Automatische betrouwbaarheidsinschatting van webinhoud mogelijk maken op basis van herkomstinformatie

Enabling Automatic Provenance-Based Trust Assessment of Web Content

Tom De Nies

Promotoren: prof. dr. ir. R. Van de Walle, prof. dr. ing. E. Mannens
Proefschrift ingediend tot het behalen van de graad van
Doctor in de Ingenieurswetenschappen: Computerwetenschappen

Vakgroep Elektronica en Informatiesystemen
Voorzitter: prof. dr. ir. R. Van de Walle
Faculteit Ingenieurswetenschappen en Architectuur
Academiejaar 2015 - 2016

UNIVERSITEIT GENT

ISBN 978-90-8578-898-0
NUR 983, 988
Wettelijk depot: D/2016/10.500/30

For Sabine Hebbelinck, my mother
1961 - 2014

I don't believe in the sort of eureka moment idea. I think it's a myth. I'm very suspicious that actually Archimedes had been thinking about that problem for a long time. And it wasn't that suddenly it came to him.

Tim Berners-Lee

Preface

The above quote from Tim Berners-Lee – inventor of the Web – applies especially to this thesis. I did not come up with all this work in one *"eureka!"* moment. It was a process, that ended up taking five years. These last five years included some of the best moments of my life, and some of the worst. Good or bad, those moments are what eventually brought me here; I think it is only fitting to say that they make up the *provenance* of this dissertation. Here, I want to thank all of the people who are part of that provenance.

First and foremost, I want to thank my whole family for all their love and support. Especially my father Erwin, my brother Sam, and my mother Sabine – who was taken from us way too soon. Together with my grandparents, they have always made sure that my brother and I got the opportunity to achieve anything we wanted, and have always put our well-being in front of their own. For my grandparents, I'll gladly switch to Dutch for a moment, to thank them. *Pepe en meter, voor jullie schakel ik graag even over naar het Nederlands. Bedankt voor alles wat jullie voor mij en jullie andere kleinkinderen doen, en om steeds zo hard in mij te geloven.*

I wrote this dissertation at the office, at home, on airplanes, on trains, in hotel rooms, at conferences, in coffee houses, etc. Quite literally, my research took me around the world, and I've got the frequent flyer card to prove it. It's this flexibility of my job that I love most, and it wouldn't be possible without the infinite support of my supervisors Erik and Rik. Erik, thank you for unconditionally supporting all your PhD students, and for your endless enthusiasm about our work. Under your wing, I've seen the number of colleagues working on Semantic Web triple and almost quadruple, and our group became a major player in the field. I'm proud I could be part of that. Rik, thank you for inspiring me even during my master student years. When you initiated a class trip to the Flemish broadcaster in my 3rd bachelor year, it showed me how easily doors open for a computer science engineer. I remember thinking: *"one day, I want to work together with VRT"*, and since I joined MMLab, I've worked with them on several projects.

A big thanks also goes to my colleagues and ex-colleagues, whose flexibility, feedback and collaboration is indispensable to make this research work. Ruben, Sam, Davy, Wesley, Miel, Laurens, Anastasia, Pieter, Ben, Fréderic, ... the list goes on. Everyone from MMLab – now Data Science Lab: thank you! Ellen and Laura, thank you as well for taking a huge administrative and logistic load off of our hands, helping us focus on our work.

I can't forget to thank the people I've collaborated with over the years. I'd like to thank the people at iMinds-MICT, like Evelien, Peter, Matthias and Ellen, who are always ready to help put our work into a social context. To Sam Decrock, Matthias De Geyter, Mike Matton, Luk Overmeire, and Steven van Assche from the former VRT Medialab: thanks for helping me remember how fun coding can be, and for the great collaboration. I'd also especially like to thank Paul Groth, Sara Magliacane, Davide Ceolin and Frank van Harmelen for welcoming me as a visiting researcher at VU Amsterdam in 2014. Since then, Amsterdam really became a second hometown for me. During the month I stayed there, I got the chance to clear my thoughts, experience a different way of working, and lay the foundations for what would eventually become a chapter in this dissertation – I wish I could have stayed longer. On that note, other chapters became a reality thanks to other fruitful international collaborations. Special thanks go out to Io Taxidou and Peter Fischer from the University of Freiburg; to Christian Beecks from RWTH Aachen; to Robert Meusel, Kai Eckert, and Dominique Ritze from University of Mannheim; and to Luc Moreau from University of Southampton. Also thanks to Luc, Paul, and the other members of the W3C Provenance Working Group, for welcoming a junior researcher into their midst, and thereby skyrocketing my research. Finally, I thank Edzard Höfig, for giving me the opportunity to take over as chair of the METHOD workshop.

I also thank the members of the examination committee – prof. Patrick De Baets, prof. Guy De Tré, prof. Bettina Berendt, dr. Paul Groth, dr. Femke Ongenae, and prof. Peter Fischer – for their detailed reading reports and insightful remarks, that made this thesis so much better.

I probably do not want to know the effects all the stress, irregular lifestyle and comfort food associated with this PhD had on my Crohn's disease, but it probably isn't pretty. Therefore, perhaps a bit unusually, I'd also like to thank my physicians, prof. Peeters and dr. Hendrick, for their professional care and keeping me afloat even under these circumstances. Speaking of keeping me afloat, I also want to thank all of my friends for the much needed moments of relaxation, barbecue, and whisky: Christophe, Hendrik, David, Peter, Bilhah, Bert, Frank, Irene, ... thank you all!

And last but absolutely not least, I thank my wife Claudia. I would have thanked her together with my family, but she deserves a special spot of her own. She stuck by me through all the crazy deadlines, all the nights I spent working at the office or at home, all the moments of doubt, all my health issues, and all the stress, even when the chips were down for her as well. She also made many of the conferences unforgettable, by joining me in traveling the world. Heck, she even volunteered to listen to me practice several of my presentations, for which she simply deserves a medal. There is no one I would have rather made this journey with, and no one I would rather write the next chapter with.

Tom De Nies
Ghent, May 2016

Table of Contents

List of Figures

List of Tables

List of Acronyms

ADL	Advanced Digital Learning
AMT	Amazon Mechanical Turk
ASR	Automatic Speech Recognition
CSV	Comma Separated Values
EMD	Earth Mover's Distance
FOAF	The *Friend Of A Friend* ontology
HIT	Human Intelligence Task
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
IR	Information Retrieval
IRI	Internationalized Resource Identifier, a generalization of URI
ISBN	International Standard Book Number
JSON	JavaScript Object Notation
JSON-LD	JSON for Linked Data
LOD	Linked Open Data
MAP	Mean Average Precision
MRR	Mean Reciprocal Rank
N3	Notation3, an RDF serialization
NER	Named-entity recognition
NFD	Normalized Freebase Distance
NGD	Normalized Google Distance
NSWD	Normalized Semantic Web Distance
OPM	Open Provenance Model
ORCID	Open Researcher & Contributor ID
OWL	Web Ontology Language
PROV	The W3C family of documents for interoperable provenance
PROV-AQ	Provenance Access and Query
PROV-DM	The PROV Data Model
PROV-N	The PROV Notation
PROV-O	The PROV Ontology
SPARQL	SPARQL Protocol and RDF Query Language
SQFD	Signature Quadratic Form Distance
SMD	Signature Matching Distance
SQL	Structured Query Language

STT	Speech-To-Text
RDF	Resource Description Framework
RDFa	RDF Annotations – a notation used to embed RDF in (X)HTML pages
RDFS	RDF Schema
RecSys	Recommendation Systems
TF-IDF	Term Frequency - Inverse Document Frequency
Turtle	Terse RDF Triple Language, an RDF serialization
URI	Universal Resource Identifier
VSM	Vector Space Model
W3C	World Wide Web Consortium
WIQA	Web Information Quality Assessment
XHTML	Extensible HyperText Markup Language
xAPI	Experience API
XML	Extensible Markup Language

Summary

In the last two decades, the Web has evolved from a novelty to an integrated part of our daily lives. With this evolution, the amount of content on the Web has increased exponentially, creating a new problem for content authors, publishers, and consumers: *how do we decide which content to trust?*

In this dissertation, we argue that the key to answering this question is in the content's history, also known as its *provenance*. Our proposed solution is to offer a user insight into *who* contributed to a piece of content, *what* they contributed, *when*, *how*, and *where*. To achieve this, there are several aspects to be considered.

The first aspect is identifying the right technology to tackle the problem with. Since the problem is native to the Web, it is only fitting that a Web-native technology be used. The perfect candidate for our solution is the so called *Semantic Web*, a layer of the Web intended to be interpreted by machines, in addition to the layer visible to humans. In the Semantic Web, data is made self-descriptive, by linking it to other data, similar to the way that traditional Web pages use hyperlinks to link to other pages. This data is then referred to as Linked Data. The difference with traditional data is that now, these links can also be interpreted by machines. This enables us to apply much more complex reasoning to the data, in a way that would be too overwhelming for a human. Note that this reflects exactly the kind of problem we are trying to solve. The amount of data – on the Web or in databases – is too much for a human alone to judge in terms of its trustworthiness. By making it machine-interpretable, a computer can process it, reason over it, and present a human user with more manageable amounts of data and (preliminary) trust assessments.

The second aspect is how to use Semantic Web technology to model provenance. Before 2013, models to describe provenance were very diverse and often proprietary to one specific system, resulting in a situation where the provenance of the same thing was differently represented in different systems, and a large amount of information is lost. Recently, a standardization effort we contributed to at the World Wide Web Consortium (W3C) has made it possible to tackle this problem once and for all. The result of this effort was PROV-DM, the recommended data model for interoperable provenance. PROV-DM is a highly generic model, intended to allow the modeling of provenance in any use case. However, at the same time, it is often *too* generic, lacking the expressiveness to model domain-specific aspects of provenance. Therefore, we propose two extensions to PROV: one to model the provenance of information diffusion on social media, and one to model uncertain provenance and provenance of uncertainty.

The third aspect to consider is that there are cases where provenance is already present on the Web, albeit in a non-interoperable form. In most of these cases, a specific system is needed to access the provenance, and even then, the provenance is locked in a specialized format. A typical example of such a case is a version control system, which stores the complete version history – or in essence, provenance – of source code (and other data). We investigate how to expose this sort of provenance as W3C PROV. To this end, we propose a generic workflow, and illustrate its merit in three distinct use cases. First, we expose the provenance of version control systems, and illustrate it using a tool called Git2PROV. Second, we look into learning experiences logged using the so-called Tin Can API, and expose their provenance using TinCan2PROV. Third and last, we expose the provenance of a mapping quality assessment and refinement workflow, used to generate high-quality Linked Data. Each of these use cases, together with other applications and services exposing interoperable provenance, can be seen as a node in a *provenance ecosystem*, contributing to the common goal of adding provenance into the Semantic Web and enabling a *Web of Trust*.

A fourth aspect is that for most content on the Web, the provenance is incomplete or missing altogether. Therefore, we also propose a method to tackle the challenging task of reconstructing this lost provenance. More specifically, we propose to do this using semantic similarity. Our experiments show that this is possible with up to 68.2% precision and 73% recall when reconstructing the provenance inside a news archive. While this is far from perfect, it does provide the opportunity for a human user to verify the reconstructed provenance, which is vastly more convenient than manually reconstructing it. Furthermore, we show that when reconstructing provenance of information diffusion on social media, a significant amount of new influences can be discovered, in addition to the ones exposed by the social media APIs themselves.

Finally, the fifth aspect is how to use all of this accumulated provenance to generate statements about the trustworthiness of the content it is associated with. We provide several tools to do this, ranging from providing easy access to provenance on the Web, to assessing its validity and the reputation of the agents involved. All these tools are then combined to implement our version of the so called "Oh, Yeah?"-button. This button was envisioned by one of the inventors of the Web, Tim Berners-Lee, to be integrated into every browser for the user to press when he or she loses the feeling of trust. We focus on the aspects of the "Oh, Yeah?"-button that can be realized based on the provenance of a Web page, and provide users with information that helps in their decision whether or not to trust the page. This adheres to the philosophy that it is more useful to help a user identify *distrust events* than to provide a single, non-informative *trust score*.

To conclude the dissertation, we discuss how our proposed methods fit into a greater context of automatic content *value* assessment on the Web. An essential part of this work will include the assessment of relevance, through semantic similarity. Therefore, we briefly outline the experiments we have conducted using novel semantic similarity measures, which form the basis for our future work.

Samenvatting

In de laatste twee decennia is het wereldwijde web uitgegroeid van een leuke nieuwigheid tot een integraal deel van ons dagelijks leven. Door deze evolutie is de hoeveelheid inhoud op het web exponentieel toegenomen, wat een nieuw probleem heeft veroorzaakt voor auteurs, uitgevers en consumenten van deze informatie: *hoe beslissen we welke inhoud we kunnen vertrouwen?*

In dit doctoraat benadrukken we dat de sleutel tot het antwoord op deze vraag ligt in de geschiedenis van de inhoud, iets waar in het Engels naar gerefereerd wordt als *provenance*, wat in het Nederlands het dichtst vertaald kan worden als *herkomstinformatie*. Onze voorgestelde oplossing is om de gebruiker een inzicht te bieden in *wie* er bijgedragen heeft tot een bepaald stuk inhoud, *wat* ze bijgedragen hebben, *wanneer*, *hoe*, en *waar*. Om dit te bereiken, zijn er verschillende aspecten van het probleem te overwegen, die we één voor één zullen benaderen.

Het eerste aspect is een juiste technologie te vinden om het probleem mee aan te pakken. Aangezien het om een probleem eigen aan het web gaat, lijkt het maar logisch om ook voor een technologie te kiezen die eigen is aan het web. De perfecte kandidaat hiervoor is het zogenaamde *semantische web*, een onderliggende, complementaire laag van het web bedoeld om geïnterpreteerd te worden door machines. Op het semantische web wordt data zelf-beschrijvend gemaakt door ze te linken aan andere data, zoals webpagina's bedoeld voor mensen hyperlinks gebruiken om naar andere pagina's te linken. Deze data wordt dan benoemd als Linked Data. Het verschil met gewone data is dat deze links ook interpreteerbaar zijn voor machines. Dit maakt het mogelijk om veel complexere redeneringen te maken met de data, op een manier die veel te overweldigend zou zijn voor een mens. Merk op dat dit exact overeenkomt met het soort probleem dat wij proberen oplossen. De hoeveelheid data – op het web of in databanken – is te groot om door een mens alleen te worden ingeschat op gebied van betrouwbaarheid. Door deze data machine-interpreteerbaar te maken kan een computer deze verwerken en er over redeneren, en de menselijke gebruiker meer beheersbare hoeveelheden data voorschotelen, samen met een aantal (voorbarige) inschattingen van betrouwbaarheid.

Het tweede aspect is dan om deze semantische web technologie te gebruiken om herkomstinformatie te modelleren. Tot 2013 waren de modellen om herkomstinformatie te beschrijven zeer divers en vaak beperkt tot één specifiek systeem. Dit resulteerde in een situatie waarbij de herkomstinformatie van dezelfde dingen op verschillende manieren werd gerepresenteerd in verschillende systemen, waardoor veel nuttige informatie verloren ging. Een recente standaardisatie waar we aan bijgedragen hebben bij het wereldwijde web consortium (W3C) maakt het nu

echter mogelijk om dit probleem voor eens en altijd aan te pakken. Het resultaat hiervan is PROV-DM, het aangeraden data model voor interoperabele herkomstinformatie. PROV-DM is een zeer generiek model, bedoeld om de modellering van herkomstinformatie in vrijwel elk geval mogelijk te maken. Dit betekent echter dat tegelijkertijd, het model soms *te* generiek is, en de nodige expressiviteit mist om domein-specifieke aspecten van herkomstinformatie te modelleren. Daarom stellen wij twee extensies aan PROV voor: één om de herkomstinformatie van informatiediffusie op sociale media te modelleren, en een tweede om onzekere herkomstinformatie en herkomstinformatie van onzekere zaken te modelleren.

Een derde aspect om te overwegen is dat er gevallen zijn waarbij herkomstinformatie al op het web aanwezig is, zij het onder een niet-interoperabele vorm. In de meeste van deze gevallen is een specifiek systeem vereist om toegang te krijgen tot de herkomstinformatie, en meestal is deze dan ook in een gespecialiseerd formaat. Een typisch voorbeeld is versiebeheer, waarbij de volledige historiek van code (en andere data) wordt bijgehouden. Wij onderzoeken hoe we deze herkomstinformatie kunnen ontsluiten als W3C PROV. Om dit te doen, stellen we een generieke werkwijze voor, waarvan we de toepasbaarheid tonen in drie verschillende gevallen. Als eerste ontlsuiten we de herkomstinformatie uit versiebeheersystemen, wat we illustreren met de Git2PROV applicatie. Als tweede, beschouwen we de herkomstinformatie van leerervaringen bijgehouden met de zogenaamde Tin Can API, en ontsluiten we de herkomstinformatie hiervan met TinCan2PROV. Als derde en laatste geval, publiceren we de herkomstinformatie van een werkwijze om de kwaliteit van transformaties van data naar Linked Data na te gaan en te verfijnen. Elk van deze voorbeelden kan gezien worden als een component in een ecosysteem van herkomstinformatie, samen met andere systemen die interoperabele herkomstinformatie ontsluiten. Elk van deze systemen heeft hetzelfde doel: herkomstinformatie in het semantische web integreren, en een "web van betrouwbaarheid" mogelijk maken.

Het vierde aspect is dat voor de meeste inhoud er enkel gedeeltelijke of helemaal geen herkomstinformatie beschikbaar is op het web. Daarom stellen we een methode voor om de uitdagende taak aan te gaan om deze verloren herkomstinformatie te reconstrueren. Specifieker stellen we voor dit te doen op basis van semantische gelijkenis. Onze experimenten tonen tot 68.2% precisie en 73% recall met deze methode, toegepast op nieuws. Hoewel dit verre van perfect is, geeft het een menselijke gebruiker wel de kans om de gereconstrueerde herkomstinformatie na te gaan op correctheid, wat veel gemakkelijker is dan ze volledig manueel te reconstrueren. Verder zien we ook dat onze methode, toegepast op informatiediffusie op sociale media, een groot aantal nieuwe connecties aan het licht kan brengen, die zelfs voor de API's van de sociale media sites zelf verborgen waren.

Tot slot is er het vijfde aspect: hoe al deze geaccumuleerde herkomstinformatie te gebruiken om uitspraken te genereren over de betrouwbaarheid van de inhoud waarmee ze geassocieerd is. We bieden verscheidene technieken aan om dit mogelijk te maken. Deze gaan van het aanbieden van gemakkelijke toegang tot de herkomstinformatie op het web, tot de validiteit ervan nagaan en de reputatie van de betrokken partijen na te trekken. We combineren deze technieken vervolgens

om onze versie van de zogenaamde “Oh, Yeah?”-knop te implementeren. Deze knop werd voorgesteld door één van de uitvinders van het web, Tim Berners-Lee, en is bedoeld om in elke webbrowser geïntegreerd te worden, zodat er op gedrukt kan worden als een gebruiker het gevoel van vertrouwen verliest. We spitsen ons toe op de aspecten die op basis van de herkomstinformatie van een webpagina kunnen worden gerealiseerd, en tonen gebruikers informatie die kan helpen bij het al dan niet vertrouwen van deze pagina. Dit stemt overeen met de filosofie dat het nuttiger is om gebruikers indicaties van onbetrouwbaarheid te helpen detecteren, in plaats van een enkele, niet-informatieve betrouwbaarheidsscore te tonen.

Om de thesis af te ronden, bespreken we nog hoe onze voorgestelde methodes in een grotere context passen van automatische waardebepaling van inhoud op het web. Een essentiële component van dit werk zal zijn om automatisch relevantie in te schatten, op basis van semantische gelijkenis. Daarom schetsen we kort de experimenten die we al op dit gebied hebben uitgevoerd, gebruik makend van nieuwe manieren om semantische gelijkenis te meten. Deze vormen dan ook de basis voor ons toekomstig onderzoek.

Somewhere, something incredible is waiting to be known.

Carl Sagan

1 Introduction

Due to the abundance of content on the Web, content authors, publishers, and consumers have a pressing need for systems that select content that is trustworthy. However, it is not always clear what exactly makes content trustworthy. Is trust purely *objective*, purely *subjective*, or a mix of both? Is it something only a human can assess, or is it possible to automate the assessment partly or even fully, and if so, what is needed to make this happen? In essence, all these questions are closely related to the central vision of the Semantic Web: *machines that can interpret and analyze all content on the Web*. Therefore, in this doctoral research, we investigate the extent to which it is feasible to use Semantic Web technologies to enable the automatic assessment of trustworthiness of content that is digitally published. More specifically, we investigate the history of Web content, better known as its *provenance*, and how it contributes to trust.

In this first, introductory chapter, we explain the need and context for such research in Section 1.1, followed by a brief introduction to the terminology and key concepts needed to fully understand the remaining chapters of this dissertation in Section 1.2. In Section 1.3, we sum up the concrete research questions and hypotheses we investigate. This section also makes clear that while each chapter in this dissertation is intended to be understandable on its own and could be read in no particular order, it is also part of the "bigger picture" of automatic trust and value assessment.

This chapter is partly based on the following publications:

Tom De Nies. Assessing content value for digital publishing through relevance and provenance-based trust. In *The Semantic Web – ISWC 2013 (Doctoral Consortium)*, pages 424–431. Springer, 2013

Tom De Nies, Christian Beecks, Wesley De Neve, Thomas Seidl, Erik Mannens, and Rik Van de Walle. Towards named-entity-based similarity measures: Challenges and opportunities. In *Proceedings of the 7th International Workshop on Exploiting Semantic Annotations in Information Retrieval*, pages 9–11. ACM, 2014

1.1 The Need for Provenance and Trust on the Web

The explosion of content on the Web has resulted in an inevitable side-effect: the Web is littered with untrustworthy content, such as false information, spam, malicious pages, etc. In many cases, it is very difficult for a consumer to distinguish this untrustworthy content from trustworthy content. Moreover, information consumers and curators rarely receive enough time to process the continuous stream of information they are presented with. This leads to people – and machines – consuming and producing false information, trusting untrustworthy parties with their data, and making bad decisions. This is confirmed by Li et al. [129], who showed that an abundance of content can generate so-called *distrust events* for its users. In other words, there is a clear need for a (semi-)automatic solution to assess trustworthiness of content on the Web.

Intuitively, the history of a piece of content – who made it, how, when, where, and why? – is an important influencing factor when making a decision on whether to trust this content or not [88, 134]. This history is more formally known as *provenance*, defined as *information about entities, activities, and people involved in producing a piece of data or thing* [149]. Together with reputation, provenance plays an important role in making assessments about the trustworthiness of any content. The need for provenance has also been illustrated in several surveys and reports [89, 145], as well as the need to reconstruct it when it is incomplete or missing [135]. However, due to the relatively young nature of the field, the existing research to address these needs is very sparse, and in most cases highly domain-specific [145]. This has caused us to make the modeling, exposure, and reconstruction of provenance of Web content in general the main focus of our research. It also led us to contribute to its standardization as W3C PROV [149] in 2013. Since then, applications consuming and producing interoperable provenance have slowly started to emerge, forming a *provenance ecosystem*. In this dissertation, we show that such an ecosystem effectively enables the automatic assessment of trustworthiness on the Web, in a way that was not possible before.

1.2 Terminology and Key Concepts

Before diving into our methods towards providing provenance and enabling automatic trust assessment on the Web, we recommend that non-expert readers familiarize themselves with a number of essential concepts, which we briefly introduce in this section.

1.2.1 The Semantic Web

In 2001, Tim Berners-Lee, James Hendler, and Ora Lassila published an article in *Scientific American*, entitled *"The Semantic Web"* [20]. In this article, they explain – in popular terms – their vision of a *Web of Data*, which can be interpreted and used by intelligent software agents performing various tasks for humans. An important thing to understand is that this Web of Data is not meant to exist separately from the human-understandable Web, as we all know it, but rather as an extension of it. The Semantic Web gives well-defined meaning and structure to information on the Web, and thus enables computers and humans to work together in a better way.

1.2.2 Knowledge Representation as Graphs

A key component to realize the vision of the Semantic Web is the ability to represent knowledge in structured collections of information and inference rules that computers can access and *reason* over. The field of knowledge representation existed in artificial intelligence before, but it is only by applying the principles of the Web that it became applicable on a large scale. The most important of those principles is *decentralization*, which encompasses a compromise: give up the ideal of total consistency to allow for unlimited, exponential growth. This means the (Semantic) Web operates under the assumption of an *open world*: it is never possible to be sure that everything is found. However, it also means that the languages used are very flexible, and can support every desired scenario.

RDF The language to express meaning – and actually, everything – on the Web of Data is RDF, short for the *Resource Description Framework*. In RDF, everything is encoded in sets of *triples*, which express relations in the form *subject – predicate – object*, as illustrated by Figure 1.1.

RDF has several notations, including XML and Turtle. The Turtle notation is commonly preferred, thanks to its relatively easy-to-read syntax for humans. For example, we can express that the book "A Game of Thrones" was written by George R.R. Martin in Turtle as follows:

Example 1.1

```
:A_Game_of_Thrones  :author  :George_R._R._Martin .
```

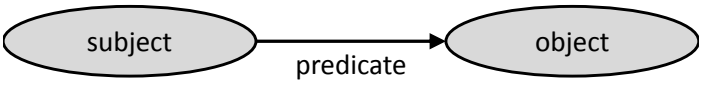

Figure 1.1: Conceptual illustration of an RDF triple.

However, this does not carry much meaning yet for a computer. Therefore, in RDF, the subject, predicate, and object are each identified by a *Universal Resource Identifier* – a URI for short[1]. This means that anyone can introduce new concepts or predicates, simply by defining a URI on the Web, which is what makes RDF such a flexible and expressive language. It also means that for each of these subjects, predicates and objects, there can be no confusion with concepts bearing the same name, as URIs are completely *unambiguous*. For example, `dbpedia.org` hosts URIs to represent the concepts from our earlier triple:

Example 1.2

```
<http://dbpedia.org/resource/A_Game_of_Thrones>
  <http://dbpedia.org/ontology/author>
  <http://dbpedia.org/resource/George_R._R._Martin> .
```

Now, when a computer wants to know more about a certain resource, its URI is *dereferenced*, meaning that the computer can navigate to the URI using HTTP and request its representation intended for machines. The result is a set of RDF triples describing the properties of that resource. Note that this is very similar to the way a human uses a Web browser to navigate to a URI and the browser uses HTTP to request the representation of the resource intended for humans, commonly resulting in an HTML page. A computer can then deduct *meaning* from these triples, since RDF is backed by a logic known to all systems that conform to the standard.

Because this notation with URIs is not very user-friendly for a human to read, Turtle allows the specification of *URI prefixes*. The above example is equivalent to the following, much better readable example:

Example 1.3

```
@prefix dbpedia: <http://dbpedia.org/resource/> .
@prefix dbpedia-owl: <http://dbpedia.org/ontology/> .

dbpedia:A_Game_of_Thrones
  dbpedia-owl:author  dbpedia:George_R._R._Martin .
```

Note that both the resources for ‘A Game of Thrones’ and ‘George R.R. Martin’ have the same URI prefix `http://dbpedia.org/resource/`. In this case, it is said that these concepts belong to the same *namespace*.

[1] Except when the object is a so called *literal*, e.g., a piece of text, a number, etc.

When many of these triples are put together in a *triple store*[2], *knowledge graphs* are created. For example, the following is only a small excerpt from the data available in the `dbpedia.org` knowledge graph for the resource `http://dbpedia.org/resource/A_Game_of_Thrones`:

Example 1.4

```
@prefix  dbpedia:     <http://dbpedia.org/resource/> .
@prefix  dbpedia-owl: <http://dbpedia.org/ontology/> .
@prefix  dbpprop:     <http://dbpedia.org/property/> .

dbpedia:A_Game_of_Thrones
  dbpprop:name       "A Game of Thrones" ;
  dbpedia-owl:author dbpedia:George_R._R._Martin ;
  dbpedia-owl:isbn "ISBN 0-553-57340-3(US paperback)";
  dbpedia-owl:series dbpedia:A_Song_of_Ice_and_Fire .

dbpedia:George_R._R._Martin
  dbpprop:name       "George R.R. Martin" ;
  dbpprop:website    <http://www.georgerrmartin.com> .

dbpedia:A_Song_of_Ice_and_Fire
  dbpprop:name       "A Song of Ice and Fire" .
```

For each of these triples, a subject – predicate – object relation can be drawn as in Figure 1.1, resulting in the graph from Figure 1.2. Note that in our example, the URI prefixes are different for the concepts – '`dbpedia:`' – and the properties – '`dbpedia-owl:`' and '`dbpprop:`'. This happens to separate the actual data from its structure, better known as its *ontology*.

1.2.3 Ontologies

If everyone can define their own URIs to describe concepts, it is obvious a mechanism must exist to specify the relationships between all these URIs, and to allow for reuse of existing descriptions. This is where *ontologies* come into play. An ontology is a set of descriptions that formally defines the relations among terms[3]. Typically, an ontology is composed of a *taxonomy* and a set of *inference rules*.

Taxonomy In the taxonomy, classes of objects are defined, as well as the relations between them, such as subclass-relations and properties. For example, a class `Book` may be described by the properties `author`, `title` and `isbn`. The

[2] a database optimized to store and query triples

[3] More formally, an ontology is defined as an explicit specification of a conceptualization [100]

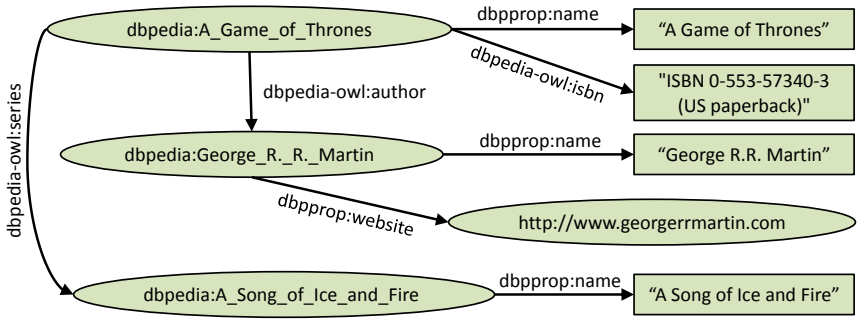

Figure 1.2: Graphical representation of an excerpt from the the data available in the dbpedia knowledge graph for the resource `http://dbpedia.org/resource/A_Game_of_Thrones`.

`author` property may point to an object of the `Person` class, the `title` and `isbn` properties may point to a plain text entry. In this case, the class `Book` is part of what is referred to as the *domain* of the `author` property, and the class `Person` is part of what is referred to as its *range*. The domains of `title` and `isbn` include `Book` as well, and their ranges include the so called *literal* class.

Ontologies are expressed in an interoperable way using two W3C Recommendations: RDF Schema (RDFS) [27] and the Web Ontology Language (OWL). RDFS is a basic data-modeling vocabulary for RDF data, whereas OWL [138] and its successor OWL 2 [189] provide more complex constructs. Both RDFS and OWL were developed as vocabulary extensions for RDF. In other words, the ontologies used to describe things in RDF, are described in RDF as well! The taxonomy from the above example is expressed in RDFS/OWL as follows:

Example 1.5

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

:Book
  a             owl:Class .
:author
  a             owl:ObjectProperty ;
  rdfs:domain   :Book ;
  rdfs:range    foaf:Person.
:isbn
  a             owl:ObjectProperty ;
  rdfs:domain   :Book ;
  rdfs:range    rdfs:Literal .
```

```
:title
  a              owl:ObjectProperty ;
  rdfs:domain    :Book ;
  rdfs:range     rdfs:Literal .
```

Note that we did not introduce a new `:Person` class, but use the `foaf:Person` class from the existing FOAF[4] ontology instead. On the Web of Data, reuse of data and terminology is encouraged, so existing data can be enriched with new knowledge. For example, if we put the RDFS/OWL from the above example book ontology in a file and make it dereferenceable at a namespace (e.g., `http://example.org/bookontology#`), other data publishers can use these concepts as well (e.g., by using `http://example.org/bookontology#Book` as a type for objects in their dataset that refer to books).

Inference Rules The inference rules in an ontology enable the expression of even more meaningful relations. Inference rules are commonly used to express the characteristics of the properties in an ontology, such as symmetry, transitivity, (inverse) functionality and inverse properties. For example, say we want to introduce a new relation `:wrote` to our example book ontology:

Example 1.6

```
:wrote
  a                owl:ObjectProperty ;
  rdfs:domain      foaf:Person ;
  rdfs:range       :Book .
```

To humans, it is obvious that the properties `:author` and `:wrote` are very related (they're *inverse properties*). However, a machine cannot know that until we formally express that relation. Therefore, an inference rule may express "IF a person is author of a book, THEN that person wrote that book". This inference rule could be expressed in Notation3 (N3) – a superset of Turtle supporting formulas, variables and quantification – as follows:

Example 1.7

```
{
  ?book   :author  ?author .
}
=>
{ ?author   :wrote  ?book . }
```

[4]The name FOAF was derived from the acronym for 'Friend of a Friend'. Its full specification is available at `http://xmlns.com/foaf/spec/`

We'd then also have to express the other direction of this inference rule:

Example 1.8

```
{
  ?author   :wrote  ?book .
}
=>
{ ?book   :author  ?author . }
```

Since this would quickly become a hassle, OWL includes a number of properties that allow us to express this kind of inferences rules more efficiently. For example, the two inference rules above can be expressed by one triple stating that `:author` and `:wrote` are inverse properties:

```
:wrote owl:inverseOf :author .
```

Since `owl:inverseOf` is already specified by OWL to be a symmetric property, this triple is all we need. We can then write one inference rule covering all cases of inverse properties:

Example 1.9

```
{
  ?property1   owl:inverseOf  ?property2 .
  ?resourcea   ?property1  ?resourceb .
}
=>
{ ?resourceb   ?property2  ?resourcea . }
```

Some triple stores already include generic inference rules such as this one, and provide the option for their query engine to take inferenced data into account.

Querying data A knowledge graph can be queried in very rich ways. Similar to the way relational databases are queried through the Structured Query Language (SQL), RDF has its own query language, called SPARQL [158].

Queries written in SPARQL resemble N3, only with additional clauses such as `SELECT` and `WHERE`. For example, to query the DBpedia knowledge graph for *who wrote the book "A Game of Thrones"*, we use the following SPARQL query:

Example 1.10

```
SELECT ?author WHERE {
  dbpedia:A_Game_of_Thrones
  dbpedia-owl:author
  ?author .
}
```

Triple stores generally expose their data through a public API that accepts SPARQL queries, more commonly referred to as a *SPARQL endpoint*. Executing the above query on the public SPARQL endpoint of DBpedia[5] results in the result set shown in Table 1.1.

author
http://dbpedia.org/resource/George_R._R._Martin

Table 1.1: DBpedia result set for the query 'who wrote the book "A Game of Thrones"?'.

Naturally, SPARQL queries can be made much more complex than that. For example, if we want to get a list of book authors born in the United States that are novelists as well as screenwriters, we would use the following query:

Example 1.11

```
SELECT DISTINCT ?author WHERE {
?book   a                       dbpedia-owl:Book ;
        dbpedia-owl:author      ?author  .
?author dbpedia-owl:occupation  dbpedia:Novelist,
                                dbpedia:Screenwriter ;
        dbpedia-owl:birthPlace  dbpedia:United_States .
}
```

which on DBpedia results in the result set shown in Table 1.2.

author
http://dbpedia.org/resource/George_R._R._Martin
http://dbpedia.org/resource/Jack_Ketchum
http://dbpedia.org/resource/Robert_Crais
http://dbpedia.org/resource/Daryl_Haney
http://dbpedia.org/resource/John_Fante
http://dbpedia.org/resource/John_A._Russo

Table 1.2: DBpedia result set for the Example 1.11 query for book authors born in the United States that are novelists as well as screenwriters.

An important concept to remember when dealing with the result of SPARQL queries is the *open-world assumption*. This means that a SPARQL query result is not guaranteed to contain all possible matches on the entire Web of Data, only those considered by the SPARQL endpoint. In case the SPARQL endpoint is defined for one single dataset, the result of each query will only contain triples from within that dataset. In case a SPARQL endpoint is *federated* over multiple data

[5]Try this yourself at `http://dbpedia.org/sparql`

sources – in which case it executes a query on all data sources individually and merges the results – its query results will only contain triples from any of the considered data sources.

A full explanation of all the features of SPARQL out of scope for this thesis, and is not necessary to understand the remaining chapters of this dissertation. For more information, we refer to the full specification of SPARQL [158] and its 2013 update SPARQL 1.1 [180].

Integrating existing data The true power of the Semantic Web becomes clear when we start integrating data from different sources. We will use an example to demonstrate this: say we have a small relational database of books in the English language which we refer to as '`en`', as shown in Figure 1.3.

ISBN	Author	Title
0-553-57340-3	a_1	A Game of Thrones

ID	Name	Blog
a_1	George R.R. Martin	http://grrm.livejournal.com/

Figure 1.3: Example of a relational database '`en`' containing English books.

Because the database is composed of relations, we can easily abstract this information as a graph, reusing the concepts from our example book ontology and the FOAF ontology, as shown in Figure 1.4.

This graph can be expressed completely by the following RDF triples.

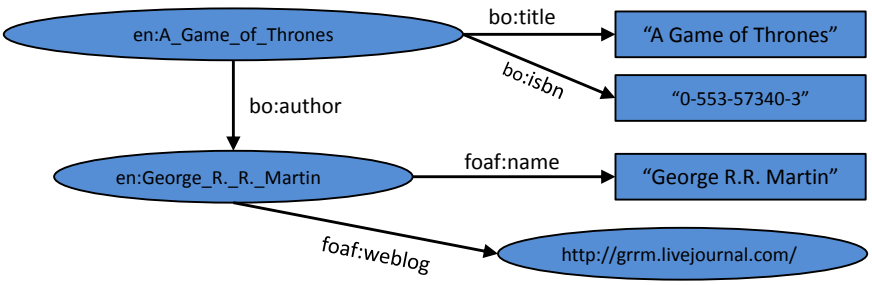

Figure 1.4: Example of a database '`en`' containing English books, abstracted as a graph.

Example 1.12

```
@prefix  en:     <http://example.org/englishbooks/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

en:A_Game_of_Thrones
  bo:title       "A Game of Thrones" ;
  bo:author      en:George_R._R._Martin ;
  bo:isbn        "0-553-57340-3" ;

en:George_R._R._Martin
  foaf:name      "George R.R. Martin" ;
  foaf:weblog    <http://grrm.livejournal.com/> .
```

This same exercise can be done with all kinds of data, regardless of whether it is stored in a relational database, an Excel file, a CSV file, or any other format. The abstracted RDF data does not even have to replace the original data format, but could be generated from the original source using a *mapping*, and exist in parallel.

If we examine Figure 1.4 and Figure 1.2, we see that the graphs in `en` and DBpedia show a lot of similarities. As a human, this is easy to see, but a machine needs to be told explicitly that `en:A_Game_of_Thrones` refers to the same thing as `dbpedia:A_Game_of_Thrones`. In RDF, this is done using an `owl:sameAs` link:

Example 1.13

```
en:A_Game_of_Thrones
  owl:sameAs
  dbpedia:A_Game_of_Thrones .
```

Now any machine that dereferences `en:A_Game_of_Thrones`, knows that more information is available at `dbpedia:A_Game_of_Thrones`, since both resources refer to the same thing. We can go even further, and add more cross-dataset semantics to `en`:

Example 1.14

```
bo:isbn   owl:equivalentProperty dbpedia-owl:isbn .
bo:author owl:equivalentProperty dbpedia-owl:author .
```

A machine will now be able to interpret that if a resource has a certain value for the property `bo:isbn`, it also can be assumed to have the same value for `dbpedia-owl:isbn` (ditto for `bo:author` and `dbpedia-owl:author`).

This means that a new graph can be inferred by combining the graph in `en` with the graph on DBpedia, as illustrated in Figure 1.5. Note that both original graphs get enriched with new information by this simple merge. For example,

now we can use this graph for the query 'give me the website of the author of the book with ISBN 0-553-57340-3', while this is information that was in *neither* of the original graphs.

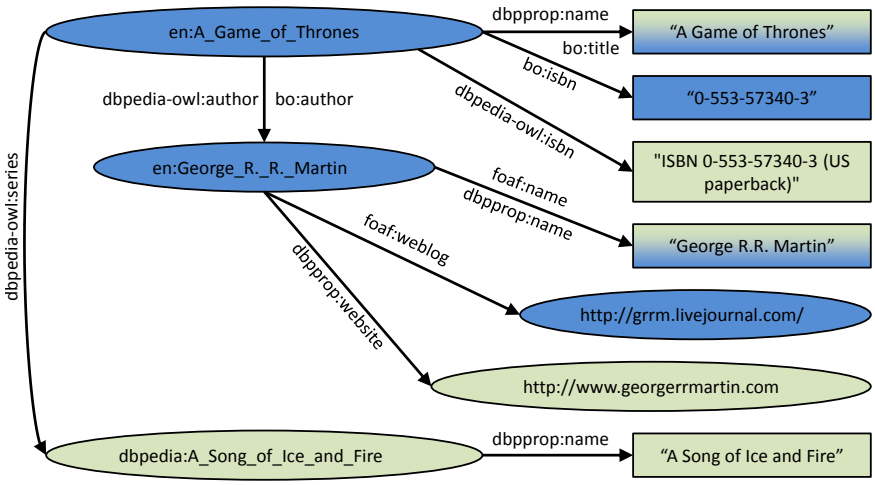

Figure 1.5: Inferred graph by merging the `en` *and DBpedia graphs.*

While this example demonstrates the enormous potential of the Semantic Web, it also raises a number of important questions. *How can we know where the data used to answer a certain query came from? Who is responsible for this data? Can the source be trusted? What if we integrate a data source polluted with faulty* `owl:sameAs` *links?* These are exactly the kind of questions that we address in this PhD thesis.

1.2.4 Linked (Open) Data

After the idea of the Semantic Web was coined in 2001, the academic and industrial communities did not adopt it as eagerly as they did with the original World Wide Web. As a result, the Semantic Web did not expand as fast as its inventors had hoped [169]. While there are likely multiple reasons behind this, an important contributing factor was the following stalemate: *application developers did not have enough data to work with, and data providers did not want to provide data until there were applications to use it*.

To break this stalemate, Tim Berners-Lee proposed what he called "the four rules of *Linked Data*" [19] in 2006:

1. Use URIs as names for things.
2. Use HTTP URIs so that people can look up those names.

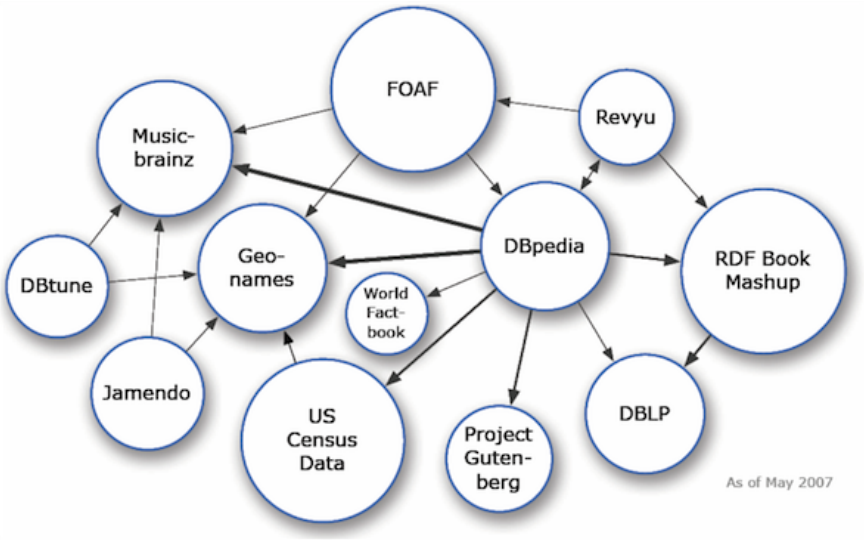

Figure 1.6: LOD cloud diagram 2007, by Max Schmachtenberg, Christian Bizer, Anja Jentzsch and Richard Cyganiak. Source: `http://lod-cloud.net`

3. When someone looks up a URI, provide useful information, using the standards (RDF, SPARQL).

4. Include links to other URIs. so that they can discover more things.

In other words, Linked Data is simply what the Semantic Web is filled with. Through these rules, and the efforts of Linked Data evangelists, data providers were convinced to start providing Linked Data, and thus developers could start building Semantic Web applications [24]. In fact, some data providers went even further and also provided their data under an open license, to be reused for free. This collection of datasets that were open and linked to each other is what became known as the *Linked Open Data (LOD) cloud*. In 2007, it was still very limited, with only 12 datasets – as illustrated by Figure 1.6. After this, the LOD cloud expanded very quickly to 570 datasets in 2014, as illustrated by Figure 1.7[6].

As could be expected, such rapid growth was accompanied by a serious decline in data quality. Businesses started to incorporate Linked Open Data in their marketing strategies, even when they only had a few Excel files on a site somewhere. To make sure all this data could eventually be used by the Semantic Web agents he had envisioned, in 2010 Tim Berners-Lee added the *five stars of Linked Open Data* to his earlier article on Linked Data [19]:

[6]Although it is hard to discern at this scale, Ghent University – iMinds – Multimedia Lab (now known as Data Science Lab) even hosts its own dataset with publication data in this version of the LOD cloud, available at `http://data.mmlab.be/`

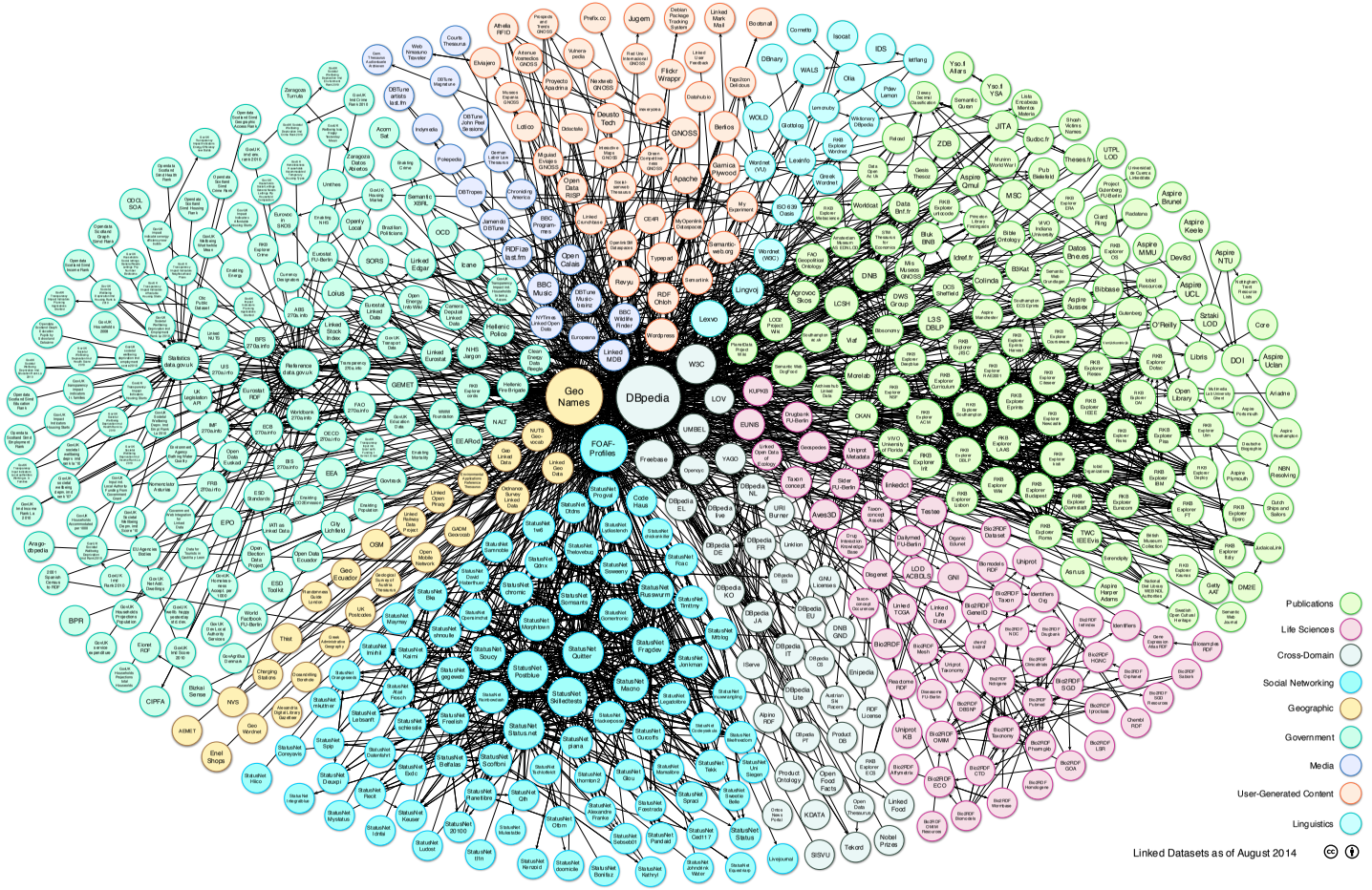

Figure 1.7: LOD cloud diagram 2014, by Max Schmachtenberg, Christian Bizer, Anja Jentzsch and Richard Cyganiak. Source: `http://lod-cloud.net`

⋆ Available on the web (whatever format) but with an open licence, to be Open Data.

⋆⋆ Available as machine-readable structured data (e.g., excel instead of image scan of a table).

⋆⋆⋆ All the above plus: in a non-proprietary format (e.g., CSV instead of excel).

⋆⋆⋆⋆ All the above plus: use open standards from W3C (RDF and SPARQL) to identify things, so that people can point at your stuff.

⋆ ⋆ ⋆ ⋆ ⋆ All the above, plus: link your data to other people's data to provide context.

These five stars are now the guidelines for any institution that publishes Linked Open Data. Of course, there are still plenty of scenarios where Linked Data is not open (and should not be), but is still useful in the private sector. For example, think of businesses enriching each other's data by providing links, before selling that data to their customers through their own applications. It is true that publishing data as Linked Open Data is the quickest way to get it adopted by Semantic Web applications, but it is by no means a necessity.

1.2.5 Semantic Annotations

In Sections 1.2.1 and 1.2.4, we discussed how to model and publish structured data on the Semantic Web. However, we did not yet raise the question: what about all the *unstructured* content already available on the Web?

The largest problem when connecting unstructured, textual content to the Semantic Web is *ambiguity*. For example, if the word "apple" is encountered in a document, its meaning can be either the fruit apple or the company Apple. When annotating that document by linking it to, e.g., `dbpedia.org`, correct disambiguation should ensure that the fruit "apple" is linked to `http://dbpedia.org/resource/Apple`, and that the company "apple" is linked to `http://dbpedia.org/resource/Apple_Inc.`. In recent years, the amount of research on the Semantic Web has increased and new methods for exposing machine-interpretable annotations and metadata have been developed, growing increasingly accurate and useful. Several forms of semantic annotation exist, ranging from categorization and topic detection to recognition of named entities and linking (parts of) content to the LOD cloud. One key element is crucial in all of these scenarios: *accurate disambiguation*.

RDFa Another issue we are faced with when annotating documents on the Web is the *format* in which we encode these annotations, and how we allow applications to retrieve them. In Section 1.2.1, we discussed RDF and its various notations.

In 2008, RDFa was recommended by the W3C as an RDF notation specifically intended for annotating Web pages [5]. Since then, the specification was updated to RDFa 1.1 in 2013 [3] and again in 2015 [4]. As a full explanation of all the features of RDFa is beyond the scope of this introduction, we refer the interested reader to the latest version published by W3C [4] or the introductory RDFa Lite document. In essence, RDFa makes it possible to embed machine-interpretable RDF data into the structure of human-interpretable (X)HTML pages[7]. For example, say we want to make a personal Web page accessible for humans as well as machines. If we are only interested in helping humans understand who the site belongs to, we could write the following HTML code:

Example 1.15

```
<p>My name is Tom De Nies.<p>
```

However, we could also easily add a number of RDFa annotations:

Example 1.16

```
<p resource="http://tomdenies.be/#Tom"
prefix="foaf: http://xmlns.com/foaf/0.1/"
typeof="foaf:Person">My name is
  <span property="foaf:name">Tom De Nies</span>
</p>
```

Note that by doing this, we have not changed anything to the way a human will perceive this HTML code, since neither attributes nor the `<span>` tag are visible when viewing this code in a browser. However, we have made the code interpretable for machines, since now it contains the following RDF triples:

Example 1.17

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

<http://tomdenies.be/#Tom>
   a foaf:Person;
   foaf:name "Tom De Nies" .
```

In other words, by adding those RDFa annotations, we made all machines that visit this Web page and have access to an *RDFa processor* understand that the content refers to a person, with the name "Tom De Nies".

This technology is essential to bridging the human-interpretable Web and the Semantic Web, since machines are no longer limited to the pure Web of Data. In fact, RDFa allows HTML pages to connect to the Web of Data, which allows us to consider them for the scenarios and methods discussed in this doctoral thesis.

[7] A popular vocabulary used for this purpose is `http://schema.org/`

1.2.6 Named Entities

Several advances relevant to this dissertation were recently made in the field of *named-entity recognition (NER)*. Named entities are words in a text that refer to persons, locations, companies, objects, events, etc. NER services try to recognize these named entities and their correct semantics. For example, ideally, a NER service would be able to discern that in the sentence *"I saw George in Washington the other day"*, "George" refers to a person, and "Washington" to a place, and that in the sentence *"I saw George Washington the other day"*, "George Washington" refers to the first president of the United States, even though the two sentences are very syntactically similar. Ideally, these sentences should be automatically annotated to allow a machine to make this distinction, e.g., with RDFa as described in Section 1.2.5.

NER and disambiguation is the subject of many ongoing research efforts [160, 181, 183], both in academia and on the industrial level, as indicated by initiatives such as the Microsoft ERD Challenge [140]. Popular commercial NER solutions include AlchemyAPI[8] and OpenCalais[9] by Thomson Reuters, whereas open source initiatives such as DBpedia Spotlight[10] also exist. Proposed techniques to improve disambiguation are very diverse, and vary from mining additional training data [130] to exploiting Linked Data structure [39]. It is evident that lower accuracy of the annotation process will inevitably lead to lower accuracy of any Semantic Web applications using these annotations.

1.3 Research Questions, Hypotheses and Outline

In Section 1.1, we discussed the need for automatic trust assessment on the Web. Therefore, the main research question in our work will come as no surprise.

Research Question 1
How can we enable automatic assessment of the trustworthiness of content on the Web?

From all approaches that we've come across in literature, and as already mentioned in Section 1.1, two things can be distilled: the provenance of a piece of content is a key component in assessing its trustworthiness, as is its reputation. Therefore, we investigate the following main hypothesis:

Hypothesis 1
Basic automatic trustworthiness assessments can be made by accessing the content's provenance and the reputation of the entities, agents and processes involved.

[8] http://alchemyapi.com
[9] http://opencalais.com
[10] http://spotlight.dbpedia.org/

Before we investigate whether this hypothesis is valid in Chapter 5, we must address a number of new sub-questions that this hypothesis raises. We must first ask ourselves:

Research Question 2
How can provenance be modeled in an interoperable way across multiple use cases?

We address this question in Chapter 2, with the following hypothesis:

Hypothesis 2
Provenance can be modeled in an interoperable way by using the W3C PROV standard, extended when needed for specific use cases.

However, during our research, we observed that for the vast majority of content on the Web, the provenance information is obscured in a non-interoperable way, incomplete or missing. This leads to two new research questions. The first additional research question is:

Research Question 3
When provenance information is obscured in a non-interoperable way, how can we expose it?

In Chapter 3, we address this question by investigating the following hypothesis:

Hypothesis 3
When obscured in a non-interoperable way, provenance can be exposed by automatically mapping it to an interoperable form.

The second additional research question is:

Research Question 4
When provenance information is incomplete or missing, how can we reconstruct it?

This leads to the following hypothesis, which we investigate in Chapter 4:

Hypothesis 4
When (partially) missing, provenance can be reconstructed based on the content's semantic similarity to other content.

The latter hypothesis also caused us to investigate semantic similarity, and overall content relevance assessment. This led to the realization that our research also fits within a broader picture of automatic content value assessment on the Web, by also considering the relevance of the content to the consumer. Methods to automatically assess relevance and semantic similarity are already intensively investigated in the Information Retrieval (IR) and Recommendation Systems (RecSys) communities. However, the increased popularity and availability of Semantic Web technologies such as those introduced in Section 1.2.1 caused us to ask the following question:

Research Question 5
How can we improve existing methods to automatically assess semantic similarity and/or relevance using Semantic Web technologies?

This has lead to the following hypothesis, which we briefly discuss in Chapter 6:

Hypothesis 5
Existing methods to automatically assess semantic similarity and/or relevance of content can be improved based on extracted semantic features.

Each of these research questions and corresponding hypotheses forms a subject on its own, and thus, each chapter in this dissertation is written in a self-contained way so it can be read individually and in arbitrary order. The exception to this rule is Chapter 7, where all the lessons learned during this doctoral research are summed up, followed by an outlook to future work.

The very ink with which all history is written is merely fluid prejudice.

Mark Twain

2 Modeling Provenance

Before 2013, the majority of users who wanted to assert provenance did so in their own, proprietary way. The research community responded to this by proposing several models for provenance, in the hope that more consistency would be obtained in the provenance asserted across multiple applications. For example, one of the models that emerged from the community, was the Open Provenance Model (OPM) [146]. All these modeling efforts eventually lead to the creation of a Provenance Working Group at the W3C [147], in which we actively participated. In April 2013, this group finalized a family of documents referred to as PROV [99]. In this chapter, we introduce the concepts needed to understand and work with PROV. Additionally, we discuss two extensions we created for the PROV model: one to model information diffusion on social media, and one to model uncertainty.

This chapter is partly based on the following publications:

Io Taxidou, Tom De Nies, Ruben Verborgh, Peter M. Fischer, Erik Mannens, and Rik Van de Walle. Modeling information diffusion in social media as provenance with W3C PROV. In *Proceedings of the 24th international conference on World Wide Web Companion – MSM 2015*, pages 819–824, 2015

Tom De Nies, Sam Coppens, Erik Mannens, and Rik Van de Walle. Modeling uncertain provenance and provenance of uncertainty in w3c prov. In *Proceedings of the 22nd international conference on World Wide Web Companion – Posters*, pages 167–168, 2013

Document	Type	Description
PROV-OVERVIEW	Note	provides an overview of all 12 documents
PROV-PRIMER	Note	provides an intuitive entry point to PROV
PROV-DM	Rec	describes the full PROV Data Model and all its features
PROV-CONSTRAINTS	Rec	describes the constraints that govern PROV-DM
PROV-SEM	Note	discusses formal semantics of PROV-DM
PROV-DICTIONARY	Note	extends PROV-DM with a specific type of collection, consisting of key-entity pairs
PROV-LINKS	Note	extends PROV-DM with a mechanism to link across different PROV bundles
PROV-N	Rec	formally describes the PROV Notation, a human-interpretable serialization of PROV-DM
PROV-O	Rec	describes the PROV Ontology, a machine-interpretable serialization of PROV-DM modeled in OWL2
PROV-XML	Note	describes an XML serialization of PROV-DM
PROV-DC	Note	describes a mapping between the PROV-O and Dublin Core ontologies
PROV-AQ	Note	describes how to refer to, access, and query provenance

Table 2.1: Overview of the PROV Family of Documents.

2.1 The W3C PROV Family of Documents

In this section, we provide a brief introduction to W3C PROV. The full PROV family is comprised of 12 documents: 4 *Recommendations* –which means that these are fully endorsed by the W3C – and 8 *Notes* – which means that these are documents that are considered useful by the Working Group, but are not (yet) formally endorsed by the W3C. The full list of documents is shown in Table 2.1. In this introduction, we focus on the essential concepts of PROV-DM, its recommended notations (PROV-N and PROV-O), and its constraints. PROV-N is a notation especially developed for PROV, intended to be easily readable for humans. PROV-O, on the other hand, is an ontology to represent the PROV concepts in RDF, intended for machines.

PROV is built around three essential concepts: *entities*, *activities*, and *agents*. In order to describe their provenance, these concepts can be connected with each other through various relationships. Figure 2.1 shows the relationships included in PROV-DM, and how they interconnect the three central concepts. The symbols used in the figure correspond to the conventions established in PROV: ellipses for `prov:Entity`, rectangles for `prov:Activity`, pentagons for `prov:Agent`, and directed arrows for relations between them. Note that in PROV, one is always

talking about the past. Therefore, the direction of the relations – and thus, the arrows in Figure 2.1 – is inverse to that of the actual process or workflow, which might seem counter-intuitive at first glance. For example, in a typical workflow we would assert "activity 1 generates entity 2", whereas the provenance of this workflow is logged as "entity 2 was generated by activity 1". In the rest of this section, we briefly discuss every concept and its relations.

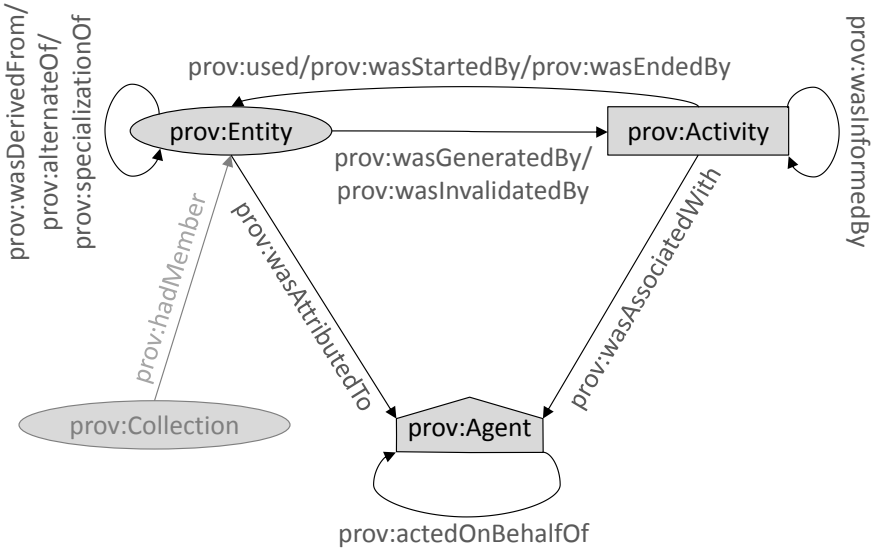

Figure 2.1: All relations included in PROV-DM, and how they interconnect the concepts Entity, Activity, and Agent.

2.1.1 Entities, Derivations and Alternates

In PROV-DM, the concept *Entity* is defined as:

Definition 2.1
*An **entity** is a physical, digital, conceptual, or other kind of thing with some fixed aspects; entities may be real or imaginary.*

This is – purposely – a very broad definition, that allows virtually anything to be described as an entity. For example, a webpage could be an entity, but a building could be as well, or even an idea someone had at some point. In PROV-N, an entity is written as `entity(id, [attr1=val1, ...])`, with `id` an identifier, optionally appended with a list of attribute-value pairs representing additional information about the fixed aspects of this entity. In PROV-O, this translates as:

```
@prefix : <#> .
@prefix prov: <http://www.w3.org/ns/prov#> .

:id   a       prov:Entity ;
      :attr1  "val1" .
```

Entities are defined by the fixed aspects of the thing they represent. This means that in PROV, it is possible that two separate entities actually fix different aspects of the same thing. For example, one entity could be defined as the digital PDF of this dissertation, while another entity could be defined as the print copy of the same dissertation. In PROV, two such entities `e1` and `e2` are consider as *alternates*, denoted by `alternateOf(e1, e2)` in PROV-N, and by the following triple in PROV-O:

```
:e1  prov:alternateOf  :e2 .
```

A *specialization* is a special kind of alternate. According to PROV-DM, *"an entity that is a specialization of another shares all aspects of the latter, and additionally presents more specific aspects of the same thing as the latter"*. A typical use case is versioning: e.g., the version of an online news page on a specific day, which is a specialization of that news page in general. In PROV-N, a specific entity `e1` that is a specialization of a general entity `e2` is described by `specializationOf(e1,e2)`, which translates to the following triple in PROV-O:

```
:e1  prov:specializationOf :e2 .
```

The main purpose of having entities in PROV is of course to describe their provenance, including which other entities were involved in their production. An important relationship included in PROV for this purpose is *derivation*, defined as *"a transformation of an entity into another, an update of an entity resulting in a new one, or the construction of a new entity based on a pre-existing entity"*. When an entity `e1` was derived from another entity `e2`, this is denoted in PROV-N by `wasDerivedFrom(e1, e2, [prov:type='derivationType'])`. Optionally, a *derivation type* may be specified. Note that when this is the case, the derivation relationship has more than two arguments, which makes it more complex to represent in triple form. To represent such *n-ary* relations in PROV-O, the *Qualification Pattern* [75] is used. This means that the first resource is linked to a new instance of an intermediate class that represents the relation between two resources, in this case using `prov:qualifiedDerivation`.

```
:e1  prov:wasDerivedFrom       :e2 ;
     prov:qualifiedDerivation  [
     a  derivationType .
 ] .
```

PROV-DM includes three derivation types: Revision, Quotation, and PrimarySource. When `e1` is a revised version of `e2`, this is denoted in PROV-N or PROV-O by substituting `derivationType` by `prov:Revision` in the notation above, or by using a shorthand property `prov:wasRevisionOf` (only in PROV-O). When `e1` is the repeat of (some or all of) `e2` by someone who may or may not be its original author, this is considered a *quotation* in PROV. Quotations are denoted in PROV-N or PROV-O by substituting `derivationType` by `prov:Quotation` in the notation above, or by using the shorthand property `prov:wasQuotedFrom` (only in PROV-O). When `e1` was derived from `e2`, and `e2` refers to something produced by some agent with direct experience and knowledge about the topic, at the time of the topic's study, without benefit from hindsight, `e2` is considered as the primary source of `e1`. This is represented in PROV-N/PROV-O by substituting `derivationType` by `prov:PrimarySource` in the notation above. In PROV-O, the shorthand `prov:hadPrimarySource` can also be used.

Finally, there is one special type of entity: a *collection*. `prov:Collection` is an entity type that provides a structure to its members, which are entities themselves. In PROV-N, a collection `c` with members `m1` and `m2` is described as follows:

```
entity(m1)
entity(m2)
entity(c, [prov:type='prov:Collection'])
hadMember(c, m1)
hadMember(c, m2)
```

This translates to PROV-O as follows:

```
:m1  a  prov:Entity .
:m2  a  prov:Entity .
:c   a  prov:Collection ;
     prov:hadMember  :m1, :m2 .
```

There are no restrictions to the number of members a collection can have. When a collection had no members, it can be given the type `prov:EmptyCollection`.

With these relations and entity types, a fair amount of provenance can already be provided for an entity, albeit on a fairly coarse-grained level. For example, say we consider the CNN front page `cnn`, which contained an article `article1` on August 31st, 2015 which was revised to an article `article2` on September 1st. Only by using the concepts discussed up to here, we can already model the provenance of that page and both articles in PROV-N, as shown in Example 2.1.

Example 2.1

```
entity(cnn, [url="http://cnn.com/"])
entity(cnn-20150831, [prov:type='prov:Collection'])
entity(cnn-20150901, [prov:type='prov:Collection'])
specializationOf(cnn-20150831, cnn)
specializationOf(cnn-20150901, cnn)

entity(article1)
entity(article2)
hadMember(cnn-20150831, article1)
hadMember(cnn-20150901, article2)
wasDerivedFrom(article2, article1,
                        [prov:type='prov:Revision'])
```

To go into more detail, we need more fine-grained concepts, such as activities.

2.1.2 Activities, Usage and Generation

In PROV-DM, the concept *Activity* is defined as:

Definition 2.2
*An **activity** is something that occurs over a period of time and acts upon or with entities; it may include consuming, processing, transforming, modifying, relocating, using, or generating entities.*

Activities are *disjoint* with entities, as specified in PROV-CONSTRAINTS [32]. This means that when something is asserted as an activity, it can never be asserted as an entity as well. As the definition suggests, *time* is a very important factor when talking about activities. Therefore, PROV-DM allows the (optional) specification of the start time and end time of an activity when it is expressed. In PROV-N, an activity is denoted as `activity(id, startTime, endTime, [attr1=val1, ...])`, which translates to the following PROV-O triples:

```
@prefix : <#> .
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix xsd:  <http://www.w3.org/2001/XMLSchema#> .

:id   a                  prov:Activity ;
      prov:startedAtTime "startTime"^^xsd:dateTime
      prov:endedAtTime   "endTime"^^xsd:dateTime
      :attr1             "val1" .
```

As Definition 2.2 states, activities act upon or with entities. To remain as widely applicable as possible, PROV-DM models three generic types of these interactions:

an activity may *use* entities, *generate* them, and *invalidate* them. A usage, generation, or invalidation is an instantaneous event, that describes *the beginning of the utilization of an entity*, *the completion of its production*, or *the start of its expiry*, respectively. Note that one activity may use, generate, and/or invalidate multiple entities between its start and end time, as long as a number of constraints to ensure consistency are maintained[1]. Usage, generation, and invalidation are described using the PROV relations `prov:used`, `prov:wasGeneratedBy`, and `prov:wasInvalidatedBy`, respectively.

When one activity generates an entity which is used by a second activity, this is known in PROV as *communication* between these two activities. In other words, at a certain point between its start and end time, the latter activity used information generated by the former. This concept is modeled by the relation `prov:wasInformedBy`.

To illustrate these concepts, let's revisit Example 2.1 and see what we can append to it. We know that `article1` was generated by the writing activity on August 31st, and that `article2` was generated by the writing activity on September 1st. We also know that the staff working on the writing activity on September 1st used the content of `article1` in order to revise it. In PROV, this means that the writing activity on September 1st was informed by the writing activity on August 31st. Say the revision was necessary because of a mistake in `article1`, which was corrected in `article2`, after which `article1` was taken offline. In this case, `article1` was invalidated by the writing activity on September 1st. All of this information can be modeled in PROV-N, as shown in Example 2.2.

Example 2.2

```
activity(writing-20150831, 2015-08-31T06:00:00,
                                  2015-08-31T23:59:59)
activity(writing-20150901, 2015-09-01T06:00:00,
                                  2015-09-01T23:59:59)
wasGeneratedBy(article1, writing-20150831)
wasGeneratedBy(article2, writing-20150901)
used(writing-20150901, article1)
wasInformedBy(writing-20150901, writing-20150831)
wasInvalidatedBy(article1, writing-20150901)
```

With these concepts, we can go into detail about *how* and *when* entities were produced. To know *who* was involved, we need the concept of agents.

[1] Formally describing these constraints here is out of scope for this thesis. We refer the interested reader to the official PROV-CONSTRAINTS Recommendation [32]

2.1.3 Agents and Attribution

In PROV-DM, the concept *Agent* is defined as:

Definition 2.3

*An **agent** is something that bears some form of responsibility for an activity taking place, for the existence of an entity, or for another agent's activity.*

In other words, an agent can be a person, but also a software agent, an organization, etc. In PROV-N, an agent is denoted as `agent(id, [attr1=val1, ...])`, which translates to the following PROV-O triple:

```
@prefix : <#> .
@prefix prov: <http://www.w3.org/ns/prov#> .

:id   a       prov:Agent ;
      :attr1  "val1" .
```

Note that this bears a lot of similarity with how entities are defined. In fact, it is perfectly possible for something in PROV to be both an agent and an entity. This is particularly useful when the provenance of the agent itself also needs to be asserted. There is also no formal disjointness between activities and agents. However, one should keep in mind that some specific types of agents may be suitable as activities (e.g., software agents), while others may not (e.g., persons).

In PROV, an agent can be *associated* with one or more activities, indicating that the agent had a – possibly unspecified – role in these activities. If an agent is involved in the creation of an entity, this entity is *attributed* to that agent. Association of an activity with an agent is expressed by the `prov:wasAssociatedWith` relation, whereas attribution of an entity to an agent is expressed with the relation `prov:wasAttributedTo`. The role an agent had in its association with an activity or its attribution to an entity is specified using the `prov:role` attribute.

In many cases, more than one agent is responsible for a certain activity, or one agent is acting in another's stead. The latter case is referred to as *delegation*, where one agent is assigned authority and responsibility (by itself or by another agent), while the agent it acts on behalf of retains some responsibility. In PROV, this is modeled using the `prov:actedOnBehalfOf` relation.

With these relations, we can now add the concept of responsibility to Example 2.1 and Example 2.2. Say `article1` was written by a writer named Alice. Alice wrote the article on August 31st, and realized she made a mistake on September 1st. However, Alice was unavailable to correct the mistake due to a lack of internet access, so she called her editor Bob and asked him to correct it. Note that Bob only acted on behalf of Alice in the context of editing this article, not always[2]. In PROV-N, this is modeled as shown in Example 2.3.

[2]To specify this, the `prov:actedOnBehalfOf` relation must also specify the contextual activity. In PROV-O, the `prov:qualifiedDelegation` property is used for this, as shown in Example 2.5.

Example 2.3

```
agent(alice, [prov:label="Alice"])
agent(bob, [prov:label="Bob"])
wasAssociatedWith(writing-20150831, alice,
                              [prov:role="Writer"])
wasAttributedTo(article1, alice)
wasAssociatedWith(writing-20150901, bob,
                              [prov:role="Editor"])
wasAttributedTo(article2, bob)
actedOnBehalfOf(bob, alice, writing-20150901)
```

2.1.4 Influence

Most of the relations between entities, activities and agents in PROV specify some form of influence. Therefore, a generic notion of influence was defined in PROV-DM as follows:

Definition 2.4

Influence *is the capacity of an entity, activity, or agent to have an effect on the character, development, or behavior of another by means of usage, start, end, generation, invalidation, communication, derivation, attribution, association, or delegation.*

The only three relations in PROV-DM that are not considered to be influences are `alternateOf`, `specializationOf`, and `hadMember`. For the relations that are influences, there is always one party who is deemed to be the *influencee*, and another to be the *influencer*.

2.1.5 Bundles

One final concept we did not discuss yet, is how to group provenance descriptions together. For this purpose, PROV-DM provides *bundles*. Apart from grouping a set of provenance descriptions, bundles are also an important mechanism to support provenance of provenance, which is important to assess the trustworthiness of the provenance itself.

A bundle is specified in PROV-N using a constructor and an identifier:

```
bundle id
description1
description2
...
endBundle
```

In PROV-O, a bundle is simply the set of RDF statements that describe the provenance. Other than that, PROV-O does not specify how to encode bundles in

RDF. However, it is suggested to use the *named graph* construct from the TriG syntax [22] for this purpose. The above example in PROV-O with TriG syntax is encoded as follows:

```
:id {
  description1
  description2
  ...
}
```

The final step in completing our running example is to group all the statements from Examples 2.1, 2.2, and 2.3 together in a bundle, so we can specify its provenance – for example, that we generated it.

Example 2.4

```
bundle runningExample
entity(cnn, [url="http://cnn.com/"])
entity(cnn-20150831, [prov:type='prov:Collection'])
entity(cnn-20150901, [prov:type='prov:Collection'])
specializationOf(cnn-20150831, cnn)
specializationOf(cnn-20150901, cnn)

entity(article1)
entity(article2)
hadMember(cnn-20150831, article1)
hadMember(cnn-20150901, article2)
wasDerivedFrom(article2, article1,
[prov:type='prov:Revision'])
activity(writing-20150831, 2015-08-31T06:00:00,
 2015-08-31T23:59:59)
activity(writing-20150901, 2015-09-01T06:00:00,
 2015-09-01T23:59:59)
wasGeneratedBy(article1, writing-20150831)
wasGeneratedBy(article2, writing-20150901)
used(writing-20150901, article1)
wasInformedBy(writing-20150901, writing-20150831)
wasInvalidatedBy(article1, writing-20150901)

agent(alice, [prov:label="Alice"])
agent(bob, [prov:label="Bob"])
wasAssociatedWith(writing-20150831, alice,
[prov:role="Writer"])
wasAttributedTo(article1, alice)
wasAssociatedWith(writing-20150901, bob,
[prov:role="Editor"])
wasAttributedTo(article2, bob)
actedOnBehalOf(bob, alice, writing-20150901)
endBundle
```

```
agent(tom, [prov:label="Tom De Nies"])
wasAttributedTo(runningExample, tom)
```

As can be seen, a significant amount of information that helps determining the trustworthiness of the articles can be distilled from this provenance trace: we know that the articles are part of a Web page with URL `http://cnn.com/`, who made them, when, under which roles, and that a revision was made that invalidates the first article. However, as mentioned at the start of this section, PROV-N is intended for human use. For a machine to understand this information, it is better to encode the provenance in PROV-O, as shown in Example 2.5.

Example 2.5

```
@prefix : <http://example.org/> .
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

:runningExample {
  :cnn a prov:Entity ; :url "http://cnn.com/" .
  :cnn-20150831 a prov:Entity , prov:Collection ;
    prov:specializationOf :cnn .
  :cnn-20150901 a prov:Entity , prov:Collection ;
    prov:specializationOf :cnn .
  :cnn-20150831 prov:hadMember :article1 .
  :cnn-20150901 prov:hadMember :article2 .

  :article1 a prov:Entity ;
   prov:wasGeneratedBy :writing-20150831 ;
   prov:wasInvalidatedBy :writing-20150901 ;
   prov:wasAttributedTo :alice .
  :article2 a prov:Entity ;
   prov:wasGeneratedBy :writing-20150901 ;
   prov:wasAttributedTo :bob ;
   prov:wasDerivedFrom :article1 ;
   prov:qualifiedDerivation [
    a prov:Revision ;
    prov:entity :article1
  ] .

  :writing-20150831 a prov:Activity ;
    prov:startedAtTime
         "2015-08-31T06:00:00.000+01:00"^^xsd:dateTime ;
    prov:endedAtTime
         "2015-08-31T23:59:59.000+01:00"^^xsd:dateTime ;
    prov:wasAssociatedWith :alice ;
    prov:qualifiedAssociation [
      a prov:Association;
```

```
      prov:agent   :alice;
      prov:hadRole "Writer"
    ] .
  :writing-20150901 a prov:Activity ;
    prov:startedAtTime
         "2015-09-01T06:00:00.000+01:00"^^xsd:dateTime ;
    prov:endedAtTime
         "2015-09-01T23:59:59.000+01:00"^^xsd:dateTime ;
    prov:used :article1 ;
    prov:wasInformedBy :writing-20150831 ;
    prov:wasAssociatedWith :bob ;
    prov:qualifiedAssociation [
      a prov:Association;
      prov:agent   :bob;
      prov:hadRole "Editor"
    ] .

  :alice a prov:Agent ;
    rdfs:label "Alice" .
  :bob a prov:Agent ;
    rdfs:label "Bob" ;
    prov:qualifiedDelegation [
      a prov:Delegation;
      prov:agent :alice ;
      prov:hadActivity :writing-20150901
  ].
}
:runningExample a prov:Bundle ;
  prov:wasAttributedTo :tom .
:tom a prov:Agent ;
  rdfs:label "Tom De Nies" .
```

2.1.6 Extending PROV

The PROV concepts discussed in this section already provide the means to specify provenance in many use cases. However, in some cases it is necessary to expand the possibilities of PROV-DM to model specific scenarios, or it is desired to provide more fine-grained provenance than PROV-DM allows. To account for these cases, PROV was kept deliberately general, and made particularly easy to extend. PROV-DM provides the following extensibility points [149]: (1) *sub-types* and *sub-relations*; (2) application and domain specific *roles*; (3) application-specific *attributes*. This means that by simply creating new data types, roles, and/or attributes, it is still possible to assert valid provenance, yet provide sufficient detail for any use case. We illustrate this in the next sections by describing two extensions of our own.

2.2 Modeling Provenance of Information Diffusion on Social Media

We created two extensions to PROV-DM to cater to the needs of our work. The first extension – PROV-SAID, described in this section – allows us to model the provenance of information diffusion on social media. The second extension – UP, described in Section 2.3 – allows to specify the provenance of uncertain things and uncertain provenance.

In recent years, *information diffusion* in social media has attracted the attention of researchers, since the produced data is fast, massive and viral [101]. Additionally, the *provenance* of such data is equally important because it helps to judge the relevance and trustworthiness of the information enclosed in the data. However, social media currently provide insufficient mechanisms for provenance, while models of information diffusion use their own concepts and notations, targeted to specific use cases. In this work, we propose a model for information diffusion and provenance, based on *W3C PROV*. The advantage is that PROV is a Web-native and interoperable format that allows easy publication of provenance data, and minimizes the integration effort among different systems making use of PROV.

2.2.1 Introduction to Information Diffusion

Social media such as online social networks (e.g., Facebook), micro-messaging services (e.g., Twitter) or sharing sites (e.g., Instagram) provide the virtual space in which a significant part of social interactions takes place. Many real-life situations, such as elections, are reflected by social media. In turn, social media shape these situations by forming opinions or strengthening trends, or by spreading reports on emerging situations faster than conventional media. Furthermore, word of mouth plays an important role in shaping users' attitudes and behavior. Most importantly, social media provide a huge audience (some users maintain millions of connections) where information can be easily spread and consumed by others. This phenomenon is referred to as *information diffusion* [102].

Because there exists a plurality of opinions and multiple sources of information in social media, the need for judging the relevance and trustworthiness of such information is becoming urgent. The understanding of how a piece of information propagated in social media provides additional context, including the source and its properties, the intermediate forwarders and the modifications that this piece of information has undergone. A social media user can take advantage of this context to assess how much value, trust, and validity such information carries.

For example, online journalists need to understand the cycle of information diffusion in a timely manner, by assessing the source and intermediate forwarders, predicting information *virality* as well as determining the impact of their own pub-

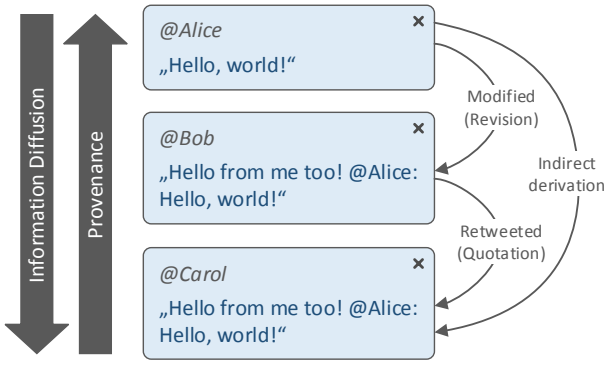

Figure 2.2: Information Diffusion and Provenance.

lications. Additionally, the detection of rumors is feasible not only by discovering the sources but also by analyzing the properties of the diffusion process [122] and the intermediate steps. When it comes to massive amounts of negative opinions expressed in social media, companies, politicians and celebrities need to understand who is propagating certain information and who is influencing others.

This kind of analysis actually refers to the reverse process of information diffusion: *information provenance*, that seeks the paths back to the sources. While provenance is a well researched topic in domains like workflows [92] or databases [31], it has received limited attention in the context of social media, compared to classical information diffusion. Likewise, existing models of information diffusion are insufficient to model provenance, while the current structure of social media provides limited or no mechanism to its users to judge received information [12]. For example, for retweets on Twitter, only the source of information is provided but not the intermediate steps (forwarders). However, it has been shown that forwarders play an equally important role in the information diffusion [11].

To further clarify the relation between information diffusion and provenance, we provide an example in Figure 2.2. Three Twitter users are emitting a similar message: Alice is the source of information diffusion, as she emits an original message. At a later point in time, user Bob modifies the original message and then user Carol copies and forwards (retweets) the message of Bob. In this process, it is important to understand how the message was modified and forwarded. User Carol was indirectly influenced by user Alice, since her message was *indirectly derived* from the source (two-step procedure). This means that the trustworthiness of all three users involved should be judged, since they participate in the diffusion and modification of this message.

Despite the variety of models of information diffusion, there is currently no unified, conceptual model for information diffusion and provenance that can be ap-

plied to different datasets and set-ups, while remaining both expressive and generic enough to cover many use cases. In this work we provide such a conceptual model. More specifically, we introduce **PROV-SAID**, a model to assert the **Prov**enance of **S**ocial medi**A** **I**nformation **D**iffusion based on PROV-DM.

The Web-native, interoperable design of PROV-DM is very useful in cases where data needs to be combined from different (social media) sources that do not share the same concepts and notations. Additionally, PROV-DM is domain-agnostic, but it has the benefit of extensibility, allowing domain-specific information to be included.

As our main contribution in this work, we introduce a number of new attribute values to extend PROV-DM, and relevant extensions to PROV-Constraints [32] to govern the use of these attributes values. In more detail, we provide: (1) a structured ontology for information diffusion and provenance on social media; (2) extensions of entities and activities relevant for information diffusion and provenance; (3) introduction of the use of these new concepts as attributes and roles in PROV assertions; and (4) extensions for the generic concept of *Influence* in PROV-DM. Note that on the one hand, our model allows the representation of social connections among users, since information flows through them in the majority of cases. On the other hand, the model is generic enough to assert the provenance of information diffusion even without the presence of social connections.

2.2.2 Motivation and Related Work

Information Diffusion in social media and networks has been a well researched topic. A review of relevant models can be found in [101]. Until now, the focus has been on the design of models with specific goals [12] (e.g., assessing the probability that certain users are being reached). Such research is mostly driven by data mining techniques to analyze specific datasets. This sort of analysis is useful in use cases such as marketing, solving the problem of maximizing the spread of information by targeting specific users (i.e., the *influence maximization* problem [116]).

While provenance is a thoroughly investigated topic in other domains [31, 92], existing models of information diffusion do not provide the means to express it. A review of challenges and methods for provenance on social media can be found in [12]. Authors propose their own method inducing both user attributes and network structure. Our work is complementary, since we provide a general model for provenance to be used in different use cases and they propose certain metrics and algorithms to assess provenance. An example of using simple user attributes (e.g. authority score) to express provenance is the Twitcident system [1], that traces emergent events in Twitter. However, information concerning modifications that tweets undergo and intermediate steps, is not being exposed. The work of [176] and [177] presents a system and visualizations for reconstructing diffusion paths

in real-time on social media. The proposed algorithm searches for all possible diffusion paths back to the sources and offers the possibility of different influence models in case it is not clear which paths the information took.

Finally, a PROV extension for the quantitative measurement of influence was proposed in [83]. There, the authors introduce the attribute `influenceFactor`, that allows users to attach a real or discrete value to a PROV influence relationship, to indicate the amount of influence an entity, activity or agent had over another. The concept of quantifying the level of influence is an interesting one, and could be applied complementary to the subtypes of influence we introduce in this chapter.

2.2.3 PROV-SAID Model Overview

In Sections 2.2.3, 2.2.4 and 2.2.5, we describe our model with its relevant extensions and constraints[3]. PROV has formal semantics [33], which cover our model as well, since our extensions and constraints are fully compliant with PROV.

Throughout the text, we provide a full example that covers all aspects of the model. To improve clarity, this example is unfolded incrementally and the reader should take into account information provided in previous examples.

Overview The PROV-SAID model can be applied to any social network where information propagates from user to user in the form of messages. Messages can be transmitted though *social connections*, but the model is general enough to capture *external influence* as well, as often happens in social media [150]. For example, Twitter users might publish information that has been seen on the public timeline without any direct social connection. Furthermore, our experience with provenance on Twitter shows that information does not flow only from social connections, but there is an external influence in approximately 20% of the messages [176]. The last observation derives from experiments with reweets where diffusion is explicit, while this percentage is much higher for non-explicit diffusion (propagation of Twitter hashtags).

Our model includes *activities* and *relationships* connected with information diffusion, such as exchanging *messages*, finding the *source* of diffusion, and expressing which *changes* the message has undergone through this procedure. User *influence* plays a key role in information diffusion since it drives information flow. The concept of influence in PROV-DM is vaguely defined and it is recommended to use more specific terms when possible. This is sensible since influence can take many forms in different use cases. However, for our use case the influence relationship has its own merit. Therefore, we define and extend the concept of influence, expressed through different activities, types and user roles.

[3]For detailed specification and formal constraints, see `http://semweb.datasciencelab.be/ns/prov-said/PROV-SAID.html`

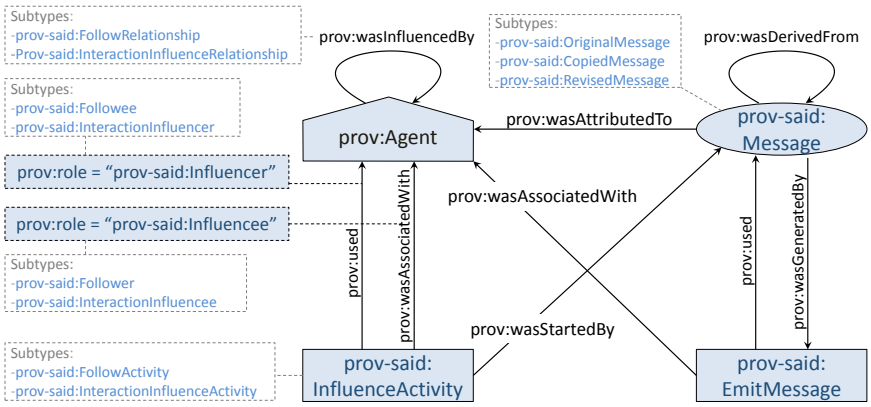

Figure 2.3: The PROV-SAID model.

Figure 2.3 shows a high-level overview of the PROV-SAID model. The proposed extensions to the standard are written in a blue font. In the next sections, we will describe each component in detail. Throughout this description, the prefix *prov:* refers to the PROV namespace[4] and the prefix *prov-said:* refers to the new PROV-SAID namespace[5]. Users who emit messages on social media are represented by the *prov:Agent* concept.

Design Decisions The purpose of PROV-SAID is to offer an easily reusable model that covers and infers different aspects of information diffusion and provenance. Note that the goal is not minimizing the relationships in the model, but offering maximum expressiveness, as is the case for PROV-DM.

Also, adhering to the extensibility points provided in PROV-DM, we use existing PROV concepts wherever possible, and define our own extensions for specific use cases. This way we improve clarity and we encourage reusability of the model. One example of extending the model, is the concept of *prov:Influence*. We differentiate the cases in the context of information diffusion and provenance and we give a clearer meaning to them.

Since social connections are the main carriers of information [176] we need to specify whether a message was propagated through them or whether there was some external influence. Therefore, other than just modeling information diffusion, we also implicitly model social graph connections (unidirectional relationships) produced through *follow* activities.

Next, we describe the components and relationships of our model step by step.

[4] `http://www.w3.org/ns/prov#`
[5] `http://semweb.datasciencelab.be/ns/prov-said/`

2.2.4 Modeling Messages

In order to model messages that are emitted by users, we propose the following extensions that are subtypes of *prov:Entity*:

- *prov-said:Message*: denotes the general class of messages. Messages in social media may be original messages, copied messages or revised messages.

To model this distinction, we define the following subtypes of *prov-said:Message*:

- *prov-said:OriginalMessage* denotes an original message that is not derived from any other message. The user who emitted it is the initiator of information propagation for that specific message.
- *prov-said:CopiedMessage* denotes a message which is based on another message that has been published in the past and was forwarded as an exact copy. Users who emit copied messages comply fully with the content and opinions of the original message. For example, Twitter offers the *retweet* function were users can easily forward copies of messages emitted by others.
- *prov-said:RevisedMessage* denotes a message that is produced by modifying an existing message. This means that the user who emits such a message may or may not share the original opinion of the original message. It is possible that the information carried by the original message is altered.

With these three types, we have covered the main cases of information diffusion through messages. Next, we need to specify how to model attribution, emission, and derivation of messages.

Message Attribution A *prov-said:Message* is always attributed to a *prov:Agent* using the relationship *prov:wasAttributedTo*. Example 2.6 illustrates the use of messages and attribution for the Twitter social network.

Example 2.6

```
prefix twitter: <http://twitter.com/>
prefix alice-status:
   <http://twitter.com/Alice/status/>
prefix bob-status:
   <http://twitter.com/Bob/status/>
prefix carol-status:
   <http://twitter.com/Carol/status/>

// User @Alice tweeted a message "Hello, world!"
prov:entity(alice-status:123,
            [prov:type='prov-said:OriginalMessage',
             prov:label="Hello, world!"])
```

```
// User @Bob modified and re-emitted the message
prov:entity(bob-status:456,
           [prov:type='prov-said:RevisedMessage',
            prov:label="Hello from me too!
                        MT @Alice: Hello, world!"])

// User @Carol retweeted (copied) the revised message
prov:entity(carol-status:789,
            [prov:type='prov-said:CopiedMessage',
             prov:label="Hello from me too!
                         MT @Alice: Hello, world!"])

// alice-status:123 was emitted by twitter:Alice
prov:wasAttributedTo(alice-status:123, twitter:Alice)
```

Message Emission Next we define the following activity that refers to message emission and is a subtype of *prov:Activity*

- *prov-said:EmitMessage* denotes a generic emission of a message. It must generate a *prov-said:Message*, and may use another *prov-said:Message*.

Note that the subtype of the generated *prov-said:Message* (original, copied or revised) can be inferred from the usage of another *prov-said:Message* by the *prov-said:EmitMessage*. If the content of the generated message is identical to that of the used one, it is a *prov-said:CopiedMessage*. If the content of the generated message was altered from that of the used one, it is a *prov-said:RevisedMessage*.

Message Derivation Whereas an original message does not have dependencies on other messages, copied and revised messages can be traced back to their original sources through derivation. PROV-DM already provides most of the concepts needed to model this, in the form of *prov:Quotation*, *prov:Revision*, and *prov:PrimarySource*, as illustrated by Example 2.7.

Example 2.7

```
// bob-status:456 was derived from alice-status:123
// (which is also its primary source)
// (in the context of Twitter)
prov:wasDerivedFrom(bob-status:456, alice-status:123,
                    emit-456, gen-456, use-123,
                    [prov:type='prov:Revision',
                     prov:type='prov:PrimarySource'])
```

```
// carol-status:789 was quoted from bob-status:456
// (which is not its primary source)
prov:wasDerivedFrom(carol-status:789, bob-status:456,
                    emit-789, gen-789, use-456,
                    [prov:type='prov:Quotation'])
```

We observe that *carol-status:789* was derived from *alice-status:123*, albeit indirectly. To model this special kind of dependency, we introduce the concept *prov-said:IndirectDerivation*. This way, we can model multi-step provenance and trace how messages are being derived, without being restricted to the previous step only. We illustrate this in Example 2.8.

Example 2.8

```
// carol-status:789 was indirectly derived from
// alice-status:123
prov:wasDerivedFrom(carol-status:789,alice-status:123,
           [prov:type='prov-said:IndirectDerivation'])
```

At this point, we express the following constraints:

- An *prov-said:OriginalMessage* cannot be derived from a *prov-said:Message*.
- A copied or revised message should always be derived from another message. A *prov-said:EmitMessage* that generates a *prov-said:CopiedMessage* and uses a *prov-said:Message* implies that the first message was derived from the latter by means of *prov:Quotation*. Analogously, when a prov-*said:RevisedMessage* is generated and a *prov-said:Message* is used by a *prov-said:EmitMessage*, this implies that the first message was derived from the latter by a *prov:Revision*.

These new provenance types and relations already allow us to model the way messages are diffused. However, to truly grasp all the aspects of information diffusion on social media, we must also consider modeling the types of *influence* users have on each other when propagating messages.

2.2.5 Modeling Influence

Influence plays a key role since it drives information flow in social media. Users are often being influenced by external factors such as traditional media, and consequently react in social media [150]. In this section, we model influence in the closed world of social media. Influence on social media has two ways of being expressed: through establishing social connections and through exchanging messages. Influence in PROV-DM is vaguely defined and we need a more expressive modeling to capture its different forms. We propose extensions for influence types, influence activities and influence roles.

Influence types First, we define a relationship *prov-said:InfluenceRelationship* to express influence in general, and two subtypes to specify the ways that such influence can be expressed.

- *prov-said:InfluenceRelationship* is a subtype of *prov:Influence*. It denotes an influence between agents in the context of social media.

We define the following two subtypes of *prov-said:InfluenceRelationship*:

- *prov-said:FollowRelationship* denotes that one agent was influenced by another agent, by establishing a unidirectional (follow) relationship. In the context of social media, that practically means being exposed to the messages emitted by the latter. For example on Twitter, this is the only way of connecting with users. Facebook also gives its users the possibility of unidirectional connection by subscribing to the messages of other users, apart from the bidirectional *Friendship* relation. Here, we assume that once an agent starts to follow another agent, he or she is exposed to the latter's old messages and future ones (if there are any). This is also the case on Twitter and Facebook. As a result, we do not model the influence that derives from exposure/subscription to messages explicitly, since it is implied by the *prov-said:FollowRelationship*. Furthermore, note that such a relationship entails a certain degree of uncertainty, since it can not be asserted whether an agent has seen the messages of another in reality.
- *prov-said:InteractionInfluenceRelationship* denotes that an agent was influenced by another agent, by having quoted or revised messages of the latter. As mentioned in Section 2.2.4, such a relationship can be discovered by investigating the similarity of messages of the former to the latter.

We illustrate the aformentioned influence types in Example 2.9.

Example 2.9

```
// User @Alice followed user @Carol, so a
// prov-said:FollowRelationship existed between them
prov:wasInfluencedBy(twitter:Alice,
                     twitter:Carol,
          [prov:type='prov-said:FollowRelationship'])

// User @Bob revised a message from @Alice, so a
// prov-said:InteractionInfluenceRelationship
// existed between them
prov:wasInfluencedBy(twitter:Bob,
                     twitter:Alice,
                      [prov:type='prov-said:
                 InteractionInfluenceRelationship'])
```

By following these influence types, both the social graph and the interaction graph [190] can be reconstructed at a certain point in time by using provenance. The interaction graph aggregates interactions (e.g., emission of messages) among users as weighted edges.

Influence Activities Additional to the influence types expressed as relationships among users (subtypes of *prov:Influence*), we explicitly model the corresponding activities. This design decision offers greater expressiveness by providing more information about the start and end time of influences, what triggered them, etc. For these purposes, we introduce three subtypes of *prov:Activity*:

- *prov-said:InfluenceActivity* is a subtype of prov:Activity. It denotes the activity of one agent influencing another with the following two subtypes:
- *prov-said:FollowActivity* denotes the activity of one agent that established a unidirectional connection with another. Once such an activity starts, the former agent is exposed to the (future and past) message emissions of the latter. Because the connection lasts for a prolonged period of time, this activity has a start time that denotes the time of establishing the connection and an optional end time in case the agent removes the connection with regard to the other agent.
- *prov-said:InteractionInfluenceActivity* denotes the activity of one agent to influence another, so that the latter interacts by forwarding the messages of the first. Note that here, the *prov-said:InteractionInfluenceActivity* is instantaneous, and thus has the same start and end time. This way, we can model multiple interactions of agents by generating multiple *prov-said:InteractionInfluenceActivity* instances. If we had chosen to allow *prov-said:InteractionInfluenceActivity* to be asserted only once, without an end time, we would have come to contradiction with the principles of information diffusion, where the significance of past interactions fades quickly over time.

Example 2.10 illustrates the subtypes of *prov-said:InfluenceActivity*.

Example 2.10

```
// A prov-said:FollowActivity started at the moment
// user @Alice followed user @Carol.
// Since @Alice was still following @Carol at the time
// of assertion, there is no end time for the activity

activity(alice-follows-carol, 2015-01-09T13:00:00, - ,
   [ prov:type='prov-said:FollowActivity'])

// A prov-said:InteractionInfluenceActivity was
// started at the moment user @Bob modified and
// re-emitted the message of @Alice.
```

```
activity(bob-influencedby-alice,
 2015-01-09T13:05:00, 2015-01-09T13:05:00,
 [prov:type='prov-said:InteractionInfluenceActivity'])
wasStartedBy(bob-influencedby-alice, bob-status:23456,
             emit-23456, 2015-01-09T13:05:00)
wasEndedBy(bob-influencedby-alice, bob-status:23456,
           emit-23456, 2015-01-09T13:05:00)
```

Influence Roles Analysts of information diffusion and influence in social media make use of specific roles for their agents [10]. To model this, we need to specifically define values for the *prov:role* attribute in the context of *prov:Usage* and *prov:Association*. This way, we clarify the roles of agents involved in a *prov-said:InfluenceActivity*. We define the following role-values:

- *prov-said:Influencer* denotes the role of an agent that was used by an *prov-said:InfluenceActivity* that was associated with another agent. This means that the first agent influences the latter.

- *prov-said:Influencee* denotes the role of an agent that was associated with an *prov-said:InfluenceActivity*. This agent is being influenced by another agent used by the same *prov-said:InfluenceActivity*.

To specify these roles even further, we define two additional subtypes of *prov-said:Influencee* and two subtypes of *prov-said:Influencer*. First, we model the follow relationship with the roles *Follower* and *Followee*. Second, we model the activity of interaction by exchanging messages with the roles *InteractionInfluencee* and *InteractionInfluencer*. Note that these roles are pairwise complementary by revealing the active behavior of one agent in order to establish connections and to forward messages (Follower, InteractionInfluencee) and the passive behavior of another (Followee, InteractionInfluencer) who exerts some influence on the first.

- *prov-said:Followee* is the role of an agent used by a *prov-said:FollowActivity* associated with another agent. This means that the latter followed the first.
- *prov-said:Follower* denotes the role of an agent that was associated with a *prov-said:FollowActivity*. It means that the agent established a unidirectional connection with another agent. In other words, the first followed the latter.
- *prov-said:InteractionInfluencer* denotes the role of an agent that was used by a *prov-said:InteractionInfluenceActivity* associated with another agent. It means that the first agent is influencing the latter in the a way that the latter propagates the messages of the first.
- *prov-said:InteractionInfluencee* denotes the role of an agent that was associated with an *prov-said:InteractionInfluenceActivity*. This means that the agent is being influenced by another agent by forwarding the messages of the latter.

Note that agents can be associated with multiple influence activities, with a potentially different role in each activity. For example, in the case of Facebook the *friend* relationship is symmetric, so when two agents establish a friend relationship, they both get the Follower role as well as the Followee role, albeit in two separate instances of FollowActivity.

We demonstrate the influence roles in Example 2.11.

Example 2.11

```
used(alice-follows-carol, twitter:Carol,
   [prov:role='prov-said:Followee'])
wasAssociatedWith(alice-follows-carol,
   twitter:Alice,
   [prov:role='prov-said:Follower'])

used(bob-influencedby-alice,
   twitter:Alice,
   [prov:role='prov-said:InteractionInfluencer'])
wasAssociatedWith
   (bob-influencedby-alice,
   twitter:Bob,
   [prov:role='prov-said:InteractionInfluencee'])
```

At this point we express the following constraints:

- A *prov-said:InfluenceRelationship* always implies that there exists a corresponding *prov-said:InfluenceActivity*, *prov:Usage* and *prov:Association.* According to the type of *prov-said:InfluenceActivity*, specific *prov:roles* are being used.
- A *prov-said:InteractionInfluenceActivity* starts (and ends, since it is defined to be instantaneous) with the emission of a *prov-said:CopiedMessage* or *prov-said:RevisedMessage*.

With these concepts, we have covered the model of influence with its possible expressions in activities, relationships and roles.

2.2.6 Future Additions

While the concepts described in this section already provide a detailed provenance model for social media analysis, further detail is always possible (although it might be argued whether this is desirable). Furthermore, the social media landscape is very dynamic, and new functionality and ways to influence other users are added regularly. For example, since its launch, Twitter has added the functionality to *reply* to and *quote* messages, as well as explicitly *mention* other users. Apart from

these explicit means of interaction, *implicit interactions* are also possible, which are not exposed by social media APIs. For example, users could propagate similar messages due to participation in the same event, in which case there is an *external influence*. Users could also re-propagate their own messages, for example to modify earlier statements or for promotional purposes. In this case, we could speak of *self-influence*. It is important to identify what kind of influence these interactions imply, under which circumstances. This will be possible by capturing the stream of messages in specific circumstances (e.g., an event) and observing which types of interactions and influences occur. To this end, we have already started capturing the Twitter stream during a number of conferences we attended, and confirmed the necessity for modeling self-influence and external influence [178].

As a next step, we will investigate in what ways the work on quantifying influence described in [83] can be applied to our model, and be used in the context of social media.

2.3 Modeling Uncertain Provenance and Provenance of Uncertainty

Our second extension to PROV-DM deals with the uncertain aspects of provenance. Currently, the PROV does not model uncertainty, which is a good thing, because that would make the model unnecessarily complex for those who do not need to model it. For asserting provenance of provenance, PROV already has a mechanism in place: *bundles*. However, in most cases, bundles contain many provenance statements, which makes it very hard, if not impossible, to talk about the provenance of individual statements. Whereas bundles enable coarse-grained provenance of provenance, this section illustrates how to model finer-grained Uncertainty Provenance (UP) using a lightweight approach. Three new attributes with clearly defined values and semantics are proposed. Modeling this information is an important step towards the modeling and derivation of trust from resources whose provenance is described using PROV.

2.3.1 Uncertainty Attributes

One of the advantages of PROV is its flexibility when it comes to attributes. PROV allows almost all provenance statements to be annotated with optional attributes, with a few exceptions, which are discussed in Section 2.3.1.3. Therefore, the most straightforward way of modeling UP is to specify one or more optional attributes for the existing constructs that allow them. First, we explain how to model uncertain provenance and how to allow provenance consumers to make trust assessments about the provenance itself. Then, we discuss how to model uncertainty of the content whose provenance is asserted, which is currently not possible in

PROV. In total, three attributes are proposed, with predefined values and semantic guidelines.

2.3.1.1 Modeling of Uncertain Provenance

According to [111], there are three aspects to uncertainty of provenance:

1. the assigned truth value of the asserter to a statement;
2. the truth value in the eyes of the consumer;
3. the trust relation between asserter and consumer.

When modeling uncertain provenance using PROV, only the first and last aspects apply, since the truth value a provenance consumer assigns to a statement is not meant to be asserted in PROV (and if it is, the consumer becomes an asserter, and we are back to the first aspect). Therefore, we define the following attributes[6], allowing a *degree* and *type* of (un)certainty to be specified for each PROV statement:

up:assertionConfidence This attribute has a *numerical value between 0 and 1*, and signifies the confidence assigned to a provenance statement by the asserter.

up:assertionType This attribute describes the type of uncertainty associated with a provenance statement. In our vocabulary, we predefined several values for this attribute. The values *up:HumanAsserted*, *up:MachineGenerated* and *up:MachineCollected* specify whether a provenance statement was generated by a human asserter, or generated or collected by an automated process. *up:Complete* and *up:Incomplete* signify whether all information about this statement is known. For example, this could mean that the statement has missing (optional) arguments, or that a collection has unknown members other than those asserted. *up:Future* signifies that the provenance describes a process that is yet to be executed, or entities that do not exist yet at the time of assertion. And finally, the values *up:Trusted* and *up:Untrusted* describe whether the provenance comes from a trusted or untrusted source.

2.3.1.2 Modeling of Uncertainty

It is important to distinguish the difference between uncertainty of asserted provenance itself, and asserting the uncertainty of information using provenance. This last concept is what we model in this section. Similar to the assertion confidence from the previous section, we define a new attribute:

up:contentConfidence This attribute specifies a confidence score, denoting how confident a user or application was about the content whose provenance is asserted. It has a *numerical value between 0 and 1*.

[6] for readability, we shorten `http://semweb.datasciencelab.be/ns/up/` as prefix `up:`

This type of provenance is useful in cases where applications or users make fuzzy decisions, and want to assert the provenance of these decisions. Typical examples of such use cases are named-entity recognition, automatic speech recognition (ASR), visual concept detection, etc.

2.3.1.3 Relations without Optional Attributes

Using the three attributes described above, we provide uncertainty information about almost all provenance concepts defined in PROV-DM. However, there are three relations in PROV that do not support optional attributes: *specialization*, *alternate* and *membership*. Here, the solution lies in specifying an additional entity, with the optional attributes, as a **specialization** of the specializing, alternate or member entity. This principle is illustrated in Example 2.13.

2.3.2 Use Case Examples

In this section, we will clarify the use of the attributes defined in Section 2.3.1, by providing a number of use cases where uncertain provenance is asserted.

Example 2.12
Provenance Reconstruction

```
entity(ex:document1)
entity(ex:document2)
entity(ex:document3)
wasDerivedFrom(`d1'; ex:document3, ex:document1,
  [up:assertionConfidence="0.6",
   up:assertionType='up:MachineGenerated'])
wasDerivedFrom(`d2'; ex:document3, ex:document2,
  [up:assertionConfidence="0.9",
   up:assertionType='up:HumanAsserted'])
```

In this example, `ex:document3` is derived from two different documents. While this is technically possible, the derivation `'d1'` was automatically generated with a relatively low confidence score, whereas `'d2'` was asserted with high confidence by a human. Applications consuming these provenance assertions now have the option to accept or reject the automatically generated assertions, if they decide not to trust them.

Example 2.13
Named-entity Recognition

```
entity(ex:document)
entity(ex:namedEntities,
  [prov:type='prov:Collection'])
```

```
activity(ex:NER)
wasDerivedFrom(ex:namedEntities,ex:document,ex:NER)
entity(dbpedia:New_York)
entity(dbpedia:Joe_Biden)
entity(ex:New_York,
  [up:contentConfidence="0.6"])
entity(ex:Joe_Biden,
  [up:contentConfidence="0.8"])
specializationOf(ex:New_York, dbpedia:New_York)
specializationOf(ex:Joe_Biden, dbpedia:Joe_Biden)
hadMember(ex:namedEntities, ex:New_York)
hadMember(ex:namedEntities, ex:Joe_Biden)
```

Here, we model the confidence the NER algorithm `ex:NER` had when extracting the named entities `dbpedia:New_York` and `dbpedia:Joe_Biden` from `ex:document`. Normally, this information is stored with the content, causing overhead for those users that are not interested in the provenance.

Example 2.14
Automatic Speech Recognition

```
entity(ex:word1, [prov:value="this"])
entity(ex:word2, [prov:value="it's"])
activity(ex:ASR)
entity(ex:transcript)
used(ex:ASR, ex:word1,
  [up:contentConfidence="0.8"])
used(ex:ASR, ex:word2,
  [up:contentConfidence="0.2"])
wasGeneratedBy(ex:transcript, ex:ASR,
  [up:contentConfidence="0.8"]))
```

In this last example, we model the process of the detection of a spoken word by an ASR algorithm. The two words “this” and “it’s” are very similar, and the algorithm had to choose an option based on the likelihood of it being the correct word. Logically, “this” was chosen because it had the highest confidence, but that does mean that the generation of the transcript only has the same confidence score of 0.8.

These examples show that there are plenty of use cases for fine-grained uncertain provenance. The attributes we introduced provide a flexible means of asserting this kind of provenance, while preserving the validity of the assertions in conformance with the PROV standards. Adaptation of these lightweight attributes opens an array of possibilities regarding trust assessment of both content and provenance

information. However, it might be argued that the lightness of the extension might also be a weakness, since the semantics of the confidence indication in the $[0, 1]$ interval are not always clear. The annotations with regards to the origin of these confidence indications – through the assertion type or other, custom attributes – play a crucial role in comparing and aggregating them. Therefore, it will be interesting to consider a number of constraints or recommendations for semantics of these annotations in future work.

2.4 Conclusion

In this chapter, we showed the broad applicability and flexibility of the PROV-DM, developed by W3C. Thanks to its generic core specification, multiple use cases related to provenance can be modeled in an interoperable way. This means that application builders, scientists, data publishers, etc. no longer have an excuse to lock in their provenance in a way that only they can understand. Through the usage of the interoperable PROV model, provenance traces from all these use cases can now be intertwined, exposing connections that were previously hidden from information consumers. This is further illustrated in Chapter 3, where we expose the provenance of three distinct systems as W3C PROV.

We have shown that in those cases where PROV does not provide sufficient detail, the model can easily be extended to accommodate for more fine-grained provenance, while still remaining compliant with the standard.

PROV-SAID, our first extension to the W3C model, enables systems that analyze social media to incorporate provenance data in their information diffusion analysis. This has the potential to relieve the massive human-centric efforts for judging relevance and trustworthiness of information by exposing its sources and intermediate steps. The true test of our extension to the PROV model will come through its usage. As the model is adapted in more use cases (e.g., the provenance reconstruction approach described in Section 4.7), it will become possible to evaluate its expressiveness and effectiveness.

With UP, we created a light-weight extension to the PROV-DM for users who wish to model uncertainty aspects of their provenance, without reverting to other existing, unnecessarily complex models. Apart from in our own applications, UP has already been adapted in use cases such as Dutch ships and sailors Linked Data [42] and Linked Data apps for Smart Cities [133].

Naturally, our extensions to model the provenance of information diffusion on social media, and uncertain provenance will still need to prove their merit through further use. However, in the cases where we have tested them, we did not encounter any obvious faults in terms of expressiveness or effectiveness. In Chapter 4, we apply both extensions to our provenance reconstruction approaches.

Everything in this world has a hidden meaning.
Nikos Kazantzakis

3
Exposing Provenance

Provenance is currently available for various types of content. However, in the vast majority of cases, provenance is obscured in a domain- or technology-specific way. This means that all tools and applications that wish to use this provenance must be domain- or technology-specific as well. In this chapter, we focus on exposing provenance from various sources in an interoperable form, so it may be used by generic approaches, applicable across many domains.

This chapter is based on the following publications:

Tom De Nies, Sara Magliacane, Ruben Verborgh, Sam Coppens, Paul T. Groth, Erik Mannens, and Rik Van de Walle. Git2PROV: Exposing version control system content as W3C PROV. In *International Semantic Web Conference (Posters & Demos)*, pages 125–128, 2013

Tom De Nies, Frank Salliau, Ruben Verborgh, Erik Mannens, and Rik Van de Walle. TinCan2PROV: Exposing interoperable provenance of learning processes through Experience API logs. In *Proceedings of the 24th international conference on World Wide Web Companion – LILE 2015*, pages 689–694, 2015

Tom De Nies, Anastasia Dimou, Ruben Verborgh, Erik Mannens, and Rik Van de Walle. Enabling dataset trustworthiness by exposing the provenance of mapping quality assessment and refinement. In *4th International Workshop on Methods for Establishing Trust of (Open) Data (METHOD 2015)*, 2015

3.1 Introduction

Provenance has been around for a long time. Institutions such as art merchants, libraries, archives, etc. have been recording the provenance of their inventories as part of their daily activities [174]. More recently, scientists have been recording provenance about their workflows and results in order to keep them reproducible [171]. However, before the standardization at the W3C, these stakeholders were lacking a uniform, interoperable way of expressing and exchanging provenance [145]. In other words, there is a vast amount of legacy provenance on the Web, obscured in some proprietary format, and thus locked in its own domain.

Furthermore, there are many applications that unknowingly generate provenance of the data they handle, without exposing it interoperably. The ultimate example of this is the version control system Git. Millions of programmers, writers and designers use Git, and upload their repositories to the public `github.com` website. All of this version history constitutes provenance, but it remains exposed only in the Git logs, which are difficult to interlink with other data.

Because there are so many use cases, it is impossible to create a "one size fits all"-solution to expose hidden provenance. Every use case has its own specific provenance concepts and relations, and may or may not demand incorporating extensions to the PROV model. However, the general workflow to expose provenance in a certain use case is always roughly the same:

1. Identify the entities, activities, and agents present in the existing data.
2. Identify information in the existing data that holds characteristics of PROV relations between the entities, activities, and agents.
3. Determine whether there are remaining provenance relations in the data that cannot be modeled in sufficient detail using the PROV Data Model. If yes, use or create an extension to the PROV Data Model (as described in Section 2.1.6). If no, continue to step 4.
4. Write down the full PROV – in a serialization of your choice – that is about to be exposed, using generic placeholders for the concept identifiers and relation attributes identified in steps 1, 2 and 3.
5. Automate the mapping, instantiating the placeholders in the provenance from step 4 for each individual data item.
6. Evaluate the process by exposing the PROV in a sample of the data.

The rest of this chapter is structured as follows: we briefly discuss related work in Section 3.2, after which we apply our generic workflow to three distinct real-world use cases. These use cases are: version control systems in Section 3.3, learning experiences in Section 3.4, and data mapping in Section 3.5.

3.2 Related Work

We observe two broad categories of work related to exposing interoperable provenance. The first category includes approaches that **map other data models** to an interoperable model such as W3C PROV. The second category includes the systems that have mechanisms built in to **expose PROV as part of their workflow**.

For the first category, a number of other initiatives exist to map non-standard provenance to W3C PROV. Notable examples include: a Dublin Core Mapping to PROV [84], the mapping of the revision history of Wikipedia to OPM (an ancestor of PROV) [154], and a mapping from PROV to Datalog [143]. Additionally, Sharma et al. [121] exposed bibliographical data in the MARC 21 format as Linked Data, including provenance. Our first and second use case could also be classified under this category, since they map W3C PROV from the provenance already available in version control systems and learning experience logs, respectively.

The second category is applicable to our third use case, where we expose the provenance of a data mapping workflow. A number of other systems with similar mechanisms built in are available in literature. For example, the DEEP system [191] exposes PROV for an executable document environment for scientific research. Curcin et al. [37] explain their recommendations on how to implement interoperable provenance using OPM and/or PROV in biomedical research. To expose geospatial data provenance, Yuan et al. [193] used the older Provenir [164] model, but claim that their approach is easily adapted to W3C PROV. Lagoze et al. [124] investigated how to expose the provenance metadata for social science datasets. Korolev et al. [119] used our Git2PROV [63] approach as a component in their PROB tool for tracking provenance of Big Data experiments. Finally, Sharma et al. [170] expanded their approach of publishing MARC 21 data to a full provenance tracking system for RDF resources, and evaluated it on Harvard Library Bibliographic Datasets.

Note that none of these approaches are in competition with our own, as they are restricted to different domains than our use cases. In fact, each of these initiatives – as well as our use cases – can be seen as a node in a *provenance ecosystem*, each contributing to the common goal of integrating provenance data into the Semantic Web. While the separate domains and use cases might seem rather disconnected from each other, their provenance is exposed in a common interoperable format. This allows new connections to be made that were not trivial to consider before. For example, learning experience logs could be linked to bibliographic data of the learning materials, source code history of educational software, or even changes on Wikipedia pages referred to by students. The combination of all this information – through PROV – potentially enables new insights into complex problems, such as why students are passing or failing certain classes that require use of particular books, software, or Wikipedia references at different points in time.

3.3 Use Case 1: Version Control Systems

As a first use case, we consider the provenance locked in a Version Control System (VCS). Because of their widespread use, VCSs offer a virtual gold mine in untapped provenance information. In this section, we provide an approach to expose this provenance information as W3C PROV, including a live demonstrator.

3.3.1 Introduction

VCSs have a long history in computing. The first such system was the Source Code Control System, developed in 1972 [161]. Nowadays, VCSs are widely popular and becoming more so with the advent of cloud-based services, such as Github[1] and Bitbucket[2], that both simplify the management of the VCS and expose their information through Web interfaces. For example, at the time of writing, Github has over 12 million users and maintains over 31 million projects. Github is built around the immensely popular VCS Git. Git is an example of a distributed VCS. This means that each user has a full copy of the entire repository (including all previous versions) at all times. When a user makes a change, he or she *commits* that change locally, and merges his or her copy of the repository with the others by *pushing/pulling* changes to/from a remote repository, which acts as a synchronization point for all the users involved. As a more detailed explanation of Git is out of scope for this thesis, we refer to [132] for an overview.

In essence, versioning of data is an aspect of *provenance*. Thus, the aim of the system described in this use case is to enable the provenance within a VCS to be exposed in a *Web-native and interoperable format*, such as W3C PROV. This provenance can then be consumed by other PROV enabled tools. The potential impact of this is immense, not only because all 31 million projects on Github can now be exposed as PROV, but also considering that many components in the Linked Data publishing chain are maintained using a VCS as well. Indeed, given that the software used to create many Linked Data sets is available through public version control systems[3], we believe that the tool can be used to enrich the provenance of many of these datasets.

In the rest of this section, we briefly discuss related work and present a mapping from the Git version control system to PROV. This is followed by a description of the implementation and demonstration of our system. In particular, we illustrate how the resulting information can be consumed by other PROV enabled systems. The demonstration (live and video) is available at the following URI: `http://git2prov.org`.

[1] `http://www.github.com`

[2] `http://www.bitbucket.com`

[3] e.g., `https://github.com/dbpedia` or `https://github.com/jimmccusker/twc-healthdata`

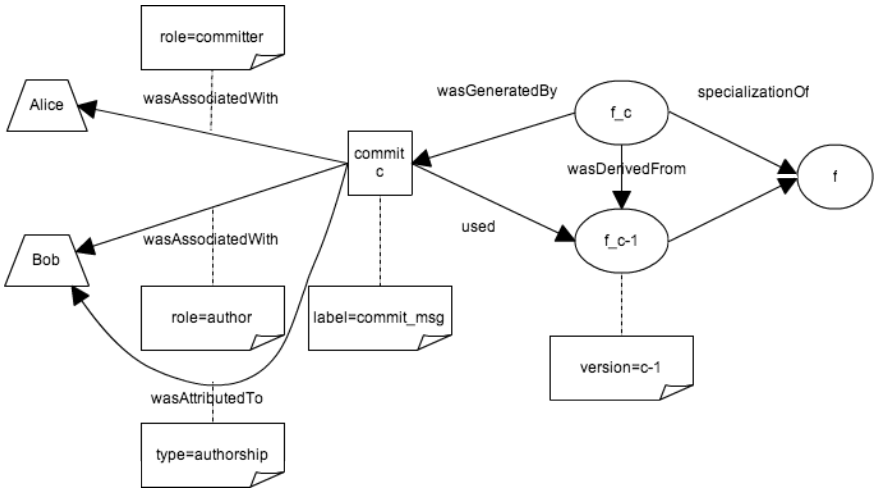

Figure 3.1: Mapping of Git operations to PROV concepts.

3.3.2 A Mapping from VCS to PROV

Our mapping was created by identifying whether the data could represent information about entities, activities and agents, as specified in Section 3.1. We then identified three classes of relations that interconnect these concepts. The three classes we used are identified below. For each class, we describe how provenance can be expressed using concepts from the PROV Data Model.

- Dependency - a dependency between two objects expressed as the relationship between two prov:Entity objects using `prov:wasDerivedFrom` and `prov:specializationOf`. For example, if a file f_c was derived from another previous file f_{c-1}, both are a specialization of a certain file f;

- Activities - a process expressed as a `prov:Activity` that connects two `prov:Entity` objects, expressed through the relations `prov:used` and `prov:wasGeneratedBy`. For example, a commit c uses a file f_{c-1} and generates a file f_c;

- Attribution - attribution information expressed as the `prov:Agent` that created a `prov:Entity` using the relations `prov:wasAttributedTo` and `prov:wasAssociatedWith`, modeling the two potentially distinct roles of an author and a committer.

Note that these classes reflect the three use-case perspectives on provenance identified by the W3C Provenance Primer [90]: *object-oriented*, *process-oriented* and *agent-oriented*.

The mapping is shown in Fig. 3.1. Note that the activity *start* and *end* concepts of PROV are not depicted, and correspond to, respectively, the author time and the commit time of each commit.

3.3.3 Implementation

Because we want our conversion tool to be as flexible as possible, we chose to build a Web service for this purpose. This service is directly available at the following URL: `http://git2prov.org/git2prov?giturl=<your_git_url>`.

The only required input for this service is a URL **giturl** that refers to a Git repository. In this proof-of-concept implementation, only openly accessible repositories are supported. However, adding support for secure repositories is a matter of implementing an authentication and authorization layer, which does not affect the basic principle of the mapping. In addition to **giturl**, the service accepts a number of optional parameters, with the default value in bold:

serialization *(possible values: [**PROV-N**, PROV-O, PROV-JSON])* This parameter is used to specify the desired PROV serialization.

shortHashes *(possible values: [**false**, true])* This parameter forces the service to use the short commit hash in the exported provenance, to increase readability for human users

ignore *(possible values: a provenance relation)* This parameter is used to filter the specified relation from the converted provenance.

Note that each provenance document generated by the Git2PROV service includes a link to the complete document (without any restricting parameters).

Upon receiving a request, the service *clones* the Git repository to a temporary location, and performs a `git log` command on it, which lists the entire version history of the repository. Using the output of this log, all files that are or were once present in the repository are identified. A second `git log` command is then performed to retrieve the full revision history for each of these files. The output of the file-specific logs is then *mapped* to PROV as described in Sect. 3.3.2, and written to the HTTP response in the requested serialization.

To illustrate our approach, a public demonstrator has been made available at `http://git2prov.org`, as illustrated by Figure 3.2. A video with instructions on how to use it is also available at `http://vimeo.com/70980809`. The full source code has been made available as well at `https://github.com/mmlab/Git2PROV`, under a GPLv3 open source license.

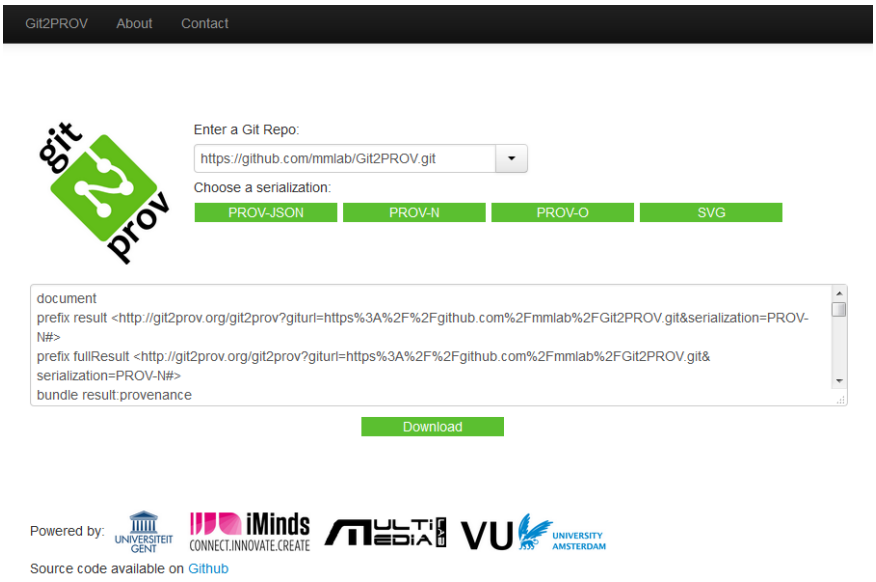

Figure 3.2: Screenshot of the Git2PROV demonstrator.

3.3.4 Impact and Future Work

We believe that systems such as Git2PROV have the potential to become an important enabler of the widespread interchange of standardized provenance. With our proof-of-concept implementation, we have shown that it is feasible to build a lightweight Web service to convert versioning systems into PROV. This is confirmed by T. Lebo's hg2PROV[4] and svn2PROV[5] implementations. These tools apply the mapping in Section 3.3.2 for two additional version control systems: Mercurial[6] and Subversion[7], respectively.

Furthermore, Git2PROV has been applied in several use cases since its release. For example, it was integrated into a workflow to automatically generate a time-consistent Web API [184], providing an interface to specific versions of Web resources through the Memento framework [182]. As mentioned in Section 3.2, Git2PROV was also used as a component for a system tracking the provenance of Big Data experiments [119]. In future work, we aim to improve our work by including more semantic annotations in combination with the provenance to allow further reasoning over it, with the prospect of deriving trust assessments.

[4] https://github.com/timrdf/pvcs/wiki/hg2prov
[5] https://github.com/timrdf/pvcs/wiki/svn2prov
[6] https://www.mercurial-scm.org/
[7] https://subversion.apache.org/

3.4 Use Case 2: Learning Experiences

As a second use case, we look at the domain of *e-learning*. Specifically, we consider e-learning platforms and applications that offer interactive exercises to students, and store the results. In many cases, these e-learning system do not only store a result by itself, but also contextual information on *how* the student achieved that result (e.g., how long it took to complete the exercise, which information sources were used, which difficulty level was enabled, etc.). In other words, these e-learning systems log part of the *learning process*. When a learning process is logged, this log describes which resources, which actions, and which people were involved in producing a certain result. In other words, this log can be seen as being part of the *provenance* of a learning process.

Knowing this, we could investigate all the aspects of logging learning processes, and create a data model based on PROV. However, a significant effort has already been made in this field, namely by the Advanced Distributed Learning (ADL) organization, in the form of the Experience API (xAPI) [179] (also referred to as the Tin Can API), a specification to structure experience logs in the JSON format. In its most basic form, an xAPI statement corresponds to the sentence “I did this, and it resulted in that". In the xAPI, the “I" is modeled as an *actor*, the “did" as a *verb*, the “this" as an *object*, and the “that" as a *result*. Apart from these basic concepts, various pieces of context information can be added to each xAPI statement. The xAPI is already widely adopted by organizations in the educational field[8].

Instead of re-inventing the wheel, we specify a conversion approach between the xAPI and W3C PROV. The approach consists of the following components, each signifying a contribution on their own: (1) an OWL ontology of the xAPI vocabulary, (2) a context document to interpret xAPI statements as JSON-LD [173], (3) a mapping to convert xAPI JSON-LD statements into PROV, and (4) a tool implementing this mapping. This way, developers are offered a choice in technology and serialization when it comes to logging, and the resulting Linked Data is more easily published in a scalable way and made interoperable with other provenance repositories.

The rest of this section is structured as follows: first, we discuss the context of this use case and its related work. Next, we provide a general overview of our approach, after which we describe each of the aforementioned components in detail and provide a link to an online demonstrator. Finally, we evaluate the approach before concluding with a brief discussion and outlook to future work.

[8] http://tincanapi.com/adopters/

3.4.1 Context and Related Work

The merit of interoperable provenance in the field of education has already been illustrated in literature. For example, it has been shown to help instructors to be more effective and to improve the learning experience [41]. We argue that it can provide teachers and students with an unseen amount of valuable information about the learning process. For example, the speed and continuity at which students complete a task – intermittent or all at once – may already indicate a need to revise the task. If information such as that could be linked to the lineage of the study material itself, it would become possible to observe the direct effect of changes in the material on the learning experience. The possibilities become even greater when also taking into account the provenance of the teaching staff (e.g., teachers leaving/joining), the inventory of the IT infrastructure (e.g., the acquisition of a new device), etc. Connections that would never be apparent upon first glance would appear automatically, all because the provenance of all these aspects is made interoperable.

Unfortunately, current models to track learning processes are often designed with one particular use case in mind, and their data is siloed (often for good reasons, such as privacy). For example, Yeh et. al. [192] built an e-learning system that keeps learning records such as grades, reading time, login times, and online discussions. The purpose of their system was to measure the effect of blended e-learning. Similarly, the authors of [109] measure patterns in a Web 2.0 learning environment.

A more comprehensive approach was proposed by Mazza et al. [137] in the form of *MOCLog*, a tool to analyze and present log data on a server running *Moodle*, an open-source PHP-based learning management system. While the rationale behind their approach is similar to ours – namely that all data that can be logged has potential value for analysis –, their system is catered towards one specific technology. This prevents other sources of external information to be interlinked with the logged data. In fact, mapping the MOCLog data to PROV might be an interesting case for future research efforts.

For a more extensive review of current student monitoring technologies, we refer to Corbi & Burgos [36], who provide insights on standards such as the *Caliper* framework by IMS [114], IEEE standard 1484.11.1/2 [113], JSON Activity Streams [2], and the xAPI.

Of all the learning process monitoring technologies mentioned above, Tin Can seems to be the most developer-friendly, which explains its wide adoption by the industry. Therefore, exposing its data in a complementary way, by mapping it reversibly to an interoperable model is a logical step.

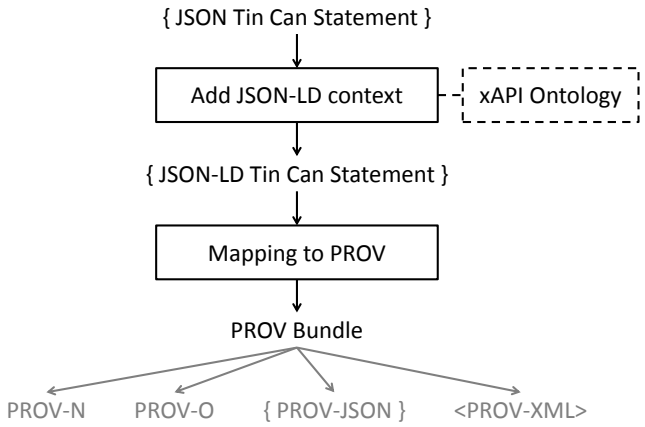

Figure 3.3: High-level overview of TinCan2PROV.

3.4.2 Approach

Figure 3.3 provides a high-level overview of our approach. The workflow starts with a Tin Can statement in the JSON format, which needs to be converted to PROV. We could just map every Tin Can property to a corresponding PROV concept. However, to allow for the mapping process to be reversed (i.e., making it possible to convert the provenance back to Tin Can), this would require an annotation in each PROV statement, indicating the original Tin Can property. While this is easily achieved by introducing an optional attribute (for example, `tincan2prov:property='actor'`), there is a more elegant solution.

This solution consists of first converting the Tin Can statement into proper Linked Data. The most straightforward approach to do this is by providing a *JSON-LD context*[9] as explained in Section 3.4.4, mapping each term in the Tin Can statement to a IRI (Internationalized Resource Identifier) describing that term. This allows the original JSON object representing the Tin Can statement to remain unchanged, while providing us with the identifiers (IRIs) necessary to map the statement to PROV. This way, an xAPI actor object mapped to a PROV Agent can be associated with both types, with no need to introduce extra attributes.

Unfortunately, the IRIs provided by the ADL organization for the basic Tin Can terms point to PDF and GitHub URIs, making them not machine-interpretable. Ideally, the IRIs should be dereferenceable to a human-readable (e.g., HTML) or machine-interpretable (e.g., OWL) representation, depending on which type is requested. As this is currently not the case, we created our own instance of the xAPI ontology created by ADL, to be referred to from the JSON-LD context. This

[9] `http://www.w3.org/TR/json-ld/#the-context`

is described in detail in Section 3.4.3. If ADL would host its own instance of such an ontology in the future, the IRIs could easily be adapted[10].

Once the JSON-LD context is in place, each concept in the xAPI ontology is then mapped to its corresponding PROV representation, as explained in Section 3.4.5. Finally, this representation is serialized in one of the PROV serializations, as described in Section 3.4.6.

3.4.3 xAPI Ontology

At the time of writing, the official specification of the xAPI is hosted in two places: one PDF document [179] specifying version 1.0.1 and one GitHub repository[11] where the ongoing development is managed. Unfortunately, neither of these provide a machine-interpretable version of the xAPI, leaving their IRIs unsuitable to be used as Linked Data.

The verbs and activities vocabularies are specified in a better way. Possible values for the term `verb` suggested by ADL[12] are listed at `http://www.adlnet.gov/expapi/verbs/`. Analogously, possible values for `activity` suggested by ADL are listed at `http://www.adlnet.gov/expapi/activities/`. Each verb and activity has its own IRI, dereferenceable to a (human-readable) description of the concept. No machine-interpretable description is provided at this IRI at the time this thesis was written. However, in July 2015, the W3C Experience API (xAPI) Vocabulary & Semantic Interoperability Community Group started to create a semantic vocabulary for the xAPI. To allow for our proposed workflow to be executed until this ontology is completed, we created a temporary formal version of the xAPI ontology as specified by ADL. Specifically, we hosted our own version of the specification, in a human- and machine-interpretable way.

Our formal ontology corresponds for the most part to the official xAPI specification. We constructed it by going trough sections 4.0 and 5.0 of the xAPI document on ADL's GitHub repository, and creating an OWL ontology following two simple rules. First, whenever an `objectType` was encountered, a corresponding `owl:Class` was created and – if applicable – linked to its superclass by `rdfs:subClassOf`. Second, whenever a property was encountered, a corresponding `owl:ObjectProperty` was created. In both cases the value of the `rdfs:isDefinedBy` property was set to the IRI of the xAPI.md document on GitHub (followed by a #), and `rdfs:label` was set to the name of the `objectType`. Finally, every activity listed at `http://www.adlnet.gov/expapi/activities/` was made a subclass of `:ActivityType`. An example of a simple xAPI statement, modeled in the ontology is illustrated in Figure 3.4.

[10] In fact, shortly after starting this work, we became involved in the W3C Experience API (xAPI) Vocabulary & Semantic Interoperability Community Group, whose goal is creating such an ontology.

[11] `http://GitHub.com/adlnet/xAPI-Spec/`

[12] Note that other, user-specified values are possible as well.

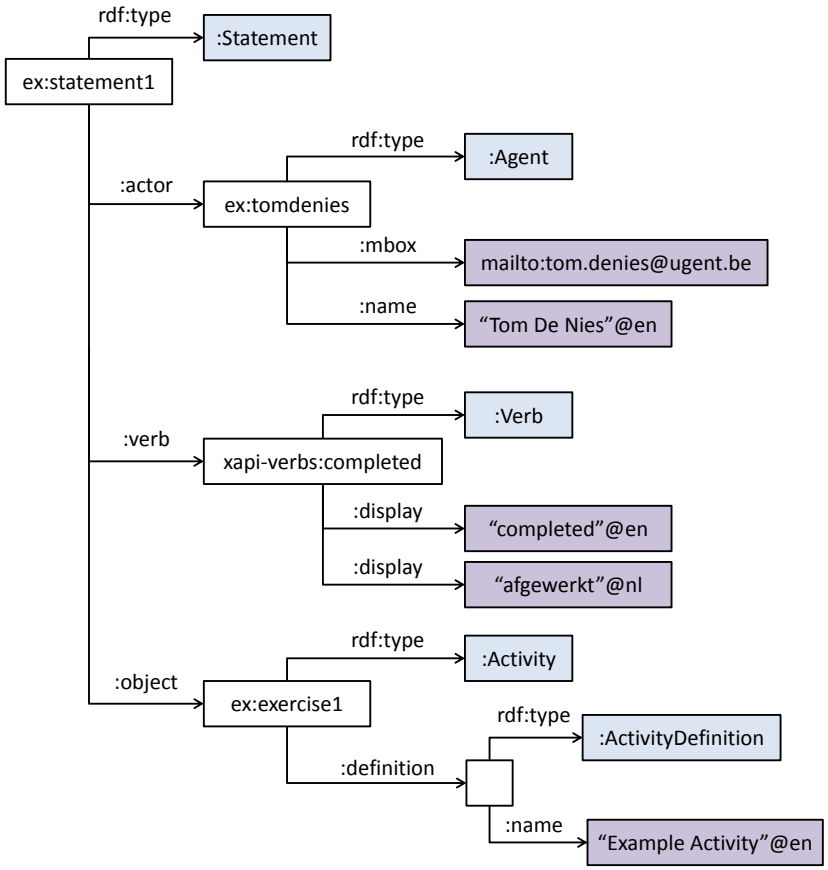

Figure 3.4: Example of a simple statement in the xAPI ontology.

In a number of cases, an extra class was created to support the modeling of more complex objects that are not supported by default in OWL or RDF Schema. For example, the range of the `:display`, `:name` and `:description` properties includes a `:LanguageMap`. We used a similar approach to model the `:extensions` property.

We made a full description of the ontology available at `http://semweb.datasciencelab.be/ns/tincan2prov/`, which is abbreviated using the prefix `xapi:` throughout the rest of this section. When navigating to this ontology with a browser, an HTML representation of the ontology will be shown. However, when an RDF media type[13] is specified in the `Accept` header of the HTTP request for the same IRI, an RDF (OWL) description will be returned. Ideally, such an

[13] `http://www.w3.org/2008/01/rdf-media-types`

ontology should be hosted at the ADL organization itself in the future, for example at `http://www.adlnet.gov/expapi/`.

3.4.4 Adding JSON-LD Context

In order to convert a JSON document to JSON-LD, we have to design and specify a JSON-LD context (`@context`). Such a context document maps all terms that may occur in a document to their corresponding IRIs in the ontology. Our JSON-LD context document is available at `http://semweb.datasciencelab.be/ns/tincan2prov/tincan2prov.jsonld`. To convert a Tin Can JSON statement to Linked Data, a `@context` entry referencing this document is added to the root of the JSON, as well as an `@type` entry with value `xapi:Statement`. Furthermore, the following snippet is added to every `:verb` and `:object` property: `"@context": { "id": "@id"}`. This is illustrated in Example 3.1.

Example 3.1
xAPI Statement in JSON-LD

```
{   "@context": "http://semweb.datasciencelab.be/ns/
tincan2prov/tincan2prov.jsonld",
    "@type": "http://semweb.datasciencelab.be/ns/
tincan2prov/Statement",
    "actor": {
        "mbox": "mailto:tom.denies@ugent.be",
        "name": "Tom De Nies",
        "objectType": "Agent" },
    "verb": {
    "@context": { "id" : "@id" },
        "id": "xapi-verbs:completed",
        "display": { "en": "completed",
                     "nl": "afgewerkt"
        }
    },
    "object": {
        "@context": { "id" : "@id" },
        "id": "http://www.example.org/exercise1",
        "objectType": "Activity",
        "definition": {
            "name": { "en": "Example Activity" }
        }
    }
}
```

A few additional conventions are necessary to ensure a smooth conversion, the first of which regarding **language**. The xAPI conforms to RFC 5646 [157] language tags for internationalization, while JSON-LD conforms to the older RFC 4646 [156], which includes a slightly different set of languages. In other words, to avoid unexpected behavior, all language tags must be changed (if necessary) upon conversion to comply with RFC 4646. The default language in our context document is set to "`en`".

The second convention concerns **extensions** and **attachments**. The xAPI allows the addition of extra JSON maps as extension to the vocabulary. However, since the keys of these maps are unknown, it is impossible for us to define a proper JSON-LD context for them. Therefore, when extensions are used, developers wishing to convert their xAPI statements to JSON-LD must provide this context themselves. In our ontology, we provided the generic `:Extension` class, described by the properties `:key` and `:value`, which could be used in such a context document.

Example 3.2 shows what happens when the statement from Example 3.1 is converted to an RDF notation such as Turtle[14].

Example 3.2
The same xAPI Statement in Turtle

```
[] xapi:actor [
     a xapi:Agent;
     xapi:name "Tom De Nies"@en;
     foaf:mbox <mailto:tom.denies@ugent.be>
 ];
   xapi:verb  xapi-verbs:completed ;
   xapi:object
     <http://www.example.org/exercise1> .

xapi-verbs:completed
   xapi:display "completed"@en ,
                        "afgewerkt"@nl .
<http://www.example.org/exercise1>
   a xapi:Activity ;
   xapi:definition [
       xapi:name
           "Example Activity"@en
 ] .
```

[14] Prefixes omitted for clarity.

Statement property	condition/property	Action taken | PROV concept asserted
:actor		prov:Agent <value of :verb> prov:wasAssociatedWith <this prov:Agent>
	:name	prov:label
	:member	prov:hadMember
:verb		prov:Activity <this prov:Activity> prov:used <value of :object>
	:display	prov:label with value for every language
:object		prov:Entity
	:name	prov:label with value for every language
	:type	rdf:type
:result		prov:Entity <this prov:Entity> prov:wasGeneratedBy <value of :verb>
:score		prov:Entity
:context		prov:Entity <value of :verb> prov:used <this :Context>
	:statement	<root statement id> prov:wasInfluencedBy <this :Statement>
:contextActivities		prov:Collection with all :Activity objects below as prov:hadMember.
	:parent	<value of :context> prov:wasInfluencedBy <this :Activity>, with prov:label="Parent"
	:grouping	<value of :context> prov:wasInfluencedBy <this :Activity>, with prov:label="Grouping"
	:category	<value of :context> prov:wasInfluencedBy <this :Activity>, with prov:label="Category"
	:other	<value of :context> prov:wasInfluencedBy <this :Activity>, with prov:label="Other"
:timestamp		<value of :verb> prov:qualifiedStart <prov:Start with same time>
:stored		prov:wasGeneratedBy
:authority		<this value> rdf:type prov:Agent <statement id> prov:wasAttributedTo <this prov:Agent>
:attachments	:Attachment	prov:Entity
	:display	prov:label with value for every language

Table 3.1: Actions taken and PROV concepts asserted for each observed property of a xAPI statement. In all cases, any remaining properties are kept as attribute-value pairs to the corresponding PROV concept.

3.4.5 Mapping xAPI to PROV

In this section, we describe a mapping between our formal instance of the xAPI ontology, and the PROV Ontology (PROV-O) [126]. By doing this, we are effectively mapping every Tin Can concept to a PROV concept.

We start from an RDF representation of an xAPI statement, obtained by following the steps described in Section 3.4.4. For each :Statement, a bundle is created using named graphs denoted using the TriG [22] syntax, as specified in PROV-Links [148]. This bundle will contain all triples for this statement, including those created during the JSON-LD conversion.

Then, PROV concepts are inferred and asserted for each property of the statement. The details of all the inferred PROV concepts are listed in Table 3.1. Note that during the JSON-LD conversion, class instances[15] are created as the values for the properties :actor, :verb, :object, :result, :context, :attachments, and :contextActivities, respectively. The remaining properties that do not map to any PROV concepts are kept as they are, and will be asserted as attribute-value pairs in serializations other than RDF.

[15] :Actor, :Verb, :Activity or :(Sub)Statement, :Result, :Context, :Attachment, and :ContextActivitiesObject

The result is an RDF document of mixed PROV-O and xAPI ontology concepts, which conforms to the PROV Data model. As explained in Section 3.4.6, it is now possible to translate this document into one of the other PROV serializations. Figure 3.5 shows a simplified version of such a provenance graph, representing the same xAPI statement as in Figure 3.4.

3.4.6 Serialization

As described in Section 3.4.5, we restrict our implementation of the mapping to the RDF (PROV-O) serialization. For the other serializations, we refer to the excellent ProvTranslator[16] by the University of Southampton, which – at the time of this thesis – supports PROV-N, PROV-O, PROV-JSON, PROV-XML, Turtle, TriG, and SVG.

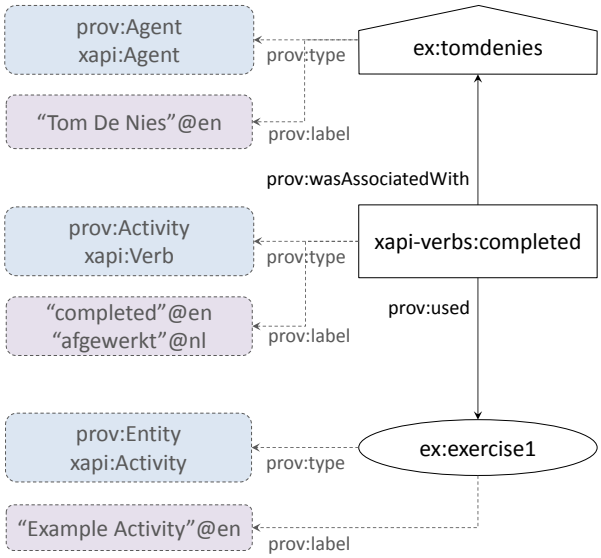

Figure 3.5: Example of an xAPI statement converted to PROV.

3.4.7 Demonstrator

An online demonstrator of the workflow described above is available at `http://tincan2prov.org`. The demonstrator, as illustrated in Figure 3.6, provides a form where a user can enter a Tin Can statement in JSON format, which – upon submission – is then converted to JSON-LD, RDF, and PROV-O. As the JSON-LD

[16] `https://provenance.ecs.soton.ac.uk/`

to RDF conversion process was not our primary focus, we relied on the jsonld[17] and n3[18] libraries for Node JS for this step. At the time of writing, advanced features such as extensions and attachments are not yet fully supported due to the arbitrary nature of their properties. This remains as a challenge for future work. All updates regarding the ongoing development and improvements are published at the same URI.

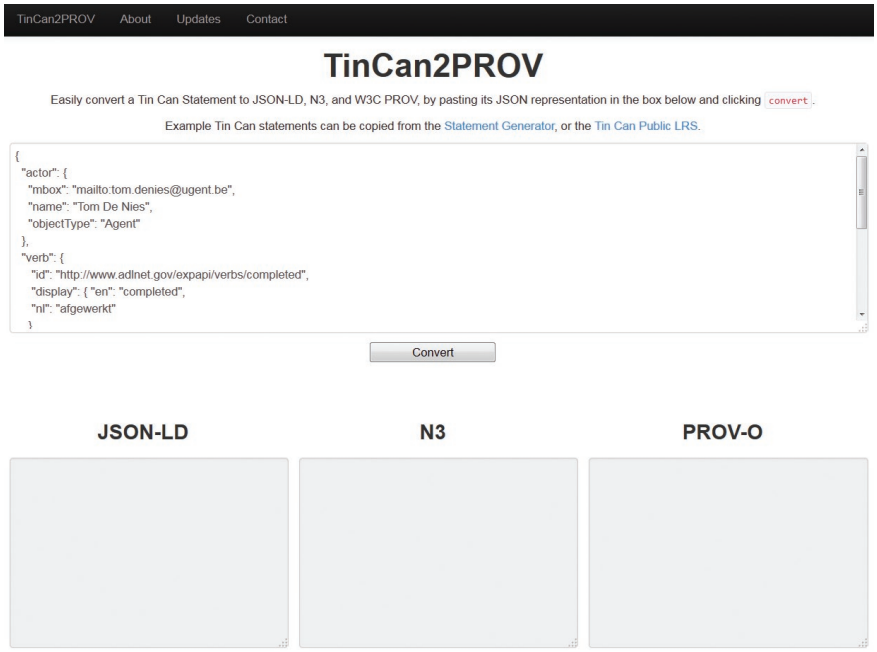

Figure 3.6: The user interface of the TinCan2PROV demonstrator.

3.4.8 Evaluation

A mapping can be deemed successful if it converts data from one representation to another, without losing any information. In this use case, we introduced two separate mappings. On the one hand, we introduced a workflow to convert Tin Can statements to Linked Data using the xAPI ontology. On the other hand, we created a mapping between this xAPI ontology and W3C PROV. It's important to keep this distinction in mind when interpreting the evaluation results.

[17] `https://www.npmjs.com/package/jsonld`
[18] `https://www.npmjs.com/package/n3`

We evaluated the mapping demonstrator by performing a limited empirical study. We copied 20 diverse statements[19] from the Tin Can Public LRS[20], and converted them first to JSON-LD, and then to PROV using the online demonstrator provided. Upon successful conversion, we then manually inspected each of the representations for loss of information. By 'loss of information', we mean that data present in one representation, can no longer be found in another representation.

The detailed data and results of the evaluation are provided at the following URL: `http://tincan2prov.org/evaluation.html`. On this page, the original Tin Can statements are listed as they were copied from the public LRS, as well as their JSON-LD form and their PROV-O form. Additionally, the PROV graph of successful results can be viewed, courtesy of the ProvTranslator. We provide a summary of the most important observations here.

During this preliminary evaluation, we discovered a number of technical challenges with regard to robustness to user error. For example, one statement did not convert from JSON to RDF, due to incorrect URL-encoding of an identifier in the original JSON statement, which means the mapping tool was not at fault. In another statement, the key `"ar-SA@calendar=gregorian"` was used in an attempt for internationalization. However, this does not result in a valid Language Map when converted to RDF. Therefore, these keys were filtered out during conversion to JSON-LD.

Converting the Tin Can RDF representation to PROV went smoothly. For the 19 statements that did successfully convert to N3, we observed **no loss of valid information** in the PROV-O representation. This means that while invalid elements (such as the aforementioned internationalization tags) are lost in the conversion, all other information could be expressed in Tin Can again, should this be needed.

3.4.9 Discussion and Future Work

With our mapping workflow, we have increased the interoperability of Tin Can, without sacrificing its information content. Even apart from the inferred PROV, the JSON-LD conversion step had merit on its own: after this step, Tin Can data can now be exposed as Linked Data (after anonymization). The release of a formal ontology by the W3C xAPI Community Group in the future will improve the situation even more. In their recently released Companion Specification for xAPI Vocabularies [6], ADL already provide a number of recommendations for representing xAPI vocabularies as Linked Data.

As for our own future work, we will continue the evaluation and development of the mapping tool to increase robustness, e.g., by providing a JSON-LD context for commonly used extensions and attachments.

[19] After 20 statements, it became increasingly difficult to find more statements on the public LRS with enough difference in content and structure compared to those already included.

[20] `http://tincanapi.com/public-lrs/`

3.5 Use Case 3: Mapping Refinements

Our third and final use case deals with exposing the provenance of a data mapping process, more specifically of its quality assessment and refinement steps.

The ever increasing adoption of Linked Data caused data owners to look for ways to efficiently publish their data on the Web. However, as in most cases this data is unstructured or semi-structured, a mapping process has to be applied to obtain their semantically enhanced representation. Even though mapping data to the RDF data model and publishing them as Linked Data is mainly performed by data owners, it is up to the data consumers to assess the quality of the datasets they consider to use, and decide whether they trust the dataset for further use or not. Especially when dealing with a new version of the same dataset, it is important to inform the data consumer whether or not to trust the new version.

In most cases, Linked Data Quality Assessment (QA) is focused and applied to data that is already published and it is performed by each interested party whenever necessary. As a consequence, different data quality assessment solutions use different criteria and different forms to output their results. The lack of a common form to describe the output of the tests makes it hard to compare the quality assessments of a dataset, and thus, assess its trustworthiness.

In the case of data mapped to the RDF data model, the most crucial moment to assess them for their quality is after the data is mapped and before it is published. Then, the data publisher can better react to the quality violations of the dataset, as structural adjustments can still be easily applied, improving the RDF dataset eventually generated. Until recently, there were no attempts to systematically incorporate quality assessment in the mapping and publishing procedure and such information was not considered to be included among the dataset's metadata. However, we contributed to a new approach to automatically assess and refine mapping documents to improve dataset quality [71]. In fact, this approach was even shown to be more effective than assessing and refining the quality of a dataset directly. In this section, we expand upon this idea and incorporate the capture of provenance information during the quality and assessment process of mappings. By exposing this provenance, we are providing valuable metadata to help a data consumer interpret the relative trustworthiness of mapped data.

3.5.1 Related Work

Related work exposing the provenance of mapping data to RDF, or of its quality assessments and refinements in interoperable form is scarce. Relevant work includes Green et al. [97], who proposed a system to track the provenance of updates that propagate between peers, related by database schema mappings, and to filter updates based on trust conditions that consider this provenance. Furthermore, the TRAMP [91] system helps to understand the transformations performed in com-

plex schema mappings, through their provenance. However, both of these mostly focus on relational database schemas, and predate the W3C PROV standard. More recently, LinkLion [152], a repository for links – or mappings – between knowledge bases, exposes the PROV of their mappings.

3.5.2 Mapping Assessment and Refinement Workflow

In previous work, our lab contributed to the RML Validator [71], a uniform, iterative, incremental assessment and refinement workflow that produces a high-quality RDF dataset. The RML Validator is based on applying the RDFUnit validation framework [118] to mappings described with the RDF Mapping Language (RML).

RML [72] is an extension of R2RML [40], the W3C-recommended language for defining mappings of data in relational databases to RDF. However, RML also covers mappings from sources in different semi-structured formats, such as CSV and JSON. RML mapping definitions specify how structured input data can be represented in RDF. Sets of RML mapping definitions consist of so called *Triples Maps* which define how triples are generated.

RDFUnit [118] is a validation framework for RDF, inspired by the unit tests commonly applied in software development. In RDFUnit, the SPARQL language is used to define a set of data quality test cases for every vocabulary, ontology, dataset or application. This means that apart from generic, pre-defined test cases, users can create their own test cases as well. By using SPARQL, violations can be easily identified because they can be directly queried for.

RML Validator [71] incrementally assesses the quality of an RDF dataset, covering both the mappings and the dataset itself. The solution relies on mapping definitions specified with RML. Since RML mapping definitions are expressed as regular RDF documents, the RDFUnit validation framework can apply its test cases to RML mapping definitions in the same way as it would be applied to an RDF dataset. Thus, the same set of schema validation patterns normally applied to the RDF dataset is also applicable on the mapping definitions.

The RML Validator covers a set of quality assessment measures which are implemented with the workflow visualized in Figure 3.7. An initial mapping document could potentially generate error-prone RDF. Therefore, first, the RML mapping definitions are assessed against quality assessment measures. The violations identified during this Mapping Quality Assessment (MQA) are reported and are taken into consideration to refine the definitions. The MQA may be repeated until the mapping definitions can not be further refined. The refined mapping definitions are finally used to generate the RDF representation, applied to either a sample of

the data or the complete data. The generated RDF dataset is assessed, using the same quality assessment framework. The Dataset Quality Assessment (DQA), intertwined with the MQA or not, can also be repeated until a refined version of the mapping definitions is generated. The latter is then used to perform the actual mapping.

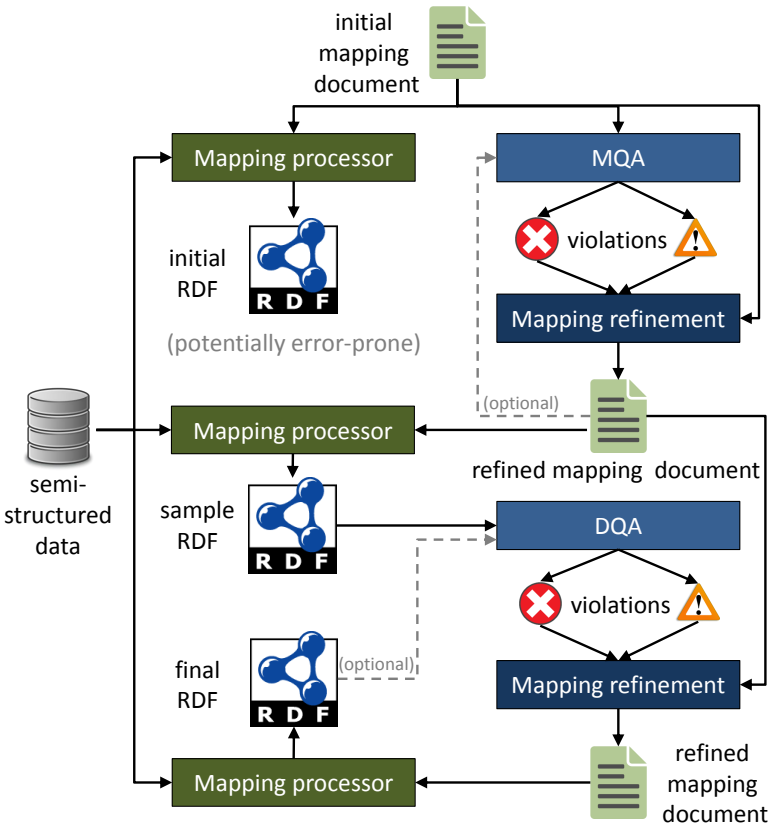

Figure 3.7: Visualization of the mapping assessment and refinement workflow.

3.5.3 Provenance of Mappings

There are several occasions in the workflow described in Section 3.5.2 where provenance may be logged. On a high level, the four stages are: A) mapping the original data to RDF using the original mapping document[21]; B) assessing and refining the quality of the mapping document on its own (MQA); C) assess-

[21] Note that in order to assess and refine a mapping, step A) does not actually need to be executed. However, it is essential to record the provenance of the data published using the original mapping.

ing and refining the mapping document quality even further through a data sample (DQA); D) mapping the data to new RDF using the improved mapping document. In Figure 3.8, we provide a general overview of the provenance logged during these four stages. The symbols used in Figure 3.8 correspond to those used in the W3C PROV specifications (ellipses for `prov:Entity`, and rectangles for `prov:Activity`, and directed arrows for relations between them). In PROV, the direction of the relations – and thus, the arrows in Figure 3.8 – is inverse to that of the actual workflow, which might seem counter-intuitive at first glance. Note that in a realistic scenario, the MQA in step B might be repeated a number of times to refine the mapping document optimally (two repetitions are shown here). Additionally, note that the original data does not change, only its RDF representations generated using the different mappings. By reasoning over the provenance of these RDF representations, a different level of trust (delta) may be assigned to each of them.

3.5.3.1 Provenance of Original Mapping Document

In Figure 3.8A, we see which provenance elements are generated when the original data is mapped without any quality assessment or refinement. In this step, the original data is retrieved and used by the mapping activity, which also uses a mapping document in order to generate RDF. This provenance corresponds to the following PROV-O triples:

```
:originalData a prov:Entity .
:dataRetrieval a prov:Activity ;
      prov:used :originalData .
:data a prov:Entity ;
      prov:wasGeneratedBy :dataRetrieval .
:mapDoc a prov:Entity .
:mapping a prov:Activity ;
      prov:used :data, :mapDoc .
:rdf a prov:Entity ;
      prov:wasGeneratedBy :mapping .
```

3.5.3.2 Provenance of Mappings Quality Assessment and Refinement

The next step is to assess and refine the quality of the aforementioned mapping document. The first MQA activity generates a number of violations. These are then used by the first refinement activity, which generates a new mapping document. To see whether this new mapping document actually represents an improvement over the old one, its quality is assessed again. This second MQA activity generates a new set of violations, which can then be compared to the previous ones, and used for a second refinement activity which generates a second new

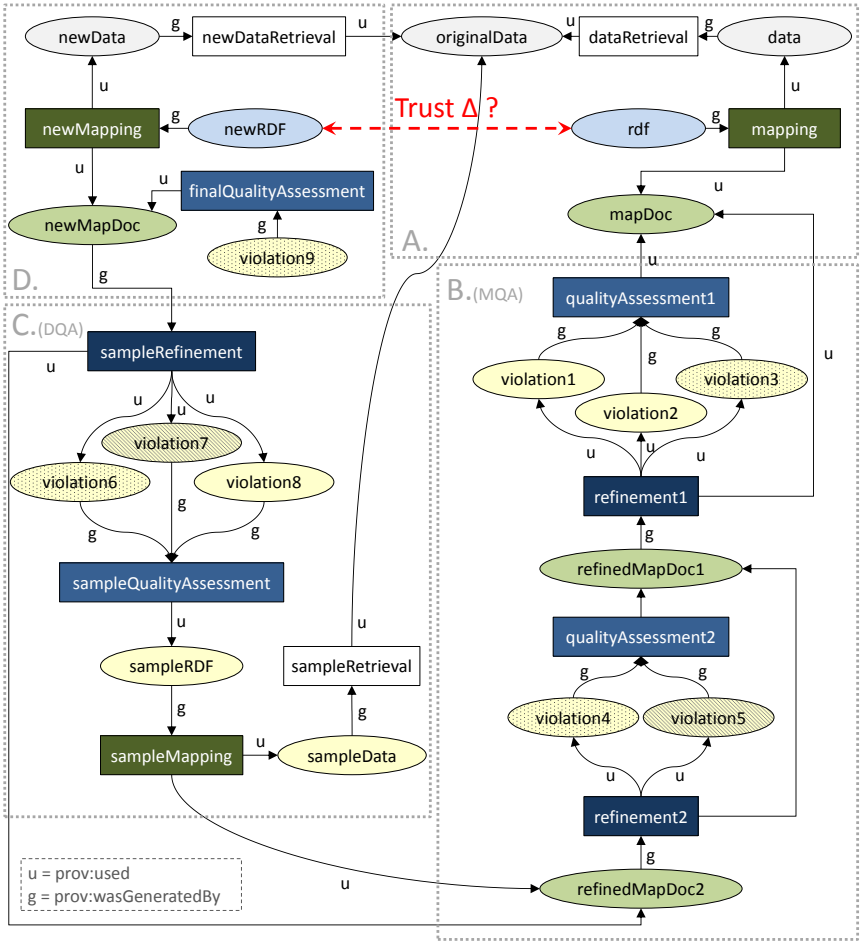

Figure 3.8: Overview of the provenance of data generated while using, assessing and refining a mapping document. The figure describes A. the normal mapping situation (without refinement); B. the quality assessment and refinement workflow of the mapping as such (MQA); C. the DQA of the mapping (through the mapping of a data sample); D. the mapping of new data using the final refined mapping document.

mapping document. This process is typically repeated until an optimal situation is achieved (e.g., when the violations remain the same). In Figure 3.8B, two such repetitions are shown. The PROV-O exposed by this process is:

```
:mapDoc a prov:Entity .
:qualityAssessment1 a prov:Activity ;
        prov:used :mapDoc .
:violation1 a prov:Entity, :violationTypeA ;
        prov:wasGeneratedBy :qualityAssessment1 .
:violation2 a prov:Entity, :violationTypeA ;
        prov:wasGeneratedBy :qualityAssessment1 .
:violation3 a prov:Entity, :violationTypeB ;
        prov:wasGeneratedBy :qualityAssessment1 .
:refinement1 a prov:Activity ;
        prov:used :mapDoc,
        :violation1, :violation2, :violation3 .
:refinedMapDoc1 a prov:Entity ;
        prov:wasGeneratedBy :refinement1 .

:qualityAssessment2 a prov:Activity ;
        prov:used :refinedMapDoc1 .
:violation4 a prov:Entity, :violationTypeB ;
        prov:wasGeneratedBy :qualityAssessment2 .
:violation5 a prov:Entity, :violationTypeC ;
        prov:wasGeneratedBy :qualityAssessment2 .
:refinement2 a prov:Activity ;
        prov:used :refinedMapDoc1,
        :violation4, :violation5.
:refinedMapDoc2 a prov:Entity ;
        prov:wasGeneratedBy :refinement2 .
```

Note that, apart from their `prov:Entity` type, each violation also has its own specific violation type. In Figure 3.8B, this is indicated by the background pattern of the entities. This gives us information about how effective the first refinement step was. For example, the figure shows that even though `refinement1` eliminated the violation type of `violation1` and `violation2`, the violation type of `violation3` is still present in `violation4` after the refinement, and a new violation type is even introduced through `violation5`. This means that the new mapping document has less violations than the old one, however we do not know if they are more or less severe. This information is important for the trust assessment of the refined mapping document, as further discussed in Section 3.5.4.

3.5.3.3 Provenance of Dataset Quality Assessment

The next step in the process is to retrieve a sample of the original data (or even the entire dataset), and use this in a sample mapping activity, together with the refined mapping document from step B. This process is illustrated by Figure 3.8C. The sample mapping activity generates a sample of RDF data, which is then used by a sample DQA activity. This DQA activity also generates violations, which are then used in a sample refinement activity of the mapping document, if possible. This generates a final, refined mapping document to be used in the final mapping step. The triples generated by this step are described by the following PROV-O:

```
:sampleRetrieval a prov:Activity ;
      prov:used :originalData .
:sampleData a prov:Entity ;
      prov:wasGeneratedBy :sampleRetrieval .
:sampleMapping a prov:Activity ;
      prov:used :sampleData, :refinedMapDoc2 .
:sampleRDF a prov:Entity ;
      prov:wasGeneratedBy :sampleMapping .

:sampleQualityAssessment a prov:Activity ;
        prov:used :sampleRDF .
:violation6 a prov:Entity, :violationTypeB ;
        prov:wasGeneratedBy :sampleQualityAssessment .
:violation7 a prov:Entity, :violationTypeC ;
        prov:wasGeneratedBy :sampleQualityAssessment .
:violation8 a prov:Entity, :violationTypeA ;
        prov:wasGeneratedBy :sampleQualityAssessment .
:sampleRefinement a prov:Activity ;
        prov:used :refinedMapDoc2,
                :violation6, :violation7, :violation8 .
:newMapDoc a prov:Entity ;
        prov:wasGeneratedBy :sampleRefinement .
```

3.5.3.4 Final Mapping

The final mapping, as shown in Figure 3.8D, is performed in the same way as described in Section 3.5.3.1, except that the new mapping activity uses the new mapping document instead of the original one to generate new RDF representation from the original data. Additionally, to ensure that we have a complete provenance trace, a final MQA step is performed on this final, refined mapping document to find out which violations remain. This is valuable information for the trust

interpretation, as described in Section 3.5.4. In other words, the PROV-O triples that are generated during this final step are:

```
:newDataRetrieval a prov:Activity ;
      prov:used :originalData .
:newData a prov:Entity ;
      prov:wasGeneratedBy :newDataRetrieval .
:finalQualityAssessment a prov:Activity ;
      prov:used :newMapDoc .
:violation9 a prov:Entity ;
      prov:wasGeneratedBy :finalQualityAssessment .
:newMapping a prov:Activity ;
      prov:used :newData, :newMapDoc .
:newRDF a prov:Entity ;
      prov:wasGeneratedBy :newMapping .
```

We now have all the provenance recorded to make a detailed assessment of the difference in trustworthiness between the RDF generated using the original mapping document, and the new RDF generated using the refined mapping document.

3.5.4 Trust Interpretation

When interoperable, machine-interpretable provenance of refined mappings is exposed as described in Section 3.5.3, this information can be used to give the consumer of the data generated using these mappings a number of valuable trust assessments. One way to achieve this is by creating reasoning rules or queries over the exposed provenance, combined with the semantic information available on the various violations. Instead of just providing a single, non-informative trust assessment score in the form of "dataset A is X% more trustworthy than dataset B", we argue for a more informative trust report, that allows the data consumer to weigh his or her options.

To illustrate this concept, we provide an example of such reasoning in two categories: *count-based* and *semantics-based*. Count-based rules or queries simply observe the number of refinements and violations to suggest a trust assessment to the consumer.

For example, to sort the RDF datasets based on their number of violations in the example provenance in Figure 3.8, we can use the following SPARQL query:

```
PREFIX : <http://example.org/>
PREFIX prov: <http://www.w3.org/ns/prov#>
SELECT ?rdf (COUNT(?violation) as ?violations) WHERE {
  ?rdf prov:wasGeneratedBy ?mapping .
  ?mapping prov:used ?mapDoc .
```

```
  ?mqa prov:used ?mapDoc .
  ?violation a :Violation .
  ?violation prov:wasGeneratedBy ?mqa .
} ORDER BY ASC(?violations)
```

For the example provenance in Figure 3.8, this would return the results listed in Table 3.2, which tell us that it is probably better to trust `:newRDF` than `:rdf`, since it has less violations. This result can then be further reasoned upon, for example by classifying the RDF datasets into levels of trustworthiness.

RDF	violations
http://example.org/newRDF	1
http://example.org/rdf	3

Table 3.2: Results for the example query to sort the RDF datasets based on their violation count in the example provenance in Figure 3.8.

However, it could be that in the above example, the type of `:violation9` is actually more severe than the violation types of `:violation1`, `:violation2` and `:violation3` combined. Therefore, we propose to also create semantics-based rules, which go deeper and report on the types of refinement performed, and the gravity of the violations (e.g., errors or warnings). To achieve this, a large part of the responsibility lies with the quality assessment approaches. In order to obtain meaningful trust assessments, the descriptions of the violations generated by quality assessment approaches need to be available in a machine-interpretable format, and semantically rich. There have been promising initiatives towards such a description: e.g., the `errorClassification` property of the Test-Driven Data Validation Ontology[22] associated with the RDFUnit system. However, at the time of writing, we do not have access to sufficient test data using such an ontology yet to investigate this approach.

3.5.5 Discussion and Future Work

In Section 3.5.4, we showed that it is definitely feasible to enable the inference of trust assessments by exposing the provenance of a mapping quality assessment and refinement workflow. However, a number of challenges remain. On the one hand, richer semantics are needed to describe the results (e.g., violations) of quality assessments, and the implications they have on the trustworthiness of generated data. On the other hand, a suitable ontology needs to be identified to represent the trust assessments that are the result of the interpretation process as described in Section 3.5.4. One option to be investigated in future work is to infer trust statements

[22] `http://rdfunit.aksw.org/ns/core`

modeled in the ontology from [29], which also incorporates the complexities of belief and warrants associated with trustworthiness.

A final challenge lies in evaluating our approach, since there are no approaches to compare with, and there are no benchmarks available that provide a ground truth as far as the provenance of a mapping refinement workflow is concerned. Therefore, the effectiveness of our approach will have to be measured through usage and user feedback. At the time of writing, we are in the process of integrating our approach into a graphical user interface for editing RML mapping documents, known as the RMLEditor [108]. This editor will include an option to visualize the provenance of a mapping refinement, as well as the trust assessments inferred from it as described in Section 3.5.4. A user study including expert and non-expert users is planned to evaluate the understandability and usefulness of these trust assessments, which will give us more insight into the effectiveness of our approach.

3.6 Interoperability Example

While the systems described in Sections 3.3, 3.4, and 3.5 each have merit on their own, their real value becomes apparent when they enrich each other through the interoperability of the provenance they expose. Here, we illustrate the interplay between Git2PROV and TinCan2PROV.

As an example, we take the online course materials for the Web Fundamentals module of the Ghent University course on Internet Technology, as provided by our colleague, dr. Ruben Verborgh. These materials are publicly available on GitHub: `https://github.com/RubenVerborgh/WebFundamentals.git`.

This means that not only the raw learning materials are available to students, but also the complete history of commits made to the repository. Using Git2PROV, we can expose this history as PROV-O[23]. This PROV-O trace would be much too large for a human to process, as it already contains more than 7000 lines at the time of writing. However, by exposing it as PROV, a machine can process it for us. As an example, let's focus on a small excerpt from the PROV-O:

Example 3.3

```
@prefix result: <http://git2prov.org/git2prov?giturl=https
%3A%2F%2Fgithub.com%2FRubenVerborgh%2FWebFundamentals.git&
serialization=PROV-O#>

result:file-linked-data-publishing-index-html
  a prov:Entity ;
  rdfs:label "linked-data-publishing/index.html"@en .
```

[23]The full PROV-O is available at `http://git2prov.org/git2prov?giturl=https\%3A\%2F\%2Fgithub.com\%2FRubenVerborgh\%2FWebFundamentals.git&serialization=PROV-O#`

This tells us that there is a file named *linked-data-publishingindex.html* in the repository. Upon examining the repository, we discover that this is a slide deck that teaches students about the principles of Linked Data Publishing (as the file name already suggests). The latest version of the slides can be downloaded directly from the GitHub repository at the following URL: `https://raw.githubusercontent.com/RubenVerborgh/WebFundamentals/gh-pages/linked-data-publishing/index.html`.

Furthermore, the PROV-O generated by Git2PROV also tells us about changes made to this slide deck, for example that a slide about master theses was added on March 23rd, 2016.

Example 3.4

```
result:file-linked-data-publishing-index-html_commit-
cfb6db55c330933da4d58a57057702db5dc72d08
 a prov:Entity ;
 prov:specializationOf
  result:file-linked-data-publishing-index-html ;
 prov:wasGeneratedBy
  result:commit-cfb6db55c330933da4d58a57057702db5dc72d08 .

result:commit-cfb6db55c330933da4d58a57057702db5dc72d08
 a prov:Activity ;
 prov:endedAtTime "2016-03-23T22:39:09.000Z"^^xsd:dateTime ;
 prov:wasAssociatedWith  result:user-Ruben-Verborgh ;
 rdfs:label  "Add slide about theses."@en .
```

For the sake of this example, we assume that the students of the Web Fundamentals module are using some form of e-learning system, that always pulls the latest version of the slides from the GitHub repository, and logs all learning activities using the Tin Can API. This means that if a student named Alice viewed the aforementioned slide deck on March 22nd, 2016, the following Tin Can statement was generated by the e-learning system:

Example 3.5

```
{
 "timestamp": "2016-03-22T13:00:00+01:00",
 "actor": {
   "mbox": "mailto:alice@example.com",
   "name": "Alice",
   "objectType": "Agent"
 },
 "verb": {
   "id": "http://adlnet.gov/expapi/verbs/experienced",
    "display": {
      "en-US": "experienced"
    }
 },
```

```
 "object": {
  "id": "https://raw.githubusercontent.com/RubenVerborgh/
WebFundamentals/gh-pages/linked-data-publishing/
  index.html",
  "definition": {
   "name": { "en-US": "Slide Deck" },
   "type": "http://id.tincanapi.com/activitytype/slide-deck"
  },
  "objectType": "Activity"
 }
}
```

Through TinCan2PROV, this translates to the following PROV-O:

Example 3.6

```
@prefix xapi-verbs: <http://adlnet.gov/expapi/verbs/> .
@prefix xapi: <http://semweb.datasciencelab.be/ns/
tincan2prov/> .
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

_:b0 a xapi:Statement;
 xapi:actor _:b1;
 xapi:verb xapi-verbs:experienced;
 xapi:object <https://raw.githubusercontent.com/RubenVerbo
rgh/WebFundamentals/gh-pages/linked-data-publishing/index.
html>;
 xapi:timestamp "2016-03-22T13:00:00+01:00"^^xsd:dateTime .

_:b1 a prov:Agent, xapi:Agent;
  foaf:mbox "mailto:alice@example.com"@en;
  prov:label "Alice"@en .

xapi-verbs:experienced xapi:display "experienced"@en-us;
    a prov:Activity;
    prov:wasAssociatedWith _:b1;
    prov:label "experienced"@en-us;
    prov:used <https://raw.githubusercontent.com/RubenVerb
orgh/WebFundamentals/gh-pages/linked-data-publishing/index
.html>;
   prov:qualifiedStart [
    a prov:Start;
    prov:atTime "2016-03-22T13:00:00+01:00"^^xsd:dateTime .
         ] .

<https://raw.githubusercontent.com/RubenVerborgh/WebFundam
entals/gh-pages/linked-data-publishing/index.html>
    a prov:Entity;
```

```
    xapi:definition [
     xapi:name "Slide Deck"@en-us;
     xapi:type "http://id.tincanapi.com/activitytype/slide
-deck"^^xapi:Activity .
    ] .
```

To a human, it is obvious that because Alice accessed the slides on March 22nd, she did not read the information about the master theses, which was added on March 23rd. To make this obvious to a machine, the only thing we need to do is to specify that the URI referring to the slide deck in the TinCan2PROV output and the URI referring to the slide deck in the Git2PROV output are actually referring to the same thing. As we saw in Section 1.2.1, this can be done using an `owl:sameAs` link.

Example 3.7

```
<https://raw.githubusercontent.com/RubenVerborgh/WebFundament
als/gh-pages/linked-data-publishing/index.html>
owl:sameAs result:file-linked-data-publishing-index-html .
```

This link could be generated by a reasoner that knows the structure of GitHub, based on the rdfs:label provided in Example 3.3, or by a future version of the Git2PROV tool, for example. Regardless of how it is generated, it allows any machine reasoning over these two provenance traces to view them as talking about the same thing, and take action based on this information.

In our example, a reasoner that is part of the e-learning system could monitor the provenance of all its learning resources through Git2PROV. It could then, for example, trigger an action after each commit, to notify any students that recently accessed the files that were modified in that commit that something has changed. In our case, this would mean that the e-learning system sends Alice a notification that the slide deck located at *linked-data-publishing/index.html* has been changed, with the commit message *"Add slide about theses."*. This already provides Alice with useful information, generated without any human intervention, based only on the PROV traces of her learning experiences and the course materials.

Naturally, this is only one simple example of the value of interoperable provenance shared between systems. Other, more complex examples can be found that illustrate the advantages of interoperable provenance generated by Git2PROV, TinCan2PROV, our RML refinement workflow, and other systems exposing PROV. In future work, we will focus on exploiting these advantages, and showcasing the interplay between our different use cases.

3.7 Conclusion

In this chapter, we have illustrated that provenance can be exposed as W3C PROV in a variety of use cases, by following a number of intuitive steps. Services such as these will be an essential step to promote the widespread use of PROV, especially by established systems. The added value is that now, the provenance of these systems is no longer locked in its own domain. This opens up the way to combine the PROV generated by all these systems, and create a distributed ecosystem of provenance-enabled applications. Such an ecosystem allows information consumers to retrieve as much detailed provenance as desired, even when it transcends their own domain. Assisted by automatic reasoners, such as the approach discussed in Chapter 5, this effectively enables a *"Web of Trust"* to be established across these domains.

Wisdom is not wisdom when it is derived from books alone.

Horace

4

Reconstructing Provenance

As interest in provenance grows among the Semantic Web community, it is recognized as useful information to collect in many domains. However, most existing provenance collection techniques either rely on (low-level) observed provenance, or require that the user discloses formal workflows. Additionally, provenance information on the Web is often missing or incomplete. In this chapter, we propose a new approach for automatic reconstruction of provenance, at multiple levels of granularity. To accomplish this, we detect possible entity derivations, relying on clustering algorithms and semantic similarity. While the proposed approach is purposely kept general, allowing adaptation in many use cases, we provide an implementation for two of these use cases, namely discovering the sources of news articles, and of messages on social media. Finally, we present a gold standard dataset to evaluate approaches such as ours.

This chapter is based on the following publications:

Tom De Nies, Sam Coppens, Davy Van Deursen, Erik Mannens, and Rik Van de Walle. Automatic discovery of high-level provenance using semantic similarity. In *Provenance and Annotation of Data and Processes – IPAW 2012*, pages 97–110. Springer, 2012

Tom De Nies, Io Taxidou, Anastasia Dimou, Ruben Verborgh, Peter M Fischer, Erik Mannens, and Rik Van de Walle. Towards multi-level provenance reconstruction of information diffusion on social media. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management*. ACM, 2015

4.1 Introduction

Nowadays, as interest in provenance grows among the Semantic Web community [89], media content authors are faced with a dilemma. While they clearly see the advantages of providing provenance information with their data, the process of manual annotation is labor intensive and dull work, especially for those without a technical background [95]. When provenance information is already present in a system, but obscured in a non-interoperable format, the proposed techniques from Chapter 3 can be used to expose it. However, in a large number of cases, the provenance is partially or completely missing. Clearly, there is a need for automated ways to add provenance to such existing content.

Another point to observe is the *granularity* of provenance produced in existing systems. Most automatic provenance collection techniques described in literature focus on either *observed provenance*, or *disclosed provenance* [197]. This means that these techniques either observe *all* activity on the target resources (observed provenance), or require that the users specify formal workflows which are used to create and modify the resources (disclosed provenance) . The first techniques often result in a low-level, very fine-grained view of the provenance associated with a resource, since these approaches do not necessarily understand the semantics of their observations. This is not always suitable for the provenance consumers, who can then become overwhelmed with unnecessary details (e.g., in the use cases described in this chapter) [26]. The latter approach requires significant effort from the user, and is not always applicable, since many creative processes are difficult, if not impossible, to formally describe.

In this chapter, we propose an approach for automatic discovery of provenance from limited information, at multiple levels of granularity. Whereas low-level provenance denotes the exact change at the finest granularity (e.g., at the character level), higher-level provenance denotes changes at a coarser granularity (e.g., at the document level). To achieve this, we detect inter-document derivations, using clustering methods based on semantic similarity, resulting in provenance complementary to the observed and disclosed kind. We apply the approach to two specific use cases: online news versioning and social media information diffusion. We attempt to reconstruct missing provenance, solely based on the content and timing information, allowing us to track down the original source of a document.

The rest of this chapter is structured as follows: first, we discuss related work. Next, we explain our view on the granularity levels of provenance, and how this is reflected in the PROV model. We then provide an in-depth explanation of the generic approach. In Sections 4.6 and 4.7, we describe our use case implementations. Finally, we propose a gold standard dataset to further evaluate approaches such as these in Section 4.8

4.2 Related Work

Although very few people in the world are working on this specific problem, a number of domain-specific techniques used to reconstruct lost or missing provenance information do exist. For example, Zhao et al. [195] predict missing provenance based on semantic associations in the domain of reservoir engineering. Zhang et al. [194] exploit the logging capabilities of existing relational database management systems to retrieve lost source provenance traces. The work of [128] focuses on tracing news and quotes (referred to as *memes*) on the Web over time. The focus is on temporal patterns, mutations (alterations) that online phrases undergo and properties of the life cycle of online news. A subsequent work using the same datasets and methods [172] shifts the focus on fine-grained content alterations. However, these techniques all predate the PROV standard and do not offer a generic solution such as ours.

The most recent related work on provenance reconstruction is by Aierken et al. [7]. They participated in the 2014 Provenance Reconstruction Challenge – which we hosted – with their multi-funneling approach to provenance reconstruction. More precisely, they apply three techniques: one based on *IR techniques and the Vector Space Model (VSM)* similar to our approach [55], one based on the *machine learning and topic modeling*, and one based on *dynamic programming and matching the longest common subsequence*. Like us, they achieve results that show promise for certain use cases, but still have room for improvement.

This brings us to the largest problem in this field: evaluation. There are little to no provenance datasets available that are suitable to evaluate reconstruction approaches. Magliacane and Groth [136] surveyed existing benchmark corpora that could be adapted for this purpose, but so far no one has implemented this yet. This is why we collaborated with them to create two *gold standard* datasets – one human-generated and one machine-collected – for the Provenance Reconstruction Challenge in 2014, as further explained in Section 4.8.

4.3 Provenance Granularity Levels

In the research described in this chapter, we make an important distinction between *low-level* and *high-level* provenance. What we call low-level provenance is the sort of provenance which is very *fine-grained*. For example, this type of provenance is expected from capturing systems and versioning systems. A typical example is that of a programmer's versioning system, where the provenance of each document is stored as a list of characters that where changed, together with their position in the document. High-level provenance is what we call more *coarse-grained* provenance, e.g., at the *document level*. An example of high-level provenance might be: "Document A is a revision of document B".

Apart from these two provenance types, an intermediary approach might be desirable in certain cases as well. For example, we might want to record: "Document A is a derivation of document B, with concept 'Magistrate' in document A narrowed down to 'Prosecutor' in document B". We will label this as provenance at the *semantic level*, providing more details than at the document level, but remaining high-level, at a coarser granularity than the typical low-level systems.

In an ideal scenario, different provenance granularity levels are combined in one dataset. When this is the case, the different levels of granularity can be seen as *views* on the provenance. The user can then choose to *zoom in* to get a more fine-grained, low-level view on the provenance, and *zoom out* to get a more coarse-grained, high-level view, with as many intermediary steps as is required by the use case.

4.4 Proposed Approach

In this section, we provide an in-depth description of how we aim to discover provenance derivations, using semantic similarity. We choose to follow this approach since it mimics the way humans look for the source of a piece of information (i.e., by looking for other, prior information that talks about the same subject). As outlined in Section 4.2, alternative approaches to reconstruct provenance after the fact are scarce. Those that do exist, mostly focus on detecting temporal patterns and/or matching (parts of) phrases. In other words, they also employ some form of similarity measure, albeit purely based on text. The exception to this are the approaches based on machine learning. However, these do not offer any transparency as to how the provenance was reconstructed, or to which degree it is certain. This is where we aim to differ, by providing a generic method for provenance reconstruction, allowing transparency in terms of the similarity measure used, as well as the degree of similarity, which in turn indicates a degree of certainty.

While we want to keep our approach as general as possible, it is necessary to make some assumptions about the data we will be providing provenance for. We will assume that the data essentially consists of two types of entities. We define a *document* as an entity that is characterized by multiple other entities, which we will refer to as *semantic properties*. Both documents and semantic properties can be modeled as a prov:Entity, and thus can be connected through activities and/or entity-entity relations. In our news use case, an example of a document would be a news article, whereas examples of semantic properties would be the descriptive metadata annotations of this article. We also assume that timing information (i.e., date of creation) is available for all documents.

The general goal of our research is to analyze documents to automatically reconstruct their provenance information. Since this is very general, we will narrow it down to 2 sub-goals. Starting from a set of documents S, we aim to:

1. Reconstruct high-level derivations at a *coarse granularity*.

2. Reconstruct additional, low-level derivations at a *finer granularity*.

We will make a distinction between *single-step* derivations, and *multi-step* derivations, defined as follows:

Definition 4.1
*A **single-step derivation** indicates that one document was derived from another directly, without any intermediary documents.*
*A **multi-step derivation** indicates that one document was derived from another, but there may or may not be a (possibly unknown) path of one or more intermediary single-step derivations between them.*

In earlier versions of the PROV-DM, this distinction could be modeled using the `prov:steps` attribute, with either *"single"* or *"any"* as value for single-step and multi-step derivations, respectively. However, this was removed from the final version of the data model because it was deemed too confusing for general users. Therefore, we have chosen to model these attributes ourselves in a mini-ontology "PROVR"[1], since they are still useful for our approach.

Below, we describe how we achieve the two sub-goals.

4.4.1 Reconstructing Coarse-grained Provenance

To discover provenance at the coarsest granularity, we rely on the semantic similarity of documents. Since it is safe to assume that derivations of the same document are semantically similar to each other, our hypothesis is that in many cases (but naturally, not always), the inverse also holds: *if documents are very similar to each other, it is likely that they are also (indirect) derivations of the same document*.

First, we group (or *cluster*) all semantically similar documents into clusters S_i, so that for all documents $doc_a \in S_i$:

$$doc_a \in S_i \Leftrightarrow \forall doc_b \in S_i : sim_D(doc_a, doc_b) > T_s \tag{4.1}$$

with T_s an empirically determined *similarity threshold*, and $i \in \{1, 2, ..., N\}$ with N the number of clusters[2]. Sim_D is a similarity metric, which enables semantic comparison of documents. Note that this similarity metric is interchangeable, and a more accurate similarity metric will result in better clustering (for example in our implementation of the news use case, semantic similarity of documents is based on the comparison of their semantic properties). To avoid clusters becoming too large, resulting in poor derivations, all clusters larger than a *clustering threshold*

[1] hosted at `http://semweb.datasciencelab.be/ns/provr#` – prefix `provr:`
[2] Note that overlap between clusters is possible.

T_c, can (optionally) be re-clustered with a higher similarity threshold T_s to achieve better accuracy.

Next, we order all documents in each cluster according to their date of creation. For each cluster, we assume that the document doc_1 that was created first is the original source of all other documents in the cluster[3]. This means that we can now connect each document of the cluster to doc_1 by a **multi-step** derivation, as illustrated by Fig. 4.1(a).

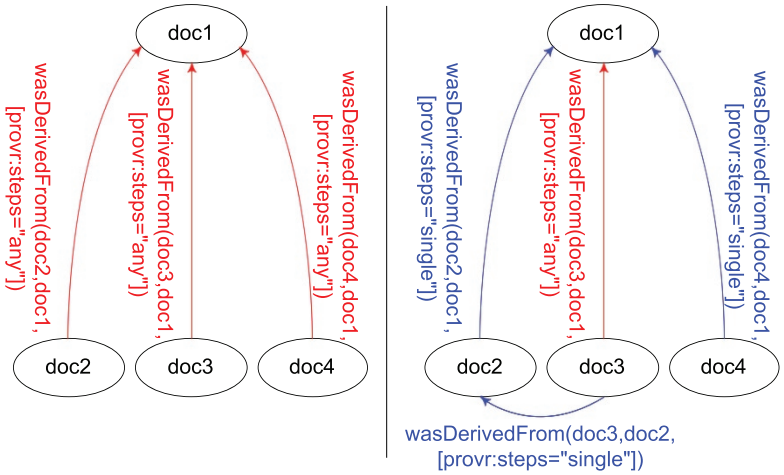

Figure 4.1: Example of how documents doc_2, doc_3 and doc_4 within one cluster are related (a) to the original source doc_1 by multi-step derivations, and (b) to each other by single-step derivations.

A significant challenge now lies in creating **single-step** derivations, since multiple revisions can be based on a single document, regardless of timing. Therefore, simply considering the timing and connecting successive documents with single-step derivations is not a correct approach. Our proposal to do this is to take both the inter-document similarity and timing information into account. In each set S_i, for each document $doc_a \in S_i \setminus \{doc_1\}$ (in other words, not the oldest document), we find the semantically most similar document $doc_b \in S_i$, for which Equation 4.2 holds.

$$\forall doc_k \in S_i \setminus \{doc_a\} : sim_D(doc_a, doc_b) \geq sim_D(doc_a, doc_k) \tag{4.2}$$

[3]Note that there is a degree of uncertainty here, since this assumption will not always be correct. This uncertainty could be annotated using the UP ontology we proposed in Section 2.3. For example, we could specify *up:MachineGenerated* as the *up:assertionType*, and the degree of similarity as the *up:assertionConfidence*.

We then connect doc_a and doc_b with a single-step derivation, of which the direction depends on which document was created first, as per Inference 4.3 and 4.4.

$$\begin{aligned} & time(doc_b) < time(doc_a) \\ \Rightarrow\ & wasDerivedFrom(doc_a, doc_b, [provr : steps = \text{“}single\text{”}]) \end{aligned} \tag{4.3}$$

$$\begin{aligned} & time(doc_b) > time(doc_a) \\ \Rightarrow\ & wasDerivedFrom(doc_b, doc_a, [provr : steps = \text{“}single\text{”}]) \end{aligned} \tag{4.4}$$

In Fig. 4.1(b), we apply this method to the example from Fig. 4.1(a). We assume that $time(doc_i) < time(doc_j) \Leftrightarrow i < j$. Here, doc_2 is most similar to doc_1, doc_3 most similar to doc_2 and doc_4 most similar to doc_1. Even though doc_4 was created after doc_3, it was directly derived from doc_1.

4.4.2 Reconstructing Finer-grained Provenance

In a finer-grained view, derivations can specify an *activity*, responsible for using the original entity, and generating the derived entity. Converting the derivations reconstructed using the approach in Section 4.4.1 to this form is done by defining a *revision* activity for each derivation, as illustrated by Fig. 4.2.

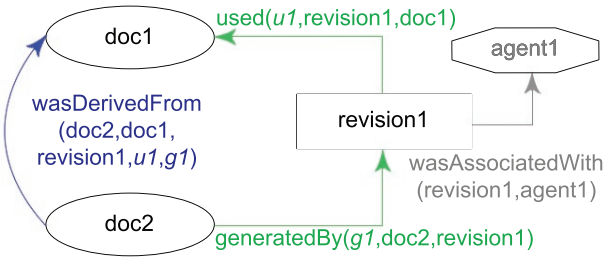

Figure 4.2: The fine-grained derivation of doc_2 from doc_1 specifying an activity $revision_1$, which uses doc_1 and generates doc_2, and is associated with an agent $agent_1$.

Specifying this activity enables us to model *responsibility* for the revision by specifying an agent, if available. In the best case scenario, this agent is found in the document's metadata, as the annotated author or editor. In the worst case, when no agent can be found, the provenance of the revision can still be asserted, without an agent. In other cases it might be possible to find the correct agent by querying other data sources and finding a matching document, with author information available. However, in this work, we will keep the focus on reconstructing derivations only.

To obtain provenance at an even finer granularity, we propose using the semantic properties characterizing the documents. As a document is revised, some of its semantic properties change, and others remain the same. Changes might imply *replacements*, *generalizations* or *specializations*. Some properties might be *omitted*

from the document, whereas new ones may be *added*. While it might be more desirable to use a separate ontology for this (e.g., for version control), these changes can technically be modeled using PROV-DM. We start from the coarse-grained provenance bundle associated with a set of related documents, as generated in the previous steps, and create a new, fine-grained bundle, enclosing it.

How the semantic properties of a document are identified is dependent on the type of data, and may vary for each use case. In our news use case in Section 4.6, this is achieved through named-entity recognition. Once the properties are identified, we link each of them to their document by a `prov:hadMember` relation.

Next, the properties of each document pair related by a single-step derivation are semantically compared. Once again, this comparison is dependent of the type of data and use case. However, it is important that the comparison can model **replacements**, **generalizations** and **specializations**. Additionally, we will model **additions** and **omissions**.

In PROV-DM, **replacements** or synonyms can be modeled by the *alternateOf* relation. The replaced property p_i is *used* by the revision activity, which *generates* the new property p_j. **Specialization** is modeled by the PROV-DM *specializationOf* relation. The more general property p_i is *used* by the revision activity, which *generates* the specialized property p_j. **Generalization** is modeled as an inverse specialization. Here, we model **Addition** by a revision activity that *generates* a property p_i, but does not use a replaced, specialized or generalized property. Similarly, we model **omission** by a revision activity that *uses* a property p_i, but does not generate a replacing, specializing or generalizing property[4].

As an example, we consider the coarse-grained bundle associated with two documents doc_1 and doc_2, as illustrated by Fig. 4.2. Suppose we were able to identify three properties p_1, p_2, p_3 of doc_1 and three properties p_3, p_4, p_5 of doc_2. Figure 4.3 shows the usage activities linking these properties with doc_1 and doc_2. When comparing the properties, it was discovered that p_4 is a specialized concept of p_2. This is modeled by the usage of p_2 and generation of p_4 by $revision_1$, and the specialization relation between p_4 and p_2. p_1 was omitted from the revised document, which is modeled by the usage of p_1 by $revision_1$ and the lack of a generation of a related property. p_5 was added to the revised document, which is modeled by the generation of p_5 by $revision_1$ and the lack of a usage of a related property. Storing these assertions into a new, fine-grained bundle, encompassing the original, coarse-grained bundle, provides us with a multi-level view of the provenance of doc_1 and doc_2.

[4]Note that by modeling addition and omission this way, we remain technically compliant with the PROV-DM specification, which is recommended by the W3C. However, it might be more intuitive to use the insertion and removal properties of the PROV-Dictionary Note [57].

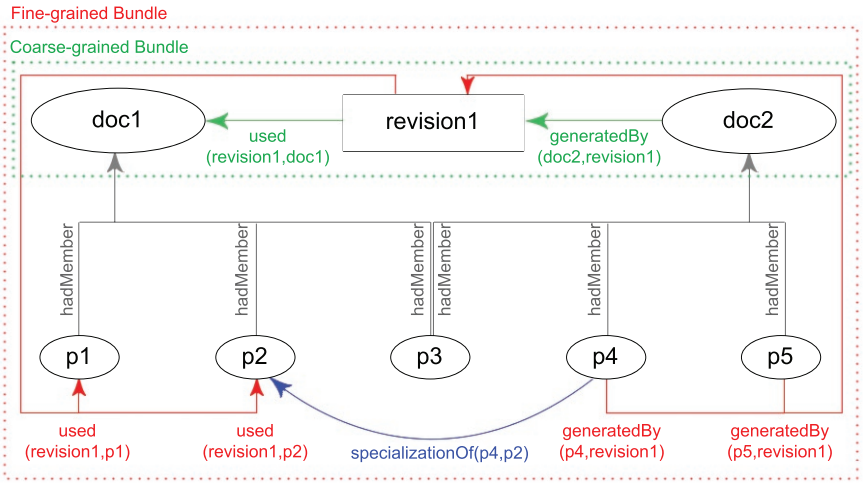

Figure 4.3: Finer-grained derivations indicating which changes occurred in a document.

4.5 Use Cases and Evaluation

We kept the description of our approach as general as possible, to ensure its broad applicability. However, for clarification and evaluation purposes, we customize it to two different use cases: reconstructing the provenance of different versions of news articles, and reconstructing the provenance of information diffusion on social media, described in Sections 4.6 and 4.7, respectively. Additionally, we describe the datasets we created for the 2014 Provenance Reconstruction Challenge in Section 4.8, enabling us to compare our approach to others in future work.

4.6 Use Case 1: News Versioning

In today's news industry, specification and justification of sources are key factors for producing high quality journalism. Unfortunately, due to the strong time constraints inherent to news production, provenance information is often incomplete or omitted. The consumers' need for near-immediate reporting also results in an abundance of very similar publications by all leading news organizations, often slightly modified versions of the same article, with limited to no possibility to determine the original source, or to determine which modifications were made to the content. This is exactly where our approach fills the gap. By detecting the derivation of one revision into another, our approach makes it possible to find the original source of an article, as well as the intermediary revisions. In this section, we describe how our approach is implemented for this use case, using the knowledge and data we gathered from the Belgian news agency *Belga*.

4.6.1 Documents and Properties

For the implementation of our approach, we need to identify "documents" and "properties", as described in Section 4.4. As documents, we use *news stories*, provided in different *revisions*. At news agencies, a news story generally starts as a simple, short *alert*, which can then be expanded into a *short story*, a *brief article*, and finally a *full article* (in some cases one or more of these stages are skipped). In Belgium, the articles are available in several languages, so multiple brief articles can be derived from one short story, etc.

As semantic properties, we use *named entities* associated with the news stories. These can be manually added, or automatically extracted from the content. In either case, the named entities are enriched, linking them to unique resources in the Linked Open Data Cloud. For the implementation of our approach, the named entities are also modeled as entities in PROV-DM, with each news article linked to the entities corresponding to the metadata by a *usage* activity.

4.6.2 Extracting Properties by Named-Entity Recognition

When news articles are not annotated with sufficient descriptive metadata, as is often the case in real-world scenarios, we need to automatically generate this metadata ourselves. The availability of accurate metadata associated with the documents will be beneficial to the resulting provenance.

To achieve this, we use publicly available named-entity recognition services. These services accept regular text as input, and output a list of linked named entities, detected in the text. The NERD [160] comparison tools allow us to evaluate the services and select the most fitting one for our work. For our implementation, we choose to use OpenCalais[5], a well-established, thoroughly tested [112] and freely available NER service. Note that as OpenCalais does not support Dutch, nor French at the time of writing, an automatic translation step is performed before sending the data, using the Microsoft Bing API[6].

4.6.3 Similarity Measure

Traditionally, semantic similarity between two documents is measured using the so called Vector Space Model (VSM) [166], also known as the "bag of words" model. In this model, objects are represented by vectors of weights, created based on their features[7]. For example, in the case of textual content, the weights in a vector are calculated using the Term Frequency - Inverse Document Frequency (TF-IDF) scheme. TF-IDF weights signify the importance of each term in the document.

[5] http://www.opencalais.com
[6] http://www.microsofttranslator.com/dev/
[7] For a more extensive explanation of the VSM, see Section 6.2.1

We adapt the VSM approach to work with named entities instead of words. This will allow two documents containing similar concepts, but of significantly varying length, to receive a high similarity score, whereas the classic TF-IDF approach would yield a lower score, due to the difference in text length.

When comparing two documents A and B, we create two *vector representations* $\mathbf{a} = (a_1, ..., a_n)$ and $\mathbf{b} = (b_1, ..., b_n)$ of their named entities, where the relevance score of named entity i in document A – as determined by OpenCalais during the NER step – is chosen as a_i (analogous for B). The similarity between the documents is calculated as the *cosine similarity* of these vectors. When no named entities were detected, we revert to the classic "bag of words" approach, using TF-IDF weights for every word in the text. Note that it would also be possible to calculate the similarity using a different method and/or different weights. Here however, we want to illustrate the feasibility of our provenance reconstruction approach, not the performance of a complex weighting scheme. We go into further detail about the VSM and other semantic similarity measures in Section 6.2.

4.6.4 Coarse-Grained Provenance through Clustering

As described in Section 4.4, we obtain the first, coarse-grained provenance by clustering sufficiently similar documents together. Using the similarity-measure in Section 4.6.3, we cluster the total set of news articles into sets of closely related articles. As shown in [8], clustering with a lower bound on similarity is an NP-Hard optimization problem. Fortunately, the authors of [8] also provide a greedy heuristic, SimClus, to approximate a solution to this problem. We choose this heuristic to cluster our dataset.

The applied algorithm is summarized as follows. The set of possible cluster centers S_{pc} initially contains all elements (with at least three named entities, to ensure accuracy of the similarity measure) of S. We compute the complete similarity matrix of the dataset S, which is then used to determine a *cover-set* S_u for each item $u \in S$. S_u contains all elements of S *covered* by u, which means their similarity to u is above an empirically determined threshold T_s. We now choose the cluster centers as follows:

1. Choose the item $u \in S_{pc}$ with the largest cover-set S_u as the next cluster center (if multiple items are tied, choose the one with the most properties; if there is still a tie, choose arbitrarily).
2. Remove all elements of S_u from S_{pc}.
3. Repeat step 1.

The algorithm terminates when there are no items left to choose as cluster center. The dataset is now divided into (possibly overlapping) clusters, corresponding to

the cover-sets of each cluster center. As an optimization, clusters with more items than a predetermined upper bound T_c are clustered again with a higher similarity threshold T_s. In our implementation, we choose $T_c = 10$, since news items rarely have more than ten revisions. For each cluster, we now add the multi-step and single-step derivations according to the method described in Section 4.4.1. Next, we construct the activities as in Section 4.4.2, resulting in fine-grained single-step derivations.

4.6.5 Finer-Grained Provenance

Starting from the coarse-grained provenance bundle from Section 4.6.4, we can create a finer-grained bundle in the manner described in Section 4.4.2. Note that the semantic properties are already identified in the NER step (see Section 4.6.2). Since these properties are linked to the LOD Cloud, information regarding synonyms, specializations and generalizations is available by following (or dereferencing) these links to popular datasets such as DBpedia, WordNet, Freebase, etc. Synonym relationships include *owl:sameAs* and *skos:exactMatch*, whereas examples of links specifying generalization and specialization are (respectively) *skos:broader* and *skos:narrower*. Using the methods in Section 4.4.2, we create the correct derivations, usages and generations linked to the revision activities from the coarse-grained provenance, and create a new, finer-grained provenance bundle, encompassing the original.

In Figure 4.4, this is illustrated for one news item. The news item starts as an English alert $news_1$, which is then translated into a Dutch alert $news_2$. Soon after that, a short story $news_3$ is written based on the English alert. Finally, the short story is revised to a brief story $news_4$, replacing the word "magistrate" with "prosecutor".

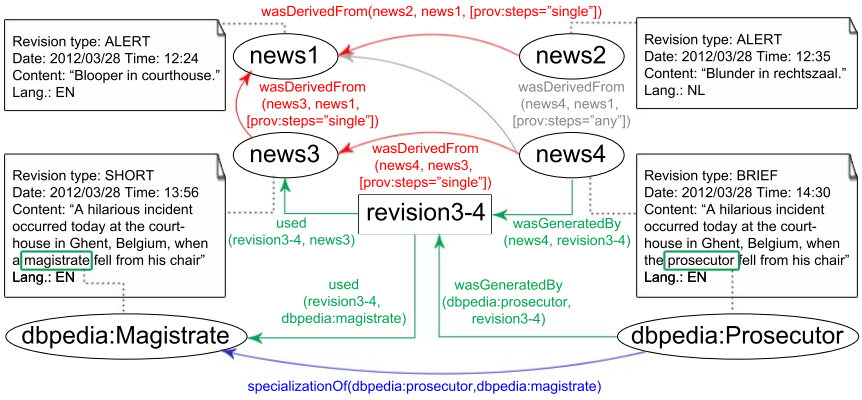

Figure 4.4: Example of discovered provenance in the news use case.

4.6.6 Evaluation

Our evaluation data consists of a set of 410 news stories, corresponding to 100 news items, in up to two different languages (Dutch and French), acquired from Belga[8], a professional Belgian news agency, over the course of one week.

The original provenance for the news stories, as specified by the content provider, is limited to the *revision types*, *original sources* and *multi-step derivations*. The source of a news item is always the earliest news story associated with that news item (usually an alert or short story). All following stories about that news item are (directly or indirectly) derived from its source (as a multi-step derivation).

Since there is no formal workflow to describe the creative process of news production, indisputably correct single-step derivations are nearly impossible to determine, even for the content providers (which is why our approach is so useful to them). Therefore, we restrict the evaluation to multi-step derivations.

We constructed coarse-grained provenance using the approach described in Section 4.6.4, based only on the (enriched) content and timing information of the news stories in our dataset. We can now compare the detected clusters, sources and multi-step derivations to the original information provided by the news agency. For each of the detected sources and multi-step derivations, we calculated the precision p as in Equations 4.5 and 4.6, and the recall r as in Equations 4.7 and 4.8, respectively.

$$p_{source} = \frac{\textit{number of correctly detected sources}}{\textit{total number of detected sources}} \tag{4.5}$$

$$p_{multi-step} = \frac{\textit{number of correctly detected multi-step derivations}}{\textit{total number of detected multi-step derivations}} \tag{4.6}$$

$$r_{source} = \frac{\textit{number of correctly detected sources}}{\textit{total number of sources in ground truth}} \tag{4.7}$$

$$r_{multi-step} = \frac{\textit{number of correctly detected multi-step derivations}}{\textit{total number of multi-step derivations in ground truth}} \tag{4.8}$$

	$T_s = 0.2$	$T_s = 0.3$	$T_s = 0.4$	$T_s = 0.5$	$T_s = 0.6$	$T_s = 0.7$	$T_s = 0.8$
p_{source}	68.0%	67.3%	**69.2**%	68.2%	64.3%	59.3%	57.7%
r_{source}	70.0%	68.0%	72.0%	**73.0**%	72.0%	70.0%	71.0%
$p_{multi-step}$	56.3%	61.6%	67.6%	**72.3**%	71.5%	57.1%	57.9%
$r_{multi-step}$	45.8%	**48.1**%	45.2%	44.5%	41.3%	28.4%	26.1%

Table 4.1: Accuracy of the provenance reconstruction in the news use case with similarity threshold $T_s \in \{0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8\}$ and cluster threshold $T_c = 10$.

[8] http://www.belga.be

In Table 4.1, the results are shown for different initial similarity thresholds T_s. In the optimal case, with $T_s = 0.5$, we were able to detect 73% of the original news sources, with 68.2% precision. The multi-step derivations constructed from these sources have a precision of 72.3% and a recall of 44.5%.

An explanation for these results is found when examining the clustered news stories. First, we consider the cluster precision $p_{cluster}$, which indicates the percentage of the detected clusters where for each cluster, all news stories it contains belong to the same news item and thus, share provenance. Second, we consider the news item recall $r_{newsitem}$, which indicates the percentage of original news items for which all news stories were cataloged into the same cluster by the algorithm. In Table 4.2, it is shown that for nearly all clusters (96% with $T_s = 0.5$), the news stories in the cluster all belong to the same original news item. This means that 96% of the clusters contain news stories that indeed share common provenance. However, $r_{newsitem}$ shows that many of the original news items are spread across more than one cluster, which creates more than one cluster per news item. This results in lower overall accuracy of the detected provenance, since there can only be one original source per news item, and the algorithm detects multiple.

	$T_s = 0.2$	$T_s = 0.3$	$T_s = 0.4$	$T_s = 0.5$	$T_s = 0.6$	$T_s = 0.7$	$T_s = 0.8$
$p_{cluster}$	83.8%	86.0%	93.3%	96.0%	97.7%	98.0%	100%
$r_{newsitem}$	30.0%	37.0%	32.0%	31.0%	26.0%	11.0%	8.0%

Table 4.2: Percentage $p_{cluster}$ of clusters for which all news stories originally belong to the same news item and percentage $r_{newsitem}$ of original news items that were cataloged into a single cluster.

Since we do not have a ground truth available for finer-grained provenance than the multi-step derivations, we could not evaluate the accuracy of this step of our proposed approach. However, the accuracy of the finer-grained provenance reconstructed using our approach depends strongly on the correctness of the detected named entities, and the quality of their links to ontologies that describe alternates, specializations and generalizations. Therefore, by analyzing the accuracy of the named-entity recognition step, we can at least get an indication of how well our approach can perform. When processing the 410 news stories, OpenCalais extracted 722 distinct named entities. Upon manual evaluation of these named entities we labeled 20 of them as incorrectly detected, resulting in 97.2% precision. Criteria for labeling a property as incorrectly detected were *non-existence* (no such concept exists) and *incorrect disambiguation* (linked to the wrong resource). These results are consistent with those of a larger performance analysis of OpenCalais, described in [112]. Of the 722 named entities, 47 were automatically linked to a resource in the LOD Cloud by OpenCalais. This means that while not perfect, the named-entity recognition step is definitely accurate enough for our finer-grained provenance reconstruction step to be deemed feasible.

4.7 Use Case 2: Social Media Information Diffusion

Nowadays, information from social media is frequently analyzed and processed for professional use. Examples include online journalism [106], rumor detection [123], and viral marketing [127]. In all these cases, it is important for the consumer to know the level of trust and relevance that the information carries.

In order to assess the trustworthiness of information on social media, a consumer needs to understand where this information comes from, and which processes were involved in its creation, i.e., its provenance. To model provenance for information diffusion on social media, we specified PROV-SAID [175], an extension to the W3C PROV model, explained in Section 2.2. Using this model, the social and influence graphs can be represented in an interoperable way. However, automatically reconstructing the aforementioned graphs based on the APIs that most social media provide poses a challenge, since current social media APIs cannot always capture the full lineage of every message. This leaves the consumer with incomplete or missing provenance, crucial for judging the trust it carries.

Most current methods are designed to only model direct, high certainty influence edges, caused by *explicit* re-emission of messages (e.g., retweets) and combined with connections between users (social graph), in order to unveil who was influenced by whom. These methods do not consider the large amount of *inexplicit* influences that are less certain, and thus more difficult to detect automatically (e.g., a user adapting another user's message, without explicitly referring to it). In this case, the provenance must be reconstructed, unraveling the unobserved references that users are using but not giving credit to, and revealing their influencers.

Therefore in this section, we propose an approach to reconstruct the provenance of messages on social media on multiple levels, in collaboration with the University of Freiburg. We combine a fine-grained, high-certainty approach, with a coarse-grained, less certain approach for provenance reconstruction. To obtain a fine-grained level of provenance, we use an approach from Taxidou et al. [176] to reconstruct information cascades with high certainty, and map them to PROV using PROV-SAID. To obtain a coarse-grained level of provenance, we adapt our similarity-based, fuzzy provenance reconstruction approach – previously applied on news. We illustrate the power of the combination by providing the reconstructed provenance of a limited social media dataset gathered during the 2012 Olympics, for which we were able to reconstruct a significant amount of previously unidentified connections.

Our contributions in this section are: 1) an approach for creation and integration of multi-level provenance; 2) a real-world application and evaluation of the PROV-SAID model; 3) a mapping in order to convert input data from social media into RDF; 4) a novel application of our previous work on similarity-based provenance reconstruction in the context of social media.

4.7.1 Related Work on Information Diffusion

While *information diffusion* in social media has received significant attention, in particular its modeling [102], there is limited work on the reverse procedure, i.e., *information provenance*, which is the focus of this section. We divide the state-of-the-art in this area in the following categories: (i) provenance reconstruction through social graph connections; (ii) provenance reconstruction through user profile metadata.

(i) Traditional information diffusion research includes tracing a piece of information back to its sources through social connections, revealing the concepts of influence and trust among the users involved. The work of [80] recovers information recipients sub-graphs given a small fraction of known recipients. In [103] unknown recipients are identified under the assumptions of degree and closeness propensity: nodes with a higher degree and closer to the sources are more likely to propagate information. [13] describes a provenance reconstruction method through social connections based on well established information diffusion models. Finally, the authors of [176] automatically reconstruct *information cascades*, that show which paths information took, given a piece of information that propagates over a social graph. Information cascades are graphs that model how information is being diffused from user to user; in other words, the approach in [176] reconstructs the paths of users who propagate information back to the sources by finding intermediate influencers.

(ii) Provenance can also be derived through user profile metadata, attributing relevance and trust to the information emitted according to the characteristics of the contributor. The work of [104] implements a tool for collecting such user information from different media sites, while not providing any information on the provenance paths and sources.

Our work reveals provenance paths by extending and adapting the solution proposed in [176] and our own as proposed in Section 4.4. The results are modeled and combined in an interoperable way using the PROV-SAID model. To recapitulate: PROV-SAID provides a rich description of provenance with regard to information diffusion concepts such as: *direct* and *indirect derivations*, *copied* and *modified messages*, and influence types such as *follow relationships* and *interaction influences*.

4.7.2 Combined Provenance Reconstruction Approach

As illustrated in Figure 4.5, we reveal provenance paths on two levels: (1) *low-level (fine-grained)*, based on structure as in [176] and (2) *high-level (coarse-grained)*, based on content similarity as in our proposed approach from Section 4.4. These methods are then combined using *PROV-SAID*. In order to convert the XML-based influence graph of [176] into PROV-SAID, we use the *RML mapping lan-*

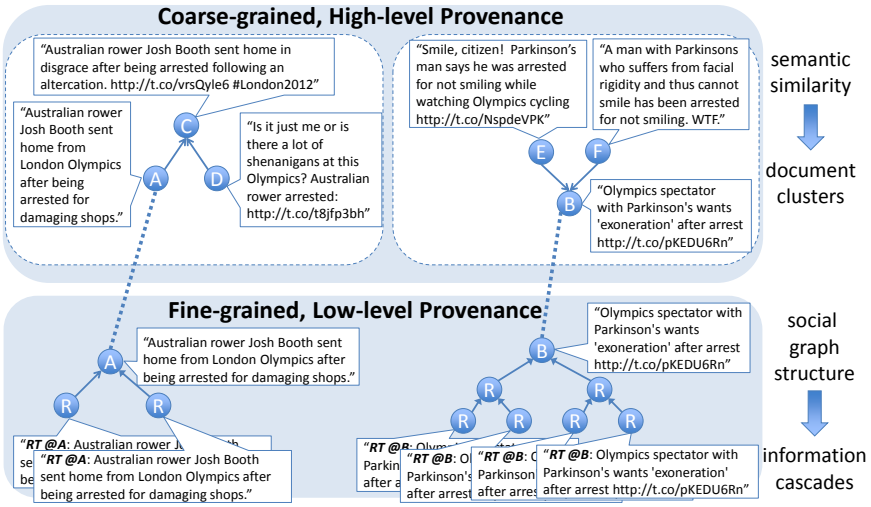

Figure 4.5: Overview of integrated, multi-level provenance. The arrows for the low-level provenance refer to `prov:wasQuotedFrom` *for all copied messages (retweets); for the high-level provenance they refer to* `prov:wasRevisionOf` *for all modified messages.*

guage [72]. RML is used in combination with a processor to convert proprietary data – such as XML – to RDF. In our case the data is converted to PROV-O, which is the ontology that expresses the PROV Data Model. Note that by using RML, we ensure that any input can be converted to PROV-O, rendering our method interoperable and reusable in many applications.

4.7.2.1 Low-level, fine-grained provenance

To obtain low-level provenance, we build upon [176] to reconstruct the so-called *information cascades* found in social media. Diffusion paths are reconstructed according to who is influenced by whom given messages that propagate over a social graph, with the assumption that users propagate identical messages (e.g., by retweeting) and identify possible influencers. When applied to the Twitter dataset described in Section 4.7.3, the reconstructed information cascades comprise of retweets, where users give credit only to the initial source of a message, not the intermediate source that exposes the message to them. In other words, it remains unclear which paths information took from the sources to the recipients. Therefore, the algorithm leverages the social graph in order to reconstruct the intermediate diffusion paths and find influencers, given the assumptions that information flows over the social graph and users are influenced by their connections in order to propagate a piece of information.

The algorithm outputs *edges*, directed from a *tweet A* to a *tweet B*. For each tweet, we have access to the *tweet-id*, *timestamp* and *userid*. When we map this to PROV-SAID using RML, we obtain the following PROV-O sub-graph for each edge:

```
status:tweetA_id  a  prov-said:Message ;
  prov:wasAttributedTo  user:tweetA_userid ;
  prov:wasGeneratedBy  _:emit-tweetA_id ;
  prov:generatedAtTime  tweetA_time .

status:tweetB_id a prov-said:CopiedMessage;
  prov:wasAttributedTo  user:tweetB_userid ;
  prov:wasQuotedFrom  status:tweetA_id ;
  prov:wasGeneratedBy  _:emit-tweetB_id ;
  prov:generatedAtTime  tweetB_time .

user:tweetA_userid  a  prov:Agent .
user:tweetB_userid  a  prov:Agent ;
  prov:wasInfluencedBy user:tweetA_userid ;
  prov:qualifiedInfluence  [
    a prov-said:FollowRelationship ;
    prov:agent user:tweetA_userid . ] ;
  prov:qualifiedInfluence  [
    a prov-said:InteractionInfluenceRelationship;
    prov:agent user:tweetA_userid . ] ;

_:emit-tweetA_id a prov-said:EmitMessage .
_:emit-tweetB_id a prov-said:EmitMessage ;
  prov:used  status:tweetA_id .
```

Note that the prefixes `status:` and `user:` refer to `https://twitter.com/statuses/` and to `https://twitter.com/intent/user?user_id=`, respectively, and that the prefixes `prov:` and `prov-said:` refer to their respective namespaces. This representation of the information cascades as provenance is now suitable to be merged with other interoperable provenance, such as the high-level provenance described in Section 4.7.2.2.

4.7.2.2 High-level, coarse-grained provenance

To obtain high-level provenance, we consider what is missing from the dataset generated in Section 4.7.2.1. Since the approach in Section 4.7.2.1 only relies on relationships exposed through a social media API, it does not consider all messages that were copied or revised without this being tracked by the social media software (e.g., when a user copy-pastes a message instead of retweeting it). To reconstruct this kind of information diffusion, we adapt our proposed similarity-based provenance reconstruction approach to be usable with social media content.

The core assumption of our approach is: *"if two messages are highly similar, there is a high probability that they share some provenance"*. The adapted approach consists of the following steps:

1. remove all tracked copied messages from every information cascade as generated in Section 4.7.2.1, keeping only the root messages;
2. index this reduced dataset using a feature model and semantic similarity function (e.g., TF-IDF and the cosine similarity), and compute the full similarity matrix of all messages;
3. apply a similarity-based clustering algorithm such as SimClus [8] to divide the dataset into (possibly overlapping) clusters of messages that all have a similarity to each other higher than a predetermined threshold;
4. for each cluster:
    - identify the oldest message as the root message of that cluster;
    - connect all other messages to the root message:
        - if the message is identical to the root message, using a `prov:wasQuotedFrom` relationship;
        - if the message is not identical to the root message, using a `prov:wasRevisionOf` relationship.

The expected result of this approach is that the vast majority of messages will be clustered as a singleton, meaning that no new relationships are introduced. Nonetheless, for those messages that do get clustered together, we know that they exhibit a high similarity. We use their temporal information to estimate their provenance relationship, thereby enriching the dataset and exposing previously hidden knowledge about the information diffusion. When we integrate this result in the next step, we are effectively re-connecting entire information cascades, whose connection was lost to the social media API. Note that due to the calculation of the full similarity matrix, this approach will have an quadratic complexity w.r.t. the number of messages considered, so it should always be applied on a pre-filtered dataset (e.g., a search result).

4.7.2.3 Integration of Multi-level Provenance

Because both algorithms output interoperable PROV, the integration of the two aforementioned levels of provenance consists of simply merging the two sets of RDF statements. However, it is important to understand the new structure this will give to the data. We clarify how the data is enriched by the combination of the two reconstructed provenance sets using Figure 4.5.

Each level of provenance differs in precision and granularity. The fine-grained, low-level provenance is very detailed, and was constructed with high certainty, since it consists solely of copied messages exposed by a social media API (in our

case: the Twitter API). The coarse-grained, high-level provenance, however, was constructed in a much less certain way, relying on semantic similarity to reconstruct connections that were lost to the social media API. The two levels enrich each other, providing previously unidentifiable connections between messages for data consumers (e.g., social media analysts) to explore.

4.7.3 Evaluation

As a preliminary evaluation, we tested our approach on a dataset gathered using the Twitter Streaming API during the 2012 Olympics. We chose Twitter because it provides trace information for copied messages (retweets). The dataset was collected by following the keywords 'Olympics2012' and 'London2012'. We limited the dataset by only considering tweets with a certain keyword, in our case: *'arrest'*. This simulates a realistic scenario where a social media analyst first searches for a broad keyword (e.g., a trending topic), and then investigates the information diffusion paths among the results. Complementary, we desire to avoid messages not carrying important information, for example: "I am watching the Olympics". This way, we include relevant events that attract attention both by individual users and mass media, while yielding information cascades by being retweeted. The final dataset consists of 9047 tweets, of which 5174 are copied messages (retweets), and 3873 are original messages according to the Twitter API. However, a number of these 3873 'original' messages are in fact also derived from each other, as will become clear using the high-level provenance reconstruction approach.

4.7.3.1 Low-level Provenance Reconstruction

We identified 31 cascades using the low-level reconstruction approach from Section 4.7.2.1, resulting in a skewed distribution from 5 to 1771 recorded retweets with the root tweet contained in the dataset (out of the total of 5174 retweets). This approach has already been thoroughly evaluated in [176], so we can safely assume that the identified cascades are correct.

4.7.3.2 High-level Provenance Reconstruction

Using the approach described in Section 4.7.2.2, we clustered the 3873 original messages from the dataset based on their semantic similarity. More specifically, we used the TF-IDF approach from traditional information retrieval to model all messages as vectors, and computed their similarity using the cosine similarity. We then executed the SimClus algorithm [8], which we also used for our news versioning use case, as explained in Section 4.6.4. Essentially, SimClus divides the set of messages into clusters of messages that all exhibit a similarity higher than a predefined threshold to their respective cluster center. To use the clusters to reconstruct

provenance as described in Section 4.7.2.2, the major challenge lies in identifying the optimal similarity threshold. The threshold must be high enough to ensure that only messages that actually share provenance get clustered together, while it must also be low enough to avoid that too many messages are clustered as singletons, which would result in missed connections. Ideally, the optimal threshold would be found empirically by analyzing the precision and recall of the provenance reconstruction approach, as it was done for the news versioning use case. Here however, we do not have access to a ground truth. However, we can investigate the influence of the similarity threshold on the number of clusters and their size, which at least gives us an idea of its behavior.

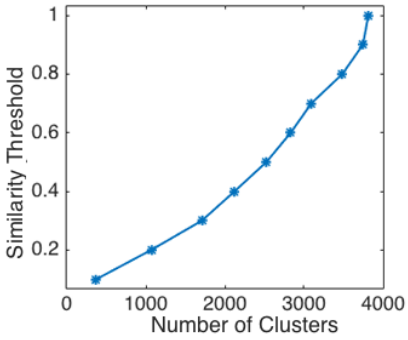

Figure 4.6: Total number of clusters for each similarity threshold.

As illustrated by Figure 4.6, the total number of clusters is approximately proportionate to the similarity threshold. This means that if we use a low threshold, we will have a small number of relatively large clusters. On the other hand, if we use a high threshold, we can expect a high number of smaller clusters. When the threshold is set to 1, only identical messages will be clustered together, and therefore only retweets – no modified tweets – missed by the Twitter API will be identified. This is further confirmed by our observations of the number of clusters per cluster size, as illustrated by Figure 4.7. Here, we see that for the lower thresholds (0.3 and 0.5), the cluster size varies highly, whereas there are less different cluster sizes for the threshold 0.7. These observations are an indication that for the lower thresholds, many clusters are incorrectly merged, which will affect the precision of the reconstruction. On the other hand, we see that if the threshold is set too high (e.g., 0.9), that the larger clusters are split, resulting in missed provenance relationships – and thus affecting the recall. In all cases above 0.3, we see that the number of singletons does not vary significantly, which means that messages that do not belong together will most likely not be clustered together, regardless of the similarity threshold. While it is too early to make a definite decision regarding the optimal threshold without a content-based evaluation, these results lead us to expect that

the optimum will be somewhere in the vicinity of 0.7. Using this threshold (0.7), we generated a set of 3094 clusters, and used the 206 non-singletons to reconstruct 879 provenance relationships. Of these 879, 62 relationships were detected more than once, leaving us with 817 provenance relationships (31 quotations and 786 revisions).

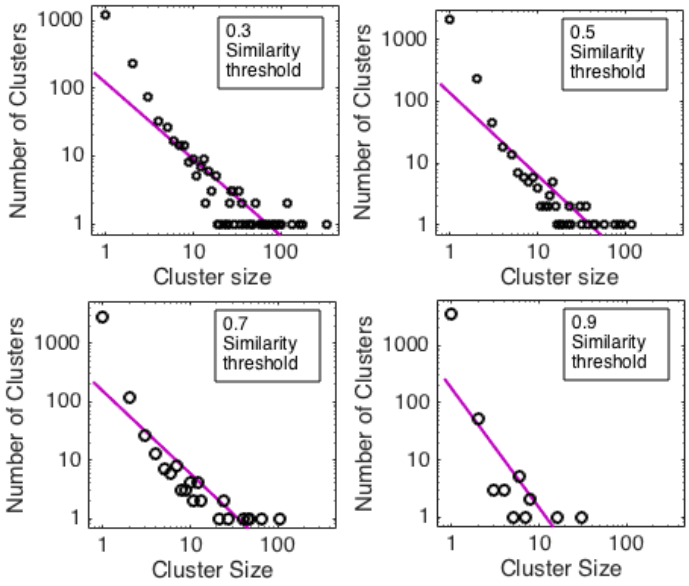

Figure 4.7: Distribution of the number of clusters per cluster size.

In other words, when we integrate this high-level provenance with the 31 cascades discovered by the low-level provenance reconstruction, we effectively introduce 817 connections that were previously unidentified. This creates much larger graphs for the consumers of the provenance data to analyze, and provides an enriched view on the information diffusion process. The entire reconstructed provenance graph can be downloaded at `http://semweb.datasciencelab.be/ns/prov-said/cikm2015.ttl`

4.7.4 Discussion & Future Work

The evaluation shows that our method has the potential to augment the provenance of messages on social media. This is especially the case when there is external influence not deriving from one single source (in our case: Twitter) or for copied messages that do not give credit to their initial contributors. In these cases, an obvious influencer is not exposed by the social media API. Such messages do not produce large cascades resulting in low-level provenance, but are clustered together in the high-level provenance reconstruction of our approach.

In future work, a more extensive evaluation is necessary to get an idea of the accuracy and impact of our approach on diverse datasets and combined data from different social media. Additionally, we will improve our method by applying more suitable metrics of message similarity for micropost text.

4.8 Provenance Reconstruction Challenge

As we have shown in this chapter, there is still a plethora of data that lacks associated data provenance. To help solve this problem, a number of research groups – including our own – have been looking at reconstructing the provenance of data using the computational environment in which it resides. This research however is still very new in the community, and datasets suitable for evaluation are rare. Thus, together with the Data2Semantics project from VU Amsterdam, we initiated the 2014 Provenance Reconstruction Challenge[9]. The aim of this challenge was to help spur research into the reconstruction of provenance by providing a common task and datasets for experimentation.

Challenge participants received an open data set and the corresponding provenance graphs (in W3C PROV format). They could then work with the data trying to reconstruct the provenance graphs from the open data set. The data consists of two distinct sets: one machine-generated, and one human-generated. This way, we are able to evaluate the reconstruction accuracy for provenance that was automatically collected based on observations, and provenance that was generated based on information provided by humans, which could not be captured automatically. For each dataset, we provide the raw data, and the ground truth provenance serialized in PROV-O.

4.8.1 Dataset 1: Version Controlled Documents

The first, machine-generated dataset is available at: `http://git2prov.org/reconstruction/machine-generated-dev.zip`.

The ground truth (*groundtruth.ttl*) for the first dataset was generated from a number of GitHub repositories using the Git2PROV tool. As raw data, it includes every version of each file that was ever present in the repository (including deleted files). However, the filenames are randomized, to simulate a scenario where all provenance was lost. Due to these randomized filenames, the timing metadata associated with the files may differ from the original. The correct timings can be found in the ground truth provenance (see the prov:atTime property of the qualified generations).

The main goal is to reconstruct the derivation graph of the original files, serialized as PROV-O. Participants were encouraged to make their generated prove-

[9] `http://www.data2semantics.org/prov-reconstruction-challenge/`

nance as complete as possible to obtain the best result. By this, we mean that it is advised to elaborate on complex relations such as `prov:wasDerivedFrom`, `prov:wasGeneratedBy`, etc., by also providing their qualified forms, i.e., `prov:qualifiedDerivation`, `prov:qualifiedGeneration`, etc.

To execute an approach, any information embedded in the files or external information may be used, save from the ground truth or the GitHub repositories themselves. For example, crawling repository hosting websites such as GitHub would not classify as a valid approach. It is assumed that the timing information of the raw data has also been lost and needs to be reconstructed. However, if an approach relies heavily on correct timing information, the `prov:atTime` properties of the qualified generations in the ground truth can be used. Naturally, if this is the case, it needs to be explicitly mentioned when describing the results.

Results using the dataset should report at a minimum two types of evaluation criteria:

- derivation recall: precision/recall of the `prov:wasDerivedFrom` relations;
- overall recall: precision/recall of all provenance relations mentioned in the ground truth.

Also, when reporting results, it is advised to make the distinction clear as to whether timing information was used.

4.8.2 Dataset 2: Human-Generated News

The second, human-generated dataset is available for download at: `http://git2prov.org/reconstruction/human-generated-dev.zip`.

The ground truth for the second dataset was created using the sources mentioned in news articles from *WikiNews*. The link between news articles and their sources is modeled using the `prov:hadPrimarySource` relation. The raw data consists of the entire HTML of the WikiNews articles, without the sources, and a list of URIs (*human_sources.txt*). In other words, the goal of this task is to match the source URIs from this list to the correct WikiNews article.

Approaches may use any information embedded in the files or external information as you see fit, save from the ground truth or WikiNews, for obvious reasons. Evaluations should report at a minimum the results of precision/recall of the `prov:hadPrimarySource` relations.

4.8.3 Results

The results were discussed during Provenance Week 2014[10], where unfortunately, only one participant submitted a full evaluation: Aierken et al. [7]. Therefore, the challenge ended up being more of a technical meet-up discussion various approaches for provenance reconstruction, rather than a benchmark. Nonetheless, the challenge datasets are now publicly available, and can be used as a gold standard for provenance reconstruction. This means that there is now a possibility to externally evaluate provenance reconstruction approaches, such as our own.

Our Similarity-based Approach We applied our method as described in Section 4.4.1 only on the human-generated dataset, for which our approach was primarily designed, and which is harder to capture in an automatic way[11]. As parameters, we used the **cosine similarity with TF-IDF weighting**, **10 different the similarity thresholds** T_s, and **no cluster-size threshold** (so no re-clustering). Additionally, the following considerations were made during the implementation:

- A number of documents do not include a date on the article page. In these cases, the original WikiNews articles were consulted, and the date reported there was used. In certain cases, this is the date of access by the writers of the article. Because a number of sources provide the full date and time, while others only provide the day of publishing, *only the day of publishing* was used for all articles.

- We re-formatted the dataset to be usable with our software. To do this, the text, and date had to be extracted from each HTML document, without advertisements, images, videos, etc. To obtain results that reflect the performance of our approach, not influenced by automatic text extraction methods, we performed this extraction manually, thereby assuming an *'ideal' text extractor*.

The results are shown in Table 4.3. At first glance, our method only achieved a rather disappointing maximum precision of 27% and recall of 16%. However, these results can be explained by looking deeper into how the human-generated dataset was constructed, and how our method tries to reconstruct it.

T_s	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Precision	0.30	0.14	0.20	0.21	**0.27**	0.25	0	0	0	0
Recall	0.12	0.13	**0.16**	0.15	0.12	0.066	0	0	0	0

Table 4.3: Results of our method as described in Section 4.4.1 on the human-generated 2014 Provenance Reconstruction Challenge dataset.

[10] `http://provenanceweek.dlr.de/`

[11] While reconstructing machine-generated provenance also has its merit, we focus on the adaptation of workflows to expose this type of provenance in an interoperable form, as described in Chapter 3.

In our method as described in Section 4.4.1, we assume the *oldest document* in a cluster to be the (indirect) source of *multiple documents* – i.e., all others in the cluster. However, the ground truth dataset was constructed in exactly the opposite way: the *newest document* is derived from *multiple sources*. This means that with a very minor adjustment to our method, we might be able to achieve much better results. Therefore, we extended our method for this benchmark, by including a **new parameter** that allows the algorithm to select the *newest document* in every cluster instead of the oldest, and making all other documents in the cluster primary sources of the former. When we ran our reconstruction algorithm with this parameter enabled, it confirmed our suspicions, and we achieved much better results, as shown in Table 4.4. Now, our method achieves 86% precision and 59% recall with $T_s = 0.4$.

T_s	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Precision	0.52	0.54	0.70	**0.86**	0.77	0.69	0.2	0	0	0
Recall	0.26	0.51	0.57	**0.59**	0.33	0.18	0.016	0	0	0

Table 4.4: Results of our slightly adjusted method on the human-generated 2014 Provenance Reconstruction Challenge dataset.

Comparison to Related Work Aierken et al. [7] reported a precision of 77% and a recall of 47% for reconstructing human-generated provenance, and a precision of 78% and recall of 68% for machine-generated provenance. However, since their method relies heavily on training data, they used the human-generated challenge dataset as a training set for their method, and created a new WikiNews dataset using the same procedure for their evaluation. This means that while at first glance, our reported results seem to outperform theirs, the numbers are not entirely comparable. Moreover, in our communications with the authors of [7], they are reporting new results on the challenge dataset of up to 95% precision and 73% recall using an improved version of their multi-funneling method, which should be published in the near future.

While some harmonization efforts of the various evaluation methodologies remains to be done, the results of Aierken et al. [7], together with our results on the challenge dataset and those we measured on our news dataset in Section 4.6.6, can at least be interpreted as an indication of the level of accuracy that is achievable with the current state of the art in this field. To maintain an overview of the advances in this field over the coming years, we initiated `http://provenancereconstruction.org`, a website where information regarding new provenance reconstruction research and evaluation datasets can be gathered.

4.9 Discussion and Future Work

We applied our approach for provenance reconstruction to two specific use cases, and one benchmark. The results of these evaluations show that our approach is feasible and provides good foundations for future work. We suspect that a more semantically aware similarity measure is likely to have a significant impact on the overall accuracy (e.g., as in [110]). To accommodate such a measure, extracted semantic properties need to be accurately linked to the Semantic Web. Luckily, NER and disambiguation techniques are continuously being investigated and improved in the scientific and industrial community. Recommending an optimal similarity measure is not possible, since its performance will be highly dependent on the use case. Therefore, even though it would make the approach less generic, considering domain specific information might prove worthwhile, as it may significantly improve accuracy and levels of granularity of the reconstructed provenance.

Thanks to the generic nature of our proposed provenance reconstruction method, several other use cases are possible. Examples of possible applications include plagiarism detection, provenance of code snippets and the tracing of information sources used for quotes in online content, such as blogs. Implementation of one or more of these use cases will allow us to further evaluate the approach, and provide more meaningful fine-grained provenance assertions.

The trust of the innocent is the liar's most useful tool.

Stephen King

5 Provenance-based Trust

In the previous chapters, we have proposed methods to gather as much provenance about content on the Web as possible. In this chapter, we will explain how all that provenance information enables us to make trust assessments about the corresponding content. More specifically, we discuss the methods to access PROV on the Web, how to check its validity, and how to infer trust from it.

This chapter is based on the following publications:

Tom De Nies, Sam Coppens, Ruben Verborgh, Miel Vander Sande, Erik Mannens, Rik Van de Walle, Danius Michaelides, and Luc Moreau. Easy access to provenance: an essential step towards trust on the Web. In *IEEE 37th Annual Computer Software and Applications Conference Workshops (COMPSACW) – METHOD 2013*, pages 218–223, 2013

Tom De Nies, Robert Meusel, Dominique Ritze, Kai Eckert, Anastasia Dimou, Laurens De Vocht, Ruben Verborgh, Erik Mannens, and Rik Van de Walle. A lightweight provenance pingback and query service for Web Publications. In *Provenance and Annotation of Data and Processes – IPAW 2014*, pages 203–208. Springer, 2014

5.1 Introduction

In the research community, provenance has been established as an essential contributing factor establishing trust for information resources [88, 145]. In this chapter, we present our ideas to enable making basic trust assessments of information

on the Web, based on the availability, quality, and content of its provenance. We argue that the aspects of trustworthiness of Web content are much too complex to be compressed into one single *trust score*, which would be difficult for a user to interpret. Instead, our main goal is to present the provenance of information on the Web in such a way that a non-expert user can easily understand it, and make a decision whether or not to trust the information.

5.2 Related Work

In literature, a significant amount of work is available on provenance as well as trust assessment on the Web. Here, we limit ourselves to those works describing a combination of both, as those are most relevant to our goals. For a detailed survey on trust in computer science and on the Web in general, we refer to Artz & Gil [9].

Content trust is defined by Gil & Artz [88] as *a trust judgment on a particular piece of information in a given context*. Most approaches in literature agree that *reputation* is an essential component for making trust assessments [88, 145]. In [94], a system is proposed that generates recommendations for content, based on the trust a user has in the agents that produced and/or published the content. The added value of this system, is that it not only makes use of the general reputation of a person or organization, but also of the trust relationships in the user's FOAF[1] profile, resulting in a very personalized trust assessment. The FOAF profile lists the user's contacts, and therefore provides context for the trust assessment, as the user is likely to trust someone he or she considers a friend. This is an important consideration to keep in mind for our future work.

However, as Ceolin et al. [28] argue, assessing reputation alone is not enough. They present an approach to compute the trustworthiness of user-annotated tags in a video corpus, based on a combination of reputation and provenance specified in W3C PROV. Their main goal is to provide reasoning and information retrieval software with automatically generated information regarding the trustworthiness of data. Another combination of reputation and provenance is described in [131], where the trustworthiness of sensor network data is assessed based on the reputation of network nodes in the provenance trace of this data.

In [129], events are identified that invoke distrust for a user, and it is described how these events relate to the provenance of the distrusted information. In Section 5.6, we apply a similar logic by checking for indicators that might generate these distrust events in the PROV associated with a Web resource.

We observe that most approaches focus on reputation of the source, in two cases in combination with provenance. Our approach directly considers the influence of the provenance associated with information on the Web on its trustworthiness level, allowing us to make trust assessments based on this information alone.

[1] Friend Of A Friend: `http://www.foaf-project.org/`

5.3 Accessing Provenance

The first step to using the provenance of Web content to assess trust for that content is to access it. The Provenance Working Group has published a note named PROV-AQ [117], stating the recommended methods to associate provenance to a document.

In the PROV-AQ specification, three mechanisms are proposed for a provenance provider to supply information that may assist a provenance consumer to locate the provenance descriptions related to a document: the HTTP *Link* header, the HTML *<link>* tag, and *RDF(a)*. Provenance descriptions for a resource can be provided in two ways: either by using a *provenance resource* that contains a set of provenance descriptions about the resource, or by using a *provenance query service*, where provenance for the resource can be retrieved.

For a resource accessible using HTTP, the provenance descriptions may be linked from the Link header included in the HTTP response to a GET or HEAD request, as specified in [153]. To this end, link relation types have been created: `has_provenance` and `has_query_service`. They are used as follows:

```
Link: <provenance-URI>; rel=
 "http://www.w3.org/ns/prov#has_provenance";
 anchor="target-URI",
 <provenance-service-URI>; rel=
 "http://www.w3.org/ns/prov#has_query_service";
 anchor="target-URI"
```

Here, the `provenance-URI` is used to indicate the provenance descriptions associated with the document, in which the document itself is referred to as `target-URI`. If no anchor parameter is provided in the header, `target-URI` is assumed to be the URI of the requested resource in the HTTP request. The `provenance-service-URI` refers to a service description that provides the consumer with the necessary information to submit a query to retrieve the provenance descriptions for the `target-URI`. Multiple `has_provenance` fields are permitted per Link header.

For resources represented as HTML, a provenance resource may be linked to by appending a `<link>` element to the HTML `<head>` element of the document. Three link relation types are defined: `has_provenance`, `has_anchor` and `has_query_service`. The placeholders `provenance-URI`, `target-URI` and `provenance-service-URI` have the same meaning as specified for the Link header above.

```
<html>
<head>
 <link href="provenance-URI" rel=
 "http://www.w3.org/ns/prov#has_provenance">
 <link href="target-URI" rel=
```

```
 "http://www.w3.org/ns/prov#has_anchor">
 <link href="provenance-service-URI" rel=
 "http://www.w3.org/ns/prov#has_query_service">
</head>
<body> ... </body>
</html>
```

Finally, a resource identified by a `resource-URI` and represented as RDF (in any syntax, including RDFa) may contain triples that relate the resource to its own provenance. Therefore, the relations `has_provenance`, `has_anchor` and `has_query_service` may also be used as RDF properties to create these triples.

```
@prefix prov: <http://www.w3.org/ns/prov#>.
<resource-URI>
 prov:has_provenance <provenance-URI>;
 prov:has_anchor <target-URI>;
 prov:has_query_service <provenance-service-URI>;
```

5.4 Alternative Provenance Access: Pingback

In all the access methods described in Section 5.3, the representation of the resource is directly linked to its corresponding provenance, so that only the publisher of the resource is in control of which provenance information is provided. This type of *"packaged"* solution gives rise to multiple issues, particularly when the owner of the resource is not in control of the publication process. In reality, most publishers lack incentives to publish the provenance of resources, even if the owner would like such information to be published. Currently, it is very intricate to link existing resources to new provenance information, either provided by the owner or a third party. In this section, we present a solution for this problem by implementing a lightweight, read/write provenance query service, integrated with a pingback mechanism, while still following the recommendations in PROV-AQ.

To make this less abstract, we will focus on a particular use case, in which the aforementioned issues are experienced every day: the domain of *scientific publishing*. The need of providing additional provenance information in this domain has long been identified [41, 196]. In the domain of scientific publishing, the resource (usually a PDF document) is published by the publisher, whereas its provenance (e.g., datasets, processes, and/or software used) is generally controlled by the author, as illustrated in Figure 5.1. Besides provenance information created at publication time, additional information such as pointers to corrections or derivations – forward-links in the provenance chain – should be added to enhance the value and the trustworthiness of the resource. The process of most publishers is currently not designed for this kind of updates, as they do not include information about the creation process at all. For example, an empirical study for economics journals shows that of all 141 considered journals, over 70% do not have any policy dealing with the data used in the journal publications [188].

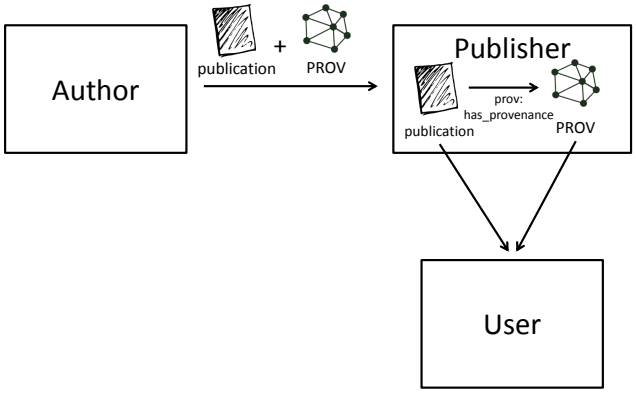

Figure 5.1: The current situation in the scientific publishing domain when it comes to provenance.

While general approaches to store and query workflow provenance have been introduced, cf. [38, 74, 98], these solutions date from before the publication of the W3C PROV standard, and/or constitute highly customized architectures. Additionally, in these solutions, the responsibility for publishing the provenance still lies either with the author or publisher, with no method to establish a *pingback* or *backlink* to the other party. Despite the PROV-AQ description and the possibility to apply basic technologies, we have not come across lightweight, distributed solutions for provenance storage and querying yet.

A possible, fully distributed solution to this problem is the concept of *provenance pingback*, as introduced in PROV-AQ. Provenance pingback enables the establishment of forward-links: e.g., to get to know which resources are based on a certain resource or who makes use of the resource. This solution, however, also highly relies on the goodwill and technological know-how of publishers to provide a pingback URI. Additionally, this would require the publishers to implement a management system aiding in the decision of which provenance is accepted to be published with the associated resource(s). Since this is outside of their core business, it would be better if publishers could rely on a third party to take the responsibility for this selection.

These facts justify the creation of a *lightweight* and *flexible* solution, in the form of an independent *provenance pingback service*. An independent service has the advantage that it does not rely on the cooperation of the publishers and enables all authors to use this service. This way, the responsibility is divided between the author, the publisher, and the pingback service, as illustrated by Figure 5.2. This scheme effectively allows each party to focus on their core capabilities. Publishers can focus on the publications themselves, whereas the community producing and consuming these publications can keep track of the various processes involved. The distributed nature of the Semantic Web makes this technically possible. Such

a service needs to allow the storage and retrieval of provenance links for published resources, thereby enriching them with information that is otherwise hard to expose. PROV-AQ defines a mechanism for this concept, named *provenance query services*. In the remainder of this section, we introduce our implementation of such a service targeted at the domain of scientific publishing. Furthermore, we show the advantages of our solution in this application domain.

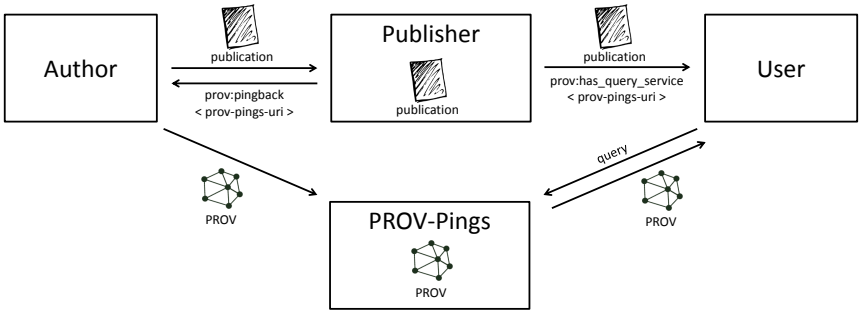

Figure 5.2: Our proposal for provenance management through a provenance pingback and query service, with all parties focusing on their core capabilities.

5.4.1 Lightweight Distributed Provenance Service

We propose a lightweight, RESTful web service for linking resources published on the Web with their provenance information. The solution allows pushing and querying of provenance information. This way, a seamless integration with existing publication management systems, such as *Research Gate*, *Mendeley*, *Google Scholar*, etc., is achieved. Figure 5.3 shows the process diagram of our service.[2] If possible, the publisher should support a provenance service by linking to it using a *pingback URI* and *provenance query service URI* as specified in PROV-AQ, but this is not a strict prerequisite. Note that in Figure 5.3, both these URIs are represented by the `prov_service_uri`.

The steps of the process are as follows.

1. An author **POST**s provenance about a published resource, identified by the `resource_uri`, to a service, identified by the `prov_service_uri`. Both the `resource_uri` and the `prov_service_uri` are forwarded to the publisher.

2. A consumer requests (**GET**) the publication with the `resource_uri` at the publisher and gets the data about the publication, and/or the publication itself. Ideally (but not necessarily), the publisher of the resource provides the URI of our pingback service as a *provenance query service*. This way, whenever consumers access the resource through the publisher, they are provided

[2] A live demonstration of this service can be accessed at `http://prov-pings.org`

with the proper `prov_service_uri`, at which the provenance can be found. Note that if the publisher does not provide a `prov_service_uri`, this does not prevent the author from posting his/her provenance to a service of his/her choice (e.g., where provenance of the same domain is collected). We briefly elaborate on alternatives in Section 5.4.3.

3. With both the `prov_service_uri` and the `resource_uri`, the consumer **GET**s all additional provenance information of the resource provided by the author. Using the PROV Data Model allows users to provide and retrieve provenance of the resource as a whole, as well as the provenance of certain sub-parts of the publication, such as data, code, etc.

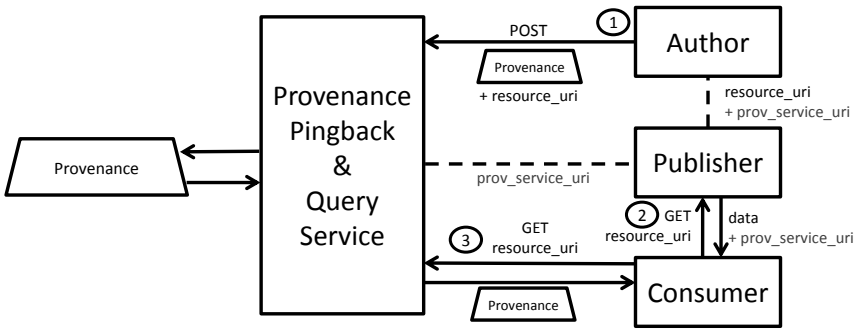

Figure 5.3: Process diagram of our proposed provenance pingback and query service.

5.4.2 Application Domains and Use Cases

Application domains that illustrate the merit of provenance query services include, but are not limited to: online news, blogs, digital books, code repositories, and data sets. In the following, we describe use cases that illustrate the different benefits provided by such a query service in our chosen domain of *scientific publications*:

Increase the trust in published results: In the area of scientific publications the typical metadata provided by the publishers are information about the authors, the proceedings or book where the publication can be found and temporal information as the year and month of the publication. It is metadata about the finalized publication, not metadata about the creation process. The metadata describing the process – the provenance data, as provided by a provenance query service – is much richer, revealing not only publications that the author has used to compile the text, i.e., the references, but also additional information about the original research data used, the methodology and the configurations of experiments to derive the results. The availability and verifyability of this information contributes to the trust in the published results.

Find related work: Beyond building trust in a specific publication, the provenance data also helps to identify *related work*, in this case work that uses the same original data or the same method. Results obtained on the same data are much more comparable. Applications of the same method on different data can demonstrate the general applicability of an approach. Contradicting interpretations of data can be found simply by the fact that both interpret the same data. Currently, information about original data can only be derived by reading the publications, which make it very time consuming or even practically impossible to find all relevant publications. With proper provenance data, this becomes trivial. To support this use case, our service specifically supports the submission of links between publications and used datasets by third parties, e.g., by an (semi-) automated process as described by Boland et al. [25].

Update and link to future work: Although the authors as well as the publishers are making huge efforts to create a final, perfect, and error-free version of a publication, it happens that published results are superseeded by future work, not to mention actual corrections in the case of errors identified after the publication. Minor updates of applied methods, adoptions to newer datasets or application versions, as well as errors in the code, dataset, and process happen more often than not. Even when the additions to existing work lead to a new publication, it is not trivial to find this newer publication. Smaller corrections, however, often do not even result in a proper new publication and an author has no reasonable way to add something to already published work. A provenance query service including the capacity of a pingback overcomes these problems, as the author is able to point to a newer, updated version of a publication. Such forward links in the provenance chain are not limited to the original author, in fact everyone can indicate that a later work builds on top of the publication.

5.4.3 Discussion

To realize the full potential of the provenance query and pingback service, there are a number of considerations to be made for its integration. The first issue to be considered is the *author verification & curation*. When a third party provides provenance information of a resource, this provenance might be inaccurate or even harmful when used to assess the trustworthiness of the resource. In order to prevent this, a form of verification should be deployed by the author upon the submission of provenance information. An already practiced solution, which is also applicable for scientific publications, is the approval of the email address which is usually associated with the publications of an author. For example, this mechanism is currently used by *Google Scholar*. Alternatively, an authorship claiming mechanism similar to `http://authorclaim.org` could also be implemented. Here, authors of information linked to provenance can claim ownership of the published provenance as well. Finally, a system such as the Open Researcher & Contributor ID (ORCID) [105] could automate part of this process, by automatically and unambiguously identifying researchers based on their contributions.

Another issue is the tracking of *provenance of the provenance*. Within a system where anyone can make claims about any resource, keeping track of the origin of submitted information and the evolution is crucial. Possible mechanisms to overcome this, can be found in version control systems, from which the provenance information can then be extracted using a mapping service such as Git2PROV [63]. A similar mapping could also support the resolution of the *provenance authoring* issues. Needles to say, that such a service needs an user-friendly way to specify provenance information, otherwise the obstacle of getting started will prevent authors and publishers to adapt the service.

Finally, the question remains what happens when the publisher does not play along and refuses to publish the link to a provenance service. A single, global provenance service is neither realistic nor desirable. Whereas a peer-to-peer communication between provenance services could be a possibility, a more straightforward solution would be a registry for provenance services or a dedicated search engine functioning as main entry point to provenance information. The investigation of all these issues remains future work.

5.5 Validating Provenance

In order to make easy, quick assessments regarding the trustworthiness of a document on the Web, a user needs more information than just the location of its provenance resource(s). The user needs to know whether the specified provenance resources actually exist, who created them, and whether they can be considered as *valid*. Valid provenance refers to W3C PROV documents that comply to the constraints defined in PROV-Constraints [32], guaranteeing well-formedness and consistency. While trust cannot directly be derived from it, validation of provenance does provide the user with an indication that the asserter of the provenance put effort into remaining compliant with the standard, and that the provenance is at least more likely to be trustworthy than invalid provenance.

5.5.1 URI Existence and Source

The first step in the validation process is to check whether the linked provenance URIs actually exist and if the provenance resources can be retrieved. This is done by sending an HTTP request for each provenance URI. If the headers of the specified provenance URI can be retrieved, the URI exists and is passed to the validator; otherwise, the URI is flagged as non-existent. Additionally, the location of the provenance is compared to the location of the original document, and it is stored regardless if they are the same or not. This happens because a user might prefer to trust a document whose provenance is stored in a trusted repository, instead of at the same location as the document itself.

5.5.2 PROV Validator

PROV Constraints states that valid PROV descriptions satisfy certain definitions, inferences, and constraints to provide a measure of consistency checking and reasoning over provenance. While validation is no guarantee for trustworthiness, it does guarantee that the supplied provenance can be consumed by all applications compliant to the standard, and therefore, it is a valuable property.

PROV-Constraints defines 56 distinct definitions, inferences, and constraints. This, in addition to the various PROV serializations, makes implementing a validator for PROV a non-trivial task. Luckily, a comprehensive, publicly available validation service[3] was developed by the University of Southampton. We will not discuss the details and inner workings of this validator here, and describe the use of its public API instead.

The API for the validator is used by sending an HTTP POST request to the validator's public submission URI[4] with the following parameters:

```
validate : 'Validate'
url : <provenance-URI>
file : <file upload>
statements : <the provenance statements>
```

Note that for our use case, only the fields `validate` and `url` will be set. No content type is specified. Content negotiation allows the following PROV representation types to be validated: `text/turle`, `text/prov-notation`, `rdf/xml`, `application/provenance+xml` and `text/json`. When the process is complete, the validator refers to an XML document, which contains the validation result. This XML document contains a child element for each validation error, and two additional elements for the provenance of the validation result itself. This means that if no error elements are present, the provenance was valid, and it can be labeled as such. To help the user understand this validation result, a link is provided to a (temporarily available) detailed validation report at the validation website.

5.6 Indicators of Trustworthiness

Providing users with access to the provenance of Web resources is an important step to allow them to make trust assessments, but this might be difficult for users who are not experts in the field of provenance or computer science in general. Therefore, an interpretation of the available information is presented to the user, in a way that he/she understands.

In literature, several indicators of trustworthiness of content have already been identified. Gil & Artz [88] observed that authority, related resources, provenance, and bias make up the four main factors contributing towards content trust. Later,

[3] `http://provenance.ecs.soton.ac.uk/validator/`
[4] `http://provenance.ecs.soton.ac.uk/validator/validation/submit`

Gamble & Goble [82] analyzed the information quality dimensions in literature, and concluded that the contributing factors can be classified in three dimensions: Trust, Quality, and Utility. They listed *reputation*, *objectivity*, *believability*, *security*, *authority*, *recommendation*, and *trustworthiness* as indicators of trust (**Trust dimensions**). Additionally, they listed the **Quality dimensions** identified in literature, such as *completeness*, *accuracy*, *consistency*, *currency*, *correctness*, *stability*, and *freedom from errors*. Finally, they listed *timeliness*, *accessibility/availability*, *conciseness*, *relevance*, *understandability*, *interpretability*, *amount of data*, *value-added*, *applicability*, *usefulness*, *cost*, and *usability* as **Utility dimensions**. The authors then argue that while the quality and utility dimensions are entirely *objective* and *subjective*, respectively, the trust dimensions are a *mix of both*. However, to us the latter is at least disputable, and the division is not so clear. For example, availability (listed as a utility dimension) can be considered objective, while completeness (a quality dimension) is very hard to measure, and can thus be considered subjective from the viewpoint of the user.

Rather than focus on objectivity or subjectivity, we asked ourselves the question: "which of these criteria are associated with provenance, and are feasible to be measured for Web content?". As a result, we compiled our own list of trust indicators, based on the provenance of a Web resource:

1. **availability**: whether there is provenance available for the resource;
2. **validity**: whether the provenance is well-structured and valid;
3. **provenance of provenance**: the source of the provenance, who asserted it, etc.);
4. **consistency**: whether the provenance is consistent with alternative sources;
5. **correctness**: whether the provenance corresponds with what's actually in the content;
6. **reputation**: the reputation of the agents and sources mentioned in the provenance.

Instead of returning a single trust score to the end-user, we choose to provide conclusions regarding each of these criteria to the end user. While a trust-score is valuable information for a machine agent making decisions on filtering or retrieving content, a human user might not understand the meaning of this score, and will possibly misinterpret it. Providing information to the user about each of the above criteria is aligned with the vision detecting distrust events, as described in [129], and will be more usable for non-expert users.

In our use case, described in Section 5.7, we provide information on criteria 1, 2, 3, and 6 to the user, because they are directly computable from the provenance associated with a Web resource using existing technology. Criteria 4 and 5 require more advanced processing of the content and provenance statements. Incorporating these two criteria is part of our future work.

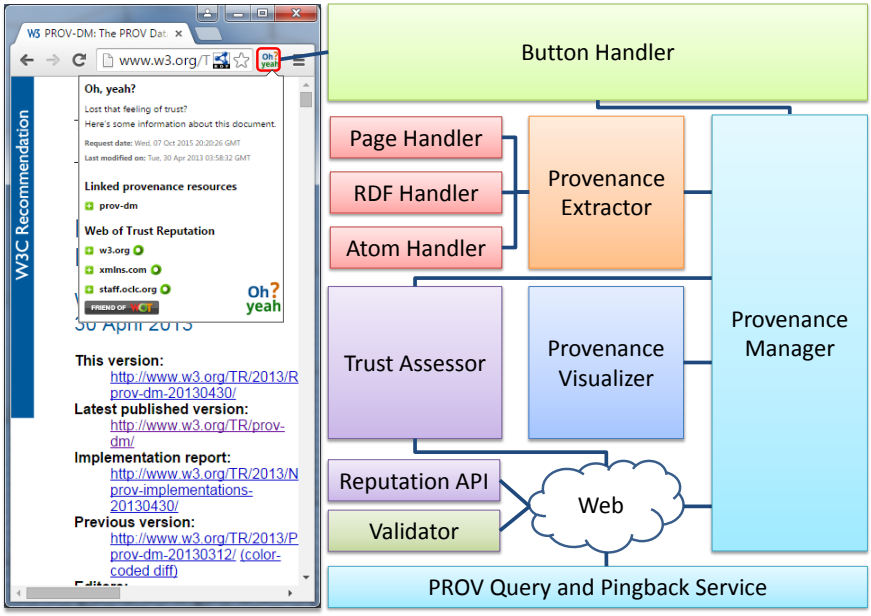

Figure 5.4: Overview of the "Oh, Yeah?"-button browser extension.

5.7 Implementation of the "Oh, Yeah?"-button

In 1997, Tim Berners-Lee proposed that each browser should have a button marked "Oh, Yeah?" [18], that a user can press when he/she loses the feeling of trust when viewing content on the Web. Upon pressing the button, information is shown about why the user should (dis)trust the document. In this section, we describe a browser extension that constitutes an implementation attempt of this "Oh, Yeah?" button. When the button is clicked, the browser acquires the provenance resources linked to by the document the user was looking at and displays the URIs, whether the provenance is valid and whether the URI actually exists, in addition to a number of automatically derived statements regarding the trustworthiness of the document. The extension is available for download at `http://research.tomdenies.be/OhYeah/`.

Although we are not the first to attempt implementation of the "Oh, Yeah?"-button, we are the first to do so based on W3C PROV, as part of a Web browser for generic Web content. Bizer & Cyganiak [23] also made an "Oh, Yeah?"-button for their Web Information Quality Assessment (WIQA) Browser, intended for exploring RDF datasets.

5.7.1 Overview of the Extension

In Figure 5.4, an overview of our application is shown. The "Oh, Yeah?" button is located at the upper right corner of the browser window. Upon pressing the button, the content and headers of the document are passed to the Provenance Manager, which then processes the information in five steps.

1. The Provenance Manager checks if any provenance is associated with the document, and if there is: where it is located[5].
2. If it is embedded in the document itself, the Provenance Extractor extracts the provenance, using a suitable method for each supported document type.
3. The linked provenance resources are fetched (if they exist), and validated using a Web-based validation service.
4. All information is interpreted by the Trust Assessor, using a web-based reputation API.
5. The results are summarized and visualized in a pop-up by the Provenance Visualizer.

In the next sections, we will explain each of these steps in detail.

5.7.2 Implementation Choices

Our application aims to bring provenance to a broad audience – not only to experts in the field of provenance – through a lightweight and understandable visualization. The application is written in Javascript, and therefore should be usable in most browsers. However, as explained in Section 5.3, provenance may be specified in the headers of HTTP requests, and our application must be able to intercept these requests. Therefore, we opted to build an extension specifically for one browser (Google Chrome), due to its easy access to Web requests (specifically, through the `chrome.webRequest` module). However, all other components of the application are browser-agnostic, and extensions could be built for Mozilla and Safari in future work.

5.7.3 Criteria for Trust and Distrust

Our current implementation of the "Oh, Yeah?"-button considers four criteria from which a distrust event can be derived. The following rules are applied:

- **Provenance availability:** If provenance is linked to, the application checks the existence of the provenance URI(s) and relays this information to the user. If no provenance is linked to or embedded, our provenance query and pingback service `prov-pings.org` is checked.

[5] It could be linked to by the document, embedded inside the document, or at a provenance query and pingback service.

- **Provenance validity:** If the provenance is successfully validated, an icon indicating this is valid provenance is displayed, if not, a warning is shown.
- **Provenance of provenance:** The location of the provenance linked to the resource is displayed to the user, indicating whether it is hosted at the same location as the resource, or at an external source. It can be argued whether either one is more cause for trust than the other. Ideal is when both a local and an external provenance record are present, as this provides a reference for consistency checking.
- **Reputation:** The domain names in all URLs referring to agents and derivation sources are extracted from the provenance statements, and their reputation is assessed by an external API, specifically the Web of Trust[6] (WOT) API. Web of Trust returns a numerical reputation score, which translates to a human readable rating, ranging from "very poor" to "excellent". These are the ratings shown to the user, as well as a clarification of the confidence, the estimated reliability of the reputation value. This way, generating unnecessary distrust events is avoided.

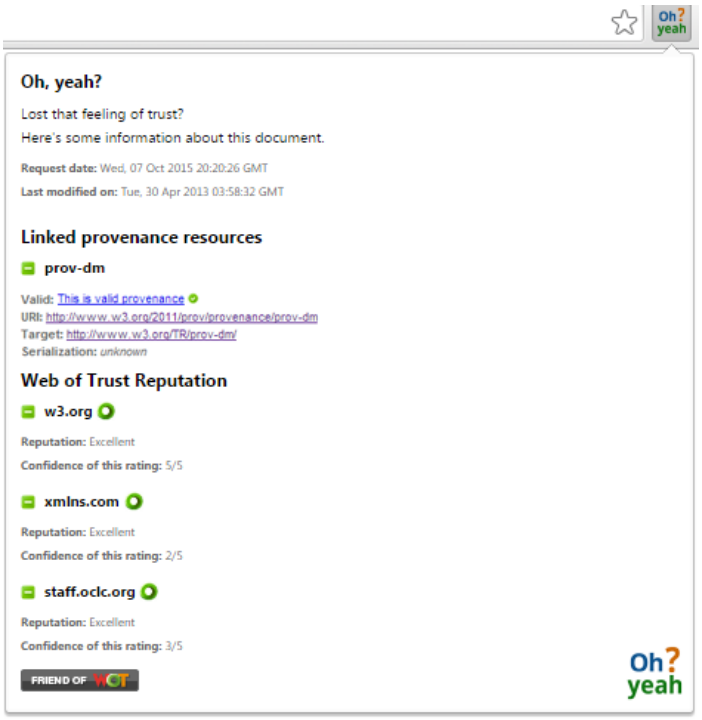

Figure 5.5: Visualization of trust assessments for the "Oh, Yeah?"-button.

[6] http://www.mywot.com/wiki/API

5.7.4 Visualization

The information acquired is relayed back to the user, by showing an unintrusive pop-up window above the document, right under the "Oh, Yeah?"-button. In Figure 5.5, this visualization is illustrated for the provenance associated with the PROV-DM specification page at `http://www.w3.org/TR/prov-dm/`.

This pop-up displays the timing details of the document (when it was requested and when it was last modified), and a list of provenance resources. Each of the items in this list contains details of the selected provenance resource. These details include: the provenance URI, whether it could be retrieved, its source, the validation result and the serialization used. Additionally, the reputation of the domains mentioned in the provenance is relayed to the user.

Note that while the information that is shown to the user might be perceived as technical, the colors used already indicate an intuitive measure of trust for non-expert users: green for aspects that inspire trust, orange for dubious aspects, and red for distrust events.

5.8 Conclusion and Future Work

The implementation of the "Oh, Yeah?"-button illustrates that thanks to the finalization of the PROV standard, we are a few steps closer to bringing trust assessments to the Web. Enabling easy access to the provenance of Web resources adds value for both consumers and providers of these resources. Consumers gain access to additional input to make an informed decision when deciding to trust the information on the Web, and providers gain an incentive to assert and publish the provenance of their resources.

In future work, we aim to research a finer-grained analysis of the provenance linked to Web resources. More specifically, this would allow us to generate statements regarding the consistency and correctness of the provenance information, based on cross-checking of the information in multiple provenance records and the content. Furthermore, the disadvantage of a centralized approach for reputation assessment is that the central service decides which sources are trustworthy, usually based on crowdsourcing. As explained in [94], a more personalized approach would be beneficial, where the preferences and relations of the user are taken into consideration when calculating the reputation assessment.

The only real valuable thing is intuition.
Albert Einstein

6

The Next Step: Assessing Content Value

When it comes to publishing and/or processing content on the Web, content producers and consumers in the digital publishing world are all facing the same problem: an abundance of content, available from an ever increasing number of sources. News sites, blogs, social media, and digital libraries are overflowing with content. However, human content creators and curators do not receive more time to filter through this continuous stream of content than before the existence of the Web. On the contrary, the need for immediate reporting increases, while the patience of the consumer decreases. This calls for a system that assists the author or publisher by automatically assessing the value of content. In this chapter, we outline how the approach for provenance-based trust assessment discussed in the previous chapters fits into a bigger picture of automatic content value assessment for digital publishing, when combined with our ongoing research on automatic relevance assessment.

This chapter refers to, and is partly based on the following publications:

Tom De Nies. Assessing content value for digital publishing through relevance and provenance-based trust. In *The Semantic Web – ISWC 2013 (Doctoral Consortium)*, pages 424–431. Springer, 2013

Tom De Nies, Pedro Debevere, Davy Van Deursen, Wesley De Neve, Erik Mannens, and Rik Van de Walle. Ghent university-ibbt at mediaeval 2012 search and hyperlinking: Semantic similarity using named entities. In *MediaEval 2012 Multimedia Benchmark Workshop*. CEUR-WS, 2012

Tom De Nies, Wesley De Neve, Erik Mannens, and Rik Van de Walle. Ghent university-iminds at mediaeval 2013: An unsupervised named entity-based similarity measure for search and hyperlinking. In *MediaEval 2013 Multimedia Benchmark Workshop*. CEUR-WS, 2013

Tom De Nies, Jasper Verplanken, Ruben Verborgh, Wesley De Neve, Erik Mannens, and Rik Van de Walle. Named-entity-based linking and exploration of news using an adapted jaccard metric. In *Proceedings of the 1st Workshop on Negative or Inconclusive Results in Semantic Web (NoISE2015)*, 2015

Fréderic Godin, Tom De Nies, Christian Beecks, Laurens De Vocht, Wesley De Neve, Erik Mannens, Thomas Seidl, and Rik Van de Walle. The normalized freebase distance. In *The Semantic Web: ESWC 2014 Satellite Events*, pages 218–221. Springer, 2014

Tom De Nies, Christian Beecks, Wesley De Neve, Thomas Seidl, Erik Mannens, and Rik Van de Walle. Towards named-entity-based similarity measures: Challenges and opportunities. In *Proceedings of the 7th International Workshop on Exploiting Semantic Annotations in Information Retrieval*, pages 9–11. ACM, 2014

Tom De Nies, Christian Beecks, Fréderic Godin, Wesley De Neve, Grzegorz Stepien, Dörthe Arndt, Laurens De Vocht, Ruben Verborgh, Thomas Seidl, Erik Mannens, and Rik Van de Walle. A distance-based approach for semantic dissimilarity in knowledge graphs. In *Proceedings of the 10th International Conference on Semantic Computing (ICSC)*. IEEE, 2016

Tom De Nies, Christian Beecks, Fréderic Godin, Wesley De Neve, Grzegorz Stepien, Dörthe Arndt, Laurens De Vocht, Ruben Verborgh, Thomas Seidl, Erik Mannens, and Rik Van de Walle. Normalized Semantic Web Distance. In *The Semantic Web – Latest Advances and New Domains (ESWC 2016)*. Springer, 2016

6.1 Indicators of Content Value

We argue that there are two aspects that consolidate the essential components of content value: *trust* and *relevance*. First and foremost, the user needs an indication whether or not the content is to be considered as trustworthy. In this dissertation, we have focused mostly on this aspect, and how to assess it automatically based on the content's provenance. However, when assessing the value of the content on a broader level, it also becomes important to determine *for whom* the content is relevant, and *which aspects* of the content make it relevant.

Note that content value cannot be expressed as a single numerical value, and it might differ depending on the target audience. For example, in the case of news

publishing, the content value of an article is primarily determined by its newsworthiness. This newsworthiness includes its relevance to a certain reader group's interests, and the trustworthiness of the information.

As this example illustrates, the problem addressed in our work is relevant to several use cases in the digital publishing world, including production of news, eBooks, digital magazines, or more open Web content, such as blogs and microposts on social media. It is also important to note that the problem is relevant to information *consumers* as well as to those on the *production* side of information. While the areas of content filtering and recommendation systems with the consumer as end-user are widely researched, the approach proposed in this chapter specifically aims to assist the content creator *during* the production process. As the information overload on the Web has the highest impact on news organizations, supporting journalists is our main use case. Our approach is especially relevant to journalists with low financial resources, such as citizen journalists and professional journalists dealing with ever-decreasing budgets.

Most works in literature deal with quality assessment of machine-generated data gathered by observation systems. For the assessment of human-generated content, only a limited number of solutions are proposed [139]. Search engines such as Google apply advanced techniques to assess the value of content to the end user, but do not make this process transparent to maintain their market advantage.

We propose a value assessment system that integrates both relevance and trust assessments on the content level, and generates a *value assessment report* for content on the Web. To achieve this, we rely on the techniques we have discussed throughout this dissertation, complemented with semantically aware relevance assessment. We consider two scenarios in which this approach is applicable. In the first scenario, the *content producer* (e.g., the author or publisher) submits his or her own content, to assess its potential value for future readers. In the second scenario, the *content consumer* (e.g., a reader or research journalist) is searching through a large dataset for content related to the document he or she is reading, or intends to publish. This large dataset can either be *closed* (e.g., the publisher's archive), or *open* (e.g., datasets on the Web). We provide a high-level overview of our proposed approach in Fig. 6.1.

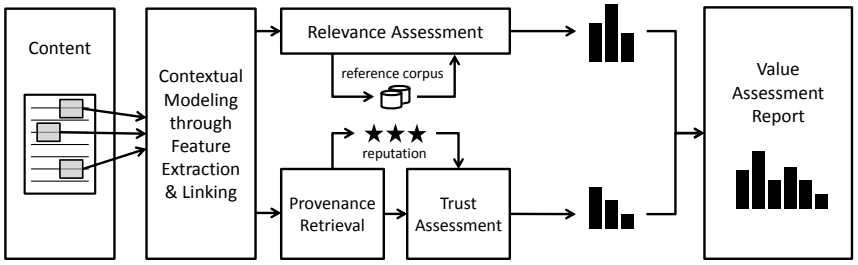

Figure 6.1: High-level overview of our value assessment approach. A contextual model is used to generate the content's relevance, reconstruct its provenance, and assess its trustworthiness.

The basis for this approach is the creation of a contextual model of the content for which the value is assessed. To achieve this, semantic feature extraction and linking methods can be relied upon, such as named-entity recognition (NER) services as discussed in Section 1.2.6. To determine the overall relevance of content, its contextual model can then be used to calculate its semantic similarity to one or more reference corpora. These corpora can include sets of recent and/or popular news articles, micropost streams from social media, or publications categorized per topic, etc. The eventual goal is to return a list of highly relevant publications, which are then used to make a statement about the overall relevance of the input content. Additionally, individual elements of the contextual model can be used to infer statements about in which aspects the content is relevant, and to whom.

In parallel, the provenance of the content can be modeled, exposed, or reconstructed as described in Chapters 2, 3, and 4, respectively. The accumulated provenance will then be used to generate a trust report, using the approach we proposed in Chapter 5.

The final step in our proposal is to present the acquired information in such a way that the user is able to determine the strong and weak points of the content. Here, it is important to provide tangible, and comparable results, to allow for automatic ranking of multiple content items. However, the reasoned statements generated in the relevance and trust assessment steps should also be made available to the user. For example, in the first scenario (value assessment), the user of our system (the content producer) requires a *fine-grained* value assessment of his or her content. This means that he or she not only requires the *raw* value assessment, but also the reasoning statements, indicating *why* the content is valuable. On the other hand, in the second scenario (content selection), the user (the content consumer seeking valuable content) requires a *coarse-grained* value assessment, used to rank the results.

Until now, the focus of this dissertation has been primarily on provenance and trust assessment. In this chapter, we discuss the other main part of our proposed approach for automatic content value assessment: assessing content relevance through semantic similarity. As seen in Chapter 4, semantic similarity also serves as an essential part of our provenance reconstruction method. Therefore, in addition to our primary research on provenance and trust, we have also investigated various methods for measuring semantic similarity. In Section 6.2 we provide an overview of this ongoing research. Finally, in Section 6.3, we reflect on the overall merit of our proposed approach, and the necessary future work to make it a reality.

6.2 Relevance Assessment and Semantic Similarity

A fully detailed description of all the similarity measurement experiments we have conducted during the course of our research would draw the focus too far away from the main topic of this dissertation. However, to give the reader an idea of our approach to the problem, we briefly introduce the techniques we have researched, and refer interested readers to the published works for more details.

6.2.1 Traditional Similarity Measures

As we briefly introduced in Section 4.6.3, semantic similarity between two digitally represented objects is traditionally measured using the so called Vector Space Model (VSM), or "bag of words" model [166]. In this model, objects are represented by vectors of weights, created based on their features. For example, in the case of textual content, the weights in a vector are calculated using the Term Frequency - Inverse Document Frequency (TF-IDF) scheme. For a document A represented by a vector $\mathbf{a} = (a_1, a_2, ..., a_n)$, the weights are calculated as follows:

$$a_i = tf_i \times idf_i, \tag{6.1}$$

where the term frequency tf_i is the number of appearances of term i in document A, and the inverse document frequency idf_i is 1 divided by the number of documents in the considered dataset term i appears in. $i \in \{1, 2, ..., n\}$ is varied over all terms in the considered dataset. This way, the idf reduces the weight of popular terms in the dataset, thereby emphasizing the importance of terms that are more unique, and thus more defining for a certain document. To obtain optimal results, the text will typically be pre-processed before this weighting scheme is applied, by generalizing the tense of verbs, converting multiples to singular forms, etc. This pre-processing step is better known as *stemming*. Finally, similarity between two documents A and B is measured by calculating the *cosine similarity* between their vector representations $\mathbf{a} = (a_1, ..., a_n)$ and $\mathbf{b} = (b_1, ..., b_n)$:

$$similarity(\mathbf{a}, \mathbf{b}) = \frac{\mathbf{a}.\mathbf{b}}{\|\mathbf{a}\| \, \|\mathbf{b}\|} = \frac{\sum_{i=1}^{n} a_i \times b_i}{\sqrt{\sum_{i=1}^{n} (a_i)^2} \times \sqrt{\sum_{i=1}^{n} (b_i)^2}} \tag{6.2}$$

Although this technique is several decades old, it is still very popular in modern Information Retrieval. For example, in the MediaEval 2012 *Search and Hyperlinking* benchmark, where video fragments needed to be searched and linked based on their transcripts, an approach using the VSM and the TF-IDF weighting scheme performed best out of all participants [78].

However, the TF-IDF weighting scheme has one major drawback: it only considers *syntactic* information, and is oblivious to the underlying *semantics* in a text document. For example, the sentences *"I saw George in Washington today"*, and *"I saw George Washington today"* are syntactically very similar (they only differ by one word), but semantically very different. A similarity measurement using TF-IDF would not be able to pick up on this difference.

6.2.2 Named-entity-based Document Similarity Measures

As described in Section 1.2.6, correctly disambiguated named entities enable semantic insight into a text. We argue that this is especially the case when it comes to similarity measures. For example, recognition of the location 'Washington' and the person 'George Washington' in the aforementioned example sentences, already provides valuable information to a machine about the semantics of these sentences.

In our work, we have experimented with two broad categories of approaches to harness the added semantic value that comes with named entities:

1. approaches that still use the cosine similarity, but with a named-entity-based weighting scheme;

2. approaches that use the Jaccard similarity, adapted for named entities.

These approaches suppose the documents to share at least one common entity in order to measure a meaningful similarity value. While related approaches have been proposed in literature, they all remain to be evaluated on a large scale in generic scenarios. The *Concept Frequency - IDF weighting scheme* [96] for instance, uses entities to create TF-IDF-like vectors which are compared using the cosine similarity in a news recommendation scenario. Additionally, an adaptation of the Jaccard metric for named entities has also been proposed [144]. We tested our proposed approaches in two scenarios: *multimedia search and hyperlinking* and *news clustering*.

6.2.2.1 Test Scenario 1: Multimedia Search and Hyperlinking

The first test scenario for our named-entity-based similarity measures was the Search and Hyperlinking task at the MediaEval benchmark, both in 2012 and 2013. The full details of the task for the 2012 and 2013 editions are explained in [79] and [76], respectively. In essence, the task consists of searching segments of video based on a short textual query, and automatically linking the results to other, related segments. This scenario is comparable to the way users explore large online video collections, such as YouTube. The task organizers relied on *crowdsourcing* to create a ground truth to evaluate the approaches of participants in the benchmark. To complete the task, participants received various descriptive features of the videos, including the audio transcripts automatically generated via speech-to-text (STT) algorithms. When only considering these transcripts, the task essentially becomes a text search and hyperlinking task, which makes it suitable to evaluate our named-entity-based approaches. Therefore, we performed a NER step on the transcripts, resulting in named entities as extra input data for the task.

Two important choices influence the results of an approach to tackle this task: the *segmentation strategy* for the videos, and the *similarity measures*. Since similarity measurement is more relevant to our research, we decided to focus our efforts on the latter choice, and use a basic segmentation strategy.

In 2012, the blip10000 collection [167] was used as a dataset for the task. This collection was crawled from the social video platform Blip.tv, and contains 4838 semi-professional videos with a total duration of 3260 hours. While the language used is predominantly English, there are also a number of French, Dutch, and Spanish videos. For the task, the videos were accompanied by two different automatic speech recognition (ASR) transcripts (generated by LIMSI [125] and LIUM [162]), textual metadata (tags) and automatically identified shot boundaries and keyframes [115].

With this dataset, we chose to use the provided shots as segments, and apply a hybrid approach to measure similarity between them [59]. This approach combined the cosine similarity with two weighting schemes: one using traditional TF-IDF weighting, and one using a new named-entity-based weighting scheme. The latter scheme uses a sparser representation of the text, and thus a faster way of assigning lower weights to common terms. The $TF(e, D)$ of a NE e in document D remains the same: the number of occurrences of e in D. However, as an alternative to the IDF, we introduced the *Inverse Support* (IS). If $support(e)$ is the number of incoming links of NE e, then the Inverse Support of e in document D is defined as:

$$IS(e, D) = \frac{\sum_{a \in D} support(a)}{support(e)}. \tag{6.3}$$

The weight of a NE e in document D is then calculated as $TF(e, D) \cdot IS(e, D)$. The purpose of this experiment was to find out if using the named entities would offer an advantage over traditional weighting schemes. However, while the weights were certainly faster to calculate, they did not result in a more accurate similarity measurement.

We achieved a Mean Reciprocal Rank (MRR) of 0.254 for the searching part of the task. This means that our approach was unable to outperform the other approaches, which used the traditional TF-IDF scheme with a better segmentation strategy, the best of which achieved a MRR of 0.470 [81]. The same pattern was found in the hyperlinking part of the task, where our approach resulted in a Mean Average Precision (MAP) of 0.171, whereas the best approach [151] achieved a MAP of 0.405. These results were confirmed with minor variations in a joint re-evaluation effort with all participants [78]. An explanation for these rather disappointing results may be found in the short nature of the textual queries for the search task – thereby resulting in a low number of named entities per query –, the quality of the automatically generated transcripts – resulting in inaccurate NER –, and a sub-optimal segmentation strategy.

For the 2013 edition of the task, a new dataset was used, comprising of 1260 hours of video provided by the BBC, primarily in the English language. Although this dataset is smaller than the one used in 2012, the videos were of better quality. Furthermore, the same transcript and metadata types were provided as in 2012, with one very important difference: apart from automatically generated STT transcripts, proper English subtitles were also provided for each video. Because the accuracy, spelling and grammar is generally much better for subtitles than for STT transcripts, this means that the quality of the NER step is potentially much better as well.

For this task, we divided the videos into fixed-length segments of approximately 30 seconds, and applied a named-entity-based version of the Jaccard similarity [58]. We calculate the Jaccard similarity between two segments A and B as follows:

$$Jaccard_{NE}(A, B) = \frac{|\{e : e \in E(A) \cap E(B)\}|}{|\{e : e \in E(A) \cup E(B)\}|}, \tag{6.4}$$

where $E(A)$ and $E(B)$ denote the sets of extracted named entities from segment A and B, respectively.

Using this approach, the results were much more promising than in 2012, especially for the hyperlinking part of the task. For the search part, our approach achieved a MRR of 0.149 using the subtitles, whereas the best approach [77] achieved a MRR of 0.376 and the lowest scoring approach [168] obtained 0.09. For the linking part of the task, our approach achieved a MAP of only 0.045, which might seem disappointing at first, especially considering the best approach [21] – when only considering the MAP – scored over 0.5[1]. However, the results are much more promising when considering an alternative evaluation criterion: the precision of the top 10 retrieved segments, abbreviated as **P@10**. Our approach achieved a P@10 of 0.35, which means that of the first 10 linked segments, 3.5 are correct on average. The best approach in this regard [165] obtained a P@10 of 0.73, and the least successful approach in this aspect [30] scored 0.107. When observing the top 5 segments, the precision (P@5) of our approach even increased to 0.38, meaning that on average, two out of the five linked segments suggested to the user, were deemed as related. Considering the best approach used absolutely all provided features, the result with our relatively uncomplicated approach using only the named entities recognized in the subtitles is certainly promising.

In conclusion, these experiments teach us that although they were not able to match the accuracy of traditional similarity measures, our proposed named-entity-based measures show enough promise to justify further investigation. Especially the adapted Jaccard similarity shows precision in the first 5 linked segments that can be considered acceptable in a practical scenario, knowing that a real-world user would typically only view the first few suggestions he or she is presented with.

6.2.2.2 Test Scenario 2: News Clustering

As a second test for our named-entity-based similarity measurement, we consider the scenario where a set of news articles needs to be divided into clusters of semantically similar articles. Note that this is exactly the scenario that we used to evaluate our provenance reconstruction approach in Section 4.6. There, the cosine similarity with an alternative, named-entity-based weighting scheme was applied and lead to promising results. Indeed, it was shown in Section 4.6.6 that in 96% of the clusters created using this similarity measure, the contained articles were derived from the same source, and thus could be considered very similar. However, while the precision was very high, the recall was 37.0% at best, meaning that the clustering algorithm did not succeed in grouping all articles that were derived from one source in one single cluster.

These findings, together with the results from the MediaEval benchmarks discussed in Section 6.2.2.1, motivated to investigate the applicability of the named-entity-based Jaccard similarity in a news clustering scenario as well. In [68], we

[1]Note that the cited working notes paper for this approach reports a MAP of 1.0, which was discovered to be a mistake by the authors during the actual benchmark.

calculate the named-entity-based Jaccard similarity between two articles a and b as follows:

$$Jaccard_{NE}(a,b) = \frac{|N(a) \cap N(b)|}{|N(a) \cup N(b)|}, \tag{6.5}$$

where $N(x)$ is the set of all entities recognized in document x.

To evaluate this approach, we gathered a set of 851 English news articles[2], over the course of one week from the online newspaper The Guardian. All articles were semantically tagged with named entities[3] using the NER service AlchemyAPI[4]. From this set, we randomly selected 30 articles, which we used as reference articles. For each of these 30 articles, we then used our approach with the named-entity-based Jaccard similarity to find four relevant articles in the dataset. In other words, we generated a set of 120 links in total (4 for each of the 30 articles).

We then performed an evaluation using the Amazon Mechanical Turk (AMT) crowdsourcing platform. In total, 120 Human Intelligence Tasks (HITs) were created, one for each linked article pair. In each HIT, the AMT worker was presented an article-pair, and was asked to rate the relatedness of the articles' content on a 5-point Likert scale. The scale had the following scores:

1	2	3	4	5
Not related at all	Slightly related	Moderately related	Highly related	Completely related (almost the same)

As a preventive measure against spam, we also asked the AMT workers to explain *why* they thought the articles were (un)related, as well as a short summary of the reference article. Additionally, we filtered out all HITs that were submitted in less than 30 seconds, had an empty explanation or summary, or exhibited an obvious indication of being automatically generated (such as identical, generic responses over multiple HITs). Each article pair was evaluated by 10 different users, leading to 1200 evaluations in total. However, after applying the spam-control measures, we dismissed 76 answers, resulting in a final set of 1124 evaluations.

When normalized between 0 and 1, the evaluations on the Likert scale allow us to quantitatively measure the difference between the human assessment of relatedness, and the automatic assessment of similarity using our approach. We define the average similarity score of all article pairs as $S_{Jaccard}$, and the average evaluation score S_{Likert} as follows:

$$S_{Likert} = \frac{\sum_{e \in E}(e-1)}{|E| \times 4}. \tag{6.6}$$

Here, E is the set of HIT evaluations for an article pair, and $e \in E$ one of those evaluations, its value ranging from 1 to 5. This means that evaluations of Likert level 5, 4, 3, 2 and 1 will correspond to scores of 1, 0.75, 0.5, 0.25 and 0, respectively. We observed an average absolute difference $|S_{Likert} - S_{Jaccard}|$ of

[2]List of URIs to the articles used in the evaluation corpus: `http://bit.ly/1hqdKi5`

[3]URIs + extracted named entities: `http://bit.ly/1bK6CoE`

[4]`http://www.alchemyapi.com/`

0.198 between all the evaluations per article pair to the assessments made by our approach, which corresponds to a difference of 1 Likert level at most. However, we also observed that the error varied positively or negatively for each article pair, meaning that it cannot be automatically corrected for.

Apart from this value, we also calculated *precision* and *recall* values for each level of the Likert scale, by observing the number of **true positives (TP)**, **false positives (FP)**, and **false negatives (FN)** per Likert level, defined as follows:

TP: number of article pairs correctly assigned to this Likert level;

FP: number of article pairs incorrectly assigned to this Likert level;

FN: number of article pairs incorrectly assigned to a different Likert level.

We assign each Likert level to a range of possible values, as indicated in Table 6.1. The precision (P) and recall (R) of each Likert level can now be calculated as $P = \frac{TP}{TP+FP}$ and $R = \frac{TP}{TP+FN}$, respectively.

Likert level	Score Range	HA	AA	TP	FP	FN	**P**	**R**
1	$[0, 0.2[$	73	61	38	23	35	**0.623**	**0.521**
2	$[0.2, 0.4[$	18	50	7	43	11	0.140	0.389
3	$[0.4, 0.6[$	18	8	1	7	17	0.125	0.056
4	$[0.6, 0.8[$	9	1	1	0	8	1.0	0.111
5	$[0.8, 1]$	2	0	0	0	2	0	0

Table 6.1: Precision (P) and recall (R) values for each Likert level, mapped to its corresponding range of assessment scores. Additionally, the no. of human assessments (HA), automatic assessments (AA), true positives (TP), false positives (FP), and false negatives (FN) is shown.

We calculated these values for each Likert level, as shown in Table 6.1. When observing these results, it is clear that the precision and recall of the Jaccard metric is only acceptable in the lowest range of relatedness as assessed by the users (Likert level 1). In fact, it seems that the majority of article pairs were classified by the approach in the $[0, 0.4[$ range. This is surprising, because the dataset consisted of the articles deemed most relevant to the reference articles by the approach. However, the average score assigned to all article pairs by the AMT workers was as low as 0.237, which corresponds to a Likert level of "2: slightly related". This means that our dataset was biased towards less related articles, and that the approach simply did not have enough highly related articles to choose from.

Another possible explanation for the lower precision in the ranges above 0.2, is that the named-entity-based Jaccard measure does not scale in the same way as the human assessment. Although the average absolute difference of nearly *20%*

between the human and the automatic assessment potentially supports this, further experiments will need to be performed in future work.

Lastly, a small correlation was observed between the minimum number of named entities recognized in the article pairs and the absolute difference in measured relatedness score and calculated similarity score. This indicates that the approach might not be suitable for texts where few or no named entities can be detected. Additionally, this stresses the importance of the quality of the NER service.

6.2.2.3 Discussion and Next Steps

With the results we have at hand, we can conclude that our proposed named-entity-based document similarity measures have potential in certain scenarios, but still lack maturity to compete with traditional approaches. A possible weak point of our approaches is that they suppose two documents to share at least one common entity in order to measure a meaningful similarity value. However, in many cases this might be sub-optimal, especially when few named entities are recognized in one or both of the documents.

As opposed to the aforementioned approaches, so called *adaptive distance-based* similarity measures [15, 16] provide the opportunity to define semantic similarity between documents in a flexible and indexable manner, even when the documents share no common entities. Examples of such distances include the Earth Mover's Distance (EMD) [163], the Signature Quadratic Form Distance (SQFD) [17], and the Signature Matching Distance (SMD) [14]. However, these distances do require a measure of similarity or dissimilarity between individual named entities. Since named entities are typically linked to a resource in a knowledge graph, this would ideally be achieved using a graph-based similarity measure or distance. Therefore, before we can adapt the aforementioned adaptive distance-based measures to work with named entities, we must first investigate these graph-based semantic distances, which constitutes a research domain on its own.

6.2.3 Graph-based Semantic Distances

Calculating the semantic distance between two resources in a knowledge graph can be achieved by different strategies. The first strategy is **ontology-based**, where the distance is calculated based on the number of edges in the shortest path between two entities in their underlying hierarchical ontology [159]. The second is **link-based**, where the distance is calculated based on the number of direct and indirect connections between two entities in their graph structured data store [155]. Finally, we developed a **shared-links-based** approach, where the distance is calculated based on the number of shared connections [93].

The latter approach is where we focus our efforts, by developing a distance called the Normalized Semantic Web Distance (NSWD) [51]. We named the distance as such, because its principle is based on the so called Normalized Web Distance (NWD) [35], which exploits the search capabilities of Web indexing engines,

such as Google [34]. The NWD infers semantic distance based on the difference in page count for two search terms occurring separately and together. The basic principle is: if two terms occur on the Web together almost as often as they do separately, their semantic distance is likely to be low. More formally the authors of [35] define the NWD as follows:

Definition 6.1
Let W be the set of pages indexed by an arbitrary search engine able to return the (approximate) number of indexed pages containing a certain search term. For each search term x, let $\mathbf{X} \subseteq W$ denote the set of pages containing x. For two search terms x and y, we define the following frequency function f:

$$f(x) := |\mathbf{X}|$$
$$f(x,y) := |\mathbf{X} \cap \mathbf{Y}|$$

The NWD is then defined as:

$$NWD(x,y) = \frac{\max\{\log f(x), \log f(y)\} - \log f(x,y)}{\log N - \min\{\log f(x), \log f(y)\}},$$

where N is the estimated total number of pages indexed by the search engine[5].

What we did for the NSWD, is expand this principle to knowledge graphs. It leverages the principle of estimating the degree of co-occurrence between two concepts while simultaneously incorporating semantic awareness into the approach. Instead of considering the human-understandable Web (accessed through a search engine), we consider a machine-understandable knowledge graph on the Semantic Web (accessed through a query client, e.g., a SPARQL endpoint). In this way, the NSWD advances from possibly ambiguous natural language terms to unambiguous concepts, which are identified by URIs, as input.

Formally, we model a knowledge graph (V, T) as a set of nodes V and a set of directed triples $T \subseteq V \times P \times V$ that is built over a set of predicates P. To calculate the NSWD, we define the following sets of nodes $V_\lambda \subseteq V$ for $\lambda \in \{in, out, all\}$ in a knowledge graph (V, T) with respect to a certain node $x \in V$:

$$V_{in}(x) := \{v \in V | (v,p,x) \in T\}$$
$$V_{out}(x) := \{v \in V | (x,p,v) \in T\}$$
$$V_{all}(x) := V_{in}(x) \cup V_{out}(x)$$

A more straightforward way to put this is that the set $V_{in}(x)$ comprises distinct nodes with at least one link – regardless of the predicate – pointing to node x, whereas the set $V_{out}(x)$ contains all distinct nodes where node x points to. Thus, the cardinality of $V_{out}(x)$ corresponds to the number of distinct object nodes in

[5]Note that ideally, N is equal to $|W|$. However, in reality it is often impossible to determine the exact number of indexed pages of a search engine, since this is extremely dynamic information. Therefore, an estimate is used.

all triples with x as subject, while the cardinality of $V_{in}(x)$ reflects the number of distinct subject nodes in all triples with x as an object. The set $V_{all}(x)$ is the union of the two. Note that in practice, the calculation of these sets is generally easy to implement using a query language for knowledge graphs, such as SPARQL. This way, links to a concept in the knowledge graph are treated as an *occurrence* of that concept. As N, (an estimate of) the total number of concepts in the knowledge graph $|V|$ is used.

Based on these sets, we can now define the *Normalized Semantic Web Distance* $\mathrm{NSWD}_\lambda(x, y)$ with respect to parameter $\lambda \in \{in, out, all\}$ between two nodes $x, y \in V$ from a knowledge graph (V, T) as follows:

Definition 6.2

$$\mathrm{NSWD}_\lambda(x, y) = \frac{\max\{\log|V_\lambda(x)|, \log|V_\lambda(y)|\} - \log|V_\lambda(x) \cap V_\lambda(y)|}{\log|V| - \min\{\log|V_\lambda(x)|, \log|V_\lambda(y)|\}}$$

As can be seen in the definition above, the NSWD_λ makes use of the direct semantic context of the nodes in their knowledge graph. The parameter $\lambda \in \{in, out, all\}$ specifies which portion of the semantic context is taken into account when determining the dissimilarity of two nodes.

As a preliminary experiment to test the feasibility of this distance, we made a first implementation of it using $\lambda = in$ based on the Freebase knowledge graph, dubbed the Normalized Freebase Distance (NFD) [93]. To calculate the NFD, we set up a Virtuoso SPARQL endpoint and used the Freebase RDF dump of March 16, 2014 containing over 1.9 billion triples.

The cardinality $|V_{in}(x)|$ for a concept with URI x is determined using the following SPARQL query:

```
SELECT COUNT(DISTINCT ?s) WHERE { ?s ?p <x> }
```

The cardinality $|V_{in}(x) \cap V_{in}(y)|$ is determined using:

```
SELECT COUNT(DISTINCT ?s) WHERE
{ ?s ?p1 <x> . ?s ?p2 <y> }
```

The returned triples were filtered for duplicates by removing triples that used the predicates `rdf:type` and `rdfs:label`, and forced the subject to be an URI.

To validate the effectiveness of the NFD, we are particularly interested in ambiguous concepts that would confuse traditional search engines. To that end, we have calculated the NFD between three types of fish, and the word *bass guitar*. Here, we expect that search engines will not see the difference between the fish species *bass* and a *bass guitar*. To test this, we also calculated the NWD for these concepts, using the search engine Bing. We provide an overview of the resulting distances in Tables 6.2 and 6.3 for the NWD and NFD, respectively.

	Salmon	**Trout**	**Bass**	**Bass guitar**
Salmon	0	0.072	0.133	0.283
Trout	0.072	0	0.123	0.247
Bass	0.133	0.123	0	**0.086**
Bass guitar	0.283	0.247	**0.086**	0

Table 6.2: Distance matrix for four concepts, using the NWD.

	Salmon (09777)	**Trout (0cqpb)**	**Bass (0cqvj)**	**Bass guitar (018vs)**
Salmon (09777)	0	0.070	0.087	0.274
Trout (0cqpb)	0.070	0	0.070	0.269
Bass (0cqvj)	0.087	0.070	0	**0.276**
Bass guitar (018vs)	0.274	0.269	**0.276**	0

Table 6.3: Distance matrix for four concepts, using the NFD. For each concept, the unique Freebase identifier is specified.

We can make two important observations from this small-scale experiment. First, we can see that the distances between the first three concepts are roughly of the same magnitude for both the NWD and NFD. This means that at least for this experiment, the NFD can be considered as a realistic distance, when compared to an established distance such as the NWD. Second, the distances between *bass guitar* and the other three concepts (the last row in each table) are of the same magnitude for both distances, except for the comparison with *bass*. There, we can observe that the NWD results in a very low distance, and thus considers *bass* and *bass guitar* to be very similar. The NFD, however, results in a distance that is similar to the distance between *bass guitar* and the other two fish. So, as expected, the NFD captures the ambiguity much better than the NWD in this case.

Additionally, we argue that this approach offers a computational advantage as well. Indeed, since the Freebase knowledge graph contains approximately 2 billion triples, using an alternative graph-based measure – such as calculating the length of the shortest path over the graph – could become a very computationally expensive task. The calculation of the NFD on the other hand, only requires the execution of three count queries, which triple stores can be optimized for. Therefore, in future research, we plan to conduct more extensive experiments, paying more detailed attention to both the accuracy and efficiency of the proposed distance metric in a variety of use cases.

Whereas our preliminary experiment focused on the use of Freebase, one could imagine applying this principle on the scale of the entire Web, using other knowledge graphs, or even combinations of multiple graphs. New developments in the field of Web-scale querying could make this possible in the near future [187]. That way, the NSWD would share much of the flexibility and power of the NWD, with the added benefit of semantic awareness.

To further investigate this, we also implemented the NSWD on the DBpedia knowledge graph, and evaluated it using a commonly used benchmark in the field of word similarity: the Miller-Charles dataset [141]. This dataset consists of 30 term-pairs that were judged for similarity by 38 people. While using lexical terms for evaluation of a distance in knowledge graphs is not ideal, and this is a relatively small dataset, it offers an insight to how humans judge the similarity between these terms, and more importantly, it gives us a number of related approaches for direct comparison, as it is very commonly used in this field. Therefore, the Miller-Charles benchmark provides the best starting point for external validation compared to established approaches. However, before we can use it, a disambiguation strategy has to be decided upon, as many of the terms are highly ambiguous, in the sense that they can correspond to more than one resource URI in the knowledge graph. To choose one to use for the NSWD calculation, we used three disambiguation strategies:

manual: manually pick a disambiguated resource URI, or suggest an alternative URI (human judgment);

count-based: use the resource URI with the highest V_{in}, V_{out} or V_{all} (depending on whether the NSWD, NSWD_{out} or NSWD_{all} is calculated, respectively);

similarity-based: use the resource URI leading to the smallest distance (only possible in the context of a pairwise comparison);

Note that it is possible that the correct disambiguation cannot be determined due to the non-completeness of the dataset. In our evaluation, we calculated the distances using all aforementioned disambiguation strategies, to see which leads to the best results.

Additionally, it must be noted that the NSWD is a *distance*, meaning that the more semantically related two concepts are, the smaller their distance is. However, in many cases – including the Miller-Charles benchmark – the opposite is desired: i.e., *similarity* must be measured. To do this, we must know the maximum value the NSWD_λ can have. Empirically, we determined that no matter what λ is set to, this value can be calculated as[6]:

$$\forall x, y \in V : \text{NSWD}_{max} = \frac{log(\lfloor \frac{|V|}{2} \rfloor + 1)}{log\,|V| - log\lceil \frac{|V|}{2} \rceil}$$

Knowing this, if the NSWD were normally distributed in the range $[0, \text{NSWD}_{max}]$, we could just scale it linearly using NSWD_{max} and subtract it from 1. However, we observed that the values that occur most frequently are in the $[0, 1]$ range. These are also the distance values that are most interesting in practical scenarios, such as recommendation systems. Scaling these values linearly using NSWD_{max} would lead to a situation where the majority of distances would be in the range of $[0, \frac{1}{\text{NSWD}_{max}}]$, which is not very useful. Keeping this in mind, we define the NSWD-based similarity Sim_{NSWD} as follows:

[6]For more details, we refer to our most recent publication on the NSWD [52].

Definition 6.3

$$Sim_{NSWD_\lambda}(x,y) := \begin{cases} 1 - d(x,y) \times (1-c), & \text{if } d(x,y) \in [0,1] \\ (1 - \frac{d(x,y)}{NSWD_{max}}) \times c, & \text{if } d(x,y) \in]1, NSWD_{max}] \end{cases}$$

$$\text{with } d(x,y) = NSWD_\lambda(x,y) \text{ and } c = \frac{1}{NSWD_{max}}$$

This way, the most semantically significant distances – those between 0 and 1 – get mapped to the similarity range $[c, 1]$ with minimal scaling, and the distances higher than 1 get mapped to the similarity range $[0, c[$ with significant scaling. Note that if NSWD_{max} is accurately calculated, $Sim_{\text{NSWD}_\lambda}(x,y)$ is normalized between 0 and 1.

Now that we can calculate similarity, our evaluation process consists of the steps below, for each of the 30 term-pairs in the Miller-Charles dataset.

1. Both terms are disambiguated, using the manual and automatic approaches. This results in 3 URI disambiguation options for each term: (a) manually selected, (b) based on the highest link-count, and (c) based on the highest similarity with the other term.

2. For each of the three URI disambiguation options, the NSWD_{in}, NSWD_{out}, and NSWD_{all} are calculated.

3. The above results in 9 distances (three for each variant of the NSWD), which are converted to similarities.

These steps result in 9 similarity assessments for each of the 30 term-pairs, each value calculated with a different combination of disambiguation option and NSWD variant. These values are compared to the human-assessed scores from the Miller-Charles dataset by calculating the Pearson correlation coefficient.

As a baseline, we added a similarity score based on the NWD to the evaluation results. This NWD-based similarity was calculated as $1 - \text{NWD}(x,y)$, with NWD(x,y) calculated using the Microsoft Bing Search API[7] as a search engine.

We calculated the Pearson correlation coefficient between the Miller-Charles scores and the NWD baseline, as well as the three NSWD variants. Each NSWD-based similarity measure was tested with three disambiguation strategies: manual (M), count-based (C), or similarity-based (S), using two widely used knowledge graphs: Freebase and DBpedia. We compare our results with the reported correlations on the same benchmark for two well-performing measures from literature: the Wikipedia Link-based Measure [142], and the Jaccard similarity as calculated in [120]. The results are shown in Table 6.4. Higher correlation indicates a stronger positive relationship between the human-assessed scores and calculated similarities. To enable reproducibility of the results, we provide online access to the files generated by our evaluation software, including all disambiguated URIs,

[7] `https://datamarket.azure.com/dataset/bing/search`

Miller-Charles scores, and similarity scores. The results are available at `http://semweb.datasciencelab.be/nswd/evaluation/`, where the JSON file `mc30_results_freebase.json` holds the results for Freebase, `mc30_results_dbpedia.json` holds those for DBpedia, and `mc30_results_bing.json` holds those for the NWD using Bing.

	$Sim_{\text{NSWD}_{in}}$			$Sim_{\text{NSWD}_{out}}$			$Sim_{\text{NSWD}_{all}}$		
	M	C	S	M	C	S	M	C	S
Freebase	0.42	0.25	0.29	0.57	0.43	0.57	0.55	0.24	**0.58**
DBPedia	0.60	0.44	0.55	0.56	**0.69**	0.62	0.66	0.58	0.68
NWD	**0.23**								
WLM*	**0.70**								
Jaccard*	**0.882**								

*using a different disambiguation strategy

Table 6.4: Pearson correlation coefficient on the Miller-Charles benchmark for the NSWD similarity variants on the Freebase and DBpedia knowledge graphs, the Normalized Web Distance using Bing, Wikipedia Link-based Measure, and Jaccard similarity.

Note that for all distance and disambiguation options, the NSWD-based similarities achieved a higher correlation than the NWD-based similarity at the time of writing, with a maximum of 0.58 for Freebase and 0.69 for DBpedia. There is no consistent trend in which disambiguation strategy performed best. Overall, the NSWD_{all} seemed to perform best, taking most of the semantic context of a node into account. None of the NSWD variants was able to perform better than the reported results of the WLM and Jaccard similarity. However, note that for these reported results, a different disambiguation strategy was applied. For the WLM as reported in [142], the disambiguation of the Miller-Charles terms was performed using a weighted combination of commonness, relatedness, and occurrence together in a sentence. However, the authors of [142] did not disclose the exact weighting scheme they used, nor the disambiguated terms. Do note that commonness and relatedness of the terms in a term-pair are factors that we also consider, by applying the disambiguation strategies using the highest link-count and highest similarity, respectively. Therefore, we can safely assume that the reported correlation of 0.70 is useful to compare with our results. In case of the Jaccard similarity as calculated in [120], disambiguation was left ad-hoc to a search engine, which makes it impossible for us to reproduce.

The quality of the knowledge graph greatly affects the performance and applicability of the NSWD. For example, during the disambiguation of the Miller-Charles dataset, we found that DBpedia often lacks a simple description of various concepts. For example, the concepts "journey" and "voyage" – resulting in the resources `dbpedia:Journey` and `dbpedia:Voyage`, repectively – both link to many disambiguation options, but none of these options capture the most straight-

forward meaning of the concepts. When inspecting the corresponding human-understandable Wikipedia pages, it becomes clear that both "journey" and "voyage" are supposed to be disambiguated to the concept "travel", with resource URI `dbpedia:Travel`. Unfortunately, these links are not currently included in DBpedia. As a result, automatic disambiguation methods (such as the count-based and similarity-based disambiguation) that only follow links included in the knowledge graph will never find the correct result, leaving manual disambiguation by a human as the only correct option in these cases. In a number of other cases, no resource exists to represent a concept, as was the case with the terms "lad" and "madhouse". The lower connectivity between concepts in DBpedia also resulted in many of the distances defaulting to NSWD_{max} during the evaluation. Freebase was found to be richer in this regard, as we found less zero-scores, and smaller variances in the similarities than in the DBpedia results. Concepts in Freebase were missing for fewer terms than in DBpedia, and there were less cases where two terms in a term-pair corresponded to the same URI. Still, terms such as "lad" and "madhouse" have no direct equivalent on Freebase.

6.3 Reflections and Future Work

Our proposed approach for automatic content value assessment combines relevance and trust assessment in a novel way. When completed, it will allow us to make fine-grained assessments, which are useful to both content producers and consumers. Traditional recommendation systems focus on providing the user with a ranked list of results. Our work augments this traditional approach, by presenting the user with the strong and weak points of content, together with references to other, relevant content. Additionally, the use of Semantic Web technologies allows for a generic, easily adaptable approach to content value, rather than the typically domain-specific approach of traditional recommender systems.

If we revisit our proposed value assessment workflow as shown in Figure 6.1, we can already instantiate many of the components using contributions made in this thesis. We considered several types of content, including news articles, social media content, software, and videos. We applied several techniques to model this content by extracting its features and linking it, including NER, the VSM, Linked Data, and our two PROV extensions. Additionally, we provided several methods to obtain the provenance associated with this content: by *exposing* it using tools such as Git2PROV, TinCan2PROV and the RML mapping refinement workflow, by *reconstructing* it when it is missing, or by *retrieving* it from services such as PROV-Pings. We also proposed a method to assess the trustworthiness of the content based on its provenance and reputation (e.g., through Web of Trust) in the form of the "Oh, Yeah?"-button. Finally, we evaluated several methods to assess the relevance of content through semantic similarity to a reference corpus (such as WikiNews), including the cosine similarity, Jaccard similarity and NSWD. We illustrate how these techniques fit in our proposed workflow in Figure 6.2.

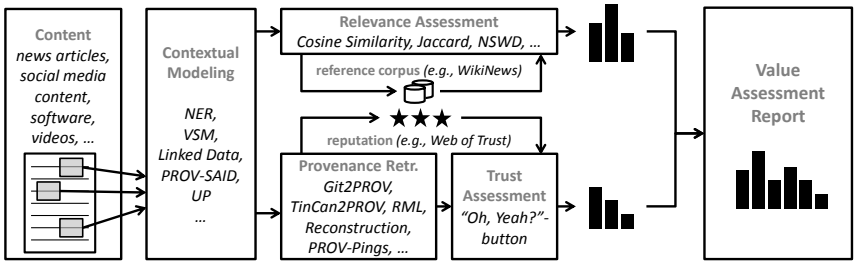

Figure 6.2: High-level overview of our value assessment approach, with the components instantiated using the techniques discussed in this dissertation.

An important aspect that remains to be addressed, is how to deal with the complex psychology behind relevance and trust. Indeed, relevance and trust are subjective, relative to the person making the inquiry. Therefore, we will have to consider modeling the interests and trust relations of the user as well.

Additionally, accurate relevance assessment remains a challenge. As can be expected, the adaptation and evaluation of adaptive distance-based similarity measures in combination with the strategies of comparing named entities over knowledge graphs, as explained in Section 6.2.3, is a challenging task which requires the definition of a generic ground truth dataset or gold standard. Unfortunately, most current benchmarks either offer a domain-specific evaluation set, which increases the complexity of evaluating the aforementioned approaches in a general scenario, or are too small and/or targeted towards traditional approaches to yield usable results. Therefore, the development and evaluation of a graph-based semantic distance suitable for use in an adaptive distance-based similarity measure for documents is a priority for our future work. To achieve this, we will further investigate the merit of the NSWD variants by applying them on more domain-specific knowledge graphs. We suspect that if the domain knowledge of the graph is high, the NSWD variants should be aware of these semantics and perform better than traditional approaches.

You can't connect the dots looking forward; you can only connect them looking backwards. So you have to trust that the dots will somehow connect in your future. You have to trust in something - your gut, destiny, life, karma, whatever. This approach has never let me down, and it has made all the difference in my life.

Steve Jobs

7 Conclusion

At the end of the first chapter of this dissertation, we posed a number of research questions, and created a hypothesis for each of them. In this final chapter, we revisit those questions and hypotheses, using the new-found knowledge gathered through our work. We also list the open questions that remain after our doctoral research, and on what we will focus our future efforts.

7.1 Review of the Research Questions

The main research question which guided our work was: *"How can we enable automatic assessment of the trustworthiness of content on the Web?"*. Our hypothesis was that basic automatic trustworthiness assessments can be made by accessing the content's provenance and the reputation of the entities, agents and processes involved. Through the implementation of the "Oh, Yeah?"-button in Chapter 5, we showed that it is indeed possible to automatically provide a user with trustworthiness assessments when the provenance of the content he or she is interested in is accessible. This validates the main hypothesis, but raises a more challenging issue: for the vast majority of content on the Web the provenance is not available in an interoperable way, incomplete, or missing altogether.

We tackled this issue by investigating three new research questions. First, we investigated: *"How can provenance be modeled in an interoperable way across multiple use cases?"*. We were not the only ones who asked ourselves this question: the W3C Provenance Working Group had already begun the process of standardizing their provenance data model now known as 'PROV', to which we actively contributed. The details of PROV are explained in Chapter 2. We pro-

vided several of the first implementations of the recommended data model, and we provided two extensions: one to model uncertain provenance and provenance of uncertainty, as explained in Section 2.3, and one to model the provenance of information diffusion on social media, as discussed in Section 2.2. Therefore, we can safely say that we validated our hypothesis that provenance can be modeled in an interoperable way by using the W3C PROV standard, extended when needed for specific use cases.

Second, we researched the question *"When provenance information is obscured in a non-interoperable way, how can we expose it?"*. Our hypothesis was that when obscured in a non-interoperable way, provenance can be exposed by automatically mapping it to an interoperable form. Indeed, as we showed in Chapter 3, this is certainly possible. We illustrated this by exposing provenance in three distinct use cases: version control systems, learning experiences, and RML mappings. Exposing the provenance in all these use cases as W3C PROV creates an important added value, since it is now no longer locked inside the scope of a specific use case. This means that provenance of learning experiences can now be enriched by the provenance of the software that supported them, or even of the process that generated the data that was used in the learning process. This corresponds to the underlying vision behind the W3C PROV data model: allowing the user to focus or *'zoom in'* on specific aspects of provenance, while maintaining a generic, *'zoomed out'* overview of workflows and creation processes.

Third, we asked *"When provenance information is incomplete or missing, how can we reconstruct it?"*. This proved to be a very challenging question, because at the time of writing, there are only a handful of researchers in the world working on this problem. We decided to investigate a seemingly counter-intuitive hypothesis: since content that shares the same provenance is often semantically similar, we made the assumption that when (partially) missing, provenance can be reconstructed based on the content's semantic similarity to other content in the same environment. In fact, we knew in advance that this hypothesis would never be 100% correct, since many counterexamples can be found where content is similar, but shares no provenance. However, in many cases, provenance that is correct *to some degree* is better than no provenance at all. Indeed, if a human is presented with a number of options for a provenance trace of a document, it becomes much easier for him or her to assess whether these options include the true provenance of that document, whereas manually searching through an entire dataset is a virtually impossible task. In Chapter 4, we showed that in many cases, semantic similarity can work to reconstruct provenance derivations to an acceptable degree of precision. We also contributed to the advancement of this research field by organizing a benchmark, providing two gold standard datasets.

Finally, the research on provenance reconstruction also caused us to investigate semantic similarity, and overall content relevance assessment. As we investigated both semantic similarity and Semantic Web technologies, we posed the following research question: *"How can we improve existing methods to automatically assess semantic similarity and/or relevance using Semantic Web technologies?"*. More specifically, we decided to investigate whether existing methods to automatically

assess semantic similarity and/or relevance of content could be improved based on extracted semantic features such as named entities. While this is still ongoing research, we did achieve some promising results with adaptations of the cosine similarity, Jaccard similarity, and the newly proposed Normalized Semantic Web Distance, as discussed in Chapter 6. As an outlook to the future, this research topic also fits into a broader picture of automatic content value assessment on the Web, when combined with our main work on provenance and trust assessment.

7.2 Future Work

In Chapter 6, we outlined our proposed automatic value assessment approach, and the building blocks we have already started research on. The main goal of our future work will be the completion of the remaining components needed to make this value assessment approach a reality. More specifically, we discern three research topics that remain to be investigated before the approach becomes applicable as a whole in a real-world scenario:

1. advancement of relevance assessment through semantic similarity measures;
2. integration into content publishing and consumption workflows;
3. subjective value assessment through trust relations and user interests.

First, we will focus our efforts on the creation of an adaptive, distance-based document similarity measure such as the SQFD using named entities. As discussed in Section 6.2.2.3, this kind of measure requires an inter-entity distance, to guarantee that documents that do not share any entities can still be compared and result in a useful similarity score. We have already started research in this direction, in the form of the NSWD. The next steps are: 1) to conduct a more thorough evaluation of the NSWD for comparing concepts, and 2) to integrate the NSWD and SQFD, and test their combined performance when comparing documents.

Second, we argue that the provenance, trust, and value assessment components of our approach will reach their highest potential when integrated into an ecosystem of content publishing and consumption workflows. Here, we especially consider the recent advancements made at our lab with regard to decentralized publishing workflows. More specifically, we investigate the integration of our proposed approaches with the RML [72] and Linked Data Fragments [187] workflows. For RML, which allows mappings from semi-structured data to Linked Data, the foundations for provenance capture and trust assessment during the mapping and validation process have already been laid, as explained in Section 3.5. In ongoing research, a similar provenance capture layer is being added to the Triple Pattern Fragments software, which enables publishing queryable Linked Data in a much more scalable way than SPARQL endpoints do. Additionally, it is being investigated how to generate provenance traces of data usage throughout this system. Ultimately, this will enable us to create an accumulative ‘trust overlay’ over query results.

Third, we will investigate the subjective nature of trust and value assessments, in collaboration with social scientists. On the one hand, we want to model the subjective aspects of trust and value assessment, using social networks. For example, people often place trust in something or someone when their close friends do. The same phenomenon occurs in professional networks: people or resources will often be recommended based on past experiences from colleagues. On the other hand, the personal interests and activities of a person will often affect his or her judgment as well. If a contextual model can be built for *people* as well as *content*, our approach for relevance assessment could be applied to mimic this behavior.

Note that besides their academic importance, all three of these research directions have industrial applications as well. Semantic similarity measurement is a key component in virtually all commercial indexing and search applications, whose importance only keeps increasing due to the ever-growing amount of content available on the Web. The strong requirements that are being imposed on data published on the Web by its consumers mean that data publishers can use all the help they can get when it comes to data quality, provenance, trust, and value assessment. Furthermore, the integration of social networks in our daily lives has resulted in rich insights into the contextual information around each individual, and is transforming the way content and products are recommended to consumers.

As a final remark, we want to make clear that the vision of automatic trust and value assessment we have worked towards throughout this dissertation will not be realized overnight, nor at one lab at one university. The Web as we know it today was developed through a process of continuous iteration, by thousands of researchers from various institutions worldwide collaborating under the wing of standardization bodies such as the W3C. The same process will be necessary to enable a Web where information can be automatically assessed for its trustworthiness and its value to a particular consumer. However, we are confident that in the next decade, the components necessary to make a trusted Web a reality will slowly but steadily evolve from research projects to Open Web standards. Eventually, they will become an essential part of browsing the Web, perceived as natural as a browser's search bar or 'share'-button.

7.3 Overview of Other Research Activities

This dissertation describes the main research track I have followed over the last five years. However, during these five years, I combined this research track with several other activities, including projects, standardization efforts, international collaborations, and parallel research directions. As can be expected, including a full description of all these activities and research results in this dissertation would severely affect its focus and readability. Therefore, I provide a brief overview of these activities in this section, as well as a list of all publications I have authored and co-authored at the time of writing.

7.3.1 Projects and Funding

In 2011 and 2012, I primarily worked on the IWT-funded project *SMIF (Smarter Media in Flanders)*, during which I first started investigating semantic similarity, provenance, and trust assessment. The results in Sections 4.6 and 6.2.2.1 were partially achieved in the scope of this project. Additionally, I contributed to the collaborative production platform CHAMP at the former VRT Medialab in 2011. In 2013 and 2014, I was involved in the project *"Uitgeverij van de Toekomst" (Publisher of the Future)*, in collaboration with Boek.be, the umbrella-organization for the Flemish book industry. Also in the context of digital publishing, I contributed in an advisory role to the iMinds-MiX project *e-Strips* on digital comic books, and in 2014 and 2015, I contributed to the iMinds-ICON project *Edutablet*, which investigated the possibilities of digital learning in schools – in particular using tablets. TinCan2PROV, as described in Section 3.4, was developed in the scope of this project. Since 2015, I have been working on the iMinds-ICON project *COMBUST*, where I am researching provenance-based trust assessment and crowdsourced veracity enhancement in a Linked Data publishing platform, as well as taking up a coordinating role. In the scope of this project, we adapted the RML refinement workflow to expose provenance, as described in Section 3.5.

7.3.2 Standardization Efforts

In 2012 and 2013, I was a member of the W3C Provenance Working Group, where – together with colleague Sam Coppens, who had been involved in the standardization process since its beginning – I actively contributed to the PROV family of documents. In particular, I contributed to the PROV Data Model [149] and PROV-Constraints [32] Recommendations, and was lead editor for the PROV-Dictionary Note [57], which extends the model to include collections that consist of key-value pairs. The details of the PROV standard are explained in Chapter 2.

In the context of the Publisher of the Future project, I also became a member of the W3C Digital Publishing Interest Group[1], which acts as a technical forum for experts in the digital publishing ecosystem. Similarly, in the context of my work on TinCan2PROV (see Section 3.4), I joined the Experience API (xAPI) Vocabulary & Semantic Interoperability Community Group[2].

7.3.3 International Collaborations

Over the years, several international collaborations played an important role in advancing the research described in this dissertation. In particular, our collaboration with VU Amsterdam resulted in Git2PROV, our approach to expose the provenance of version control systems explained in Chapter 3. Our joint publication on this approach [63] won the 'Best Demonstration' award at the International Semantic Web Conference in 2013. Furthermore, during a one-month internship

[1] `http://www.w3.org/dpub/IG/wiki/Main_Page`
[2] `https://www.w3.org/community/xapivocabulary`

at the Web & Media Group at the VU, a partnership was established to chair the 2014 and 2015 editions of METHOD: the International Workshop on Methods for Establishing Trust of (Open) Data.

Furthermore, our publications with the University of Mannheim [64] and the University of Southampton [56] provided the foundations for the approaches for accessing and interpreting provenance discussed in Chapter 5.

More recently, our collaboration with the University of Freiburg led to publications on the provenance of information diffusion in social media [66, 175]. Chapters 2 and 4 both include sections that are partially based on these publications.

Finally, our work together with RWTH Aachen on knowledge-graph-based semantic distances resulted in several publications [50, 51, 93], which form an important basis for the next steps towards an automatic value assessment approach, as explained in Chapter 6.

7.3.4 Parallel Research

Apart from the main research direction described throughout this dissertation, I have pursued other research directions as well.

Synchronization between Music and Movement For a short time after obtaining my masters degree, I continued work on my masters thesis – which investigated the interplay between music, movement, and social interaction – in collaboration with the musicology department of Ghent University. The work in this thesis was awarded with the Belgian Industrial Research & Development MSc thesis award[3], and led to two publications [69, 70].

Digital Publishing In the context of the Publisher of the Future project, I was part of a team researching the introduction of Semantic Web technologies to the digital publishing world. More specifically, we investigated the possibilities of the EPUB3 format for digital books, and how concepts from the Semantic Web could improve the reading experience [185], authoring [45], enrichment [47, 67, 85–87, 107], and discoverability [43, 44, 48] of eBooks.

7.3.5 Publications

At the time of writing, the research described in this dissertation, combined with the other activities mentioned in this section, has lead to a total of 39 publications. Of those 39, I wrote 21 publications as first author, consisting of one journal article (classified by Ghent University as A1) [69], four conference publications indexed in ISI Web of Science (P1) [49, 55, 56, 64], eleven other conference publications (C1) [50–52,60–63,65,66,68,70], and five conference abstracts (C3) [53,54,58,59, 67]. Additionally, I have contributed as a co-author to 18 publications, including

[3] http://www.birdbelgium.com/call2009/awards-call

two journal articles (A2) [43, 45], two conference publications indexed in Web of Science (P1) [86, 93], twelve other conference publications (C1) [44, 46–48, 78, 85, 87, 133, 175, 184–186], and two conference abstracts (C3) [73, 178]. At the start of each chapter in this dissertation, a list is provided of the publications that were used as a basis for part of that chapter, and/or are relevant to the research it contains.

References

[1] Fabian Abel, Claudia Hauff, Geert-Jan Houben, Richard Stronkman, and Ke Tao. Semantics + filtering + search = twitcident. Exploring information in social web streams. In *Proceedings of the 23rd ACM conference on Hypertext and social media*, pages 285–294, 2012.

[2] Activity Streams Working Group et al. JSON Activity Streams 1.0, 2011.

[3] Ben Adida, Mark Birbeck, Shane McCarron, and Ivan Herman. RDFa Core 1.1 - Second Edition. W3C Recommendation 22 August. `http://www.w3.org/TR/2013/REC-rdfa-core-20130822/`, 2013.

[4] Ben Adida, Mark Birbeck, Shane McCarron, and Ivan Herman. RDFa Core 1.1 - Third Edition. W3C Recommendation 17 March. `http://www.w3.org/TR/2015/REC-rdfa-core-20150317/`, 2015.

[5] Ben Adida, Mark Birbeck, Shane McCarron, and Steven Pemberton. RDFa in XHTML: Syntax and Processing. W3C Recommendation 14 October. `http://www.w3.org/TR/2008/REC-rdfa-syntax-20081014/`, 2008.

[6] Advanced Distributed Learning (ADL) Initiative. Companion Specification for xAPI Vocabularies. `https://adl.gitbooks.io/companion-specification-for-xapi-vocabularies/content/`, 2015.

[7] Ailifan Aierken, Delmar B Davis, Qi Zhang, Kunal Gupta, Alexander Wong, and Hazeline U Asuncion. A multi-level funneling approach to data provenance reconstruction. In *IEEE 10th International Conference on e-Science*, volume 2, pages 71–74. IEEE, 2014.

[8] Mohammad Al Hasan, Saeed Salem, Benjarath Pupacdi, and Mohammed J. Zaki. Clustering with lower bound on similarity. In *Advances in Knowledge Discovery and Data Mining*, pages 122–133. Springer, 2009.

[9] Donovan Artz and Yolanda Gil. A survey of trust in computer science and the Semantic Web. *Journal of Web Semantics: Science, Services and Agents on the World Wide Web*, 5(2):58–71, 2007.

[10] Eytan Bakshy, Jake M Hofman, Winter A Mason, and Duncan J Watts. Everyone's an influencer: quantifying influence on Twitter. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 65–74, 2011.

[11] Raquel A Baños, Javier Borge-Holthoefer, and Yamir Moreno. The role of hidden influentials in the diffusion of online information cascades. *EPJ Data Science*, 2(1):1–16, 2013.

[12] Geoffrey Barbier, Zhuo Feng, Pritam Gundecha, and Huan Liu. Provenance data in social media. *Synthesis Lectures on Data Mining and Knowledge Discovery*, 4(1):1–84, 2013.

[13] Geoffrey Barbier, Zhuo Feng, Pritam Gundecha, and Huan Liu. Provenance data in social media. *Synthesis Lectures on Data Mining and Knowledge Discovery*, 4(1):1–84, 2013.

[14] Christian Beecks, Steffen Kirchhoff, and Thomas Seidl. Signature matching distance for content-based image retrieval. In *Proceedings of the 3rd ACM conference on International conference on multimedia retrieval*, pages 41–48. ACM, 2013.

[15] Christian Beecks, Steffen Kirchhoff, and Thomas Seidl. On stability of signature-based similarity measures for content-based image retrieval. *Multimedia Tools Appl.*, 71(1):349–362, 2014.

[16] Christian Beecks, Merih Seran Uysal, and Thomas Seidl. A comparative study of similarity measures for content-based multimedia retrieval. In *ICME*, pages 1552–1557, 2010.

[17] Christian Beecks, Merih Seran Uysal, and Thomas Seidl. Signature quadratic form distance. In *Proceedings of the ACM International Conference on Image and Video Retrieval*, pages 438–445. ACM, 2010.

[18] Tim Berners-Lee. Cleaning up the User Interface - The "Oh, yeah?"-Button. `http://www.w3.org/DesignIssues/UI.html`, 1997.

[19] Tim Berners-Lee. Linked data: Design issues. `http://www.w3.org/DesignIssues/LinkedData.html`, 2006.

[20] Tim Berners-Lee, James Hendler, Ora Lassila, et al. The Semantic Web. *Scientific American*, 284(5):28–37, 2001.

[21] Chidansh A. Bhatt, Nikolaos Pappas, Maryam Habibi, and Andrei Popescu-Belis. Idiap at mediaeval 2013: Search and hyperlinking task. In *MediaEval 2013 Multimedia Benchmark Workshop*. CEUR-WS, 2013.

[22] Christian Bizer and Richard Cyganiak. The TriG Syntax. `http://wifo5-03.informatik.uni-mannheim.de/bizer/trig/`, 2007.

[23] Christian Bizer and Richard Cyganiak. Quality-driven information filtering using the WIQA policy framework. *Web Semantics: Science, Services and Agents on the World Wide Web*, 7(1):1–10, 2009.

[24] Christian Bizer, Tom Heath, and Tim Berners-Lee. Linked data-the story so far. *Semantic Services, Interoperability and Web Applications: Emerging Concepts*, pages 205–227, 2009.

[25] Katarina Boland, Dominique Ritze, Kai Eckert, and Brigitte Mathiak. Identifying References to Datasets in Publications. In *Proc.of the 2nd Int. Confernce on Thory and Practic of Digital Libraris (TPDL)*, pages 150–161, Berlin, Heidlberg, 2012. Springer.

[26] Uri Braun, Simson Garfinkel, David A Holland, Kiran-Kumar Muniswamy-Reddy, and Margo I Seltzer. Issues in automatic provenance collection. In *Provenance and annotation of data*, pages 171–183. Springer, 2006.

[27] Dan Brickley and Ramanathan V. Guha (Eds.). RDF Schema 1.1. W3C Recommendation 25 February. `http://www.w3.org/TR/rdf-schema/`, 2014.

[28] Davide Ceolin, Paul Groth, Willem Robert van Hage, Archana Nottamkandath, and Wan Fokkink. Trust evaluation through user reputation and provenance analysis. In *8th International Workshop on Uncertainty Reasoning for the Semantic Web*, page 15, 2012.

[29] Davide Ceolin, Archana Nottamkandath, Wan Fokkink, and Valentina Maccatrozzo. Towards the definition of an ontology for trust in (Web) data. In *10th International Workshop on Uncertainty Reasoning for the Semantic Web (URSW 2014)*, page 73, 2014.

[30] Shu Chen, Gareth J.F. Jones, and Noel O'Connor. DCU linking runs at mediaeval 2013: Search and hyperlinking task. In *MediaEval 2013 Multimedia Benchmark Workshop*. CEUR-WS, 2013.

[31] James Cheney, Laura Chiticariu, and Wang-Chiew Tan. *Provenance in databases: Why, how, and where*, volume 4. Now Publishers Inc., 2009.

[32] James Cheney, Paolo Missier, Moreau, Luv (Eds.), and W3C Provenance Working Group. PROV-CONSTRAINTS: Constraints of the PROV Data Model. W3C Recommendation 30 April. `http://www.w3.org/TR/2013/REC-prov-constraints-20130430/`, 2013.

[33] James Cheney and W3C Provenance Working Group. Semantics of the PROV Data Model. W3C Note 30 April. `http://www.w3.org/TR/prov-sem/`, 2013.

[34] Rudi L. Cilibrasi and Paul Vitanyi. The Google similarity distance. *IEEE Transactions on Knowledge and Data Engineering*, 19(3):370–383, 2007.

[35] Rudi L. Cilibrasi and Paul Vitanyi. Normalized Web Distance and word similarity. *arXiv preprint arXiv:0905.4039*, 2009.

[36] Alberto Corbi and Daniel Burgos. Review of current student-monitoring techniques used in elearning-focused recommender systems and learning analytics. the Experience API & LIME model case study. *International Journal of Artificial Intelligence and Interactive Multimedia*, 2(7):44–52, 2014.

[37] Vasa Curcin, Simon Miles, R Danger, Y Chen, Richard Bache, and Adel Taweel. Implementing interoperable provenance in biomedical research. *Future Generation Computer Systems*, 34:1–16, 2014.

[38] Tolga Dalman, Michael Weitzel, Wolfgang Wiechert, Bernd Freisleben, and Katharina Noh. An Online Provenance Service for Distributed Metabolic Flux Analysis Workflows. In *Proceedings of the 9th European Conference on Web Services (ECOWS)*, pages 91–98, Washington, DC, USA, 2011. IEEE Computer Society.

[39] Danica Damljanovic and Kalina Bontcheva. Named entity disambiguation using linked data. In *Proceedings of the 9th Extended Semantic Web Conference*, 2012.

[40] Souripriya Das, Seema Sundara, and Richard Cyganiak. R2RML: RDB to RDF Mapping Language. W3C Recommendation 27 September. `http://www.w3.org/TR/r2rml/`, 2012.

[41] Susan B. Davidson and Juliana Freire. Provenance and scientific workflows: challenges and opportunities. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1345–1350. ACM, 2008.

[42] Victor de Boer, Matthias van Rossum, Jurjen Leinenga, and Rik Hoekstra. Dutch ships and sailors linked data. In *The Semantic Web–ISWC 2014*, pages 229–244. Springer, 2014.

[43] Ben De Meester, Tom De Nies, Wesley De Neve, Erik Mannens, and Rik Van de Walle. Interlinking books with the world: Using the Semantic Web to create books as reliable, machine-understandable information service providers. *Journal of Electronic Publishing*, 18(1), 2015.

[44] Ben De Meester, Tom De Nies, Laurens De Vocht, Ruben Verborgh, Erik Mannens, and Rik Van de Walle. Storyblink: a Semantic Web approach for linking stories. In *ISWC 2015 Posters & Demonstrations Track*. CEUR-WS, 2015.

[45] Ben De Meester, Tom De Nies, Hajar Ghaem Sigarchian, Miel Vander Sande, Jelle Van Campen, Bram Van Impe, Wesley De Neve, Erik

Mannens, and Rik Van de Walle. A digital-first authoring environment for enriched e-books using EPUB 3. *Information services & use*, 34(3-4):259–268, 2014.

[46] Ben De Meester, Tom De Nies, Hajar Ghaem Sigarchian, Miel Vander Sande, Jelle Van Campen, Bram Van Impe, Wesley De Neve, Erik Mannens, and Rik Van de Walle. A digital-first authoring environment for enriched e-books using EPUB 3. In *Conference on Electronic Publishing (ELPUB)*, pages 68–77, 2014.

[47] Ben De Meester, Tom De Nies, Ruben Verborgh, Erik Mannens, and Rik Van de Walle. Reconnecting digital publications to the Web using their spatial information. In *Proceedings of the 24th International Conference on World Wide Web Companion – LocWeb 2015*, pages 749–754, 2015.

[48] Ben De Meester, Hajar Ghaem Sigarchian, Tom De Nies, Anastasia Dimou, Wesley De Neve, Erik Mannens, and Rik Van de Walle. Creating discoverable learning content using a user-friendly authoring environment. In *Proceedings of the Linked Learning meets LinkedUp Workshop: Learning and Education with the Web of Data*, 2014.

[49] Tom De Nies. Assessing content value for digital publishing through relevance and provenance-based trust. In *The Semantic Web – ISWC 2013 (Doctoral Consortium)*, pages 424–431. Springer, 2013.

[50] Tom De Nies, Christian Beecks, Wesley De Neve, Thomas Seidl, Erik Mannens, and Rik Van de Walle. Towards named-entity-based similarity measures: Challenges and opportunities. In *Proceedings of the 7th International Workshop on Exploiting Semantic Annotations in Information Retrieval*, pages 9–11. ACM, 2014.

[51] Tom De Nies, Christian Beecks, Fréderic Godin, Wesley De Neve, Grzegorz Stepien, Dörthe Arndt, Laurens De Vocht, Ruben Verborgh, Thomas Seidl, Erik Mannens, and Rik Van de Walle. A distance-based approach for semantic dissimilarity in knowledge graphs. In *Proceedings of the 10th International Conference on Semantic Computing (ICSC)*. IEEE, 2016.

[52] Tom De Nies, Christian Beecks, Fréderic Godin, Wesley De Neve, Grzegorz Stepien, Dörthe Arndt, Laurens De Vocht, Ruben Verborgh, Thomas Seidl, Erik Mannens, and Rik Van de Walle. Normalized Semantic Web Distance. In *The Semantic Web – Latest Advances and New Domains (ESWC 2016)*. Springer, 2016.

[53] Tom De Nies, Davide Ceolin, Paul Groth, Olaf Hartig, and Stephen Marsh. Overview of method 2014: The 3rd international workshop on methods for establishing trust of (open) data. *10th International Workshop on Uncertainty Reasoning for the Semantic Web (URSW 2014)*, page 53, 2014.

[54] Tom De Nies, Sam Coppens, Erik Mannens, and Rik Van de Walle. Modeling uncertain provenance and provenance of uncertainty in w3c prov. In *Proceedings of the 22nd international conference on World Wide Web Companion – Posters*, pages 167–168, 2013.

[55] Tom De Nies, Sam Coppens, Davy Van Deursen, Erik Mannens, and Rik Van de Walle. Automatic discovery of high-level provenance using semantic similarity. In *Provenance and Annotation of Data and Processes – IPAW 2012*, pages 97–110. Springer, 2012.

[56] Tom De Nies, Sam Coppens, Ruben Verborgh, Miel Vander Sande, Erik Mannens, Rik Van de Walle, Danius Michaelides, and Luc Moreau. Easy access to provenance: an essential step towards trust on the Web. In *IEEE 37th Annual Computer Software and Applications Conference Workshops (COMPSACW) – METHOD 2013*, pages 218–223, 2013.

[57] Tom De Nies, Sam Coppens (Eds.), and W3C Provenance Working Group. PROV-Dictionary: Modeling Provenance for Dictionary Data Structures. W3C Working Group Note 30 April. `http://www.w3.org/TR/prov-dictionary/`, 2013.

[58] Tom De Nies, Wesley De Neve, Erik Mannens, and Rik Van de Walle. Ghent university-iminds at mediaeval 2013: An unsupervised named entity-based similarity measure for search and hyperlinking. In *MediaEval 2013 Multimedia Benchmark Workshop*. CEUR-WS, 2013.

[59] Tom De Nies, Pedro Debevere, Davy Van Deursen, Wesley De Neve, Erik Mannens, and Rik Van de Walle. Ghent university-ibbt at mediaeval 2012 search and hyperlinking: Semantic similarity using named entities. In *MediaEval 2012 Multimedia Benchmark Workshop*. CEUR-WS, 2012.

[60] Tom De Nies, Evelien D'heer, Sam Coppens, Davy Van Deursen, Erik Mannens, Steve Paulussen, and Rik Van de Walle. Bringing newsworthiness into the 21st century. In *Web of Linked Entities Workshop in conjuction with the 11th International Semantic Web Conference (ISWC 2012)*, pages 106–117. CEUR-WS, 2012.

[61] Tom De Nies, Anastasia Dimou, Ruben Verborgh, Erik Mannens, and Rik Van de Walle. Enabling dataset trustworthiness by exposing the provenance of mapping quality assessment and refinement. In *4th International Workshop on Methods for Establishing Trust of (Open) Data (METHOD 2015)*, 2015.

[62] Tom De Nies, Gerald Haesendonck, Fréderic Godin, Wesley De Neve, Erik Mannens, and Rik Van de Walle. Towards automatic assessment of the social media impact of news content. In *Proceedings of the 22nd international conference on World Wide Web Companion – SNOW 2013*, pages 871–874, 2013.

[63] Tom De Nies, Sara Magliacane, Ruben Verborgh, Sam Coppens, Paul T. Groth, Erik Mannens, and Rik Van de Walle. Git2PROV: Exposing version control system content as W3C PROV. In *International Semantic Web Conference (Posters & Demos)*, pages 125–128, 2013.

[64] Tom De Nies, Robert Meusel, Dominique Ritze, Kai Eckert, Anastasia Dimou, Laurens De Vocht, Ruben Verborgh, Erik Mannens, and Rik Van de Walle. A lightweight provenance pingback and query service for Web Publications. In *Provenance and Annotation of Data and Processes – IPAW 2014*, pages 203–208. Springer, 2014.

[65] Tom De Nies, Frank Salliau, Ruben Verborgh, Erik Mannens, and Rik Van de Walle. TinCan2PROV: Exposing interoperable provenance of learning processes through Experience API logs. In *Proceedings of the 24th international conference on World Wide Web Companion – LILE 2015*, pages 689–694, 2015.

[66] Tom De Nies, Io Taxidou, Anastasia Dimou, Ruben Verborgh, Peter M Fischer, Erik Mannens, and Rik Van de Walle. Towards multi-level provenance reconstruction of information diffusion on social media. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management*. ACM, 2015.

[67] Tom De Nies, Ruben Verborgh, Miel Vander Sande, Ben De Meester, Hajar Ghaem Sigarchian, Anastasia Dimou, Wesley De Neve, Erik Mannens, and Rik Van de Walle. Towards decentralized annotations in digital books and on the Web. *Proceedings of the W3C Workshop on Annotations*, 2014.

[68] Tom De Nies, Jasper Verplanken, Ruben Verborgh, Wesley De Neve, Erik Mannens, and Rik Van de Walle. Named-entity-based linking and exploration of news using an adapted jaccard metric. In *Proceedings of the 1st Workshop on Negative or Inconclusive Results in Semantic Web (NoISE2015)*, 2015.

[69] Tom De Nies, Thomas Vervust, Michiel Demey, Marc Leman, Jan Vanfleteren, and Rik Van de Walle. Synchronizing music and movement with BeatLED: An interactive musical social game. *Journal of New Music Research*, 41(4):351–363, 2012.

[70] Tom De Nies, Thomas Vervust, Michiel Demey, Rik Van de Walle, Jan Vanfleteren, and Marc Leman. BeatLED: the social gaming partyshirt. In *8th Sound and Music Computing Conference (SMC)*, pages 526–532. Ghent University, Department of Electronics and information systems, 2011.

[71] Anastasia Dimou, Dimitris Kontokostas, Markus Freudenberg, Ruben Verborgh, Jens Lehmann, Erik Mannens, Sebastian Hellmann, and Rik Van de

Walle. Assessing and Refining Mappings to RDF to Improve Dataset Quality. In *Proceedings of the 14th International Semantic Web Conference*, 2015.

[72] Anastasia Dimou, Miel Vander Sande, Pieter Colpaert, Ruben Verborgh, Erik Mannens, and Rik Van de Walle. RML: A Generic Language for Integrated RDF Mappings of Heterogeneous Data. In *Workshop on Linked Data on the Web*, 2014.

[73] Anastasia Dimou, Miel Vander Sande, Tom De Nies, Erik Mannens, and Rik Van de Walle. RDF Mapping Rules Refinements according to Data Consumers' Feedback. In *Proceedings of the 23rd international conference on World Wide Web Companion*, pages 249–250, 2014.

[74] Li Ding, James Michaelis, Jim McCusker, and Deborah L McGuinness. Linked provenance data: A Semantic Web-based approach to interoperable workflow traces. *Future Generation Computer Systems*, 27(6):797–805, 2011.

[75] Leigh Dodds and Ian Davis. Linked Data Patterns. Qualified Relation. `http://patterns.dataincubator.org/book/qualified-relation.html`, 2012.

[76] Maria Eskevich, Robin Aly, Roeland Ordelman, Shu Chen, and Gareth J.F. Jones. Search and hyperlinking task at mediaeval 2013. In *MediaEval 2013 Multimedia Benchmark Workshop*. CEUR-WS, 2013.

[77] Maria Eskevich and Gareth J.F. Jones. Time-based segmentation and use of jump-in points in dcu search runs at the search and hyperlinking task at mediaeval 2013. In *MediaEval 2013 Multimedia Benchmark Workshop*. CEUR-WS, 2013.

[78] Maria Eskevich, Gareth J.F. Jones, Robin Aly, Roeland Ordelman, Shu Chen, Danish Nadeem, Camille Guinaudeau, Guillaume Gravier, Pascale Sébillot, Tom De Nies, et al. Multimedia information seeking through search and hyperlinking. In *Proceedings of the 3rd ACM conference on International conference on multimedia retrieval*, pages 287–294. ACM, 2013.

[79] Maria Eskevich, Gareth J.F. Jones, Shu Chen, Robin Aly, Roeland Ordelman, and Martha A. Larson. Search and hyperlinking task at mediaeval 2012. In *MediaEval 2012 Multimedia Benchmark Workshop*. CEUR-WS, 2012.

[80] Zhuo Feng, Pritam Gundecha, and Huan Liu. Recovering information recipients in social media via provenance. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 706–711, 2013.

[81] Petra Galuščáková and Pavel Pecina. Cuni at mediaeval 2012 search and hyperlinking task. In *MediaEval 2012 Multimedia Benchmark Workshop*. CEUR-WS, 2012.

[82] Matthew Gamble and Carole Goble. Quality, trust, and utility of scientific data on the web: Towards a joint model. In *Proceedings of the 3rd international web science conference*, page 15. ACM, 2011.

[83] Matthew Gamble and Carole A Goble. Influence factor: Extending the PROV model with a quantitative measure of influence. In *6th USENIX Workshop on the Theory and Practice of Provenance (TaPP '14)*, 2014.

[84] Daniel Garijo and Kai Eckert. Dublin Core to PROV Mapping - W3C Working Group Note 30 April . `http://www.w3.org/TR/prov-dc/`, 2013.

[85] Hajar Ghaem Sigarchian, Ben De Meester, Tom De Nies, Ruben Verborgh, Wesley De Neve, Erik Mannens, and Rik Van de Walle. EPUB3 for integrated and customizable representation of a scientific publication and its associated resources. *Proceedings of the 4th Workshop on Linked Science*, 2014.

[86] Hajar Ghaem Sigarchian, Ben De Meester, Tom De Nies, Ruben Verborgh, Frank Salliau, Wesley De Neve, Erik Mannens, and Rik Van de Walle. Towards making EPUB 3-based e-textbooks a first-class mobile learning environment. In *Proceedings of the 10th International Conference on e-Learning (ICEL)*, 2015.

[87] Hajar Ghaem Sigarchian, Tom De Nies, Miel Vander Sande, Wesley De Neve, Erik Mannens, and Rik Van de Walle. Towards cost-effective enrichment of EPUB3-compliant ebooks. In *W3C Workshop on Publishing using the Open Web Platform*, 2013.

[88] Yolanda Gil and Donovan Artz. Towards content trust of web resources. *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(4):227–239, 2007.

[89] Yolanda Gil, James Cheney, Paul Groth, Olaf Hartig, Simon Miles, Luc Moreau, P. Pinheiro Da Silva, Sam Coppens, Daniel Garijo, Jose Manuel Gomez, et al. Provenance XG final report. *Final Incubator Group Report*, 2010.

[90] Yolanda Gil, Simon Miles, et al. PROV Model Primer. W3C Working Group Note 30 April. `http://www.w3.org/TR/prov-primer/`, 2013.

[91] Boris Glavic, Gustavo Alonso, Renée J Miller, and Laura M. Haas. Tramp: Understanding the behavior of schema mappings through provenance. *Proceedings of the VLDB Endowment*, 3(1-2):1314–1325, 2010.

[92] Boris Glavic, Kyumars Sheykh Esmaili, Peter M. Fischer, and Nesime Tatbul. Ariadne: Managing fine-grained provenance on data streams. In *Distributed event-based systems*, pages 39–50, 2013.

[93] Fréderic Godin, Tom De Nies, Christian Beecks, Laurens De Vocht, Wesley De Neve, Erik Mannens, Thomas Seidl, and Rik Van de Walle. The normalized freebase distance. In *The Semantic Web: ESWC 2014 Satellite Events*, pages 218–221. Springer, 2014.

[94] Jennifer Golbeck and Aaron Mannes. Using trust and provenance for content filtering on the Semantic Web. In *Proceedings of the Models of Trust for the Web Workshop*, 2006.

[95] Jose Manuel Gómez-Pérez and Oscar Corcho. Problem-solving methods for understanding process executions. *Computing in Science & Engineering*, 10(3):47–52, 2008.

[96] Frank Goossen, Wouter IJntema, Flavius Frasincar, Frederik Hogenboom, and Uzay Kaymak. News personalization using the CF-IDF semantic recommender. In *Proceedings of the International Conference on Web Intelligence, Mining and Semantics*, page 10. ACM, 2011.

[97] Todd J Green, Grigoris Karvounarakis, Zachary G Ives, and Val Tannen. Update exchange with mappings and provenance. In *Proceedings of the 33rd international conference on Very large data bases*, pages 675–686. VLDB Endowment, 2007.

[98] Paul Groth, Sheng Jiang, Simon Miles, Steve Munroe, Victor Tan, Sofia Tsasakou, and Luc Moreau. An Architecture for Provenance Systems. Technical report, University of Southampton, February 2006.

[99] Paul Groth and Luc Moreau. PROV-Overview: An Overview of the PROV Family of Documents. W3C Working Group Note 30 April. `http://www.w3.org/TR/prov-overview/`, 2013.

[100] Thomas R Gruber. A translation approach to portable ontology specifications. *Knowledge acquisition*, 5(2):199–220, 1993.

[101] Adrien Guille, Hakim Hacid, Cecile Favre, and Djamel A. Zighed. Information diffusion in online social networks: A survey. *ACM SIGMOD Record*, 42(2), 2013.

[102] Adrien Guille, Hakim Hacid, Cécile Favre, and Djamel A. Zighed. Information diffusion in online social networks: A survey. *ACM SIGMOD Record*, 42(2):17–28, 2013.

[103] Pritam Gundecha, Zhuo Feng, and Huan Liu. Seeking provenance of information using social media. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, pages 1691–1696, 2013.

[104] Pritam Gundecha, Suhas Ranganath, Zhuo Feng, and Huan Liu. A tool for collecting provenance data in social media. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1462–1465, 2013.

[105] Laurel L Haak, Martin Fenner, Laura Paglione, Ed Pentz, and Howard Ratner. Orcid: a system to uniquely identify researchers. *Learned Publishing*, 25(4):259–264, 2012.

[106] Bahareh Rahmanzadeh Heravi and Jarred McGinnis. A framework for social semantic journalism. In *WebSci*, 2013.

[107] Pieter Heyvaert, Tom De Nies, Joachim Van Herwegen, Miel Vander Sande, Ruben Verborgh, Wesley De Neve, Erik Mannens, and Rik Van de Walle. Using EPUB 3 and the Open Web platform for enhanced presentation and machine-understandable metadata for digital comics. *New Avenues for Electronic Publishing in the Age of Infinite Collections and Citizen Science*, page 37, 2015.

[108] Pieter Heyvaert, Anastasia Dimou, Ruben Verborgh, Erik Mannens, and Rik Van de Walle. Towards a uniform user interface for editing mapping definitions. In *Proceedings of the 4th International Workshop on Intelligent Exploration of Semantic Data (IESD 2015)*, 2015.

[109] Raquel Hijón-Neira and Ángel Velázquez-Iturbide. From the discovery of students access patterns in e-learning including Web 2.0 resources to the prediction and enhacements of students outcome. *E-learning, experiences and future*, pages 275–294, 2010.

[110] Angelos Hliaoutakis, Giannis Varelas, Epimenidis Voutsakis, Euripides GM Petrakis, and Evangelos Milios. Information retrieval by semantic similarity. *International Journal on Semantic Web and Information Systems*, 2(3):55–73, 2006.

[111] Jingwei Huang and Mark S. Fox. Uncertainty in knowledge provenance. *The Semantic Web: Research and Applications*, pages 372–387, 2004.

[112] Francisco Iacobelli, Nathan D. Nichols, Larry Birnbaum, and Kristian J. Hammond. Finding new information via robust entity detection. In *AAAI Fall Symposium: Proactive Assistant Agents*, 2010.

[113] IEEE. Data model for content to learning management system communication, IEEE Std 1484.11.1-2004, 2005.

[114] IMS Global Learning Consortium et al. Learning measurement for analytics whitepaper, 2013.

[115] Pascal Kelm, Sebastian Schmiedeke, and Thomas Sikora. Feature-based video key frame extraction for low quality video sequences. In *Proceedings of the WIAMIS '09 Workshop*, pages 25–28, May 6-8 2009.

[116] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146, 2003.

[117] Graham Klyne, Paul Groth, Luc Moreau, Olaf Hartig, Yogesh Simmhan, James Myers, Timothy Lebo, Khalid Belhajjame, and Simon Miles. PROV-AQ: Provenance Access and Query. W3C Working Group Note 30 April. `http://www.w3.org/TR/prov-aq/`, 2013.

[118] Dimitris Kontokostas, Patrick Westphal, Sören Auer, Sebastian Hellmann, Jens Lehmann, Roland Cornelissen, and Amrapali Zaveri. Test-driven Evaluation of Linked Data Quality. In *Proceedings of the 23rd International Conference on World Wide Web*, 2014.

[119] Vlad Korolev and Anupam Joshi. Prob: A tool for tracking provenance and reproducibility of big data experiments. *Reproduce'14. HPCA 2014*, 11:264–286, 2014.

[120] Swarnim Kulkarni and Doina Caragea. Computation of the semantic relatedness between words using concept clouds. In *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval*, pages 183–188, 2009.

[121] Sharma Kumar, Marjit Ujjal, and Biswas Utpal. Exposing marc 21 format for bibliographic data as linked data with provenance. *Journal of Library Metadata*, 13(2-3):212–229, 2013.

[122] Sejeong Kwon, Meeyoung Cha, Kyomin Jung, Wei Chen, and Yajun Wang. Prominent features of rumor propagation in online social media. In *IEEE 13th International Conference on Data Mining (ICDM)*, pages 1103–1108, 2013.

[123] Sejeong Kwon, Meeyoung Cha, Kyomin Jung, Wei Chen, and Yajun Wang. Prominent features of rumor propagation in online social media. In *IEEE 13th International Conference on Data Mining (ICDM)*, pages 1103–1108. IEEE, 2013.

[124] Carl Lagoze, Jeremy Willliams, and Lars Vilhuber. Encoding provenance metadata for social science datasets. In *Metadata and Semantics Research*, pages 123–134. Springer, 2013.

[125] Lori Lamel and Jean-Luc Gauvain. Speech Processing for Audio Indexing. *Advances in Natural Language Processing. (LNCS 5221)*, pages 4–15, 2008.

[126] Timothy Lebo, Satya Sahoo, Deborah L. McGuinness (Eds.), and W3C Provenance Working Group. PROV-O: The PROV Ontology. W3C Recommendation 30 April. http://www.w3.org/TR/2013/REC-prov-o-20130430/, 2013.

[127] Jure Leskovec, Lada A. Adamic, and Bernardo A. Huberman. The dynamics of viral marketing. *ACM Transactions on the Web (TWEB)*, 1(1):5, 2007.

[128] Jure Leskovec, Lars Backstrom, and Jon Kleinberg. Meme-tracking and the dynamics of the news cycle. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 497–506. ACM, 2009.

[129] Xian Li, Timothy Lebo, and Deborah L. McGuinness. Provenance-based strategies to develop trust in Semantic Web applications. In *Provenance and Annotation of Data and Processes*, pages 182–197. Springer, 2010.

[130] Yang Li, Chi Wang, Fangqiu Han, Jiawei Han, Dan Roth, and Xifeng Yan. Mining evidences for named entity disambiguation. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1070–1078. ACM, 2013.

[131] Hyo-Sang Lim, Yang-Sae Moon, and Elisa Bertino. Provenance-based trustworthiness assessment in sensor networks. In *Proceedings of the Seventh International Workshop on Data Management for Sensor Networks*, pages 2–7. ACM, 2010.

[132] Jon Loeliger. *Version Control with Git: Powerful Tools and Techniques for Collaborative Software Development*. O'Reilly Media, Inc., 2009.

[133] Diego López-de Ipiña, Sacha Vanhecke, Oscar Peña, Tom De Nies, and Erik Mannens. Citizen-centric linked data apps for smart cities. In *Ubiquitous Computing and Ambient Intelligence. Context-Awareness and Context-Driven Interaction*, pages 70–77. Springer, 2013.

[134] Clifford A Lynch. When documents deceive: Trust and provenance as new factors for information retrieval in a tangled web. *Journal of the Association for Information Science and Technology*, 52(1):12, 2001.

[135] Sara Magliacane. Reconstructing provenance. In *11th International Semantic Web Conference (ISWC)*, pages 399–406. Springer, 2012.

[136] Sara Magliacane and Paul T. Groth. Repurposing benchmark corpora for reconstructing provenance. In *SePublica*, pages 39–50, 2013.

[137] Riccardo Mazza, Marco Bettoni, Marco Faré, and Luca Mazzola. Moclog–monitoring online courses with log data. In *Proceedings of the 1st Moodle Research Conference*, pages 14–15, 2012.

[138] Deborah L. McGuinness and Frank van Harmelen (Eds.). OWL Web Ontology Language Overview. W3C Recommendation 10 February. http://www.w3.org/TR/owl-features/, 2004.

[139] Massimo Melucci. Contextual Search: A Computational Framework. *Foundations and Trends® in Information Retrieval*, 6(4-5):257–405, 2012.

[140] Microsoft Research. Entity Recognition and Disambiguation Challenge (ERD 2014). http://web-ngram.research.microsoft.com/ERD2014/.

[141] George A Miller and Walter G Charles. Contextual correlates of semantic similarity. *Language and cognitive processes*, 6(1):1–28, 1991.

[142] David Milne and I Witten. An effective, low-cost measure of semantic relatedness obtained from wikipedia links. In *Proceedings of the AAAI Workshop on Wikipedia and Artificial Intelligence: an Evolving Synergy, Chicago, USA*, pages 25–30, 2008.

[143] Paolo Missier and Khalid Belhajjame. *A PROV encoding for provenance analysis using deductive rules*. Springer, 2012.

[144] Soto Montalvo, Víctor Fresno, and Raquel Martínez. NESM: A named entity based proximity measure for multilingual news clustering. *Procesamiento del lenguaje natural*, 48:81–88, 2012.

[145] Luc Moreau. The foundations for provenance on the Web. *Foundations and Trends in Web Science*, 2(2–3):99–241, 2010.

[146] Luc Moreau, Ben Clifford, Juliana Freire, Joe Futrelle, Yolanda Gil, Paul Groth, Natalia Kwasnikowska, Simon Miles, Paolo Missier, Jim Myers, et al. The open provenance model core specification (v1. 1). *Future Generation Computer Systems*, 27(6):743–756, 2011.

[147] Luc Moreau, Paul Groth, James Cheney, Timothy Lebo, and Simon Miles. The rationale of PROV. *Web Semantics: Science, Services and Agents on the World Wide Web*, 35:235–257, 2015.

[148] Luc Moreau, Tim Lebo (Eds.), and W3C Provenance Working Group. Linking Across Provenance Bundles. W3C Working Group Note 30 April. http://www.w3.org/TR/prov-links/, 2013.

[149] Luc Moreau, Paolo Missier, (Eds.), and W3C Provenance Working Group. PROV-DM: The PROV Data Model. W3C Recommendation 30 April. http://www.w3.org/TR/prov-dm/, 2013.

[150] Seth A. Myers, Chenguang Zhu, and Jure Leskovec. Information diffusion and external influence in networks. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 33–41, 2012.

[151] Danish Nadeem, Robin Aly, and Roeland J.F. Ordelman. Utwente does brave new tasks for mediaeval 2012: Searching and hyperlinking. In *MediaEval 2012 Multimedia Benchmark Workshop*. CEUR-WS, 2012.

[152] Markus Nentwig, Tommaso Soru, Axel-Cyrille Ngonga Ngomo, and Erhard Rahm. Linklion: A link repository for the web of data. In *The Semantic Web: ESWC 2014 Satellite Events*, pages 439–443. Springer, 2014.

[153] Mark Nottingham. Web linking (RFC 5988). `http://tools.ietf.org/html/rfc5988`, 2010.

[154] Fabrizio Orlandi and Alexandre Passant. Modelling provenance of DBpedia resources using wikipedia contributions. *Web Semantics*, pages 149 – 164, 2011.

[155] Alexandre Passant. Measuring Semantic Distance on Linking Data and Using it for Resources Recommendations. In *AAAI Spring Symposium: Linked Data Meets Artificial Intelligence*, 2010.

[156] Addison Phillips and Mark Davis. Tags for identifying languages. Technical report, BCP 47, RFC 4646, September, 2006.

[157] Addison Phillips and Mark Davis. Tags for identifying languages. Technical report, BCP 47, RFC 5646, September, 2009.

[158] Eric Prud'hommeaux and Andy Seaborne. SPARQL Query Language for RDF. W3C Recommendation 15 January. `http://www.w3.org/TR/rdf-sparql-query/`, 2008.

[159] Roy Rada, Hafedh Mili, Ellen Bicknell, and Maria Blettner. Development and application of a metric on semantic nets. *IEEE Transactions on Systems, Man and Cybernetics*, 19(1):17–30, 1989.

[160] Giuseppe Rizzo and Raphaël Troncy. NERD: a framework for unifying named entity recognition and disambiguation extraction tools. In *Proceedings of the Demonstrations at the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 73–76. Association for Computational Linguistics, 2012.

[161] Marc J. Rochkind. The source code control system. *IEEE Transactions on Software Engineering*, 1(4):364–370, 1975.

[162] A. Rousseau, F. Bougares, P. Deléglise, H. Schwenk, and Y. Estève. LIUM's systems for the IWSLT 2011 Speech Translation Tasks. In *Proceedings of the IWSLT Workshop, San Francisco, CA*, 2011.

[163] Yossi Rubner, Carlo Tomasi, and Leonidas J. Guibas. The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision*, 40(2):99–121, 2000.

[164] Satya S. Sahoo and Amit P. Sheth. Provenir ontology: Towards a framework for escience provenance management. *Kno.e.sis Publications*, 2009.

[165] Mathilde Sahuguet, Benoit Huet, Barbora Červenková, Evlampios Apostolidis, Vasileios Mezaris, Daniel Stein, Stefan Eickeler, J. L. Redondo Garcia, Raphael Troncy, and Lukas Pikora. Linkedtv at mediaeval 2013 search and hyperlinking task. In *MediaEval 2013 Multimedia Benchmark Workshop*. CEUR-WS, 2013.

[166] Gerard Salton, Anita Wong, and Chung-Shu Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[167] Sebastian Schmiedeke, Peng Xu, Isabelle Ferrané, Maria Eskevich, Christoph Kofler, Martha A. Larson, Yannick Estève, Lori Lamel, Gareth J.F. Jones, and Thomas Sikora. Blip10000: a social video dataset containing spug content for tagging and retrieval. In *Proceedings of the 4th ACM Multimedia Systems Conference*, pages 96–101. ACM, 2013.

[168] Kim Schouten, Robin Aly, and Roeland Ordelman. Searching and hyperlinking using word importance segment boundaries in mediaeval 2013. In *MediaEval 2013 Multimedia Benchmark Workshop*. CEUR-WS, 2013.

[169] Nigel Shadbolt, Wendy Hall, and Tim Berners-Lee. The semantic web revisited. *Intelligent Systems, IEEE*, 21(3):96–101, 2006.

[170] Kumar Sharma, Ujjal Marjit, and Utpal Biswas. Ptslga: A provenance tracking system for linked data generating application. *I.J. Information Technology and Computer Science*, 2015.

[171] Yogesh L Simmhan, Beth Plale, and Dennis Gannon. A survey of data provenance in e-science. *ACM Sigmod Record*, 34(3):31–36, 2005.

[172] Matthew P Simmons, Lada A Adamic, and Eytan Adar. Memes Online: Extracted, subtracted, injected, and recollected. *ICWSM*, 11:17–21, 2011.

[173] Manu Sporny, Gregg Kellogg, Markus Lanthaler (Eds.), and W3C RDF Working Group. JSON-LD 1.0: A JSON-based Serialization for Linked Data. W3C Recommendation 16 January. `http://www.w3.org/TR/2014/REC-json-ld-20140116/`, 2014.

[174] Shelley Sweeney. The ambiguous origins of the archival principle of "provenance". *Libraries & the Cultural Record*, 43(2):193–213, 2008.

[175] Io Taxidou, Tom De Nies, Ruben Verborgh, Peter M. Fischer, Erik Mannens, and Rik Van de Walle. Modeling information diffusion in social media as provenance with W3C PROV. In *Proceedings of the 24th international conference on World Wide Web Companion – MSM 2015*, pages 819–824, 2015.

[176] Io Taxidou and Peter M. Fischer. Online analysis of information diffusion in Twitter. In *Proceedings of the 23rd International Conference on WWW Companion*, pages 1313–1318, 2014.

[177] Io Taxidou and Peter M. Fischer. Rapid: A system for real-time analysis of information diffusion in Twitter. In *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*, pages 2060–2062, 2014.

[178] Io Taxidou, Peter M. Fischer, Tom De Nies, Erik Mannens, and Rik Van de Walle. Information diffusion and provenance of interactions in twitter: Is it only about retweets? In *Proceedings of the 25th international conference on World Wide Web Companion – Posters*, pages 113–114, 2016.

[179] The Advanced Distributed Learning (ADL) Initiative. Experience API, Version 1.0.1. `http://www.adlnet.gov/wp-content/uploads/2013/10/xAPI_v1.0.1-2013-10-01.pdf`, October 2013.

[180] The W3C SPARQL Working Group. SPARQL 1.1 Overview. W3C Recommendation 21 March. `http://www.w3.org/TR/sparql11-overview/`, 2013.

[181] Ricardo Usbeck, Michael Röder, Axel-Cyrille Ngonga Ngomo, Ciro Baron, Andreas Both, Martin Brümmer, Diego Ceccarelli, Marco Cornolti, Didier Cherix, Bernd Eickmann, et al. Gerbil: General entity annotator benchmarking framework. In *Proceedings of the 24th international conference on World Wide Web Companion*, pages 1133–1143. International World Wide Web Conferences Steering Committee, 2015.

[182] Herbert Van de Sompel, Michael L Nelson, Robert Sanderson, Lyudmila L Balakireva, Scott Ainsworth, and Harihar Shankar. Memento: Time travel for the Web. *arXiv preprint arXiv:0911.1112*, 2009.

[183] Seth van Hooland, Max De Wilde, Ruben Verborgh, Thomas Steiner, and Rik Van de Walle. Exploring entity recognition and disambiguation for cultural heritage collections. *Literary and Linguistic Computing*, 2014.

[184] Miel Vander Sande, Pieter Colpaert, Tom De Nies, Erik Mannens, and Rik Van de Walle. Publish data as time consistent Web API with provenance. In *Proceedings of the 23rd international conference on World Wide Web Companion*, pages 953–958, 2014.

[185] Miel Vander Sande, Tom De Nies, Wesley De Neve, Erik Mannens, and Rik Van de Walle. Improving reading experience by integrating the Semantic Web in ebooks. In *W3C Workshop on Electronic Books and the Open Web Platform*, 2013.

[186] Miel Vander Sande, Ruben Verborgh, Sam Coppens, Tom De Nies, Pedro Debevere, Laurens De Vocht, Pieterjan De Potter, Davy Van Deursen, Erik Mannens, and Rik Van de Walle. Everything is connected: Using linked data for multimedia narration of connections between concepts. In *11th International Semantic Web Conference (ISWC 2012)*, 2012.

[187] Ruben Verborgh, Olaf Hartig, Ben De Meester, Gerald Haesendonck, Laurens De Vocht, Miel Vander Sande, Richard Cyganiak, Pieter Colpaert, Erik Mannens, and Rik Van de Walle. Querying datasets on the Web with high availability. In *Proceedings of the 13th International Semantic Web Conference*, volume 8796 of *Lecture Notes in Computer Science*, pages 180–196. Springer, October 2014.

[188] Sven Vlaeminck. Data management in scholarly journals and possible roles for libraries–some insights from edawax. *Liber Quarterly–The Journal of the Association of European Research Libraries*, 23(1):48–79, 2013.

[189] W3C OWL Working Group. OWL 2 Web Ontology Language Document Overview (Second Edition). W3C Recommendation 11 December. `http://www.w3.org/TR/owl2-overview/`, 2012.

[190] Christo Wilson, Alessandra Sala, Krishna P.N. Puttaswamy, and Ben Y. Zhao. Beyond social graphs: User interactions in online social networks and their implications. *ACM Transactions on the Web*, 6(4):17, 2012.

[191] Huanjia Yang, Danius T Michaelides, Chris Charlton, William J Browne, and Luc Moreau. Deep: a provenance-aware executable document system. In *Provenance and Annotation of Data and Processes*, pages 24–38. Springer, 2012.

[192] Dowming Yeh, Chun-Hsiung Lee, Pei-Chen Sun, et al. The analysis of learning records and learning effect in blended e-learning. *Journal of information science and engineering*, 21(5):973–984, 2005.

[193] Jie Yuan, Peng Yue, Jianya Gong, and Mingda Zhang. A linked data approach for geospatial data provenance. *IEEE Transactions on Geoscience and Remote Sensing*, 51(11):5105–5112, 2013.

[194] Jing Zhang and Hosagrahar V. Jagadish. Lost source provenance. In *Proceedings of the 13th International Conference on Extending Database Technology*, pages 311–322. ACM, 2010.

[195] Jing Zhao, Karthik Gomadam, and Viktor Prasanna. Predicting missing provenance using semantic associations in reservoir engineering. In *Fifth IEEE International Conference on Semantic Computing (ICSC)*, pages 141–148. IEEE, 2011.

[196] Jun Zhao, Carole Goble, Robert Stevens, and Sean Bechhofer. Semantically linking and browsing provenance logs for e-science. In *Semantics of a Networked World. Semantics for Grid Databases*, pages 158–176. Springer, 2004.

[197] Jun Zhao, Satya S. Sahoo, Paolo Missier, Amit Sheth, and Carole Goble. Extending semantic provenance into the Web of data. *IEEE Internet Computing*, 15(1):40–48, 2011.