in Table 19, the three languages with the most labels are English (356 labels for 48 entities), Danish (26 labels for 9 entities), and French (24 labels for 9 entities). The corresponding labels per entity are 7.42, 2.89, and 2.67 for English, Danish, and French, respectively. Fewer than 10 labels were found in this condition for Spanish and Turkish.

Table 19: The number of entities in GSSO and their average number of labels that have a one-to-one correspondence with Homosaurus v3 entities considering redirection and replacement within the same WCC. In the table are the top 5 languages with the most labels.

	English	Danish	French	Spanish	Turkish
Number of labels	356	26	24	8	6
Number of one-to-one mappings	48	9	9	4	3
Avg. labels per entity	7.42	2.89	2.00	2.12	2.00

As for Wikidata, there are 5,769 entities in the integrated graph, among which 4,939 are unique in their WCCs. When further restricting to the correspondence of exactly one entity in Homosaurus v3, only 429 entities remain. Table 20 shows the top five languages with the most labels. They are English (1,692 labels for 429 entities), Spanish (951 labels for 333 entities), Chinese (893 labels for 287 entities), Portuguese (881 labels for 299 entities) and German (824 labels for 298 entities). The corresponding labels per entity are 3,94, 2.86, 3.11, and 2.95 for English, Spanish, Chinese, and Portuguese, respectively.

It was noticed that the number of entities that has a one-to-one mapping between GSSO and Homosaurus v3 is significantly smaller than that of Wikidata. Despite that the entities take more relations into consideration when retrieving multilingual labels from GSSO, the number of multilingual labels is remarkably larger for Wiki-

Table 20: The number of entities in Wikidata and their average number of labels that have a one-to-one correspondence with Homosaurus v3 entities considering redirection and replacement within the same WCC. In the table are the top 5 languages with the most labels.

	English	Spanish	Chinese	Portuguese	German
Number of labels	1,696	951	893	881	824
Number of one-to-one mappings	429	333	287	299	298
Avg. labels per entity	3.94	2.86	3.11	2.95	2.77

data (with the exception that GSSO can provide more labels in English per entity in this setting). The average number of labels is 2.56 per entity for the top 20 languages with the most labels, with an average of 268.7 entities per language for Wikidata. This indicates that Wikidata can be a better choice when considering reusing its multilingual labels to enrich Homosaurus with manual examination. Nevertheless, the these multilingual labels as suggestions cannot be directly used but remain to be assessed after manual revisions by experts for each language. Take `h3:homoit0000295` ("Coming out") for example, it has label "sortie du placard" using `rdfs:label` in GSSO. Moreover, "sortir du placard" and "coming out" are synonyms for relation `oboInOwl:hasRelatedSynonym` as well as `oboInOwl:hasSynonym`, with three labels using `owl:annotatedTarget`: "sortir du placard", "coming out", and "sortie du placard". The labels retrieved as suggestions from Wikidata are similar: there is a label of "coming out" using `rdfs:label`; some more labels using `skos:altLabel`: "coming-out", "sortie du placard", "sortie de placard", and "sortir du placard". This shows that GSSO and Wikidata have overlaps in the labels they provide. Ultimately, it is the responsibility of the developers of conceptual models to determine the preferred label, the alternative label(s), and to recognize incorrect ones.

Similarly, we can extract labels from Wikidata as suggestions for QLIT. There are 914 entities with one prefLabel each but only a total of a total of 480 altLabels. Using the method above, we extracted 775 labels in Wikidata (524 prefLabels and 251 altLabels) for 524 entities. It was noticed that, in many cases, the difference is minor, either in the upper/lower case of the first character or the upper/lower case of 'hbtqi' (the Swedish word for LGBTQI). It remains a question if Wikidata has taken advantage of QLIT for its entries, or these terms are likely to be free from concept change and ambiguity due to the restriction of exactly one WCC.

6.5 Discussion

In Section 6.4.1, we demonstrated how we can find missing links using the WCCs. A further manual examination found some correspondence of terms between QLIT and Homosaurus. For example, a link between `qlit:tm80vg73` ("Pappor till homosexuella") and `h3:homoit0000427` ("Fathers of queer people") could be included. Similarly, `qlit:iq08ee58` ("Masters (hbtqi)") could be linked to `h3:homoit0000999` ("LGBTQ+ dominants"). These missing links can lead to discrepancies between conceptual models if not fixed before new versions of QLIT are released. Our proposed WCC-based method could reduce the effort of manually finding links between conceptual models. However, our approach is limited to the entities that are linked to at least one other entity. Moreover, assessing these links requires considerable manual effort. All the links in Section 6.4.1 have been submitted to the corresponding communities for manual revision by experts. For this reason, at the time of submission, the quality of newly found links in our approach remains unknown. This is also the case for our use case 3. Given that drift and change in the concept are mixed with ambiguity and errors, it is also difficult to evaluate the output in Section 6.4.2 where a significant amount of work is required for coordination between subcommunities.

In our study, `dct:replaces` and `dct:isReplacedBy` were included in the integrated graph, despite not necessarily implying equivalence relations. The value lies in the study of the dynamics and evolution of entities in Homosaurus and the impact on other linked entities. It was noticed that the concept change in Homosaurus is partially reflected by such links. Concept drift and change can result in potential duplicate terms (see Homosaurus terms in Figure 21 for example) that could violate the Unique Name Assumption [88]. This requires further manual examination for each concept model. Using the relation `owl:differentFrom` for different entities can ease future automated examination.

Section 6.4.3 showed that GSSO cannot suggest as many labels as Wikidata. This could be due to the restriction considering WCC. A further experiment with a relaxed condition only considering redirection and replacement shows that there are suggesting labels for as many as 115 entities for Danish and 47 for French. Given that QLIT has 914 labels of `skos:prefLabel` but only 480 labels of `skos:altLabel`, Wikidata and GSSO could be considered resources to enrich QLIT with alternative labels.

Table 21: The number of entities in GSSO and their average number of labels that have a one-to-one correspondence with Homosaurus v3 entities considering redirection and replacement but not necessarily in the same WCCs. In the table are the top 5 languages with the most labels.

	English	Danish	French	Spanish	Turkish
Number of labels	5,560	261	103	91	69
Number of one-to-one mappings	1,006	115	47	43	23
Avg. labels per entity	5.52	2.27	2.19	2.12	3.00

6.6 Conclusion and Future Work

In this chapter, we studied the properties of LGBTQ+-related concepts and their links in the semantic web. With the links extracted, we constructed an integrated graph and evaluated its use in three scenarios. We illustrated how our data can be used to find missing links and outdated links. We addressed the issue of concept drift and demonstrated how WCCs can assist the community in easily locating entities with potential issues. Finally, we showcased the reuse of multilingual information for Homosaurus. Our findings indicate that Wikidata offers a substantially greater number of multilingual labels than GSSO. Handling such tasks remains semi-automatic. Manual checking and community input are essential to evaluate our approach before implementation in reality.

In practice, experts can overlook the implications of diverging or converging concepts. This chapter demonstrates how we can provide such insight to the specialists. If any results require the intervention of others, discussion in the community would be advantageous. Our code could be reused for automatic detection of outdated links in the future. Given the significant amount of manual work, an interface that supports manual revision of the WCCs could be helpful.

Heterogeneous use of relations between entities, especially that for mappings was observed (see Table 19). It could benefit the community, especially for interoperability, if they consider adapting a common FAIR Implementation Profile, a structured representation of the community's decision on knowledge representation languages, semantic models, metadata, etc [83].

Our examination in Section 6.3.3 showed that redirection happened only in Homosaurus. The 63 redirected entities in version 3 remain to be manually checked

by experts if they maintained the original semantics [49]. If correct, they shall be included (e.g. as replacement relations) in future release of Homosaurus to make it possible for other conceptual models to perform automatic updates of their links.

This chapter presents briefly the analysis of concept drift in selected conceptual models. Their labels, comments, and scopenotes could be taken into consideration for further manual analysis. Drifting concepts could be analysed by studying their use in various contexts with natural language processing techniques [74]. Adding multilingual labels could make the scenario more complex. There could be issues about bias in translation. The community could adopt or develop some best practices to enhance accuracy, reduce bias and discrimination, and improve dataset maintenance [77].

Mistakes identified in GSSO should be corrected in upcoming versions (see Section 6.3.4). The 63 redirection links in version 3 presented in Section 6.3.3 remain to be checked to ensure that they maintain the original semantics [49]. If correct, they can be included (e.g. using the replacement relation) in the future release of Homosaurus. Some other conceptual models in the semantic web, such as LCDGT [23] and DBpedia [5], have been reported to contain some LGBTQ+ terms and could be included in some follow-up research together with the mappings from QLIT to some Swedish library thesauri [46]. Other links such as `skos:narrower`, `skos:broader`, and `skos:relatedMatch` can be explored.

Given the nature of the field, there is no perfect conceptual model [23]. As addressed in Section 6.2, there is no systematic report on the quality of links, the drift of concepts, and the potential harm of outdated links. Our data could serve to ease the manual work for this task. A good starting point is a list of historical terms (see `h3:homoit0000878`) and reclaimed terms (see `h3:homoit0001559`) in Homosaurus. Additional resources can be used as references (e.g. a list of terms about transgender and diversity in LCSH [1]). While the analysis presented in this chapter is limited to selected popular conceptual models, entities of LGBTQ+-related concepts exist widely in the semantic web (e.g. DBpedia, Wikidata), exhibiting a long-tail distribution. A complete examination requires a crawl of the semantic web and some accurate filtering, which are for future work.

Finally, the issues presented in this chapter also trigger the discussion on data provenance. Table 17 shows that some versions oh Homosaurus are no longer available. Therefore, it is not possible to retrieve the labels and scope notes of the corresponding URLs. Moreover, the community lacks the use of persistent IDs. It is not always the case that the metadata is available. As addressed in Section 6.4.2, the evolutionary nature of the data makes it essential to maintain the data frequently. However, only experts of Homosaurus have persistently dedicated time to frequent maintenance. This chapter calls for community effort for the examination and maintenance of the conceptual models and their links.

7 CONCLUSION AND FUTURE WORK

> It's good to know where you come from. It makes you what you are today. It's DNA, it's in your blood.
>
> Lee Alexander McQueen

In this chapter, we conclude the thesis by providing not only a summary of how each research question has been addressed but also how each research aspect has been addressed in each chapter in Section 7.1. Some discussion is provided in Section 7.2, followed by some future work in Section 7.3. Finally, the declaration on the use of generative A.I. is in Section 7.4.

7.1 Conclusion

Recall the main research question of this thesis: How can we take advantage of the graph structure of large integrated knowledge graphs for analysis and refinement? In this thesis, we proposed a comprehensive approach by merging logical properties with structural properties to analyze and refine large integrated knowledge graphs. We have studied graphs of (pseudo-)transitive relations, identity graphs, and the evolution of entities. We applied the approaches on KGs of two domain applications: an integrated graph for Finance, Economics, and Banking, and a KG with identity-related relations in the LGBTQ+ domain. Overall, we studied various aspects of knowledge graphs and their analysis and refinement.

Next, we summarize our answers to the research questions.

RQ1.1: How can we design algorithms to make knowledge graphs acyclic with respect to specific transitive or pseudo-transitive relations, while preserving as much original information as possible?

We developed a new algorithm for the refinement of transitive relations and pseudo-transitive relations in very large knowledge graphs. Our algorithm employed an SMT solver in implementation. Apart from evaluating on a gold standard we con-

structed, we demonstrated how our algorithm extended to weighted edges and studied how the performance improved.

RQ2.1: How can we formally define and validate a Unique Name Assumption (UNA) for large integrated knowledge graphs to support identity graph refinement?

We studied existing definitions of UNA and proposed the iUNA (internal UNA). Moreover, we provided means to specify the explicit sources and implicit sources (using labels and comments).

We constructed a gold standard and performed a study on error detection using different definitions. We computed the proportion of pairs that violate different definitions of UNAs. The chance that sampled pairs violate the iUNA is significantly lower than the other two and the baseline, with an estimated chance of violation to be less than 1%. This is good enough to indicate that iUNA could be used for the detection of errors.

RQ2.2: Can the UNA be used for the design of an algorithm to detect erroneous identity links in practice reliably?

We developed an algorithm for the refinement of the identity graph. The algorithm uses various definitions of UNA for the detection of pairs that violate the specified assumption. A path is then computed between the pairs. These pairs were then encoded into logical constraints, which were handled by our specified SMT solver. The decoded results indicate the edges to be removed. This algorithm can suffer from scalability due to its dependency on an SMT solver. Furthermore, we introduced weights and demonstrated how weights can improve the evaluation results. For both the training set and the evaluation set, the improvement in evaluation results was insubstantial.

RQ3.1: How can we interpret and model the implicit semantics of IRI redirection in integrated identity graphs?

For RQ3.1, we sampled entities from three categories of connected components. We obtained the redirection chains of sampled entities. We classified the scenarios of redirection and estimated the proportion of redirection that can be interpreted as identity links. Our best approximation is that between 45.1% and 83.2% of redirection links can indeed be taken as identity links, which is far from significant enough to conclude that redirection implies identity. Therefore, we suggested that when redirection happens for an entity, the semantics need to be checked before use. This was reflected in Chapter 6 for redirected Homosaurus' entities.

RQ3.2: What are the properties and structure of the redirection graphs?

As for RQ3.2, we study the redirection graphs by performing some statistical analysis and examining their graph-theoretical properties. We found that without any redirects, only 1% of all sampled URIs return meaningful information directly, rising to 33% after redirection. This means that an estimate of 66% of all URIs end in error, failure, or timeout at the end of their redirection chain. This addressed the importance of using updated data for the analysis of the LOD cloud. More discussion on future work will be presented in Section 7.3.

RQ4.1: How can the integration of domain-specific KGs in finance and economics enhance entity descriptions and contribute to identifying errors?

We showed how the information about entities can be enriched when combining different resources with increasing in- and out-degrees. Our manual analysis shows that large connected components could be erroneous.

RQ4.2: How do refinement challenges differ between domain-specific and general-purpose knowledge graphs?

We performed analysis on how the integrated knowledge graph has some minor errors due to incorrect identity links and (pseudo-)transitive relations. We showed that the scale is small, and its quality could be improved after manual refinement. This is different from that of larger integrated knowledge graphs, whose refinement requires the development of new algorithms and manual annotation of datasets for evaluation.

RQ5.1: How can we construct a knowledge graph that captures identity-related information regarding LGBTQ+ concepts and their relations in representative conceptual models?

First, with the understanding of how identity-related properties are used in selected conceptual models, we pick 12 identity-related properties and extract the triples. Moreover, we introduce some new relations found through redirection of URIs. We construct a knowledge graph with 19k entities and 17k relations about LGBTQ+-related concepts. This knowledge graph form the basis to answer the question below.

RQ5.2: How can the constructed knowledge graph be used to examine, enrich, and maintain evolving LGBTQ+ representations, including link discovery, change tracking, and multilingual enrichment?

We demonstrated its use in three research scenarios that are driven by community needs. First, we demonstrated how the knowledge graph can be used to find missing links between conceptual models. These missing links were sent to experts who are maintaining the corresponding conceptual models. Second, by examining the

weakly connected components, we demonstrated how the knowledge graph could be used to study how ambiguity, concept drift, and mistakes could be nested into complex scenarios. Finally, we provided a primitive evaluation on the number of labels that could be reused. We highlighted the few languages with the most labels that could be reused for enriching other conceptual models. These labels would also form the basis when developing new conceptual models, instead of starting completely from scratch.

7.2 Discussion

Our main contributions in this thesis can be summarized to the following five ingredients: quantifying the scale of the problems in the large KGs (I1), information and resources beyond explicit statements in knowledge graphs for analysis and refinement (I2), algorithms that take advantage of the structure to infer information that is not explicitly stated in the knowledge graphs (I3), evaluation methods for developed algorithms (I4), and examination of the change of property and structure (I5).

Table 22 summarizes our attempts for the research questions in each chapter, regarding various aspects. Regarding the analysis of knowledge graphs, we have studied (pseudo-)transitive (including class subsumption), identity graphs, a cross-domain integrated graph, as well as a graph extracted from identity-related relations in the LGBTQ+ domain. The scale of the problems studied in this thesis (I1) is significant. Using (strongly/weakly) connected components to reduce the problems' scale proved to be an effective approach. Large connected components can be obtained in two ways: by employing RocksDB's persistent key-value store and the networkx Python library (for smaller scale). These components could be reusable for other future studies. The LOD-a-lot knowledge graph consists of 28 billion unique triples (as edges). 356.9K edges out of 11.8 million edges (i.e., around 3%) of `skos:broader` are involved in SCCs [86]. 1.4K edges out of 4.4 million edges (i.e., around 0.03%) of `rdfs:subClassOf` are involved in SCCs [82, 86]. Moreover, we provided in Chapter 2 four measures on the estimation of scale and the efforts required to resolve complex nested cycles in subgraphs. Our analysis in Chapter 6 shows that the number of newly proposed identity-related links could be manually revised for some selected CCs of interest. We also evaluated the number of multilingual labels that could be reused for some languages. Our assessment shows that in both cases, the total number of such links remains hard to assess manually by experts. This shows that the maintenance of the conceptual models and their links requires the development of some semiautomatic approaches.

Table 22: Our contribution towards each research question regarding each ingredient of research

	Chapter 2 (RQ1.1)	Chapter 3 (RQ2.1 & 2.2)	Chapter 4 (RQ3.1 & 3.2)	Chapter 5 (RQ4.1 & 4.2)	Chapter 6 (RQ5.1 & 5.2)
I1 (scale analysis)	SCCs	CCs, 4 measures	Longest redirection hops.	CCs	CCs, measure of multilingual information reuse
I2 (new resources)	Weights (inferred weights v.s. counted weights)	Sources of labels, comments, weights (number of sources)	Redirection of IRIs	None	Redirection, missing identity links, reusable multilingual information
I3 (new algorithm)	Convert to a MAXSAT problem, solved by employing an SMT solver	Convert to a MAXSAT problem, solved by employing an SMT solver	Convert to a k-terminal problem; solved by employing a graph solver	None (examples of manual refinement)	None (examples of manual refinement)
I4 (evaluation)	Number of edges removed; a gold standard (manually annotated edges)	A new gold standard (8K manually annotated entities)	An estimate of the preservation of identity; manual assessment of 100 redirection chains	None	Manual assessment of newly discovered links; Quantitative assessment of labels for multilingual reuse
I5 (changes & evolution)	None	None	Analyzed redirection of IRIs for the study of Evolution of entities	None	Versions, Redirection of IRIs, concept drift, ambiguity, change of context, etc.

Regarding I2, we discovered several new resources that were not addressed in previous research. We studied how label-like and comment-like sources can help estimate the sources of entities when the sources are not explicit. We studied weights of two kinds: inferred weights (using logical properties) and counted weights. The latter relies on sources, which are not always available for integrated graphs in practice. Our experiments were made possible by the preservation of the raw data of LOD Laundromat. Calculating these weights takes substantial time. We demonstrated how these resources can be used to infer potentially different entities. Moreover, we constructed graphs of redirection for sampled entities and studied the evolution of entities through redirection in the semantic web. The assessment of redirection in Chapter 4 inspired the assessment of outdated entities in the integrated graph in Chapter 6. Finally, we propose reusing multilingual labels in the semantic web to enrich existing conceptual models.

Regarding I3, we used many scripts and developed multiple algorithms throughout the chapters in this thesis. The main algorithms proposed follow a pattern that a) computes the (strongly/weakly) connected components to narrow down to the neighborhood, b) takes advantage of graph structures and extracts cycles and paths, c) encodes the extracted cycles and paths to an SMT solver, and d) decoding from the output of an SMT solver for removing edges. In this approach, we manage to obtain refinement results without developing new domain-specific tools.

As for I4 (evaluation), the biggest problem we encountered was the unavailability of the evaluation dataset. This is mostly due to the scale of the problem: constructing a dataset manually remains time-consuming. Moreover, given the multilingual nature of the knowledge graphs we studied, evaluating relations can be difficult and rely on resources that are beyond the knowledge of human annotators. For the research in Chapter 2 and 3, we constructed an evaluation set manually. For the latter, the ANNit tool was developed. Due to lack of gold standard, we could only provide a range as our best estimate for the preservation of identity in Chapter 4. Some additional small-scale assessment was performed manually for chains of redirection. Finally, we showed a case where evaluation requires the knowledge of domain experts, which can be done at a very small scale for domain-specific and language-specific problems in Chapter 6. Evaluating such domain applications on a larger scale remains unfeasible.

Finally, for I5, we briefly studied the dynamics of such large integrated knowledge graphs. We studied samples of entities in the identity graphs and studied how their IRIs were redirected, which forms a trajectory of their evolution. To this end, some more work was conducted for LGBTQ+-related entities in the integrated knowledge graph. We showed how this problem can get complex very quickly as we take into account multilingual issues, concept drift, and ambiguity in natural language.

As shown in the last two columns of Table 22, the experiments of the domain applications discussed in Chapter 5 and Chapter 6 differ significantly. Restricting to specific domains does not necessarily make the refinement less challenging. While Chapter 5 demonstrated how problems could be narrowed down to a scale for manual refinement (no new algorithm needs to be developed and thus no new evaluation dataset required), Chapter 6 shows that despite that the scale reduced, ambiguity and other domain-specific issues need to be taken into account. To make matters worse, the refinement progress could be bounded by the lack of experts and their limited availability. Chapter 5 did not delve into topics concerning versioning, concept drift, or other dynamic issues. This omission is primarily due to the fact that the majority of the concepts are linked to their static definitions, with other dynamic aspects being reserved for exploration in subsequent research efforts.

Taking everything into account, the proposed comprehensive approach attempted to merge logical characteristics with graph properties to analyze and refine large integrated knowledge graphs. Despite the scalability not being optimal, we demonstrated how these methodologies can be applied to particular domains, illustrating their versatility and effectiveness in domain-specific applications. In addition, we generated resources and demonstrated their use, which could contribute to future studies. These two domain applications illustrate that new algorithms are not always necessary for applications of integrated knowledge graphs. Next, we discuss limitations in our approach and its applications in domain contexts.

Throughout the development of this thesis, a multitude of valuable insights have been acquired regarding data handling and analysis processes. This thesis is structured around a central research question focusing on the analysis and refinement of integrated knowledge graphs. The experience of correcting the errors we faced initiated a deeper investigation of the root causes and sources of these errors. Some errors were due to publishing additional edges computed from transitive closure without checking [86]. Some errors were due to the poor quality of some knowledge graphs integrated [82]. A lesson we learned in chapter 6 is that, when taking into consideration the dynamics of the semantic web and the ambiguity of domain-specific concepts, the mistakes can accumulate to more complex scenarios, which can be difficult for manual refinement. Some similar issues were observed during data processing in this thesis. Although SCCs have been shown to be able to highlight potentially erroneous links, they do not offer the capability to locate all the errors because not all erroneous links necessarily make these components larger. For example, an erroneous subclass subsumption assertion could simply be wrong while not introducing any cycle or making existing cycles more complex.

Due to the scale of problems we studied, a notable bottleneck in our research is the availability of datasets that can be used for evaluation. For complete evaluation results, in Chapter 2 and 3, we manually examined some data entries and constructed

gold standards. In Chapter 4, we manually reviewed some chains of redirection. Given the scale, manually constructing such datasets can be time-consuming or unfeasible in some cases. Developing alternative evaluation methods could be helpful. The ad-hoc tool we developed for assisting with manual annotation, ANNit, is not directly applicable to other scenarios without further development. As addressed in Chapter 4, some entities are using updated IRIs, which adds more complexity to the manual annotation. Using (strongly/weakly) connected components can help the algorithms be more focused on subgraphs where the problems occur. However, the scalability remains limited by the SMT solver employed, whose performance can be reduced significantly when, in each iteration, the number of clauses (encoded from cycles detected) goes up to hundreds or thousands, from our observation. In future work, refinement algorithms could address scalability.

As addressed in the section above, we discovered new resources that could be used for refinement. However, these resources rely on the preservation of (raw) data used for developing the original graphs, which may not be available to all. Most recently, we noticed that the website of `sameAs.cc` is no longer available[1]. As addressed, some versions of Homosaurus were not available anymore, which makes some links invalid and prevents some analysis. This shows the importance of data preservation. In addition, the use of a strict license (CC BY-NC-ND) by Homosaurus and GSSO can be an issue for future work. This license prevents any derivation, which can hurt data reuse. Some recent attempts using machine translation tools for the enrichment of multilingual labels have reported suffering from making their data available due to this license [81]. In contrast, the use of `CC0` by Wikidata does not entail this issue [73].

Overall, these observations and experiences call for new hybrid approaches that take into account graph properties, semantics, dynamics, and other aspects. Next, we outline some future work.

7.3 Future Work

Similar to our definition of pseudo-transitive relation, we can define properties such as pseudo-reflexive, pseudo-irreflexive, pseudo-symmetric, pseudo-asymmetric, etc. Table 23 summarizes representative relations and their graph properties for future research. Moreover, some relations could be put together by merging the corresponding subgraphs for study, such as `skos:broader` and `skos:broaderMatch` (as in chapter 5). The corresponding subgraphs remain for future work.

1 Accessed on 11 June 2025.

Relation	Transitivity	Symmetricity	Reflexivity	Graph Property
iwwem: dependsOn	transitive	unspecified (pseudo-asymmetric)	irreflexive	no cycle of any size allowed.
skos:broader	pseudo-transitive	unspecified (pseudo-asymmetric)	unspecified (pseudo-reflexive)	self-loops are valid but redundant. Bigger cycles are likely to be wrong.
pav:hasEarlierVersion	transitive	asymmetric	unspecified (pseudo-irreflexive)	self-loops are valid but should not be present. Cycle of size two or bigger can be mistaken.
owl:bottom-ObjectProperty	transitive	asymmetric	irreflexive	cycle of any size is invalid.

Table 23: Properties of selected relations in LOD-a-lot

Moreover, some relations represent opposite meanings. As introduced in Chapter 1, `owl:differentFrom` is used for two different entities, which is the opposite of `owl:sameAs`. Given that `owl:differentFrom` is not transitive, the corresponding graph could be less erroneous than the identity graph. If there is a path in the identity graph between two entities where there is a trusted edge in the subgraph of `owl:differentFrom`, it could be the case that some edge in the path in the identity graph is mistaken. This could be used for the refinement of the identity graph. Such detected paths could be integrated as an extension into the algorithm presented in Chapter 3. Note that some inequality relations are using `owl:AllDifferent`[2], indicating that a list of individuals that are all different. Such cases require some preprocessing before being used for refinement.

Prior work has documented that 19% of unique URIs in identity graphs do not exist after only two years since publication [21]. Our redirect analysis in Chapter 4 shows that information of only around 1% of entities is still maintained at their original location, while some 33% of entities can still provide valid information when taking redirection into account, showing that redirection plays a crucial role in the analysis of dynamics in the LOD cloud. However, the examination we had remains small in scale. Chapter 4 restricts the analysis to entities in the identity graph. In future work, we would like to remove this restriction and compare against the redi-

2 https://www.w3.org/TR/owl-ref/

rection of IRIs in the LOD cloud. Constructing a much larger one about redirection and the dynamics of identity change could be a useful resource. Moreover, it could be interesting to examine how redirection can help update existing mappings. Moreover, our analysis showed that DBpedia entities are frequently redirected. This again addressed the importance of data/knowledge management, especially DBpedia. In addition, when combined with a new claw of the LOD cloud, such a redirection graph could be useful as a resource to show the dynamics in knowledge graph evolution. The use of DBpedia Databus[3] could make some contribution to data/-knowledge management and preservation as well as the study of knowledge graph evolution. An alternative solution is to consider the use of Decentralized Identifiers (DIDs) [65], which offers verifiable, decentralized digital identity. This could potentially be one of the means for better management of identity and handling redirection for outdated IRIs.

The research in this thesis relies on existing large integrated knowledge graphs that represent the semantic web from 2015, which is outdated. As indicated in Chapter 4, a significant number of entities have been updated, some of which have been captured by redirection. Over the past years, the linked data in the semantic web has adopted new standards, used new ontologies, and has increased in scale like never before. This requires upscale facilities and tools for data publishing, data preservation, data processing, and analysis. For future work on the integration of knowledge graphs, it can be beneficial to take some factors into account, such as the size of SCCs of graphs of (pseudo-)transitive relations and WCCs of identity relations. Keeping track of the errors during integration could be beneficial, especially for projects that result in KGs at a very large scale. Upon the completion and availability of a newly developed web-scale Knowledge Graph (KG), it will be feasible to conduct a series of experimental evaluations employing this resource. Subsequently, the outcomes of these experiments can be systematically compared against the benchmarks that we have established in this thesis. In addition, forthcoming data infrastructures could play a more significant role if they are not only about the publishing and preservation of data but also its validation. This function can potentially reduce the incidence of errors.

When refining knowledge graphs at scale, it can be difficult to handle multilingual labels as input for our algorithms. These generic algorithms did not have the capability to understand the multilingual labels and their associated contexts. As Large Language Models (LLMs) gained popularity in the field of knowledge graphs over the past years, they could be used for the problems we addressed. Some recent work attempted to refine noisy knowledge graphs [24]. Another recent attempt was to refine class membership relations in knowledge graphs using several LLMs [3].

3 https://databus.dbpedia.org/

These attempts demonstrated the potential of this approach. Moreover, it remains to be studied whether the use of LLMs can be combined with our approach, especially for the refinement of the identity graphs. For example, in case a pair of entities violates the UNA, a path can be retrieved. The information of pairs of entities for each edge along the path could then be evaluated using LLMs to determine which edges are most likely to be in error, and thus removed. Alternatively, when using an SMT solver, the weight of an edge could be reduced if an LLM returns a negative result for identity checking.

During the examination of concepts pertinent to the LGBTQ+ community, it became apparent that the community does not adhere to any established best practices for the publication, management, and preservation of data. This reduces findability and reusability. It was also noticed that some links remain outdated. Poor data management and preservation can harm the tracking of concept changes. Most recently, there has been an attempt to add Spanish labels[4] to Homosaurus [50]. Best practices and data management guidelines were proposed with a focus on adding multilingual labels and data publication [77,81]. Ensuring the integrity and quality of the data in the semantic web presents considerable challenges and requires active participation and contributions from various community groups. Some systematic validation and domain-specific criteria could be implemented before the publication of KGs to reduce the errors in the LOD cloud.

The usability of integrated KGs as presented in chapter 5 and chapter 6 could be further explored. Our approach could be adopted in future work to study how such entities overlap between KGs and how their perspectives change. A topic of interest is the study of interoperability in practice, especially when retrieving entities' different properties from multiple resources of knowledge graphs. Our approach could also be adopted to research at the metadata level, which could be specified differently between data infrastructures (especially data repositories and data registries). The corresponding integrated knowledge graph may be used to help find the corresponding terms in the right knowledge graph by following identity links for the conversion and enrichment of metadata. This shows the potential for use in sharing data across infrastructures and may reduce manual work.

A topic that requires further investigation is the explainability of the results. Although our methodology might not be as complex as those employing neural networks, the explainability remains to be explored and has been deferred to future research.

4 See for example, their latest release at `https://homosaurus.org/v4`.

7.4 Declaration on Generative A.I.

While preparing this thesis and the papers included, the author of the thesis used TeXGPT (through Writefull on Overleaf) and ChatGPT to paraphrase some sentences. In addition, the LaTeX code of some figures was generated with the help of ChatGPT. Neither was used to generate ideas, complete sentences, or paragraphs. Due to limited Dutch language skills, ChatGPT and Google Translate were used for the Dutch summary (translated from the English summary).

A PREFIXES OF NAMESPACES

According to the W3C Recommendation[1] on RDF 1.1 Concepts and Abstract Syntax, the IRIs in an RDF vocabulary often begin with a common substring known as a *namespace IRI*. The term "namespace" does not have a well-defined meaning in the context of RDF. It is sometimes informally used to mean "namespace IRI" or "RDF vocabulary". The *prefix* is a short abbreviation of the corresponding namespace that is used in the Turtle format of RDF files for example. See the example about the author of this thesis below.[2] In case entries of namespace prefixes in this thesis are not included in Table 24, they can be found on the website https://prefix.cc/.

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<https://shuai.ai/shuai_wang>
  a foaf:Person ;
  foaf:familyName "Wang"^^xsd:string ;
  foaf:givenName "Shuai"^^xsd:string ;
  foaf:homepage <https://research.vu.nl/en/persons/shuai-wang> ;
  foaf:knows <http://dbpedia.org/resource/Frank_van_Harmelen> .
```

Listing 1: The code in Turtule format about the author of the thesis

1 https://www.w3.org/TR/rdf11-concepts/

2 In the example, we used https://www.easyrdf.org/converter for converting code from that could be edited interactively online at https://json-ld.org/playground/.

Table 24: Namespace prefices and their corresponding IRIs in alphabetical order

Namespace Prefix	Namespace IRI
bro	http://bankontology.com/br/form/
dc	http://purl.org/dc/elements/1.1/
dct	http://purl.org/dc/terms/
dbr	http://dbpedia.org/resource/
foaf	http://xmlns.com/foaf/0.1/
fro	http://finregont.com/fro/ref/LegalReference.ttl#
fro-xbrl	http://finregont.com/fro/xbrl/
genealogy	https://example.org/genealogy/
h2	http://homosaurus.org/v2/
h3	http://homosaurus.org/v3/
iwwem	http://ubio.bioinfo.cnio.es/biotools/IWWEM/iwwem.owl#
lcsh	http://id.loc.gov/authorities/subjects/
lkif-norm	http://www.estrellaproject.org/lkif-core/norm.owl#
lkif-core	http://www.estrellaproject.org/lkif-core/
meta	https://krr.triply.cc/krr/metalink/def/
obo	http://purl.obolibrary.org/obo/
oboinowl	http://www.geneontology.org/formats/oboInOwl#
owl	http://www.w3.org/2002/07/owl#
pav	http://purl.org/pav/
qlit	https://queerlit.dh.gu.se/qlit/v1/
ro	http://www.obofoundry.org/ro/ro.owl#
rdf	http://www.w3.org/1999/02/22-rdf-syntax-ns#
rdfs	http://www.w3.org/2000/01/rdf-schema#
sdo	https://schema.org/
sioc	http://rdfs.org/sioc/ns#
skos	http://www.w3.org/2004/02/skos/core#
sxml	http://topbraid.org/sxml#
wd	http://www.wikidata.org/entity/
wdt	http://www.wikidata.org/prop/direct/
xml	http://www.w3.org/XML/1998/namespace/
xsd	http://www.w3.org/2001/XMLSchema#

B PILOT STUDY: RESOLVING CYCLIC CLASS SUBSUMPTION RELATIONS

This pilot study is based on the following paper.

- Shuai Wang, Peter Bloem, Joe Raad, and Frank van Harmelen. Submassive: Resolving subclass cycles in very large knowledge graphs, 2020. Workshop on Large Scale RDF Analytics, DOI: 10.48550/arXiv.2412.15829

Large knowledge graphs capture information about many different relations. Among them, the subclass subsumption assertions are usually present and expressed using constructs of `rdfs:subClassOf`, for example (`example:Dog`, `rdfs:subClassOf`, `example:Animal`). Ideally, such triples form a structure of hierarchy, if not considering reflexive relations. From our examination, publicly available knowledge graphs contain many potentially erroneous cyclic subclass relations, a problem that can be compounded into a more complex form when different knowledge graphs are integrated. Such cycles can be harmless. For example, reflexive cycles are simply tautologies (always true) and are therefore redundant. Due to the transitivity, cycles are only correct when all classes in the cycle are equivalent, otherwise, they represent a source of error. In practice, it is unlikely that a data engineer uses two subclass relations ($A \sqsubseteq B, B \sqsubseteq A$) or multiple (e.g., $A \sqsubseteq B, B \sqsubseteq C, C \sqsubseteq A$) to represent equivalence. In this pilot study, we present an attempt to refine such knowledge graphs by encoding the problem of cycle-resolving to a problem for an automated reasoning solver with optimization. By *resolving a cycle* we mean that at least one of its edges is deleted, resulting in an incomplete cycle. In other words, after self-loops are excluded in the preprocessing, the goal is to obtain a directed acyclic graph (DAG). Note that a DAG can be different from a hierarchy (i.e., a spanning tree), where there is assumed to be a unique root and each node has exactly one immediate super-class.

For a graph, a *simple cycle*, or an *elementary circuit*, is a closed path where no node appears twice except that the first and last node are the same. If all simple cycles are resolved, the graph becomes a DAG. By introducing a propositional variable to each edge representing whether an edge is removed or not, we can describe if there is a cycle (the propositional variables corresponding to all the edges involved in a cycle are True). Such logical descriptions are called *clauses*. To maintain as much information as possible, we formulate this problem as a Partially Weighted MAXSAT problem:

we remove as few edges as possible to resolve all the cycles. It has constraints in two forms: *soft constraints* and *hard constraints*. The fact that we need to resolve all the cyclic connections corresponds to hard constraints (as encoded above). A soft constraint is in the same form except that a weight is associated with each clause. The goal of this type of problem is to satisfy all the hard constraints while maximising the sum of the weights associated with the satisfied soft constraints, and thus it is a constrained optimisation problem. In this case, our soft constraints are simply each of the propositional variables corresponding to the edges (relations). For fairness, we assign a fixed identical weight to each edge.

We do so iteratively for each neighbourhood. For each local neighbourhood, we obtain a set of simple cycles from the subgraph. Then, in order to remove the minimum number of edges to break these cycles, we encode all the cycles as clauses and let a solver find the solution. In this way, the MAXSAT procedure will remove all cycles (i.e. satisfying the hard constraints), while keeping as many of the relations as possible (i.e. maximally satisfying the soft constraints). For the example above, the hard constraint is s as encoded above with the corresponding soft constraints as propositional variables with identical weights of 1 each. Detailed design of the algorithm can be found in the paper [82].

This pilot study served as a preliminary attempt to analyze and refine very large integrated KGs in this thesis. We learned the lesson that locating cycles in certain neighborhoods of the graph is a very essential first step before applying solving algorithms. Following this, we discovered that simply focusing on strongly connected components is enough (see the definition and explanation in Section 1.2): if there is a cycle, it must be in a connected component. This computation can take advantage of existing software packages such as the *networkx* Python package. We also see the potential to further develop this approach and apply it to relations with similar properties. This inspired the next research project to be presented in Chapter 2 on refining subgraphs corresponding to (pseudo-)transitive relations. In this study, we manually evaluated the edges removed. We recognize that there is no ground truth, and the missing gold standard will remain a drawback for evaluation in follow-up research. Given the scale, it would be impossible to compute the recall in evaluation. Moreover, as explained above, not all cycles are erroneous. Thus, the evaluation result remains an estimation rather than precise results. It demonstrated the applicability of the Partially Weighted MaxSAT solver to knowledge graph refinement. In the following sections, we use a more generic description of this kind of solver: the SMT (Satisfiability Modulo Theory) solver. More specifically, we use the Z3 [12], a state-of-the-art solver. More details can be found in the paper [82].

C SCIENTIFIC CONTRIBUTION

C.1 Publication, Presentation, and Contribution

The content of Chapter 2 has previously appeared as a standalone paper [86]:

- Shuai Wang, Joe Raad, Peter Bloem, and Frank van Harmelen. Refining transitive and pseudo-transitive relations at web scale. In Ruben Verborgh et al., editors, *The Semantic Web - 18th International Conference, ESWC 2021, Virtual Event, June 6-10, 2021, Proceedings*, volume 12731 of *Lecture Notes in Computer Science*, pages 249–264. Springer International Publishing, 2021.

CRediT author statement
Shuai Wang: Conceptualization, Methodology, Investigation, Data Curation, Formal analysis, Software, Visualization, Writing - Review & Editing, Writing - Original Draft, Validation. **Peter Bloem**: Conceptualization, Investigation, Methodology, Formal analysis, Supervision, Writing - Review & Editing. **Joe Raad**: Conceptualization, Methodology, Investigation, Supervision, Formal analysis, Writing - Review & Editing, Validation. **Frank van Harmelen**: Conceptualization, Formal analysis, Supervision, Methodology, Writing- Reviewing and Editing, Validation.

This chapter has the following changes compared with the original paper:

1. the research question is better addressed.
2. Table 2, 4, and 3 have been formatted for a clearer presentation. Figure 8 has been changed back to its colorful version.
3. Table 1 was removed in the original paper due to page limit. It was added back to Section 2.3.2 together with more details about the existing measures.
4. Figure 7 was added back with the corresponding explanation to give a better intuition about the measures introduced.
5. The discussion section (Section 2.6) was extended. Some more future work has been added to Section 2.6.3.
6. Some text has been moved to the previous chapter to avoid repeated introduction to concepts.

In addition, the pilot study in Appendix B and some text of this chapter have previously appeared as a standalone paper [82]:

- Shuai Wang, Peter Bloem, Joe Raad, and Frank van Harmelen. Submassive: Resolving subclass cycles in very large knowledge graphs, 2020. Workshop on Large Scale RDF Analytics, DOI: 10.48550/arXiv.2412.15829.

The 2020 Workshop on Large Scale RDF Analytics was a workshop co-located with the European Semantic Web Conference (ESWC) 2020 conference. Unfortunately, the proceedings that included this peer-reviewed paper have never been published as the organizers promised. Therefore, the paper has been made available on ArXiv [82] with the supplementary material (code and data) published on Zenodo [90]. For easy reuse, the resulting cycle-free graphs have been published as a standalone resource on Zenodo [75].

CRediT author statement

Shuai Wang: Conceptualization, Methodology, Investigation, Data Curation, Formal analysis, Software, Visualization, Writing - Review & Editing, Writing - Original Draft, Validation. **Peter Bloem**: Conceptualization, Investigation, Methodology, Formal analysis, Supervision, Writing - Review & Editing. **Joe Raad**: Methodology, Investigation, Supervision, Formal analysis, Writing - Review & Editing, Validation. **Frank van Harmelen**: Conceptualization, Formal analysis, Supervision, Methodology, Writing- Reviewing and Editing, Validation.

The content of Chapter 3 has previously appeared as a standalone paper [87].

- Shuai Wang, Joe Raad, Peter Bloem, and Frank van Harmelen. Refining large integrated identity graphs using the Unique Name Assumption. In Catia Pesquita et al., editors, *The Semantic Web - 18th International Conference, ESWC 2023, Hersonissou, Greece, May 28 - June 1, 2023, Proceedings*. Springer Nature, 2023.

CRediT author statement

Shuai Wang: Conceptualization, Methodology, Investigation, Data Curation, Formal analysis, Software, Visualization, Writing - Original Draft, Writing - Review & Editing, Validation. **Peter Bloem**: Conceptualization, Investigation, Methodology, Formal analysis, Supervision, Writing - Review & Editing. **Joe Raad**: Conceptualization, Methodology, Investigation, Supervision, Formal analysis, Writing - Review & Editing, Validation. **Frank van Harmelen**: Conceptualization, Formal analysis, Supervision, Methodology, Writing- Reviewing and Editing, Validation.

This chapter has the following changes compared with the original paper:

1. The research questions are summarized to two (instead of five).
2. Due to page limit, Section 3.6.4 were removed in the published conference paper. They were added back.
3. The discussion section was extended. Some more future work has been added.

4. Fig 13 was plotted with scale for visibility.
5. Some discussion on singletons was removed from the original submission due to the page limit. It has been put back.

The content of Chapter 4 has previously appeared as a standalone paper [49]:

- Idries Nasim, Shuai Wang, Joe Raad, Peter Bloem, and Frank van Harmelen. What does it mean when your uris are redirected? Examining identity and redirection in the LOD cloud. In Damien Graux et al., editors, *Proceedings of the 8th Workshop on Managing the Evolution and Preservation of the Data Web (MEPDaW) co-located with the 21st International Semantic Web Conference (ISWC 2022), Virtual event, October 23rd, 2022*, volume 3339 of *CEUR Workshop Proceedings*, pages 36–45. CEUR-WS.org, 2022.

The author of the thesis is the second author of this paper and the supervisor of the first author. The corresponding bachelor's thesis outlined the idea of research, which was further specified and developed by the author of the thesis and co-authors of the paper into complete research as presented in the publication mentioned above.
CRediT author statement
Idris Nasim: Conceptualization, Methodology, Investigation, Data Curation, Formal analysis, Software, Visualization, Writing - Original Draft. **Shuai Wang**: Conceptualization, Methodology, Investigation, Data Curation, Formal analysis, Software, Visualization, Writing - Review & Editing, Writing - Original Draft, Supervision, Validation. **Peter Bloem**: Conceptualization, Investigation, Methodology, Formal analysis, Supervision, Writing - Review & Editing. **Joe Raad**: Conceptualization, Methodology, Investigation, Supervision, Formal analysis, Writing - Review & Editing, Validation. **Frank van Harmelen**: Conceptualization, Formal analysis, Supervision, Methodology, Writing- Reviewing and Editing, Validation.

This chapter has the following changes compared with the original paper:

1. Some text and a footnote about data processing using RocksDB were removed in the publication. They were added back.
2. Figure 15 was slightly modified for a better presentation.
3. The second research question was made explicit in the chapter.

The content of Chapter 5 has previously appeared as a standalone paper [76]:

- Shuai Wang. On the analysis of large integrated knowledge graphs for economics, banking and finance. In Maya Ramanath and Themis Palpanas, editors, *Proceedings of the Workshops of the EDBT/ICDT 2022 Joint Conference, Edinburgh, UK, March 29, 2022*, volume 3135 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2022.

The 2022 International Workshop on Knowledge Graphs for Economics and Finance is a workshop co-located with the EDBT/ICDT joint conference 2022. The paper has also been presented at the ICT Open 2022 as a poster.

The author of the thesis is the only author of this paper, who has been involved at every stage of this paper: the design of the algorithm, data processing, writing, and presentation. The supervisors provided valuable feedback and helped to paraphrase the text.

CRediT author statement

Shuai Wang: Conceptualization, Methodology, Investigation, Data Curation, Formal analysis, Software, Visualization, Writing - Original Draft, Writing - Review & Editing, Validation.

This chapter extends the paper with the following:

1. A summary of the largest connected components was added in Section 5.3.2. More details about redundancy analysis were added.
2. Table 15 were added in Section 5.3.1 to list the entities with the largest in-degree and out-degree together with an analysis.
3. Following some suggestions from colleagues, some discussion about the interoperability was added to Section 5.4.
4. A small bug has been fixed and the statistics in Table 14.
5. The conclusion was made shorter and straight-to-the-point.
6. Redundant introduction of transitive relations has been removed.
7. The URIs were replaced by IRIs in the thesis, which better suited what the experiments were about.

Chapter 6 is based on a spotlight paper presented at the EKAW (Knowledge Engineering and Knowledge Management) conference. Some primitive work on the analysis of links between the conceptual models was explored in the master's thesis of the second author. Some examples regarding concept drift and ambiguity were initially observed and provided by the second author. The second author also contributed to the proofreading of the paper. Besides that, almost all the experiments, analysis, communication with domain experts, and the writing of the paper were done by the first author. Moreover, the master's thesis used an outdated version of Homosaurus, while this chapter uses the latest version of Homosaurus for all experiments by the time of the conference submission.

- Shuai Wang and Maria Adamidou. Examining lgbtq+-related concepts in the semantic web: Link discovery, concept drift, ambiguity, and multilingual information reuse. In Mehwish Alam et al., editors, *Knowledge Engineering and Knowledge Management*, pages 1–17, Cham, 2025. Springer Nature Switzerland.

Additionally, most of the results of this paper have been presented and discussed at the SWIB (Semantic Web in Libraries) conference [78]. The results were presented as a poster at the ODISSEI (Open Data Infrastructure for Social Science and Economic Innovations) conference [79]. By the time this PhD thesis was submitted, this chapter was being further extended to become a journal submission. The paper has roots in a project in the FAIR Expertise Hub, which was funded by a PDI-SSH grant. The project was conducted while the author was working for the FAIR Expertise Hub project. The author appreciates all the support from supervisors and colleagues.

CRediT author statement

Maria Adamidou: Conceptualization, Methodology, Investigation, Data Curation, Formal analysis, Validation. **Shuai Wang**: Conceptualization, Methodology, Investigation, Data Curation, Formal analysis, Software, Visualization, Writing - Original Draft, Writing - Review & Editing, Supervision, Validation.

Updates

This chapter has the following changes compared to the original paper. Reviews and feedback from the EKAW conference, the SWIB conference, as well as the ODISSEI conference, have been taken into account for this extended version.

1. The introduction and related work were extended.
2. The research questions and research scenarios were made clearer.
3. Table 17 was left out from the original paper due to the page limit. It was added back to show how version updates can have an influence on the analysis of concept change and how the lack of maintenance could lead to outdated links and inaccuracy in the semantic web.
4. Table 19, 20 were removed from the original paper due to the page limit. Table 21 was added to provide more data entries for comparison. The corresponding text was slightly adapted.
5. The images in Section 6.4 were slightly adapted to fit the template better.
6. QLIT was introduced with more details in Section 6.2.
7. Some text on the motivation of each research scenario was added to better explain the community needs and the challenge the developers of the conceptual model are facing.
8. Issues about bias in translation, best practices, and data preservation are briefly discussed.

C.2 Code and Datasets Published and Archived

During the PhD, the student learned the practice of open science and attempted to make resources and research results accessible. Some efforts were made to make the following datasets and source code available or publish them as a result of the thesis. Admittedly, not all resources are FAIR and well-documented.

For Chapter 1, datasets and code corresponding to the pilot study [82] have been published [90]. For easy reuse, datasets corresponding to cycle-free class subsumption relations (and that about properties) [82], are available on Zenodo as a standalone resource [75]. The Python script for Figure 2 can be found on GitHub[1] For the research in Chapter 2, we implemented our algorithm[2] in Python. These gold standard datasets are available on Zenodo[3] together with detailed criteria, analysis, and limitations [85]. In Chapter 3, we reused the ANNit tool for data annotation. The code can be found on GitHub[4]. The code and data are published on Zenodo[5] [89]. Chapter 4 has its source code on GitHub[6]. The datasets were published on Zenodo[7] [84]. For Chapter 5, the data and Python scripts are available on GitHub[8].

The datasets corresponding to Chapter 6 can be found on Zenodo[9]. Unfortunately, due to the strict license of CC-BY-NC-ND of Homosaurus and GSSO, it is not possible to publish all the data and the intermediate results. The intermediate results and annotated data about QLIT are provided. The datasets extracted from Wikidata and the corresponding intermediate results are provided. The remaining is available upon request from the Homosaurus team, the IHLIA, the GSSO team, and the QueerLit/QLIT team. The code and reproduction instructions can be found on GitHub[10].

Finally, datasets and related supplementary materials have been archived on YODA.

C.3 Disclaimer

This dissertation is the result of Shuai Wang's doctoral studies, fully funded by the NWO TOP grant as a part of the MaestroGraph project. The author has not received

1 The Python script is available on GitHub at https://github.com/shuaiwangvu/Logical_Inconsistency_LOD. It has not been published formally.
2 https://github.com/shuaiwangvu/Refining-Transitive-Relations
3 https://zenodo.org/record/4610000
4 https://github.com/shuaiwangvu/sameAs-iUNA
5 https://zenodo.org/record/7765113.
6 https://github.com/shuaiwangvu/redirection
7 https://doi.org/10.5281/zenodo.7225383
8 https://github.com/shuaiwangvu/EcoFin-integrated
9 https://doi.org/10.5281/zenodo.12684870
10 https://github.com/Multilingual-LGBTQIA-Vocabularies/Examing_LGBTQ_Concepts

any grant from the China Scholarship Council (CSC) or any Chinese organization, nor performed research on politically sensitive data. All the datasets and research results, except those corresponding to chapter 6, are publicly available. The author is not responsible for any misuse of the research results, data, and software.

BIBLIOGRAPHY

[1] Trans & Gender Diverse LCSH, 2024. https://translcsh.com/. The list was last accessed on 25th May, 2024.

[2] Vincent A. Traag et al. From louvain to leiden: guaranteeing well-connected communities. *CoRR*, 2018. http://arxiv.org/abs/1810.08473.

[3] Bradley P. Allen and Paul T. Groth. Evaluating class membership relations in knowledge graphs using large language models. In Albert Meroño-Peñuela et al., editors, *The Semantic Web: ESWC 2024 Satellite Events - Hersonissos, Crete, Greece, May 26-30, 2024, Proceedings, Part I*, volume 15344 of *Lecture Notes in Computer Science*, pages 14–24. Springer, 2024.

[4] Grigoris Antoniou and Frank Van Harmelen. *A semantic web primer*. MIT press, 2004.

[5] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. Dbpedia: A nucleus for a web of open data. In *international semantic web conference*, pages 722–735. Springer, 2007.

[6] Wouter Beek, Joe Raad, Erman Acar, and Frank van Harmelen. Metalink: A travel guide to the lod cloud. Lecture Notes in Computer Science, pages 481–496. Springer, 2020. 17th Extended Semantic Web Conference, ESWC 2020 ; Conference date: 31-05-2020 Through 04-06-2020.

[7] Wouter Beek, Joe Raad, Jan Wielemaker, and Frank van Harmelen. sameas.cc: The closure of 500m owl:sameas statements. In *The Semantic Web - 15th International Conference, ESWC 2018, Proceedings*, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), pages 65–80. Springer/Verlag, 2018. 15th International Conference on Extended Semantic Web Conference, ESWC 2018 ; Conference date: 03-06-2018 Through 07-06-2018.

[8] Wouter Beek, Laurens Rietveld, Hamid R Bazoobandi, Jan Wielemaker, and Stefan Schlobach. Lod laundromat: a uniform way of publishing other people's dirty data. In *International semantic web conference*, pages 213–228. Springer, 2014.

[9] Luigi Bellomarini, Marco Benedetti, Andrea Gentili, Rosario Laurendi, Davide Magnanimi, Antonio Muci, and Emanuel Sallinger. COVID-19 and company

knowledge graphs: Assessing golden powers and economic impact of selective lockdown via AI reasoning. *CoRR*, abs/2004.10119, 2020.

[10] Mike Bennett. The financial industry business ontology: Best practice for big data. *Journal of Banking Regulation*, 14(3):255–268, 2013.

[11] Jenny Bergenmar, Koraljka Golub, and Siska Humelsjö. Queerlit database: Making swedish lgbtqi literature easily accessible. In *DHNB 2022: The 6th Digital Humanities in the Nordic and Baltic Countries Conference 2022*, pages 433–437. CEUR-WS. org, 2022.

[12] Nikolaj Bjørner. Engineering theories with z3. In *Asian Symposium on Programming Languages and Systems*, pages 4–16. Springer, 2011.

[13] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment*, 2008(10):P10008, 2008.

[14] P. A. Bonatti, Aidan Hogan, Axel Polleres, and Luigi Sauro. Robust and scalable linked data reasoning incorporating provenance and trust annotations. *Journal of Web Semantics*, 9(2):165–201, 2011. Provenance in the Semantic Web.

[15] Donna Braquet. *Chapter 2 LGBTQ+ Terminology, Scenarios and Strategies, and Relevant Web-based Resources in the 21st Century: A Glimpse*, pages 49–61. 05 2019.

[16] Peter Pin-Shan Chen. The entity-relationship model—toward a unified view of data. *ACM transactions on database systems (TODS)*, 1(1):9–36, 1976.

[17] Beatrice Cherrier. Classifying economics: A history of the jel codes. *Journal of economic literature*, 55(2):545–79, 2017.

[18] Francesco Corcoglioniti, Marco Rospocher, Michele Mostarda, and Marco Amadori. Processing billions of RDF triples on a single machine using streaming and sorting. In Roger L. Wainwright et al., editors, *Proceedings of the 30th Annual ACM Symposium on Applied Computing, Salamanca, Spain, April 13-17, 2015*, pages 368–375. ACM, 2015.

[19] John Cuzzola, Ebrahim Bagheri, and Jelena Jovanovic. Filtering inaccurate entity co-references on the linked open data. pages 128–143, 09 2015.

[20] Thomas de Groot, Joe Raad, and Stefan Schlobach. Analysing large inconsistent knowledge graphs using anti-patterns. In Ruben Verborgh et al., editors, *The Semantic Web - 18th International Conference, ESWC 2021, Virtual Event, June 6-10, 2021, Proceedings*, volume 12731 of *Lecture Notes in Computer Science*, pages 40–56. Springer, 2021.

[21] Gerard de Melo. Not quite the same: Identity constraints for the web of linked data. In *AAAI*, 2013.

[22] K. Dentler and R. Cornet. Redundant elements in snomed ct concept definitions. In N. Peek et al., editors, *Artificial Intelligence in Medicine*, 2013, pages 186–195. Springer, 2013. 14th Conference on Artificial Intelligence in Medicine ; Conference date: 01-01-2013 Through 01-01-2013.

[23] Brian Dobreski, Karen Snow, and Heather Moulaison-Sandy. On overlap and otherness: A comparison of three vocabularies' approaches to lgbtq+ identity. *Cataloging & Classification Quarterly*, 60(6-7):490–513, 2022.

[24] Na Dong, Natthawut Kertkeidkachorn, Xin Liu, and Kiyoaki Shirai. Refining noisy knowledge graph with large language models. In Genet Asefa Gesese et al., editors, *Proceedings of the Workshop on Generative AI and Knowledge Graphs (GenAIK)*, pages 78–86, Abu Dhabi, UAE, January 2025. International Committee on Computational Linguistics.

[25] J. Fernández, Wouter Beek, Miguel A Martínez-Prieto, and Mario Arias. Lod-a-lot. In *ISWC*, pages 75–83. Springer, 2017.

[26] Javier David Fernandez Garcia, Wouter Beek, Miguel A Martínez-Prieto, and Mario Arias. LOD-a-lot: A queryable dump of the lod cloud. 2017.

[27] M. Fossati, D. Kontokostas, and J. Lehmann. Unsupervised learning of an extensive and usable taxonomy for dbpedia. In *International Conference on Semantic Systems*, pages 177–184, 2015.

[28] Christophe Guéret, Paul Groth, Claus Stadler, and Jens Lehmann. Assessing linked data mappings using network measures. In *Extended Semantic Web Conference*, pages 87–102. Springer, 2012.

[29] Harry Halpin, Patrick J. Hayes, James P. McCusker, Deborah L. McGuinness, and Henry S. Thompson. When owl:sameas isn't the same: An analysis of identity in linked data. In Peter F. Patel-Schneider et al., editors, *The Semantic Web – ISWC 2010*, pages 305–320, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.

[30] Harry Halpin and Valentina Presutti. An ontology of resources: Solving the identity crisis. volume 5554, pages 521–534, 05 2009.

[31] A. Harth, Sheila Kinsella, and S. Decker. Using naming authority to rank data and ontologies for web search. In *ISWC*, 2009.

[32] N. Heist and H. Paulheim. Entity extraction from wikipedia list pages. In *ESWC*, pages 327–342, 2020.

[33] S. Hertling and H. Paulheim. Webisalod: providing hypernymy relations extracted from the web as linked open data. In *ISWC*, pages 111–119, 2017.

[34] Rinke Hoekstra, Joost Breuker, Marcello Di Bello, and Alexander Boer. The lkif core ontology of basic legal concepts. pages 43–63, 01 2007.

[35] Aidan Hogan, Antoine Zimmermann, Jürgen Umbrich, Axel Polleres, and Stefan Decker. Scalable and distributed methods for entity matching, consolidation and disambiguation over linked data corpora. *Journal of Web Semantics*, 10:76–110, 2012. Web-Scale Semantic Information Processing.

[36] D. Hsu, X. Lan, G. Miller, and D. Baird. A comparative study of algorithm for computing strongly connected components. *15th IEEE International Conference on Dependable, Autonomic and Secure Computing DASC*, pages 431–437, 2017.

[37] Bernadette Hyland, Ghislain Atemezing, and Boris Villazón-Terrazas. Best Practices for Publishing Linked Data. Technical report, W3C Working Group, 2014. Online; accessed 19 October 2022.

[38] Daniel Ocic Ihrmark, Koraljka Golub, and Xu Tan. Subject indexing of lgbtq+ fiction in sweden and china. In *Knowledge Organization for Resilience in Times of Crisis: Challenges and Opportunities*, pages 379–384. Ergon-Verlag, 2024.

[39] Annamarie Jagose. *Queer theory: An introduction*. NYU Press, 1996.

[40] Ernesto Jiménez-Ruiz and Bernardo Cuenca Grau. Logmap: Logic-based and scalable ontology matching. In Lora Aroyo et al., editors, *The Semantic Web – ISWC 2011*, pages 273–288. Springer Berlin Heidelberg, 2011.

[41] Anna-Maja Kazarian and Shuai Wang. Evaluating Automated Machine Translation of LGBTQ+ Terms: Towards Multilingual Homosaurus, March 2024.

[42] Clair A Kronk and Judith W Dexheimer. Development of the gender, sex, and sexual orientation ontology: Evaluation and workflow. *Journal of the American Medical Informatics Association*, 27(7):1110–1115, 2020.

[43] R. Kudelić and N. Ivković. Ant inspired monte carlo algorithm for minimum feedback arc set. *Expert Systems with Applications*, 122:108–117, 2019.

[44] Kristine E. Lynch, Patrick R. Alba, Olga V. Patterson, Benjamin Viernes, Gregorio Coronado, and Scott L. DuVall. The utility of clinical notes for sexual minority health research. *American Journal of Preventive Medicine*, 59(5):755–763, 2020.

[45] Suhail Malik. The ontology of finance: Price, power, and the arkhederivative. In *Collapse Vol. VIII: casino real*, pages 629–811. Falmouth: Urbanomic, 2014.

[46] Arild Matsson and Olov Kriström. Building and serving the queerlit thesaurus as linked open data. *Digital Humanities in the Nordic and Baltic Countries Publications*, 5(1):29–39, 2023.

[47] A. Miles and Sean Bechhofer. Skos simple knowledge organization system reference. *W3C recommendation*, 2009.

[48] E. Mones, Lilla Vicsek, and Tamás Vicsek. Hierarchy measure for complex networks. *PloS one*, 7(3):e33799, 2012.

[49] Idries Nasim, Shuai Wang, Joe Raad, Peter Bloem, and Frank van Harmelen. What does it mean when your uris are redirected? Examining identity and redirection in the LOD cloud. In Damien Graux et al., editors, *Proceedings of the 8th Workshop on Managing the Evolution and Preservation of the Data Web (MEPDaW) co-located with the 21st International Semantic Web Conference (ISWC 2022), Virtual event, October 23rd, 2022*, volume 3339 of *CEUR Workshop Proceedings*, pages 36–45. CEUR-WS.org, 2022.

[50] Office of Communications and Marketing. An LGTBQ language thesaurus is translated to spanish, 2024. Accessed on May 19, 2024.

[51] Laura Papaleo, N. Pernelle, Fatiha Saïs, and C. Dumont. Logical detection of invalid sameas statements in rdf data. In *EKAW*, 2014.

[52] Heiko Paulheim. Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic web*, 8(3):489–508, 2017.

[53] Heiko Paulheim and Christian Bizer. Improving the quality of linked data using statistical distributions. *International Journal on Semantic Web and Information Systems (IJSWIS)*, 10(2):63–86, 2014.

[54] Roger Penrose. *The Emperor's New Mind: Concerning Computers, Minds, and the Laws of Physics*. Hunan Science Technology Publishing House, 2007. Translated to Chinese by Mingxian Xu and Zhongchao Wu.

[55] Russ Peterson. Library of congress subject headings for lgbt studies, 8 2023.

[56] F. F. Polizel, Sara J. Casare, and Jaime Simão Sichman. Ontobacen: A modular ontology for risk management in the brazilian financial system. In *Proceedings of the Joint Ontology Workshops*, 2015.

[57] J. Raad, W. Beek, F. van Harmelen, J. Wielemaker, N. Pernelle, and F. Saïs. Constructing and cleaning identity graphs in the LOD cloud. *Data Intelligence*, 2(3):323–352, 2020.

[58] Joe Raad. *Identity Management in Knowledge Graphs*. PhD thesis, University of Paris-Saclay, 2018. doctoral dissertation.

[59] Joe Raad, Wouter Beek, Frank Van Harmelen, Nathalie Pernelle, and Fatiha Saïs. Detecting erroneous identity links on the web using network metrics. In *International semantic web conference*, pages 391–407. Springer, 2018.

[60] Joe Raad, Nathalie Pernelle, Fatiha Saïs, Wouter Beek, and Frank van Harmelen. The sameas problem: A survey on identity management in the web of data. *CoRR*, abs/1907.10528, 2019.

[61] André Gomes Regino and Júlio Cesar dos Reis. Discovering semantically broken links in LOD datasets. In *Proceedings of the 6th Workshop on Managing the Evolution and Preservation of the Data Web (MEPDaW)*, 2020.

[62] Raymond Reiter. *Towards a Logical Reconstruction of Relational Database Theory*, pages 191–238. Springer New York, New York, NY, 1984.

[63] Dongxu Shao and Rajanikanth Annam. Translation embeddings for knowledge graph completion in consumer banking sector. In Amal El Fallah Seghrouchni and David Sarne, editors, *Artificial Intelligence. IJCAI 2019 International Workshops*, pages 5–17, Cham, 2020. Springer International Publishing.

[64] M. Simpson, V. Srinivasan, and A. Thomo. Efficient computation of feedback arc set at web-scale. *VLDB*, 10(3):133–144, 2016.

[65] Manu Sporny, Dave Longley, Markus Sabadello, Drummond Reed, Orie Steele, and Christopher Allen. Decentralized Identifiers (DIDs) v1.0. Technical report, W3C, 2022. Online; accessed 19 October 2022.

[66] J. Sun, Deepak Ajwani, Patrick K Nicholson, Alessandra Sala, and Srinivasan Parthasarathy. Breaking cycles in noisy hierarchies. In *Proceedings of the 2017 ACM on Web Science Conference*, pages 151–160, 2017.

[67] Jessica Tai. Cultural humility as a framework for anti-oppressive archival description. *Reinventing the Museum: Relevance, Inclusion, and Global Responsibilities*, page 349, 2023.

[68] The Homosaurus editorial Board. Homosaurus vocabulary site, 2024. Its documentation was last accessed on 24th May, 2024.

[69] Andre Valdestilhas, Tommaso Soru, and Axel-Cyrille Ngonga Ngomo. Cedal: Time-efficient detection of erroneous links in large-scale link repositories. 08 2017.

[70] Frank van Harmelen. Maestrograph: The meaning and structure of very large knowledge graphs.

[71] Jacco van Ossenbruggen, Michiel Hildebrand, and Victor de Boer. Interactive vocabulary alignment. In Stefan Gradmann et al., editors, *Research and Advanced Technology for Digital Libraries*, pages 296–307, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[72] Ruben Verborgh and Miel Vander Sande. The Semantic Web identity crisis: in search of the trivialities that never were. *Semantic Web Journal*, 11(1):19–27, January 2020.

[73] Denny Vrandečić and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85, 2014.

[74] Shenghui Wang, Stefan Schlobach, and Michel Klein. Concept drift and how to identify it. *Journal of Web Semantics*, 9(3):247–265, 2011. Semantic Web Dynamics Semantic Web Challenge, 2010.

[75] Shuai Wang. Cycle-free subclass subsumption relations of the lod cloud, February 2020. DOI: 10.5281/zenodo.3693802.

[76] Shuai Wang. On the analysis of large integrated knowledge graphs for economics, banking and finance. In Maya Ramanath and Themis Palpanas, editors, *Proceedings of the Workshops of the EDBT/ICDT 2022 Joint Conference, Edinburgh, UK, March 29, 2022*, volume 3135 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2022.

[77] Shuai Wang. Best practices and evaluation criteria for translating lgbtq+ terms in conceptual models, March 2025. DOI: 10.5281/zenodo.15082539.

[78] Shuai Wang and Maria Adamidou. Examining LGBTQ+-related concepts in the semantic web: Link discovery, concept drift, ambiguity, and multilingual information reuse, December 2024. The slides were presented at the Semantic Web in Libraries conference (SWIB'24). DOI: 10.5281/zenodo.14261883 .

[79] Shuai Wang and Maria Adamidou. Examining LGBTQ+-related concepts in the semantic web: Link discovery, concept drift, ambiguity, and multilingual information reuse, December 2024. The poster was presented at the ODSSEI conference (ODISSEI'24). DOI: 10.5281/zenodo.14261351 .

[80] Shuai Wang and Maria Adamidou. Examining lgbtq+-related concepts in the semantic web: Link discovery, concept drift, ambiguity, and multilingual information reuse. In Mehwish Alam et al., editors, *Knowledge Engineering and Knowledge Management*, pages 1–17, Cham, 2025. Springer Nature Switzerland.

[81] Shuai Wang and Maria Adamidou. Towards semi-automatic construction of multilingual lgbtq+ conceptual models. *Multilingual Digital Terminology Today*, 2025.

[82] Shuai Wang, Peter Bloem, Joe Raad, and Frank van Harmelen. Submassive: Resolving subclass cycles in very large knowledge graphs, 2020. Workshop on Large Scale RDF Analytics, DOI: 10.48550/arXiv.2412.15829.

[83] Shuai Wang, Angelica Maineri, Navroop K Singh, and Tobias Kuhn. Fair implementation profiles for social science. In *Preceeding of 17th International Conference on Metadata and Semantics Research. Springer*, 2023.

[84] Shuai Wang, Idries Nasim, Joe Raad, Peter Bloem, and Frank van Harmelen. Graphs of redirection: an examination of uris in identity graphs, October 2022. DOI: 10.5281/zenodo.7225383.

[85] Shuai Wang, Joe Raad, Peter Bloem, and Frank van Harmelen. Annotated (pseudo-)transitive relations of the lod cloud, March 2021. DOI: 10.5281/zenodo.4610000.

[86] Shuai Wang, Joe Raad, Peter Bloem, and Frank van Harmelen. Refining transitive and pseudo-transitive relations at web scale. In Ruben Verborgh et al., editors, *The Semantic Web - 18th International Conference, ESWC 2021, Virtual Event, June 6-10, 2021, Proceedings*, volume 12731 of *Lecture Notes in Computer Science*, pages 249–264. Springer International Publishing, 2021.

[87] Shuai Wang, Joe Raad, Peter Bloem, and Frank van Harmelen. Refining large integrated identity graphs using the Unique Name Assumption. In Catia Pesquita et al., editors, *The Semantic Web - 18th International Conference, ESWC 2023, Hersonissou, Greece, May 28 - June 1, 2023, Proceedings*. Springer Nature, 2023.

[88] Shuai Wang, Joe Raad, Peter Bloem, and Frank van Harmelen. Refining large integrated identity graphs using the unique name assumption. In *European Semantic Web Conference*, pages 55–71. Springer, 2023.

[89] Shuai Wang, Joe Raad, Peter Bloem, and Frank van Harmelen. Refining large integrated identity graphs using the unique name assumption, March 2023. DOI: 10.5281/zenodo.7765113.

[90] Shuai Wang, Frank van Harmelen, Joe Raad, and Peter Bloem. Submassive: Resolving subclass cycles in very large knowledge graphs, December 2024. DOI: 10.5281/zenodo.14535281.

[91] Brian M Watson. "there was sex but no sexuality*" critical cataloging and the classification of asexuality in lcsh. *Cataloging & Classification Quarterly*, 58(6):547–565, 2020.

[92] Mark D Wilkinson, Michel Dumontier, IJsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E Bourne, et al. The fair guiding principles for scientific data management and stewardship. *Scientific data*, 3(1):1–9, 2016.

[93] H. Öncü, Maher A.N. Agi, and Jérémy Guérin. A fast and effective heuristic for smoothing workloads on assembly lines: algorithm design and experimental analysis. *Computers & Operations Research*, 115, 2020.

BIOGRAFIE EN CURRICULUM VITAE IN HET NEDERLANDS

Shuai Wang geboren op 8 maart 1990 te Otog Banier, Ordos, Binnen-Mongolië, China. Hij is de zoon van Guilan Wang en Baoguo Wang, en de jongere broer van Limin Wang en Liqin Wang. Hij studeerde in Dongsheng Nr. 1 Middelbare School (Dongsheng No.1 Middle School), en in de Raket-klas (d.w.z. de klas voor de hoogbegaafden studenten) in Ordos Nr.1 School voor bovenbouw voortgezet onderwijs (Ordos No.1 High School). Hij behaalde een bachelordiploma in Kunstmatige Intelligentie aan de Universiteit van Manchester en een masterdiploma in Logica aan de Universiteit van Amsterdam.

Dit proefschrift is het resultaat van Shuai Wang's promotieonderzoek, volledig gefinancierd door de NWO TOP-subsidie als onderdeel van het MaestroGraph-project.

SIKS DISSERTATIEREEKS

2016
01 Syed Saiden Abbas (RUN), Recognition of Shapes by Humans and Machines
02 Michiel Christiaan Meulendijk (UU), Optimizing medication reviews through decision support: prescribing a better pill to swallow
03 Maya Sappelli (RUN), Knowledge Work in Context: User Centered Knowledge Worker Support
04 Laurens Rietveld (VUA), Publishing and Consuming Linked Data
05 Evgeny Sherkhonov (UvA), Expanded Acyclic Queries: Containment and an Application in Explaining Missing Answers
06 Michel Wilson (TUD), Robust scheduling in an uncertain environment
07 Jeroen de Man (VUA), Measuring and modeling negative emotions for virtual training
08 Matje van de Camp (TiU), A Link to the Past: Constructing Historical Social Networks from Unstructured Data
09 Archana Nottamkandath (VUA), Trusting Crowdsourced Information on Cultural Artefacts
10 George Karafotias (VUA), Parameter Control for Evolutionary Algorithms
11 Anne Schuth (UvA), Search Engines that Learn from Their Users
12 Max Knobbout (UU), Logics for Modelling and Verifying Normative Multi-Agent Systems
13 Nana Baah Gyan (VUA), The Web, Speech Technologies and Rural Development in West Africa - An ICT4D Approach
14 Ravi Khadka (UU), Revisiting Legacy Software System Modernization
15 Steffen Michels (RUN), Hybrid Probabilistic Logics - Theoretical Aspects, Algorithms and Experiments
16 Guangliang Li (UvA), Socially Intelligent Autonomous Agents that Learn from Human Reward
17 Berend Weel (VUA), Towards Embodied Evolution of Robot Organisms
18 Albert Meroño Peñuela (VUA), Refining Statistical Data on the Web
19 Julia Efremova (TU/e), Mining Social Structures from Genealogical Data
20 Daan Odijk (UvA), Context & Semantics in News & Web Search
21 Alejandro Moreno Célleri (UT), From Traditional to Interactive Playspaces: Automatic Analysis of Player Behavior in the Interactive Tag Playground
22 Grace Lewis (VUA), Software Architecture Strategies for Cyber-Foraging Systems
23 Fei Cai (UvA), Query Auto Completion in Information Retrieval
24 Brend Wanders (UT), Repurposing and Probabilistic Integration of Data; An Iterative and data model independent approach

25 Julia Kiseleva (TU/e), Using Contextual Information to Understand Searching and Browsing Behavior
26 Dilhan Thilakarathne (VUA), In or Out of Control: Exploring Computational Models to Study the Role of Human Awareness and Control in Behavioural Choices, with Applications in Aviation and Energy Management Domains
27 Wen Li (TUD), Understanding Geo-spatial Information on Social Media
28 Mingxin Zhang (TUD), Large-scale Agent-based Social Simulation - A study on epidemic prediction and control
29 Nicolas Höning (TUD), Peak reduction in decentralised electricity systems - Markets and prices for flexible planning
30 Ruud Mattheij (TiU), The Eyes Have It
31 Mohammad Khelghati (UT), Deep web content monitoring
32 Eelco Vriezekolk (UT), Assessing Telecommunication Service Availability Risks for Crisis Organisations
33 Peter Bloem (UvA), Single Sample Statistics, exercises in learning from just one example
34 Dennis Schunselaar (TU/e), Configurable Process Trees: Elicitation, Analysis, and Enactment
35 Zhaochun Ren (UvA), Monitoring Social Media: Summarization, Classification and Recommendation
36 Daphne Karreman (UT), Beyond R2D2: The design of nonverbal interaction behavior optimized for robot-specific morphologies
37 Giovanni Sileno (UvA), Aligning Law and Action - a conceptual and computational inquiry
38 Andrea Minuto (UT), Materials that Matter - Smart Materials meet Art & Interaction Design
39 Merijn Bruijnes (UT), Believable Suspect Agents; Response and Interpersonal Style Selection for an Artificial Suspect
40 Christian Detweiler (TUD), Accounting for Values in Design
41 Thomas King (TUD), Governing Governance: A Formal Framework for Analysing Institutional Design and Enactment Governance
42 Spyros Martzoukos (UvA), Combinatorial and Compositional Aspects of Bilingual Aligned Corpora
43 Saskia Koldijk (RUN), Context-Aware Support for Stress Self-Management: From Theory to Practice
44 Thibault Sellam (UvA), Automatic Assistants for Database Exploration
45 Bram van de Laar (UT), Experiencing Brain-Computer Interface Control
46 Jorge Gallego Perez (UT), Robots to Make you Happy
47 Christina Weber (UL), Real-time foresight - Preparedness for dynamic innovation networks
48 Tanja Buttler (TUD), Collecting Lessons Learned
49 Gleb Polevoy (TUD), Participation and Interaction in Projects. A Game-Theoretic Analysis
50 Yan Wang (TiU), The Bridge of Dreams: Towards a Method for Operational Performance Alignment in IT-enabled Service Supply Chains

2017 01 Jan-Jaap Oerlemans (UL), Investigating Cybercrime
02 Sjoerd Timmer (UU), Designing and Understanding Forensic Bayesian Networks using Argumentation

03 Daniël Harold Telgen (UU), Grid Manufacturing; A Cyber-Physical Approach with Autonomous Products and Reconfigurable Manufacturing Machines
04 Mrunal Gawade (CWI), Multi-core Parallelism in a Column-store
05 Mahdieh Shadi (UvA), Collaboration Behavior
06 Damir Vandic (EUR), Intelligent Information Systems for Web Product Search
07 Roel Bertens (UU), Insight in Information: from Abstract to Anomaly
08 Rob Konijn (VUA), Detecting Interesting Differences:Data Mining in Health Insurance Data using Outlier Detection and Subgroup Discovery
09 Dong Nguyen (UT), Text as Social and Cultural Data: A Computational Perspective on Variation in Text
10 Robby van Delden (UT), (Steering) Interactive Play Behavior
11 Florian Kunneman (RUN), Modelling patterns of time and emotion in Twitter #anticipointment
12 Sander Leemans (TU/e), Robust Process Mining with Guarantees
13 Gijs Huisman (UT), Social Touch Technology - Extending the reach of social touch through haptic technology
14 Shoshannah Tekofsky (TiU), You Are Who You Play You Are: Modelling Player Traits from Video Game Behavior
15 Peter Berck (RUN), Memory-Based Text Correction
16 Aleksandr Chuklin (UvA), Understanding and Modeling Users of Modern Search Engines
17 Daniel Dimov (UL), Crowdsourced Online Dispute Resolution
18 Ridho Reinanda (UvA), Entity Associations for Search
19 Jeroen Vuurens (UT), Proximity of Terms, Texts and Semantic Vectors in Information Retrieval
20 Mohammadbashir Sedighi (TUD), Fostering Engagement in Knowledge Sharing: The Role of Perceived Benefits, Costs and Visibility
21 Jeroen Linssen (UT), Meta Matters in Interactive Storytelling and Serious Gaming (A Play on Worlds)
22 Sara Magliacane (VUA), Logics for causal inference under uncertainty
23 David Graus (UvA), Entities of Interest — Discovery in Digital Traces
24 Chang Wang (TUD), Use of Affordances for Efficient Robot Learning
25 Veruska Zamborlini (VUA), Knowledge Representation for Clinical Guidelines, with applications to Multimorbidity Analysis and Literature Search
26 Merel Jung (UT), Socially intelligent robots that understand and respond to human touch
27 Michiel Joosse (UT), Investigating Positioning and Gaze Behaviors of Social Robots: People's Preferences, Perceptions and Behaviors
28 John Klein (VUA), Architecture Practices for Complex Contexts
29 Adel Alhuraibi (TiU), From IT-BusinessStrategic Alignment to Performance: A Moderated Mediation Model of Social Innovation, and Enterprise Governance of IT"
30 Wilma Latuny (TiU), The Power of Facial Expressions
31 Ben Ruijl (UL), Advances in computational methods for QFT calculations
32 Thaer Samar (RUN), Access to and Retrievability of Content in Web Archives

33 Brigit van Loggem (OU), Towards a Design Rationale for Software Documentation: A Model of Computer-Mediated Activity
34 Maren Scheffel (OU), The Evaluation Framework for Learning Analytics
35 Martine de Vos (VUA), Interpreting natural science spreadsheets
36 Yuanhao Guo (UL), Shape Analysis for Phenotype Characterisation from High-throughput Imaging
37 Alejandro Montes Garcia (TU/e), WiBAF: A Within Browser Adaptation Framework that Enables Control over Privacy
38 Alex Kayal (TUD), Normative Social Applications
39 Sara Ahmadi (RUN), Exploiting properties of the human auditory system and compressive sensing methods to increase noise robustness in ASR
40 Altaf Hussain Abro (VUA), Steer your Mind: Computational Exploration of Human Control in Relation to Emotions, Desires and Social Support For applications in human-aware support systems
41 Adnan Manzoor (VUA), Minding a Healthy Lifestyle: An Exploration of Mental Processes and a Smart Environment to Provide Support for a Healthy Lifestyle
42 Elena Sokolova (RUN), Causal discovery from mixed and missing data with applications on ADHD datasets
43 Maaike de Boer (RUN), Semantic Mapping in Video Retrieval
44 Garm Lucassen (UU), Understanding User Stories - Computational Linguistics in Agile Requirements Engineering
45 Bas Testerink (UU), Decentralized Runtime Norm Enforcement
46 Jan Schneider (OU), Sensor-based Learning Support
47 Jie Yang (TUD), Crowd Knowledge Creation Acceleration
48 Angel Suarez (OU), Collaborative inquiry-based learning

2018 01 Han van der Aa (VUA), Comparing and Aligning Process Representations
02 Felix Mannhardt (TU/e), Multi-perspective Process Mining
03 Steven Bosems (UT), Causal Models For Well-Being: Knowledge Modeling, Model-Driven Development of Context-Aware Applications, and Behavior Prediction
04 Jordan Janeiro (TUD), Flexible Coordination Support for Diagnosis Teams in Data-Centric Engineering Tasks
05 Hugo Huurdeman (UvA), Supporting the Complex Dynamics of the Information Seeking Process
06 Dan Ionita (UT), Model-Driven Information Security Risk Assessment of Socio-Technical Systems
07 Jieting Luo (UU), A formal account of opportunism in multi-agent systems
08 Rick Smetsers (RUN), Advances in Model Learning for Software Systems
09 Xu Xie (TUD), Data Assimilation in Discrete Event Simulations
10 Julienka Mollee (VUA), Moving forward: supporting physical activity behavior change through intelligent technology
11 Mahdi Sargolzaei (UvA), Enabling Framework for Service-oriented Collaborative Networks
12 Xixi Lu (TU/e), Using behavioral context in process mining
13 Seyed Amin Tabatabaei (VUA), Computing a Sustainable Future

14 Bart Joosten (TiU), Detecting Social Signals with Spatiotemporal Gabor Filters
15 Naser Davarzani (UM), Biomarker discovery in heart failure
16 Jaebok Kim (UT), Automatic recognition of engagement and emotion in a group of children
17 Jianpeng Zhang (TU/e), On Graph Sample Clustering
18 Henriette Nakad (UL), De Notaris en Private Rechtspraak
19 Minh Duc Pham (VUA), Emergent relational schemas for RDF
20 Manxia Liu (RUN), Time and Bayesian Networks
21 Aad Slootmaker (OU), EMERGO: a generic platform for authoring and playing scenario-based serious games
22 Eric Fernandes de Mello Araújo (VUA), Contagious: Modeling the Spread of Behaviours, Perceptions and Emotions in Social Networks
23 Kim Schouten (EUR), Semantics-driven Aspect-Based Sentiment Analysis
24 Jered Vroon (UT), Responsive Social Positioning Behaviour for Semi-Autonomous Telepresence Robots
25 Riste Gligorov (VUA), Serious Games in Audio-Visual Collections
26 Roelof Anne Jelle de Vries (UT),Theory-Based and Tailor-Made: Motivational Messages for Behavior Change Technology
27 Maikel Leemans (TU/e), Hierarchical Process Mining for Scalable Software Analysis
28 Christian Willemse (UT), Social Touch Technologies: How they feel and how they make you feel
29 Yu Gu (TiU), Emotion Recognition from Mandarin Speech
30 Wouter Beek (VUA), The "K" in "semantic web" stands for "knowledge": scaling semantics to the web

2019 01 Rob van Eijk (UL),Web privacy measurement in real-time bidding systems. A graph-based approach to RTB system classification
02 Emmanuelle Beauxis Aussalet (CWI, UU), Statistics and Visualizations for Assessing Class Size Uncertainty
03 Eduardo Gonzalez Lopez de Murillas (TU/e), Process Mining on Databases: Extracting Event Data from Real Life Data Sources
04 Ridho Rahmadi (RUN), Finding stable causal structures from clinical data
05 Sebastiaan van Zelst (TU/e), Process Mining with Streaming Data
06 Chris Dijkshoorn (VUA), Nichesourcing for Improving Access to Linked Cultural Heritage Datasets
07 Soude Fazeli (TUD), Recommender Systems in Social Learning Platforms
08 Frits de Nijs (TUD), Resource-constrained Multi-agent Markov Decision Processes
09 Fahimeh Alizadeh Moghaddam (UvA), Self-adaptation for energy efficiency in software systems
10 Qing Chuan Ye (EUR), Multi-objective Optimization Methods for Allocation and Prediction
11 Yue Zhao (TUD), Learning Analytics Technology to Understand Learner Behavioral Engagement in MOOCs
12 Jacqueline Heinerman (VUA), Better Together

13 Guanliang Chen (TUD), MOOC Analytics: Learner Modeling and Content Generation
14 Daniel Davis (TUD), Large-Scale Learning Analytics: Modeling Learner Behavior & Improving Learning Outcomes in Massive Open Online Courses
15 Erwin Walraven (TUD), Planning under Uncertainty in Constrained and Partially Observable Environments
16 Guangming Li (TU/e), Process Mining based on Object-Centric Behavioral Constraint (OCBC) Models
17 Ali Hurriyetoglu (RUN),Extracting actionable information from microtexts
18 Gerard Wagenaar (UU), Artefacts in Agile Team Communication
19 Vincent Koeman (TUD), Tools for Developing Cognitive Agents
20 Chide Groenouwe (UU), Fostering technically augmented human collective intelligence
21 Cong Liu (TU/e), Software Data Analytics: Architectural Model Discovery and Design Pattern Detection
22 Martin van den Berg (VUA),Improving IT Decisions with Enterprise Architecture
23 Qin Liu (TUD), Intelligent Control Systems: Learning, Interpreting, Verification
24 Anca Dumitrache (VUA), Truth in Disagreement - Crowdsourcing Labeled Data for Natural Language Processing
25 Emiel van Miltenburg (VUA), Pragmatic factors in (automatic) image description
26 Prince Singh (UT), An Integration Platform for Synchromodal Transport
27 Alessandra Antonaci (OU), The Gamification Design Process applied to (Massive) Open Online Courses
28 Esther Kuindersma (UL), Cleared for take-off: Game-based learning to prepare airline pilots for critical situations
29 Daniel Formolo (VUA), Using virtual agents for simulation and training of social skills in safety-critical circumstances
30 Vahid Yazdanpanah (UT), Multiagent Industrial Symbiosis Systems
31 Milan Jelisavcic (VUA), Alive and Kicking: Baby Steps in Robotics
32 Chiara Sironi (UM), Monte-Carlo Tree Search for Artificial General Intelligence in Games
33 Anil Yaman (TU/e), Evolution of Biologically Inspired Learning in Artificial Neural Networks
34 Negar Ahmadi (TU/e), EEG Microstate and Functional Brain Network Features for Classification of Epilepsy and PNES
35 Lisa Facey-Shaw (OU), Gamification with digital badges in learning programming
36 Kevin Ackermans (OU), Designing Video-Enhanced Rubrics to Master Complex Skills
37 Jian Fang (TUD), Database Acceleration on FPGAs
38 Akos Kadar (OU), Learning visually grounded and multilingual representations

2020 01 Armon Toubman (UL), Calculated Moves: Generating Air Combat Behaviour

02 Marcos de Paula Bueno (UL), Unraveling Temporal Processes using Probabilistic Graphical Models
03 Mostafa Deghani (UvA), Learning with Imperfect Supervision for Language Understanding
04 Maarten van Gompel (RUN), Context as Linguistic Bridges
05 Yulong Pei (TU/e), On local and global structure mining
06 Preethu Rose Anish (UT), Stimulation Architectural Thinking during Requirements Elicitation - An Approach and Tool Support
07 Wim van der Vegt (OU), Towards a software architecture for reusable game components
08 Ali Mirsoleimani (UL),Structured Parallel Programming for Monte Carlo Tree Search
09 Myriam Traub (UU), Measuring Tool Bias and Improving Data Quality for Digital Humanities Research
10 Alifah Syamsiyah (TU/e), In-database Preprocessing for Process Mining
11 Sepideh Mesbah (TUD), Semantic-Enhanced Training Data AugmentationMethods for Long-Tail Entity Recognition Models
12 Ward van Breda (VUA), Predictive Modeling in E-Mental Health: Exploring Applicability in Personalised Depression Treatment
13 Marco Virgolin (CWI), Design and Application of Gene-pool Optimal Mixing Evolutionary Algorithms for Genetic Programming
14 Mark Raasveldt (CWI/UL), Integrating Analytics with Relational Databases
15 Konstantinos Georgiadis (OU), Smart CAT: Machine Learning for Configurable Assessments in Serious Games
16 Ilona Wilmont (RUN), Cognitive Aspects of Conceptual Modelling
17 Daniele Di Mitri (OU), The Multimodal Tutor: Adaptive Feedback from Multimodal Experiences
18 Georgios Methenitis (TUD), Agent Interactions & Mechanisms in Markets with Uncertainties: Electricity Markets in Renewable Energy Systems
19 Guido van Capelleveen (UT), Industrial Symbiosis Recommender Systems
20 Albert Hankel (VUA), Embedding Green ICT Maturity in Organisations
21 Karine da Silva Miras de Araujo (VUA), Where is the robot?: Life as it could be
22 Maryam Masoud Khamis (RUN), Understanding complex systems implementation through a modeling approach: the case of e-government in Zanzibar
23 Rianne Conijn (UT), The Keys to Writing: A writing analytics approach to studying writing processes using keystroke logging
24 Lenin da Nóbrega Medeiros (VUA/RUN), How are you feeling, human? Towards emotionally supportive chatbots
25 Xin Du (TU/e), The Uncertainty in Exceptional Model Mining
26 Krzysztof Leszek Sadowski (UU), GAMBIT: Genetic Algorithm for Model-Based mixed-Integer opTimization
27 Ekaterina Muravyeva (TUD), Personal data and informed consent in an educational context
28 Bibeg Limbu (TUD), Multimodal interaction for deliberate practice: Training complex skills with augmented reality

	29	Ioan Gabriel Bucur (RUN), Being Bayesian about Causal Inference
	30	Bob Zadok Blok (UL), Creatief, Creatiever, Creatiefst
	31	Gongjin Lan (VUA), Learning better – From Baby to Better
	32	Jason Rhuggenaath (TU/e), Revenue management in online markets: pricing and online advertising
	33	Rick Gilsing (TU/e), Supporting service-dominant business model evaluation in the context of business model innovation
	34	Anna Bon (UM), Intervention or Collaboration? Redesigning Information and Communication Technologies for Development
	35	Siamak Farshidi (UU), Multi-Criteria Decision-Making in Software Production
2021	01	Francisco Xavier Dos Santos Fonseca (TUD),Location-based Games for Social Interaction in Public Space
	02	Rijk Mercuur (TUD), Simulating Human Routines: Integrating Social Practice Theory in Agent-Based Models
	03	Seyyed Hadi Hashemi (UvA), Modeling Users Interacting with Smart Devices
	04	Ioana Jivet (OU), The Dashboard That Loved Me: Designing adaptive learning analytics for self-regulated learning
	05	Davide Dell'Anna (UU), Data-Driven Supervision of Autonomous Systems
	06	Daniel Davison (UT), "Hey robot, what do you think?" How children learn with a social robot
	07	Armel Lefebvre (UU), Research data management for open science
	08	Nardie Fanchamps (OU), The Influence of Sense-Reason-Act Programming on Computational Thinking
	09	Cristina Zaga (UT), The Design of Robothings. Non-Anthropomorphic and Non-Verbal Robots to Promote Children's Collaboration Through Play
	10	Quinten Meertens (UvA), Misclassification Bias in Statistical Learning
	11	Anne van Rossum (UL), Nonparametric Bayesian Methods in Robotic Vision
	12	Lei Pi (UL), External Knowledge Absorption in Chinese SMEs
	13	Bob R. Schadenberg (UT), Robots for Autistic Children: Understanding and Facilitating Predictability for Engagement in Learning
	14	Negin Samaeemofrad (UL), Business Incubators: The Impact of Their Support
	15	Onat Ege Adali (TU/e), Transformation of Value Propositions into Resource Re-Configurations through the Business Services Paradigm
	16	Esam A. H. Ghaleb (UM), Bimodal emotion recognition from audio-visual cues
	17	Dario Dotti (UM), Human Behavior Understanding from motion and bodily cues using deep neural networks
	18	Remi Wieten (UU), Bridging the Gap Between Informal Sense-Making Tools and Formal Systems - Facilitating the Construction of Bayesian Networks and Argumentation Frameworks
	19	Roberto Verdecchia (VUA), Architectural Technical Debt: Identification and Management

20 Masoud Mansoury (TU/e), Understanding and Mitigating Multi-Sided Exposure Bias in Recommender Systems
21 Pedro Thiago Timbó Holanda (CWI), Progressive Indexes
22 Sihang Qiu (TUD), Conversational Crowdsourcing
23 Hugo Manuel Proença (UL), Robust rules for prediction and description
24 Kaijie Zhu (TU/e), On Efficient Temporal Subgraph Query Processing
25 Eoin Martino Grua (VUA), The Future of E-Health is Mobile: Combining AI and Self-Adaptation to Create Adaptive E-Health Mobile Applications
26 Benno Kruit (CWI/VUA), Reading the Grid: Extending Knowledge Bases from Human-readable Tables
27 Jelte van Waterschoot (UT), Personalized and Personal Conversations: Designing Agents Who Want to Connect With You
28 Christoph Selig (UL), Understanding the Heterogeneity of Corporate Entrepreneurship Programs

---

2022 01 Judith van Stegeren (UT), Flavor text generation for role-playing video games
02 Paulo da Costa (TU/e), Data-driven Prognostics and Logistics Optimisation: A Deep Learning Journey
03 Ali el Hassouni (VUA), A Model A Day Keeps The Doctor Away: Reinforcement Learning For Personalized Healthcare
04 Ünal Aksu (UU), A Cross-Organizational Process Mining Framework
05 Shiwei Liu (TU/e), Sparse Neural Network Training with In-Time Over-Parameterization
06 Reza Refaei Afshar (TU/e), Machine Learning for Ad Publishers in Real Time Bidding
07 Sambit Praharaj (OU), Measuring the Unmeasurable? Towards Automatic Co-located Collaboration Analytics
08 Maikel L. van Eck (TU/e), Process Mining for Smart Product Design
09 Oana Andreea Inel (VUA), Understanding Events: A Diversity-driven Human-Machine Approach
10 Felipe Moraes Gomes (TUD), Examining the Effectiveness of Collaborative Search Engines
11 Mirjam de Haas (UT), Staying engaged in child-robot interaction, a quantitative approach to studying preschoolers' engagement with robots and tasks during second-language tutoring
12 Guanyi Chen (UU), Computational Generation of Chinese Noun Phrases
13 Xander Wilcke (VUA), Machine Learning on Multimodal Knowledge Graphs: Opportunities, Challenges, and Methods for Learning on Real-World Heterogeneous and Spatially-Oriented Knowledge
14 Michiel Overeem (UU), Evolution of Low-Code Platforms
15 Jelmer Jan Koorn (UU), Work in Process: Unearthing Meaning using Process Mining
16 Pieter Gijsbers (TU/e), Systems for AutoML Research
17 Laura van der Lubbe (VUA), Empowering vulnerable people with serious games and gamification
18 Paris Mavromoustakos Blom (TiU), Player Affect Modelling and Video Game Personalisation
19 Bilge Yigit Ozkan (UU), Cybersecurity Maturity Assessment and Standardisation

20 Fakhra Jabeen (VUA), Dark Side of the Digital Media - Computational Analysis of Negative Human Behaviors on Social Media
21 Seethu Mariyam Christopher (UM), Intelligent Toys for Physical and Cognitive Assessments
22 Alexandra Sierra Rativa (TiU), Virtual Character Design and its potential to foster Empathy, Immersion, and Collaboration Skills in Video Games and Virtual Reality Simulations
23 Ilir Kola (TUD), Enabling Social Situation Awareness in Support Agents
24 Samaneh Heidari (UU), Agents with Social Norms and Values - A framework for agent based social simulations with social norms and personal values
25 Anna L.D. Latour (UL), Optimal decision-making under constraints and uncertainty
26 Anne Dirkson (UL), Knowledge Discovery from Patient Forums: Gaining novel medical insights from patient experiences
27 Christos Athanasiadis (UM), Emotion-aware cross-modal domain adaptation in video sequences
28 Onuralp Ulusoy (UU), Privacy in Collaborative Systems
29 Jan Kolkmeier (UT), From Head Transform to Mind Transplant: Social Interactions in Mixed Reality
30 Dean De Leo (CWI), Analysis of Dynamic Graphs on Sparse Arrays
31 Konstantinos Traganos (TU/e), Tackling Complexity in Smart Manufacturing with Advanced Manufacturing Process Management
32 Cezara Pastrav (UU), Social simulation for socio-ecological systems
33 Brinn Hekkelman (CWI/TUD), Fair Mechanisms for Smart Grid Congestion Management
34 Nimat Ullah (VUA), Mind Your Behaviour: Computational Modelling of Emotion & Desire Regulation for Behaviour Change
35 Mike E.U. Ligthart (VUA), Shaping the Child-Robot Relationship: Interaction Design Patterns for a Sustainable Interaction

---

2023 01 Bojan Simoski (VUA), Untangling the Puzzle of Digital Health Interventions
02 Mariana Rachel Dias da Silva (TiU), Grounded or in flight? What our bodies can tell us about the whereabouts of our thoughts
03 Shabnam Najafian (TUD), User Modeling for Privacy-preserving Explanations in Group Recommendations
04 Gineke Wiggers (UL), The Relevance of Impact: bibliometric-enhanced legal information retrieval
05 Anton Bouter (CWI), Optimal Mixing Evolutionary Algorithms for Large-Scale Real-Valued Optimization, Including Real-World Medical Applications
06 António Pereira Barata (UL), Reliable and Fair Machine Learning for Risk Assessment
07 Tianjin Huang (TU/e), The Roles of Adversarial Examples on Trustworthiness of Deep Learning
08 Lu Yin (TU/e), Knowledge Elicitation using Psychometric Learning
09 Xu Wang (VUA), Scientific Dataset Recommendation with Semantic Techniques

10 Dennis J.N.J. Soemers (UM), Learning State-Action Features for General Game Playing
11 Fawad Taj (VUA), Towards Motivating Machines: Computational Modeling of the Mechanism of Actions for Effective Digital Health Behavior Change Applications
12 Tessel Bogaard (VUA), Using Metadata to Understand Search Behavior in Digital Libraries
13 Injy Sarhan (UU), Open Information Extraction for Knowledge Representation
14 Selma Čaušević (TUD), Energy resilience through self-organization
15 Alvaro Henrique Chaim Correia (TU/e), Insights on Learning Tractable Probabilistic Graphical Models
16 Peter Blomsma (TiU), Building Embodied Conversational Agents: Observations on human nonverbal behaviour as a resource for the development of artificial characters
17 Meike Nauta (UT), Explainable AI and Interpretable Computer Vision – From Oversight to Insight
18 Gustavo Penha (TUD), Designing and Diagnosing Models for Conversational Search and Recommendation
19 George Aalbers (TiU), Digital Traces of the Mind: Using Smartphones to Capture Signals of Well-Being in Individuals
20 Arkadiy Dushatskiy (TUD), Expensive Optimization with Model-Based Evolutionary Algorithms applied to Medical Image Segmentation using Deep Learning
21 Gerrit Jan de Bruin (UL), Network Analysis Methods for Smart Inspection in the Transport Domain
22 Alireza Shojaifar (UU), Volitional Cybersecurity
23 Theo Theunissen (UU), Documentation in Continuous Software Development
24 Agathe Balayn (TUD), Practices Towards Hazardous Failure Diagnosis in Machine Learning
25 Jurian Baas (UU), Entity Resolution on Historical Knowledge Graphs
26 Loek Tonnaer (TU/e), Linearly Symmetry-Based Disentangled Representations and their Out-of-Distribution Behaviour
27 Ghada Sokar (TU/e), Learning Continually Under Changing Data Distributions
28 Floris den Hengst (VUA), Learning to Behave: Reinforcement Learning in Human Contexts
29 Tim Draws (TUD), Understanding Viewpoint Biases in Web Search Results

---

2024 01 Daphne Miedema (TU/e), On Learning SQL: Disentangling concepts in data systems education
02 Emile van Krieken (VUA), Optimisation in Neurosymbolic Learning Systems
03 Feri Wijayanto (RUN), Automated Model Selection for Rasch and Mediation Analysis
04 Mike Huisman (UL), Understanding Deep Meta-Learning
05 Yiyong Gou (UM), Aerial Robotic Operations: Multi-environment Cooperative Inspection & Construction Crack Autonomous Repair

06 Azqa Nadeem (TUD), Understanding Adversary Behavior via XAI: Leveraging Sequence Clustering to Extract Threat Intelligence
07 Parisa Shayan (TiU), Modeling User Behavior in Learning Management Systems
08 Xin Zhou (UvA), From Empowering to Motivating: Enhancing Policy Enforcement through Process Design and Incentive Implementation
09 Giso Dal (UT), Probabilistic Inference Using Partitioned Bayesian Networks
10 Cristina-Iulia Bucur (VUA), Linkflows: Towards Genuine Semantic Publishing in Science
11 withdrawn
12 Peide Zhu (TUD), Towards Robust Automatic Question Generation For Learning
13 Enrico Liscio (TUD), Context-Specific Value Inference via Hybrid Intelligence
14 Larissa Capobianco Shimomura (TU/e), On Graph Generating Dependencies and their Applications in Data Profiling
15 Ting Liu (VUA), A Gut Feeling: Biomedical Knowledge Graphs for Interrelating the Gut Microbiome and Mental Health
16 Arthur Barbosa Câmara (TUD), Designing Search-as-Learning Systems
17 Razieh Alidoosti (VUA), Ethics-aware Software Architecture Design
18 Laurens Stoop (UU), Data Driven Understanding of Energy-Meteorological Variability and its Impact on Energy System Operations
19 Azadeh Mozafari Mehr (TU/e), Multi-perspective Conformance Checking: Identifying and Understanding Patterns of Anomalous Behavior
20 Ritsart Anne Plantenga (UL), Omgang met Regels
21 Federica Vinella (UU), Crowdsourcing User-Centered Teams
22 Zeynep Ozturk Yurt (TU/e), Beyond Routine: Extending BPM for Knowledge-Intensive Processes with Controllable Dynamic Contexts
23 Jie Luo (VUA), Lamarck' s Revenge: Inheritance of Learned Traits Improves Robot Evolution
24 Nirmal Roy (TUD), Exploring the effects of interactive interfaces on user search behaviour
25 Alisa Rieger (TUD), Striving for Responsible Opinion Formation in Web Search on Debated Topics
26 Tim Gubner (CWI), Adaptively Generating Heterogeneous Execution Strategies using the VOILA Framework
27 Lincen Yang (UL), Information-theoretic Partition-based Models for Interpretable Machine Learning
28 Leon Helwerda (UL), Grip on Software: Understanding development progress of Scrum sprints and backlogs
29 David Wilson Romero Guzman (VUA), The Good, the Efficient and the Inductive Biases: Exploring Efficiency in Deep Learning Through the Use of Inductive Biases
30 Vijanti Ramautar (UU), Model-Driven Sustainability Accounting
31 Ziyu Li (TUD), On the Utility of Metadata to Optimize Machine Learning Workflows
32 Vinicius Stein Dani (UU), The Alpha and Omega of Process Mining

33 Siddharth Mehrotra (TUD), Designing for Appropriate Trust in Human-AI interaction
34 Robert Deckers (VUA), From Smallest Software Particle to System Specification - MuDForM: Multi-Domain Formalization Method
35 Sicui Zhang (TU/e), Methods of Detecting Clinical Deviations with Process Mining: a fuzzy set approach
36 Thomas Mulder (TU/e), Optimization of Recursive Queries on Graphs
37 James Graham Nevin (UvA), The Ramifications of Data Handling for Computational Models
38 Christos Koutras (TUD), Tabular Schema Matching for Modern Settings
39 Paola Lara Machado (TU/e), The Nexus between Business Models and Operating Models: From Conceptual Understanding to Actionable Guidance
40 Montijn van de Ven (TU/e), Guiding the Definition of Key Performance Indicators for Business Models
41 Georgios Siachamis (TUD), Adaptivity for Streaming Dataflow Engines
42 Emmeke Veltmeijer (VUA), Small Groups, Big Insights: Understanding the Crowd through Expressive Subgroup Analysis
43 Cedric Waterschoot (KNAW Meertens Instituut), The Constructive Conundrum: Computational Approaches to Facilitate Constructive Commenting on Online News Platforms
44 Marcel Schmitz (OU), Towards learning analytics-supported learning design
45 Sara Salimzadeh (TUD), Living in the Age of AI: Understanding Contextual Factors that Shape Human-AI Decision-Making
46 Georgios Stathis (Leiden University), Preventing Disputes: Preventive Logic, Law & Technology
47 Daniel Daza (VUA), Exploiting Subgraphs and Attributes for Representation Learning on Knowledge Graphs
48 Ioannis Petros Samiotis (TUD), Crowd-Assisted Annotation of Classical Music Compositions

---

2025 01 Max van Haastrecht (UL), Transdisciplinary Perspectives on Validity: Bridging the Gap Between Design and Implementation for Technology-Enhanced Learning Systems
02 Jurgen van den Hoogen (JADS), Time Series Analysis Using Convolutional Neural Networks
03 Andra-Denis Ionescu (TUD), Feature Discovery for Data-Centric AI
04 Rianne Schouten (TU/e), Exceptional Model Mining for Hierarchical Data
05 Nele Albers (TUD), Psychology-Informed Reinforcement Learning for Situated Virtual Coaching in Smoking Cessation
06 Daniël Vos (TUD), Decision Tree Learning: Algorithms for Robust Prediction and Policy Optimization
07 Ricky Maulana Fajri (TU/e), Towards Safer Active Learning: Dealing with Unwanted Biases, Graph-Structured Data, Adversary, and Data Imbalance
08 Stefan Bloemheuvel (TiU), Spatio-Temporal Analysis Through Graphs: Predictive Modeling and Graph Construction

09 Fadime Kaya (VUA), Decentralized Governance Design - A Model-Based Approach
10 Zhao Yang (UL), Enhancing Autonomy and Efficiency in Goal-Conditioned Reinforcement Learning
11 Shahin Sharifi Noorian (TUD), From Recognition to Understanding: Enriching Visual Models Through Multi-Modal Semantic Integration
12 Lijun Lyu (TUD), Interpretability in Neural Information Retrieval
13 Fuda van Diggelen (VUA), Robots Need Some Education: on the complexity of learning in evolutionary robotics
14 Gennaro Gala (TU/e), Probabilistic Generative Modeling with Latent Variable Hierarchies
15 Michiel van der Meer (UL), Opinion Diversity through Hybrid Intelligence
16 Monika Grewal (TU Delft), Deep Learning for Landmark Detection, Segmentation, and Multi-Objective Deformable Registration in Medical Imaging
17 Matteo De Carlo (VUA), Real Robot Reproduction: Towards Evolving Robotic Ecosystems
18 Anouk Neerincx (UU), Robots That Care: How Social Robots Can Boost Children's Mental Wellbeing
19 Fang Hou (UU), Trust in Software Ecosystems
20 Alexander Melchior (UU), Modelling for Policy is More Than Policy Modelling (The Useful Application of Agent-Based Modelling in Complex Policy Processes)
21 Mandani Ntekouli (UM), Bridging Individual and Group Perspectives in Psychopathology: Computational Modeling Approaches using Ecological Momentary Assessment Data
22 Hilde Weerts (TU/e), Decoding Algorithmic Fairness: Towards Interdisciplinary Understanding of Fairness and Discrimination in Algorithmic Decision-Making
23 Roderick van der Weerdt (VUA), IoT Measurement Knowledge Graphs: Constructing, Working and Learning with IoT Measurement Data as a Knowledge Graph
24 Zhong Li (UL), Trustworthy Anomaly Detection for Smart Manufacturing
25 Kyana van Eijndhoven (TiU), A Breakdown of Breakdowns: Multi-Level Team Coordination Dynamics under Stressful Conditions
26 Tom Pepels (UM), Monte-Carlo Tree Search is Work in Progress
27 Danil Provodin (JADS, TU/e), Sequential Decision Making Under Complex Feedback
28 Jinke He (TU Delft), Exploring Learned Abstract Models for Efficient Planning and Learning
29 Erik van Haeringen (VUA), Mixed Feelings: Simulating Emotion Contagion in Groups
30 Myrthe Reuver (VUA), A Puzzle of Perspectives: Interdisciplinary Language Technology for Responsible News Recommendation
31 Gebrekirstos Gebreselassie Gebremeskel (RUN), Spotlight on Recommender Systems: Contributions to Selected Components in the Recommendation Pipeline

32 Ryan Brate (UU), Words Matter: A Computational Toolkit for Charged Terms
33 Merle Reimann (VUA), Speaking the Same Language: Spoken Capability Communication in Human-Agent and Human-Robot Interaction
34 Eduard C. Groen (UU), Crowd-Based Requirements Engineering
35 Urja Khurana (VUA), From Concept To Impact: Toward More Robust Language Model Deployment
36 Anna Maria Wegmann (UU), Say the Same but Differently: Computational Approaches to Stylistic Variation and Paraphrasing
37 Chris Kamphuis (RUN), Exploring Relations and Graphs for Information Retrieval

This list will be updated before submission for printing in the final edition.

List of Figures

List of Tables

COLOPHON

This document uses the standard `scrbook` in LaTeXfor typesetting.[11] The typesetting script also used a design modified from Gonzalo Medina's code[12]. Overleaf was used for editing and compiling.

Final version as of July 22, 2025.

11 https://ctan.org/pkg/scrbook?lang=en
12 https://tex.stackexchange.com/a/86310