FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Context Enrichment of Crowdsourcing Tasks for Ontology Validation

DIPLOMARBEIT

zur Erlangung des akademischen Grades

Diplom-Ingenieur

im Rahmen des Studiums

Software Engineering / Internet Computing

eingereicht von

Stefan Gamerith, BSc.
Matrikelnummer 0925081

an der

Fakultät für Informatik der Technischen Universität Wien

Betreuung: Ao.Univ.Prof. Dipl.-Ing. Mag.rer.soc.oec. Dr.techn. Stefan Biffl
Mitwirkung: Reka Marta Sabou, MSc., PhD

Wien, 8. April 2019

Stefan Gamerith	Stefan Biffl

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.ac.at

FAKULTÄT FÜR !NFORMATIK
Faculty of Informatics

Context Enrichment of Crowdsourcing Tasks for Ontology Validation

DIPLOMA THESIS

submitted in partial fulfillment of the requirements for the degree of

Diplom-Ingenieur

in

Software Engineering / Internet Computing

by

Stefan Gamerith, BSc.
Registration Number 0925081

to the Faculty of Informatics

at the Vienna University of Technology

Advisor: Ao.Univ.Prof. Dipl.-Ing. Mag.rer.soc.oec. Dr.techn. Stefan Biffl
Assistance: Reka Marta Sabou, MSc., PhD

Vienna, 8th April, 2019

Stefan Gamerith | Stefan Biffl

Technische Universität Wien
A-1040 Wien ▪ Karlsplatz 13 ▪ Tel. +43-1-58801-0 ▪ www.tuwien.ac.at

Erklärung zur Verfassung der Arbeit

Stefan Gamerith, BSc.
Linzerstrasse 429/4215, 1140 Wien

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit – einschließlich Tabellen, Karten und Abbildungen –, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

Wien, 8. April 2019

Stefan Gamerith

Acknowledgements

First and foremost I am grateful for the support from my family, both financially and mentally. Special thanks goes to my mother Christa and father Willibald for always being there for me and giving me the strength and stability during the writing period, especially in those situations where I would otherwise give up.

Next, I want to thank my advisor Stefan Biffl and his assistance Marta Sabou for the opportunity to write this diploma thesis. As a side note, the inspiration for this thesis was actually from a seminar course I took some time ago with some of my colleagues. That was the time where I met Stefan Biffl. He showed me all the details of writing scientific texts. Most importantly, I really liked his structured and organised way of thinking and working.

Last, I want to thank all contributors from Australia, the United Kingdom and the United States of America who took part in our Crowdsourcing experiment.

Kurzfassung

Ein wichtiger Teil des Semantik Web Lebenszyklus ist die Ontologie Validierung, insbesondere bei erlernten Ontologien, die von Natur aus Fehler enthalten. Obwohl mittlerweile viele dieser Fehler von Algorithmen erkannt werden, ist dies mitunter bei komplexen Problemstellungen schwierig. Crowdsourcing stellt eine kosteneffiziente Alternative dar, die diese Aufgaben an eine Gruppe freiwilliger User (Crowd) auslagert. Dennoch gibt es bei der Ontologie Validierung mittels Crowdsourcing Verbesserungsbedarf.

Ein Lösungsansatz wäre die Zugabe kontextbezogener Informationen zu Crowdsourcing Aufgaben. Dies hätte mitunter einen positiven Einfluss auf das Validierungsergebnis.

Obwohl Fortschritte in diesem Bereich erzielt wurden, gibt es noch wenig Literatur zu diesem Thema. In dieser Diplomarbeit stellen wir 3 Methoden vor, die Kontext generieren um die Relevanz von Konzepten innerhalb einer Domäne zu überprüfen. Während der Ontology-based-Approach hierarchische Relationen verarbeitet, basiert der Metadata-based-Approach auf Annotationen. Als Basis für die letzte Methode (Dictionary-based-Approach) dienen Beispielsätze des Online Wörterbuchs WordNik.

Alle 3 Methoden wurden als Erweiterung des uComp Protege Plugin konzipiert, ein Plugin für den Ontologie Editor Protege, das die Validierung von Ontologien mittels Crowdsourcing ermöglicht. Im Rahmen von 3 Experimenten mit Datensätzen aus den Bereichen Klimawandel, Tennis und Finanzen wurden alle 3 Methoden getestet. Die Metriken Precision, Recall und F-Measure wurden für jeden Datensatz berechnet um Rückschlüsse über die Performance der getesteten Methoden ziehen zu können. Der Metadata-based-Approach lieferte die besten Validierungsergebnisse. Anhand der guten bis sehr guten Ergebnisse aller 3 Methoden (F-Measure größer 80%) wurde gezeigt, dass die Qualität der Validierung durch das Hinzufügen kontextbezogener Information gesteigert werden konnte.

Abstract

Validating the relevance of ontologies is considered an important task in the Semantic Web Lifecycle. This holds especially for learned ontologies which contain quite naturally errors. Although many errors can be tackled algorithmically, solving more complex problems by machines can be very tricky. Crowdsourcing offers a cost effective alternative in which tasks are solved by a large group of human workers. However, the performance of existing approaches that combine ontology validation with Crowdsourcing is still not satisfying.

A promising way of tackling this problem is to enrich Crowdsourcing tasks with additional contextual information to improve their understanding. This *Context* has not only a positive impact on the crowd's performance but also raises the results quality.

Even though recent research showed advances in this area, the use of Context was not explicitly targeted. In this thesis we present three novel methods that enrich Crowdsourcing tasks with contextual information to validate the relevance of concepts for a particular domain of interest. First, the Ontology based Approach processes hierarchical relations. Second, the Metadata based Approach generates descriptions based on annotations that are encoded within the ontology. Third, the idea of the Dictionary based Approach is to build up contextual information from example sentences by consulting the online dictionary WordNik.

For the analysis of all three approaches, we integrated these into the existing uComp Protege Plugin which facilitates the creation and execution of crowdsourcing tasks for ontology validation from within the Protege ontology editor. The evaluation was performed on three ontologies covering the domains of climate change, tennis and finance. For each dataset, the performance metrics Precision, Recall and F-Measure were calculated to compare the methods against the existing baseline approach that used no contextual information. The results showed that the Metadata based Approach outperformed all other methods. The other two approaches had some difficulties in certain situations, for example the Dictionary based Approach sometimes added inappropriate explanations, especially for concepts with multiple meanings associated. Likewise, the Ontology based Approach had problems with loosely connected ontologies containing just a few subsumption relations. However, all three approaches delivered results of high quality (F-Measure above 80%), indicating that adding Context to Crowdsourcing tasks is a cost-effective method of improving the crowd's performance.

Contents

CHAPTER 1

Introduction

1.1 Motivation

The advance of embedding Information Technology in all kinds of electronic devices and connecting them to collect and exchange data imposes new challenges of handling the increasing amount of data. Although many problems can be solved by machines only, there are certain tasks where humans perform better than computers. In *Crowdsourcing*, collective human intelligence is used to solve these complex tasks. [YKL11] grouped Crowdsourcing applications in 1) Voting Systems, 2) Information Sharing Systems, 3) Games with a purpose (GWAP) Systems and 4) Creative Systems. First, Voting Systems like Amazon Mechanical Turk (MTurk)[1] use majority voting to consider the answer with the highest number of votes as the correct one. Second, Information Sharing Systems enable users sharing and distributing knowledge among the crowd. Third, GWAP Systems facilitate playing small games in order to solve some meaningful tasks. Fourth, Creative Systems include tasks like labelling an image, writing algorithms or editing text.

An inherent factor of the Semantic Web is its large amount of Linked Data (e.g. DBpedia [LIJ+15]). Semantic technologies have emerged in various areas including domain modelling, data integration, enhanced search and content management [Con12]. Managing Semantic Web tasks is considered resource intensive and often requires human involvement due to its knowledge intensive and context specific nature. On the other side Crowdsourcing applications solve simple and small tasks in a cost-effective way. [SSN+15] summarises major research challenges and opportunities in combining Crowdsourcing and Semantic Web technologies. The most important challenges include 1) task and workflow design, 2) managing the quality of contributions, 3) handling multiple Crowdsourcing genres and 4) finding and managing the right crowd.

[1] https://www.mturk.com/

Whereas research shows that breaking tasks into smaller pieces and formulating the right questions has a huge impact on the outcome of Crowdsourcing tasks, it is equally important to establish a model which formally defines the required quality and skills to solve tasks. Also, there exist no general guidelines when and under which circumstances preferring small crowds with domain experts over large crowds with less qualified crowd workers is better. However, [Mor13] concluded that average crowds perform on par with domain experts in "common sense" application domains, if crowd workers are carefully selected by qualification tests and tasks are presented in the simplest possible form.

Clearly, in order to get the most qualitative responses from crowd workers, Crowdsourcing tasks need to provide enough Context that helps contributors to fully understand the task. This is especially important for ontology validation which requires expert knowledge. To support ontology engineers and domain experts, the uComp Protege Plugin [WSH16] was developed which facilitates the creation and execution of Crowdsourcing tasks for ontology validation from within the Protege ontology editor. It supports the following tasks: 1) Verification of Domain Relevance, 2) Verification of Relation Correctness, 3) Specification of Relation Type and 4) Verification of Domain and Range.

In this thesis we extend the uComp Protege Plugin and present three approaches that add contextual information to Crowdsourcing tasks.

1.2 Aim of the Work

When investigating the use of Context in Crowdsourcing tasks the first question that arise is, what is actually meant by the term »Context« . Besides referring to the need of Context to improve the understanding of Crowdsourcing tasks [SSN+15], not much literature exist yet that exclusively targets this topic. An overview of existing work as well as a conclusive definition of Context is given in Section 2.4.

Hence, the following research questions that target the need of Context in Crowdsourcing tasks for ontology validation are addressed in this work:

RQ-I *Does the crowd perform better on context enriched Crowdsourcing tasks?*

The basic question that motivates our research is whether the performance of crowd workers could be improved if Context was added to Crowdsourcing tasks in the context of ontology validation. Researchers have already stated this question [SSN+15] but not much research exists that relates to this topic.

In order to give a detailed answer to this research question, a conclusive definition of »Context« needs to be stated. While some work exists that uses Context, either implicitly or explicitly, no such definition exist yet. Whereas Section 2.4.1 gives a conclusive definition of »Context« , Section 2.4.2 examines existing approaches that use Context in Crowdsourcing tasks.

Answering this question directly leads to the next two research questions, our goal being to find generic methods applicable to similar datasets:

RQ-II *What methods can be applied that generate Context?*

In Chapter 3 we take a closer look into the approaches that generate Context suitable for extending Crowdsourcing tasks. Depending on how much manual intervention is required, Context was either generated automatically by an algorithm or manually by domain experts.

Additionally, the methods were tested against three datasets (e.g. ontologies) covering the domains of climate change, finance and tennis because an important goal was to measure the performance on a broader level. This leads to question which is stated next:

RQ-III *To what extent is it possible to transfer the investigated methods to different datasets?*

One important design goal of all approaches was to remove any bias to a particular dataset. Hence, all the approaches were evaluated on three different datasets covering diverse domains such as finance, tennis and climate change.

Furthermore, all datasets were combined to get a better picture on the performance of each method on average. To that end, we also analysed any characteristics of the used datasets to be able to make a statement on the generality of each approach. Based on this evaluation, we could answer the final research question:

RQ-IV *Which of the proposed methods works best? What are potential shortcomings and why?*

Finally, we answer the question, which of our proposed methods works best on average and under which conditions. There may be restrictions on the applicability because of the reduced size and budget of our experiment. Indeed, some issues were found that are worth mentioning and leave room for future improvements.

1.3 Contributions

By answering the research questions stated above, this work provides several contributions to advance research in the area of Crowdsourcing and the Semantic Web:

- Identification of methods that add Context to Crowdsourcing tasks
- An attempt to generalise the applicability of the proposed methods by performing an evaluation over multiple datasets
- A first definition of »Context« for Crowdsourcing tasks
- A statistical and qualitative analysis of the proposed methods evaluated on three datasets
- An extension of the uComp Protege Plugin with the proposed Context enrichment methods

1.4 Structure of the Work

The structure of this thesis is as follows:

Chapter 2 introduces the concepts that are used throughout this thesis. It includes i) a brief introduction into Crowdsourcing in Section 2.1, ii) a discussion on the interplay between Crowdsourcing and the Semantic Web in Section 2.2, iii) a detailed presentation of the uComp Protege Plugin which served as the baseline of this work (Section 2.3), and iv) a review of existing literature that addressed the use of Context in Crowdsourcing tasks which was taken together to formulate a conclusive definition of the term »Context« in Section 2.4.

Chapter 3 introduces the proposed methods that enrich Crowdsourcing tasks with Context. While in Section 3.2 the Ontology based Approach is discussed, Section 3.3 explains the Metadata based Approach and Section 3.4 presents the Dictionary based Approach. Then, Chapter 4 is dedicated to the evaluation settings. Concretely, it presents the metrics (Section 4.1) that served as performance measure and the used datasets (Section 4.2). Next, in Chapter 5 the obtained results are analysed.

In Chapter 6, the main topics of this thesis are summarised and the research questions are revisited. Finally, an outlook for future research topics is provided.

CHAPTER 2

State of the Art

In this chapter, we give a general introduction into Crowdsourcing and continue by describing its relevance in the Semantic Web area. Next, we discuss the previous approach of ontology validation using Crowdsourcing where this thesis is based on. This chapter ends with a definition of Context and how it is used in the literature.

2.1 Crowdsourcing

The term »Crowdsourcing« was initially mentioned by Jeff Howe in Wired Magazine [How06] where he described a novel model for efficiently solving problems by online workers. It was defined there as:

> *"[...] the act of taking a task traditionally performed by a designated agent (such as an employee or a contractor) and outsourcing it by making an open call to an undefined but large group of people. Crowdsourcing allows the power of the crowd to accomplish tasks that were once the province of just a specialised few."*[How08]

In other words, it means outsourcing the work to an undefined, outer workforce using an open call for participation. But in contrast to the traditional meaning of *Outsourcing*, work is distributed to a large, mostly anonymous crowd of human workers, often called the *Human Cloud*. Additionally, it sets no restrictions on the users being addressed, hence speaking of an *Open Call* to many people. Consequently, crowd workers are people with mixed skills, possibly coming from places across the globe. This does not necessarily mean that they are uneducated, instead, the crowd primarily consists of professional amateurs with valuable knowledge, education and commitment. Indeed, it needs extra motivation as the monetary reward is neglectable, being not more than a few Cents per

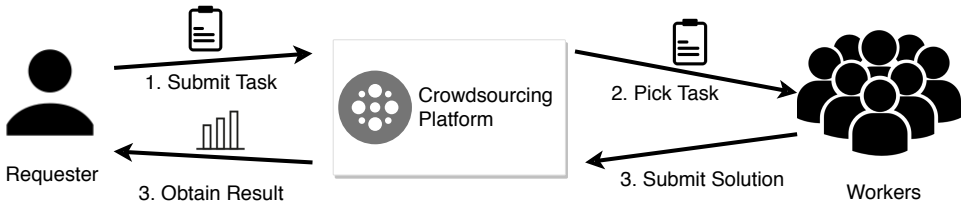


Figure 2.1: Main stakeholders of the Crowdsourcing process (adopted from [SR14, MCHJ17])

task. For many crowd workers intrinsic incentives such as gaining reputation or extending their skill set is more important though [KSV18].

A Crowdsourcing process generally involves 3 types of stakeholders (as illustrated in Figure 2.1):

1. *Requester:* A requester represents the initiator of every Crowdsourcing process. Depending on the complexity of work that needs to be done, requesters need to split the work into smaller *tasks*. Once the task is completed, requesters take the results, optionally combine them with results of completed tasks from previous runs and perform the worker payment.

2. *Workers:* The main workforce of every Crowdsourcing platform are crowd workers. They select a specific task of their interest and complete it. Once they are finished, they submit their solution to the Crowdsourcing platform.

3. *Crowdsourcing Platform:* Crowdsourcing Platforms are the central entity of the Crowdsourcing process. They enable requesters to have fast access to an on-demand, global and scalable workforce and provide workers the ability to choose from a variety of tasks.

2.1.1 Potentials and Opportunities

Crowdsourcing has been applied successfully by many companies. Harnessing human computation through Crowdsourcing and integrating it in machine computation opens up entirely new opportunities for them. Researchers have identified the following benefits in applying Crowdsourcing techniques [SG11]:

First, it can drastically **reduce costs** when work is not done by expensive in-house workers. As stated earlier, participants are mostly amateurs such as students or young graduates who want to spend their spare time doing something useful. In most cases Crowdsourcing is considered a source of additional income rather than their primary source of income.

Second, it opens **new perspectives for innovators**, especially when considering creative tasks, Crowdsourcing has positive impacts with respect to the originality of the solutions. One of the first major companies leveraging worldwide human resources was Procter & Gamble (P&G). They created a platform[1] which helps innovators in submitting new ideas to P&G's development program. Submission was open for everyone, they only required a clear and concise description of the unique features of one's solution and the status of the intellectual property.

Crowdsourcing can also have a **positive impact on network externalities** which describes the effect, when the value of a product depends on the number of users who interact with it [SCVP98]. A prime example here is OpenStreetMap[2] which dramatically profited from using Crowdsourcing, primarily because their value highly depends on the richness of the geographical content and up-to-dateness of the map data which is crowdsourced [Chi09].

Another positive aspect is that it **eliminates the risk of dependence** to a client company if work is outsourced. Companies are often lacking an overall strategy for defining contractual and transitional elements of an outsourcing initiative which can possibly ruin their business. These issues are not present because there is no strong connection between the company and the crowd workers. In many Crowdsourcing platforms, contributors are not even identified by their names, but by an artificial identifier.

Crowdsourcing enables **data collection on a large scale**. This is particularity important in academic and scientific contexts where experiments are performed with as many participants as possible to facilitate generalisation of the experiments [GMN+17].

Last, it **reduces coordination efforts** within a company. By definition, Crowdsourcing implies voluntary participation of individuals with no hierarchy or contract related constraints. Consequently, coordination by authorities as practiced in traditional working relationships is not needed anymore. Crowd workers are free to complete their tasks with a high degree of autonomy.

2.1.2 Challenges and Risks

Even though many benefits are brought through Crowdsourcing, being successful in the adoption of Crowdsourcing techniques requires awareness of its challenges and risks. In the next paragraphs major risks and challenges are presented [HTGV13]:

Probably one of the biggest challenge is related to **quality assurance**. There is plenty of literature investigating the challenges of *quality control* and *quality assessment* [ABI+13, DKC+18, HSC+13, HMS09].

Before implementing measures for improving the result quality, some metrics are needed which assign concrete values to quality attributes. For example, measuring the worker's

[1] https://www.pgconnectdevelop.com/ accessed 2018/07/26
[2] https://www.openstreetmap.org/ accessed 2018/07/26

required skills to complete certain tasks is done on a scale from 1, indicating little or now skills, to 10, requiring expert level skills. Unfortunately, this classification is often too generic and rather subjective. Substantial work was done to improve worker classification. A worker's profile includes professional experiences, the number of completed tasks, personal attributes, and other qualifications [DKC+18].

In terms of quality control, several methods have been proposed which positively impact on the output quality. For our experiments we used a *qualification test* to differentiate between useful inputs and spamming. For that, workers are required to correctly answer some questions. Another valid method is restricting access to a determined group of workers with a specific skill set. For example, Figure Eight[3] uses a 3-level *rating scheme*, ranging from level 1, setting no constraints, to level 3, selecting the most experienced contributors. A more costly method is *Reviewing*. It is either done by experts who are not members of the crowd (expert review) or by a group of workers who are part of the crowd (peer review). Expert reviews are rather expensive and time consuming but ensure high quality results. On the other hand, peer reviews are low-cost, require less time but achieve results of moderate quality. A good strategy is to use peer reviews in those situations where experts would not be able to review all outputs alone because of the sheer amount of data. The next method is primarily used for tasks with *voting* involved. A study [WC14] showed that this technique is particularly useful to elicit common knowledge, however, it fails in those situations with expert knowledge required.

An important challenge is **keeping workers motivated**. Techniques targeting the worker's motivation can be split into two groups, those trying to increase the *intrinsic motivation* and those trying to raise *extrinsic motivation*. While people motivated by intrinsic motivation are driven by personal reasons, extrinsic motivation occurs when people engage in an activity triggered by external factors.

One way of motivating crowd workers are *tailored rewards* and *payed bonuses*. Various rewarding schemes have been implemented such as volunteering, pay per time, pay per task, pay per each data unit in a task, paying tasks in bulks, to name just a few. Studies [HI11, HSSV15] showed that choosing the right amount and form of reward is essential for achieving good results. However, researchers have not yet agreed on a common strategy, guiding task designers in tweaking rewarding options to increase the worker's motivation. Paying bonuses adds extra motivation to incite contributors to deliver top results. The bonus is often added to the base reward of a task and is usually granted for reaching some defined goals or exceeding a predetermined threshold of some performance indicator.

A completely different approach in increasing the worker's motivation is to embed Crowdsourcing in a game. GWAP was first introduced by Luis von Ahn[vA06] where he had the vision of solving large-scale computational problems through online games. Participants perform tasks for joy and entertainment rather than monetary reward. Moreover, designing games that induce curiosity boosts motivation even more [LYG+16].

[3] https://www.figure-eight.com/ accessed 2018/07/27

As mentioned earlier, quantifying the worker's performance on a scale from poor to excellent helps requestors in better estimating the expected results. This comes into play when triggering the worker's motivation to reach a higher level. It is especially useful in those environments that strive for long-lasting worker engagement.

It is not enough to properly design a task and leave completion to the crowd. *Tasks with a purpose* go beyond that, they add context so that workers understand and get a clear picture of their contribution. These tasks are typically less attractive for spammers or adversarial workers because monetary reward is comparably low.

Not only assuring quality and keeping contributors motivated is challenging, evidence showed that a **proper task design** reduces the risk of incorrect or erroneous responses. One strategy is to narrow down the tasks dimensions in terms of *complexity* and *granularity*. Designing tasks in such a way that reduces cognitive complexity, that is the perceived complexity by humans, positively impacts the quality while it may lead to longer completion times. The other way to decrease task complexity is to organise work in a way that workers can concentrate on a single task rather than a sequence of related tasks.

Figure Eight[4] has a feature which controls the minimum time required to complete a page. It gives requestors the ability to control the *task duration*. It is recommended to carefully adjust this value because a contributor exceeding this limit will be stopped from completing the task. On the other hand, in certain scenarios faster completion is preferred over high quality judgements. Errors may be even desired to some extent because they will be analysed by some automated post-processing steps [KHC+16].

2.1.3 Types of Crowdsourcing Tasks

Despite the sheer amount of Crowdsourcing use cases, we focus on efforts that have been made on data processing. The common term describing that concept is *Data Mining*, defined as *"the extraction of implicit, previously unknown, potentially useful information from data."* [WFHP16] Indeed, that term primarily relates to tasks involving Artificial Intelligence (AI), which is also reflected by the fact that the book's [WF00] original title was changed from *Practical machine learning* to *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations* [BFH+10]. Even though quite some efforts were made in developing new and improving existing algorithms and approaches in AI, there are still situations in which Crowdsourcing performs better. Researchers have identified various types of data mining tasks that can be crowd-sourced which are discussed in the next paragraphs [XHSH14, BZG+12, SBS12]:

Classification Classification is the process of deciding to which class or category an item belongs to. In contrast to Clustering, all classes are known beforehand, the goal is to decide to which class an item belongs to. For example, researchers have investigated the problem of assigning images to a pre-defined set of classes [dHFRM+14]. In their

[4] https://www.figure-eight.com/ accessed 2018/07/28

study, the crowd was used to verify and possibly correct the classification done previously by an algorithm.

Clustering Clustering is a technique which groups data into classes or clusters with the goal of finding similar items. In contrast to Classification, the classes or clusters are not known beforehand. For example, in a study [KVH16] crowd workers had to compare images and judge whether they are similar or not. The goal was to group similar images by collecting labels of images, e.g, of birds or dogs of different kinds and breeds.

Semi-Supervised Learning Here, some labeled data together with some unlabeled data is used as input of a learning algorithm. The algorithm assigns labels using the information acquired from the labeled data. This data is called the training data. Semi-Supervised Learning is somewhere between Unsupervised Learning with no training data and Supervised Learning with only training data. For example, researchers [SF08] experimented with a set of images to annotate pictured humans. Their strategy for quality assurance was threefold: First, workers were required to rate annotations. Second, images included annotations from trusted users only and last, multiple annotations were collected for each image. This way, quality assessment was performed by the crowd itself at no additional costs.

Validation Likewise, humans can verify the correctness of an algorithm or predict the result on a large scale. As an example, [ALTY08] analysed 535 blog posts, finding the most active/inactive/influential/non-influential posts. The result was then compared against the top 100 voted posts on Digg[5]. Digg's content is created and maintained by the community which serves as reasonable alternative compared to other techniques.

2.2 Crowdsourcing in the Semantic Web

This section starts by briefly introducing the Semantic Web and the driving ideas in it's early stages. The central part of this section is dedicated to discussing the interplay between the Semantic Web and Crowdsourcing by the *Linked Data Life-Cycle*.

The World Wide Web (WWW) was probably one of the most influential and World changing innovation, allowing users to exchange documents without caring about the details of how they are processed or stored. The Semantic Web adds another layer on-top, enabling the use of references to real-world objects without concerning about the underlying documents in which these things are described. Therefore the Semantic Web can be seen as an extension of the WWW. It provides the means to process data in machine-readable formats, linking related properties to globally accessible schemas and offering a wide range of data interfaces [HBL10]. The adoption of Semantic Web technologies is still ongoing, many applications were developed that exploit these principles, but its full potential is just starting to be explored. This holds in particular because many

[5] http://digg.com/ accessed 2018/08/02

Paper	LD Stage	CS Contribution
[BDR17]	Stage 1 *(Data Extraction)*	Selecting named entities
[ADBB14]	Stage 2 *(Data Storage & Indexing)*	Creating listening experiences
[SWC+]	Stage 3 *(Data Revision & Authoring)*	Translating concept labels
[DDCM12]	Stage 4 *(Data Linking)*	Validating linked entities
[WBS13]	Stage 5 *(Data Classification & Enrichment)*	Validating learned ontologies
[MMJ+15]	Stage 6 *(Data Analysis & Quality)*	Validating subclass relations
[WSH16]	Stage 7 *(Data Cleansing & Evolution)*	Validating ontology parts
[Ver13]	Stage 8 *(Data Browsing & Querying)*	Creating semantic tags

Table 2.1: Overview of approaches in the Semantic Web area that showcase the application of Crowdsourcing techniques. {*LD Stage*=Stage of the Linked Data Life-Cycle (Section 2.2.1), *CS Contribution*=Contribution related to Crowdsourcing}

tasks can not be fully automated or it would be too costly. Crowdsourcing, on the other hand, facilitates distribution of tasks to a large number of contributors in a scalable and affordable way. In the remainder of this section we analyse, how Crowdsourcing can promote the adoption of Semantic Web technologies. Table 2.1 summarises the approaches in the Semantic Web area that showcase the application of Crowdsourcing techniques.

2.2.1 The Linked Data Life-Cycle

Over the years many tools and practices were developed that cover the full life cycle of weaving the Semantic Web. The stages of the Linked Data Life-Cycle are illustrated

Figure 2.2: The Linked Data Life-Cycle (consolidated from [ALNN11, ABD+12, SH08a])

in Figure 2.2. It shows the overall process of Linked Data management, starting from extracting Linked Data and ending in browsing Linked Data sources. Although the life cycle for semantic content starts with conceptual modelling (e.g. mapping unstructured data to structured or semi-structured formalisms), this is not always the case, especially if existing Linked Data should be managed as well. In that case, the first stage (Data Extraction) can be omitted. Likewise, the stages of the life cycle do not exist in isolation of each other or are passed in strict order, instead they are mutually complementary. Consequently, the examples that were given in each stage may also be relevant for other stages [SAF13].

Data Extraction When starting from scratch, data encoded in different formalisms need to be mapped to the semantic data model to facilitate semantic processing. There exist several approaches for the extraction process. When considering unstructured sources, text in particular, *Natural Language Processing (NLP)* as well as *Information Extraction (IE)* techniques have been successfully applied to gather relevant information. Three sub-disciples of NLP have emerged: *Named Entity Recognition* for discovering entity instances, *Keyword/Keyphrase Extraction* for identifying common topics and *Relationship Extraction* for linking entities to keywords. For structured data such as

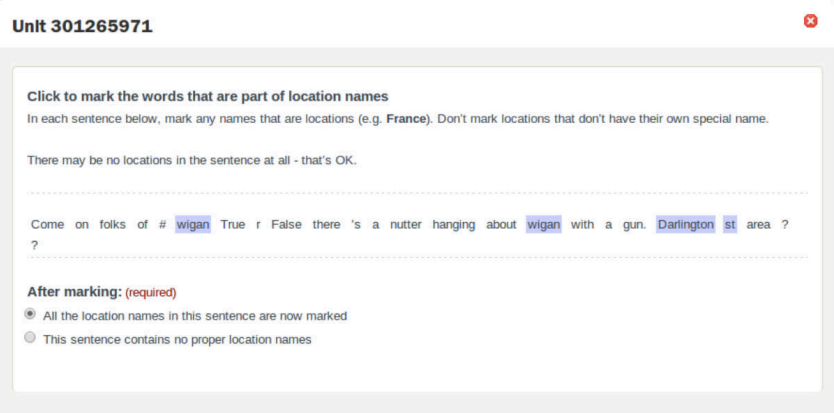


Figure 2.3: Crowdsourcing task interface for named entity recognition (adopted from [BDR17])

Extended Markup Language (XML) there exist a number of approaches. For example, the World Wide Web Consortium (W3C) published the recommendation RDB to RDF Mapping Language (R2RML)[6] which describes a common notation for mapping relational tables, views and queries to Resource Description Framework (RDF).

Even tough many problems can be solved efficiently by machines, there are still open tasks which are hard to solve algorithmically. [SH08b] summarised the major challenges for ontology construction, collecting named entities being one of them. As an example, in a study [BDR17] the authors crowdsourced a corpus containing around 10 000 tweets to extract named entities. They defined a methodical framework which combines an expert based and crowd based approach. Figure 2.3 shows the Crowdsourcing task interface for selecting named entities in tweets. To prevent spamming, an explicit confirmation step was added at the bottom of the submission form. After identifying the named entities, the next step is entity linking to find potentially ambiguous entities. DBpedia [ABK+07] was chosen as target entity linking database because of its good coverage of named entities, its frequent updates and available mappings to other Linked Data sources.

Data Storage and Indexing Up to this stage, the data was already mapped to the RDF data model but needs to be stored and indexed efficiently. Researchers have put a lot of effort into this field because efficient storage and indexing mechanisms are fundamental for the adoption of Linked Data. Due to efficient querying and storage capabilities of relational databases resulting from decades of research in this area, it

[6] http://www.w3.org/TR/r2rml/ accessed 2018/08/06

makes sense to adopt these approaches for Linked Data as well [AMMH07]. However, to support storing very large quantities of data there exist custom solutions too [BKvH02]. Targeting Linked Data indexing, various approaches and principles have been applied successfully. The common aspects of all approaches is their focus on *Data Compression* and *Data Pruning* [SM11]. Whereas the basic idea of Data Compression is to minimise the footprint of the index, Data Pruning is a technique for avoiding unnecessary data processing.

Whereas Data Indexing is traditionally done solely by machines, there are some approaches that combine Crowdsourcing and Semantic principles for Data Storage. One of them is the Listening Experience Database (LED)[7], a semantic knowledge base of accounts of listening to music in documented sources [ADBB14]. LED stores listening evidences of music across history and musical genres. Its dataset includes more than 10 000 entries collected from various sources, volunteers from the crowd being one of them.

Data Revision and Authoring In this stage users are given the opportunity to create new or modify existing semantic information. This is called *Semantic Content Authoring (SCA)* which is defined in the literature as *"a tool-supported manual composition process aiming at the creation of semantic documents."* [KA13] More generally speaking, SCA is actually embedded in a broader ecosystem for semantic content authoring as shown in Figure 2.4. The central entity of the semantic ecosystem is a semantic document which holds semantically enriched information. In information management, semantic documents serve a number of purposes such as information searching, information retrieval, information presentation, information integration, personalisation, reusability and interoperability [KA13]. For that reason there exists a research area dealing with the main fields of semantic content management. In particular, it covers the manipulation, creation and processing of semantic content. Users do not directly interact with semantic documents, but rather through a uniform User Interface (UI). A number of quality attributes for the assessment of UI-features of SCA-Systems were proposed [KA13]. The goal was to improve usability, a measure of the effectiveness, efficiency and satisfaction a user achieves.

A number of tools for semantic content authoring were developed. A good example that adds Crowdsourcing capabilities to ontology development activities is Mechanical Protege [SWC+]. The tool is a plug-in for the ontology editor Protege[8]. It allows creating classification hierarchies or labelling concepts and translating them into different languages. The user can choose from a set of pre-configured tasks, adjust parameters such as task description or reward for crowd workers and create Crowdsourcing jobs for Amazon Mechanical Turk[9]. In the example shown in Figure 2.5 the user created a task to translate English labels into German labels for the concepts *Spiciness*, *Medium*,

[7] https://led.kmi.open.ac.uk/ accessed 2019/04/01
[8] https://protege.stanford.edu/ accessed 2018/08/07
[9] https://www.mturk.com/ accessed 2019/04/03

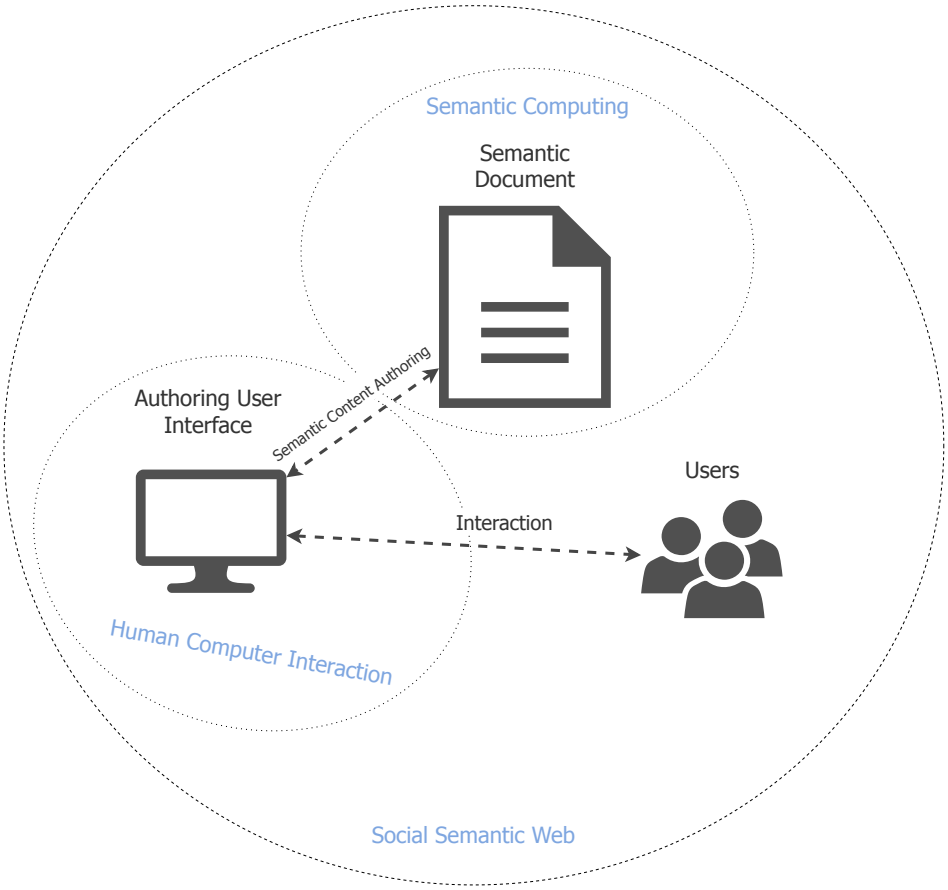


Figure 2.4: The ecosystem for semantic content authoring

Hot, *IceCream*, *Pizza* and *PizzaBase*. Furthermore, settings were adjusted to require 2 judgements for each concept and paying crowd workers 0.05 $ for task completion.

Data Linking The next principle according to the Linked Data Life-Cycle is the Data Linking principle. It is by far the most important principle because it underlines the distributed nature of Linked Data. Instead of the traditional definition of data where information is stored in silos with little or no relations to the outside, Linked Data sources are distributed, containing many links to other data sources. This paradigm perfectly fits into the distributed nature of the Web, turning it into a source of distributed information optimised for querying and browsing.

Using automated techniques of entity linking can be quite challenging, since parsing and disambiguating natural language text is considered a difficult task when done

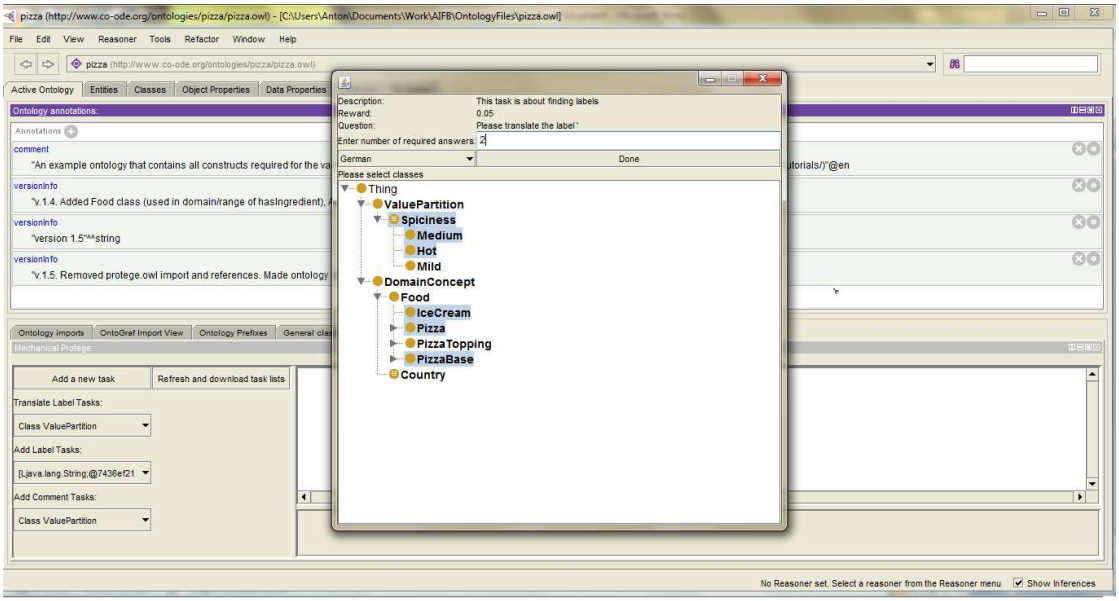


Figure 2.5: Crowdsourcing task creation in Mechanical Protege for translating concept labels

algorithmically. The process of relating entities extracted from text to their equivalents in the Linked Data space can be done automated (*Algorithmic Matching*) or with human intervention (*Manual Matching*). ZenCrowd [DDCM12] combines those two approaches by first trying an algorithmic approach and then improving the results by involving human workers. The main steps of the linking process are shown in Figure 2.6. The process takes as input a collection of HTML pages. Those pages were then inspected by the *Entity Extractors* to detect relevant textual entities. In the next step *Algorithmic Matchers* try to create links to semantically similar entities from the Linked Data set. The results of this process are stored in a *Probabilistic Network* which is taken by the *Decision Engine* to decide whether the results are useful or not. In the latter case, the HTML pages are passed to the *Micro-Task Manager* which uses Crowdsourcing to improve the results.

Data Classification and Enrichment Over time, Linked Data sources grow in size and expressiveness. This principle refers to the process of extending the expressiveness and richness of semantic knowledge bases. It means that instead of creating the structure upfront, the knowledge base evolves over time. For that, the knowledge base is typically

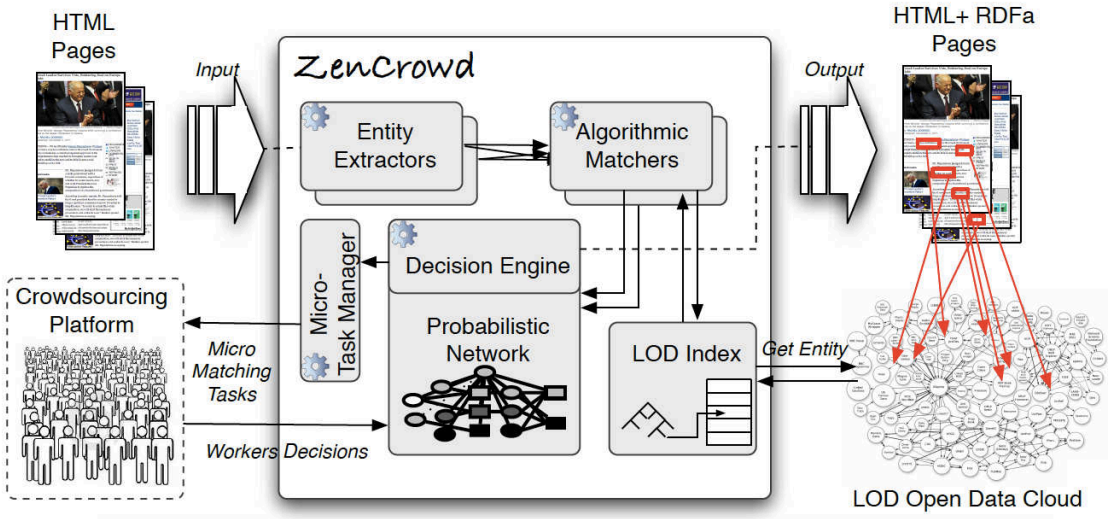


Figure 2.6: The linking process of ZenCrowd (adopted from [DDCM12])

enriched by analysing existing data to improve or extend its schema. A variety of (semi-) automatic enrichment approaches emerged over the past years. The methods span across several research areas, including machine learning, statistics, natural language processing, to name just a few.

Even though automating the process of ontology refinement greatly reduces the need for expert contribution, there are still some tasks which require human involvement. [WBS13] presented a model for evolving lightweight ontologies and a prototype which implements the model. The model describes the ontology learning process which uses Crowdsourcing to validate the concepts of the learned ontology. The algorithmic part of the process takes a seed ontology as input which is then continuously extended. This process is repeated until the target ontology reflects the semantics of the input source. The prototypical implementation only accepts textual input sources but future adaptations were planned. For the Crowdsourcing part a GWAP is used to eliminate unrelated concepts. The players of the game had to analyse the concept's relevance to the ontology's domain. The results of the Crowdsourcing part are taken to improve the quality of the learned ontology.

Data Analysis and Quality Due to the distributed nature of Linked Data where information originating from heterogeneous data sources is merged, the quality of the

resulting dataset is often varying. Therefore, a key factor for the adoption of Linked Data is ensuring its quality by identifying and fixing common problems. Before that, it is necessary to determine the quality of the existing dataset. For that, metrics which measure the quality in terms of accuracy, completeness, adequacy and degree of understandability need to be defined. [ZRM+16] defines useful quality metrics along with 26 quality dimensions that help to measure the quality of the Linked Data.

In [MMJ+15] the authors analysed 200 taxonomic relations of SNOMED CT, a widely used ontology mainly used in biomedical contexts. They used Crowdsourcing to detect inconsistencies and address the challenges of scalable ontology verification. The Crowdsourcing task interface shows the description of the evaluated relation and set of related concept definitions. The crowd worker can then either agree (by answering *yes*) or disagree (by answering *no*) on the presented statements.

Data Cleansing and Evolution After the quality is analysed and problems are detected, strategies for fixing these problems are needed. In constantly evolving datasets with millions or even billions of RDF triples it is important to keep the links between datasets. Likewise, conflicts and discrepancies in datasets can cause real trouble. Repairing such inconsistencies in overlapping datasets is called *Data Fusion*. Several works focusing on repairing problematic datasets appeared in the literature. For example, [MMB12] considers repairing with respect to data fusion and [FRV+12] investigates how provenance helps to improve the quality of Linked Data. In this context, provenance refers to where and how the data was obtained from.

The uComp Protege Plugin [WSH16] is a good example which combines Data Cleansing and Crowdsourcing. An in-depth explanation is out of scope for this section. However, a detailed description of the plugin is given in Section 2.3 as it represents the baseline of this thesis. As soon as the results from the crowdsourced ontology validation are available, the plugin guides the user to take further actions as summarised in Table 2.2. For each task the user can take advantage of the following cleansing actions: *Concept Removal* for the Verification of Domain Relevance, *Relation Removal* for the Verification of Relation Correctness, *Relation Labelling* for the Specification of Relation Type and *Domain/Range Removal* for the Verification of Domain/Range.

Data Browsing and Querying Last, users have the opportunity to explore the Linked Data available on the Web in a fast and efficient way. A prominent way to query Linked Data is the Semantic Protocol and RDF Query Language (SPARQL)[10], a query language specifically designed to retrieve and manipulate Linked Data. There are similarities between Structured Query Language (SQL) and SPARQL in terms of query structure, but there are also differences.

LexiTags [Ver13] is an example which combines Semantic Browsing and Crowdsourcing. It was initially designed as a tool for content management and emerged to a platform that

[10] https://www.w3.org/TR/rdf-sparql-query/ accessed 2019/01/17

Task Type	Cleansing Action
Verification of Domain Relevance	Concept Removal
Verification of Relation Correctness	Relation Removal
Specification of Relation Type	Relation Labelling
Verification of Domain and Range	Domain/Range Removal

Table 2.2: Data Cleansing capabilities of the uComp Protege Plugin

expose crowdsourced semantic metadata to clients, both for creation and consumption of metadata. Its main interface keeps a list of bookmarked URLs, allowing users to add and edit semantic tags. Semantic tagging is a very powerful innovation which helps users to navigate through large datasets by constructing search queries from a set of semantic tags.

2.3 The uComp Protege Plugin

In this section we present the uComp Protege Plugin [WSH16] on which our implementation builds on. The plugin was realised as a plugin for the Protege ontology editor[11]. It enables the automatic creation of Crowdsourcing tasks to support ontology validation, especially as part of Stage 6 and Stage 7 of the Linked Data Life-Cycle in Section 2.2.1. It was designed as a tool used by ontology engineers to reduce the burden of manual ontology validation.

2.3.1 Plugin Functionality

The plugin supports the following tasks which were previously performed by ontology experts in collaboration with domain experts:

Verification of Domain Relevance The goal of this task is to decide whether a given concept (or a set of concepts) is relevant for a given domain. For that, crowd workers need to answer a binary question, that is a question with a yes/no answer. The corresponding Crowdsourcing task is automatically generated by the platform and contains besides the actual concept that should be validated also the domain and optionally some additional information that is useful for answering the question.

[11] https://protege.stanford.edu/ accessed 2018/08/07

Verification of Relation Correctness Judging the correctness of relations, the plugin offers interfaces that allow the validation of subsumption and instanceOf relations. For subsumption, the crowd needs to decide whether a given concept is a subclass of another concept. For example, validating the correctness of the subclass relation *isSubClass(weather, rain)* for the domain *climate change*, crowd workers have to decide whether the concept rain is a sub-class (sub-concept) of the concept weather in the domain of climate change. For validating the correctness of instanceOf relations, the crowd needs to decide whether a given individual is an instance of a given concept (class). For example, contributors were asked to decide whether the individual *Bordeaux Region* is an instance of the concept *Region* for the *Wine* domain.

Specification of Relation Type This task is different from the others described above. Instead of answering binary questions, the crowd is asked to assign relation types to unlabeled object properties. A prerequisite for this task is that object properties that are selected for evaluation were previously labelled as *relation*. This way, the plugin knows which object properties take part in the validation process. Additionally, crowd workers can optionally suggest a new relation type if none of the suggested ones fit their needs.

Verification of Domain and Range The purpose of this task is mainly to identify problems that are relevant for reasoning rather than validating the ontology structure itself. In this task, the crowd was asked to validate domain and range restrictions as specified by Web Ontology Language (OWL). For example, the crowd needs to decide whether the object property *hasSister* maps a person (domain) to a female (range). As stated earlier, errors in range and domain restrictions have no impact on the ontology itself but are rather used by reasoners to infer additional knowledge.

2.3.2 Validation Workflow

Supporting ontology engineers was a major design goal of the uComp Protege Plugin. The workflow to create Crowdsourcing tasks that facilitate ontology validation is depicted in Figure 2.7. It involves the following steps:

Step 1 *(Task Specification)* In the beginning, the ontology engineer needs to choose from the tasks listed in Section 2.3.1. For each task a standalone interface was created which allows controlling task specific behaviour.

Step 2 *(Request Sending)* Next, the plugin gathers the information required to send a request to the uComp platform. Unfortunately, at the time of writing this thesis the platform is no longer maintained which required us to directly send the request to the Crowdsourcing provider.

Step 3 *(Crowdsourcing Job Creation)* The uComp platform then collects all the data required to create the Crowdsourcing job for the selected Crowdsourcing provider.

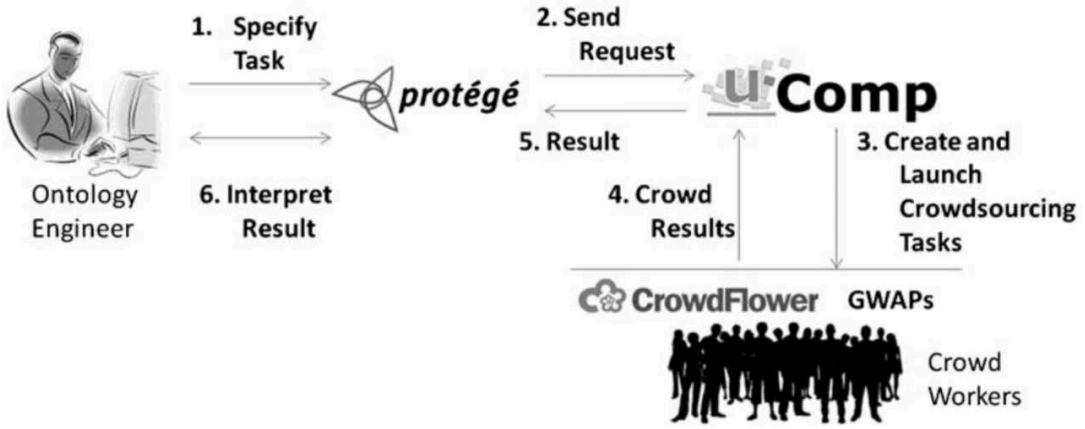


Figure 2.7: Main workflow to create Crowdsourcing tasks by the uComp Protege Plugin (adopted from [WSH16])

Initially, the uComp platform supported Crowdflower (which is now Figure Eight[12]) as the only Crowdsourcing provider. However, the creators of the uComp platform added the option to support other providers as well.

Step 4 *(Fetching Crowdsourcing Results)* After job completion (possibly lasting for several days), the results from the selected Crowdsourcing provider were collected.

Step 5 *(Aggregating Crowdsourcing Results)* The uComp platform collects all the results (possibly originating from different platforms) and then calculates the combined result.

Step 6 *(Result Interpretation)* The last step in the validation workflow is presenting the results to the ontology engineer. Depending on the chosen validation task, further actions may be taken. For example, in case the majority declined the relevance of a specific concept, the ontology engineer may decide to delete the corresponding concept. For all ontology manipulations, the plugin requires manual intervention. One of the design goals of the plugin was the guide ontology engineers in doing specific maintenance tasks rather than replacing them by performing these actions automatically.

2.4 The use of Context in Crowdsourcing Tasks

2.4.1 Context

When analysing the use of Context in Crowdsourcing tasks, we noticed that there exists no formal definition of the term »Context« . In fact, all approaches that were found use

[12]https://www.figure-eight.com/ accessed 2018/07/16

a different notion of that term. This section first investigates what context definitions these approaches use. After that, we give a consolidated definition that fits our approach of crowd-based ontology validation.

When investigating the use of Context in Crowdsourcing tasks, a good start is to look at [SSN+15]. In this work, the authors did an extensive literature study to find challenges in the context of Crowdsourcing and the Semantic Web. One of the challenges they found was a proper definition of Context as part of a complete task design. Concretely, they asked but did not answer the minimum required context a crowd needs to finish a task correctly. Unfortunately, during our studies we could not find an answer either. It seems that there exists no generic answer which applies in all contexts, it rather depends on the concrete type of task that needs to be solved.

During our literature research we found that approaches can be categorised as tasks supplying **explicit Context**, tasks supplying **implicit Context** and those providing **no Context** at all.

The most obvious work supplying *no Context* with Crowdsourcing tasks was actually done by [WSH16]. It represents the baseline of our work and motives our use of Context. A detailed explanation of this paper is out of scope for this section. However, a detailed explanation was already done in Section 2.3. In another paper, a method of collaborative ontology construction was proposed [ZGEJ17]. The actual definition of the ontology was implemented by a hybrid approach containing the definition of RDF-triples by non-experts (e.g. students) and their classification by the crowd.

Clearly, the omission of Context does not need to be problematic. Whereas crowd-based ontology validation without Context clearly has its drawbacks, it would not be beneficial if the crowd had additional information in the ontology construction example because the entities that formed the statements that were judged were simple and easily understandable by the crowd.

The other group of tasks that provide additional information are those tasks supplying *explicit Context*. The authors of [MMJ+15] and [MTH+16] supplied concept descriptions to improve the quality of the judgements. Their goal was to find inconsistencies and errors in SNOMED CT, a widely used ontology mainly used in biomedical contexts. Even though biomedical ontologies are well documented, not all entities have definitions. For that, English language definitions were manually added by domain experts.

The last category contains tasks with *implicit Context*, meaning that Context was not intentionally added. Context is rather defined implicitly, e.g. all Context is already present in the initial dataset. Hence, no additional process or algorithm is needed to define contextual information. For example, in [AZS+18] the authors used a Crowdsourcing data quality assessment tool to detect errors in Linked Data. For their analysis they used DBPedia [ABK+07] as evaluation source. Because one of the design principles of DBPedia was to derive linked information from Wikipedia[13], it seems natural to add the

[13] https://www.wikipedia.org/

link to the corresponding Wikipedia page to the Crowdsourcing task interface. To that end, no additional process or method for the enrichment of Context exists because the evaluated dataset already contains the Context.

A different approach was taken by [SWPB18, WSP+17a, WSP+17b] in which an Extended Entity Relationship (EER) diagram was verified against a software specification document in a software engineering use case. The diagram, initially created by students, was presented together with the specification text to detect and correct inconsistencies in the conceptual model. From the task description it seems obvious that no additional information is needed because Context was already given implicitly by the EER diagram. In that sense, the Context was defined in terms of the specific task that was carried out by the crowd.

Taken together all insights from above, we define »Context« in Crowdsourcing tasks as:

Definition 2.1. *(Context)* Context refers to any sort of additional information that is supplied with a Crowdsourcing task to improve its understanding in such a way that it positively affects the crowds performance and the result quality. Furthermore, we do not set a limitation on the type or format of Context that is provided. Examples are natural language descriptions, links to external content or pictures. We distinguish between Crowdsourcing tasks that 1) supply explicit Context, 2) those that supply Context implicitly and 3) those that provide no Context at all.

2.4.2 Approaches that use Context in Crowdsourcing Tasks

Based on the definition of »Context« in Section 2.4.1, in this section we give an overview of various approaches that use (or omit) Context in Crowdsourcing tasks. Table 2.3 summarises all approaches that were discussed in this section.

The first approach [AZS+18] investigates quality issues along the two dimensions *accuracy* and *interlinking*. The attributes that were evaluated by the crowd were incorrect object values, incorrect datatypes or language values and incorrect links. The evaluation was performed on a linked dataset extracted from the DBPedia corpus. The quality assessment consisted of a twofold approach. In the first stage, a group of Linked Data experts had to select possible candidates of RDF-triples that might have quality problems. In the second stage, the triples were evaluated by the crowd. Because DBPedia triples were constructed by knowledge extraction from Wikipedia, it seemed appropriate to display the link pointing to the corresponding Wikipedia page. Even though the results were promising, the full potential could possibly be reached by a hybrid approach which combines crowd-based evaluation with an automatic process that helps to reduce the number of triples that resort to Crowdsourcing.

The next work was initially proposed by [MMJ+15] and then extended by [MTH+16] to verify the quality of hierarchical relations in biomedical ontologies. They selected a random subset of 200 subsumption relations from SNOMED CT, an ontology that often serves as knowledge source in biomedical contexts. For the evaluation, the Crowdsourcing

Paper	Evaluated Unit	Evaluation Target	Context Type	Context
[AZS+18]	RDF triples	Data Quality	Implicit Context	Wikipedia Link
[MMJ+15, MTH+16]	Ontology Structure	Data Quality	Explicit Context	Concept Descriptions
[SWPB18, WSP+17a, WSP+17b]	Conceptual Model	Data Quality	Implicit Context	EER Diagram
[WSH16]	Ontology Structure	Data Quality	No Context	No Context
[ZGEJ17]	RDF triples	Data Definition	No Context	No Context

Table 2.3: Overview of approaches that Context in Crowdsourcing tasks

task was generated from subsumption relations and concept descriptions. Due to the complexity of the application domain, biomedical ontologies are well documented and naturally contain many concept descriptions. Those concepts with missing descriptions were enriched with documental information. The authors concluded that Crowdsourcing can compete with manual evaluation done by medical experts, however, certain tasks, especially more complex ones that are poorly documented, should be better done by domain experts.

A couple of researchers [SWPB18, WSP+17a, WSP+17b] investigated conceptual model verification from a Software Engineering perspective. They used Crowdsourcing techniques to verify the correctness of EER diagrams. In their first experiment, students had to create the conceptual model (EER diagram) from a specification document which was written in informal English language. The resulting models (diagrams) were then checked by the crowd to identify inconsistencies between the model and the specification text. In this setting, the EER diagram served as Context for the Crowdsourcing task. Their experiments achieved high Precision and Recall, however a few shortcomings will be addressed in their future work.

While all the Crowdsourcing tasks discussed so far had contextual information to some

extend, the approaches presented next completely omit Context. For example, [WSH16] which represents the baseline of our work and motives our use of Context. A detailed explanation was already done in Section 2.3.

The last paper that is discussed in this section is [ZGEJ17]. It covers the process of ontology construction, a costly and time consuming task that involves extensive expert participation. In this work, the authors presented a two-step approach. It consists of collaborative ontology construction by non-experts and classification of statements that were formed from RDF-triples. Special attention was paid towards the reduction or omission of subjective or biased judgements. For that, multiple viewpoints were merged to create a unified multi-viewpoint ontology. Whereas the initial task of collecting controversial subjects and creating multiple single-viewpoint ontologies was done manually by non-professionals, their classification to form one unified multi-viewpoint ontology was performed by the crowd. The results showed that no additional Context is needed if the domain is not too complex and the statements (relations) are easily understandable.

2.5 Summary

In this chapter we provided a brief overview of research fields that are relevant for our work.

In the beginning of this chapter we gave a brief introduction into Crowdsourcing (Section 2.1). We briefly discussed the potentials and risks and listed some fields in which Crowdsourcing techniques have been successfully applied.

Then, we focused on the interplay between the Semantic Web and Crowdsourcing (Section 2.2). We presented the Semantic Web life cycle and gave examples of Crowdsourcing approaches for each stage of the life cycle.

In Section 2.3 we presented the uComp Protege Plugin on which our implementation builds on. We described the plugin functionality and the supported ontology validation tasks. For the creation of Crowdsourcing tasks, we looked into the workflow of the plugin to facilitate ontology validation. Unfortunately, crowd workers often do not had enough knowledge to complete Crowdsourcing tasks. They need additional contextual information which improves their understanding. Before investigating our approaches which generate contextual information, we had to give a common definition of »Context«: Context refers to any sort of additional information that is supplied with a Crowdsourcing task to improve its understanding in such a way that it positively affects the crowds performance and the result quality. Furthermore, we do not set a limitation on the type or format of Context that is provided. Even tough there exists some approaches that use Context in Crowdsourcing tasks, they all use a different notion of Context.

CHAPTER 3

Context Enrichment Methods

In this chapter we describe our approaches of generating context descriptions in detail. All described approaches do not rely on some pre-defined settings or data, instead they are applicable to general purpose ontologies as well as specialised ones of arbitrary size.

In this chapter, we present three context enrichment methods for ontology validation. While the first one (**Ontology based Approach**), introduced in Section 3.2, takes neighbouring nodes (i.e. subclass relations) into account, the second one (**Metadata based Approach**), discussed in Section 3.3, is based on embedded context and the last one (**Dictionary based Approach**), explained in Section 3.4, uses external sources as input for enrichment.

3.1 Introduction

Previous experiments using the uComp Protege Plugin [WSH16] on which this thesis builds on, had successfully applied Crowdsourcing techniques for ontology validation and extension. They conclude that it leads to data quality comparable to that of manually performed validation done by ontology engineers while reducing the overall costs. However, there was still potential to improve the worker performance in terms of speed and quality. Studies [Mor13] confirmed this statement concluding that the best worker performance is achieved "with questions formulated in the most basic form, a domain-specific qualification, and concept definitions for context".

Based on Protege as an ontology authoring tool, we fill this gap by investigating different approaches of Context creation. All of these methods were implemented on our platform which was developed to facilitate the creation of Crowdsourcing jobs within Protege. The *first variant* (**Ontology based Approach**) uses the relations encoded by the ontology itself to generate textual definitions to serve as Context in Crowdsourcing tasks. At the current state, only subsumption relations are taken into account, but the algorithm

is rather generic which facilitates future adaptions (e.g. including other relation types for Context creation). The *second variant* (**Metadata based Approach**) depends on annotations embedded in the ontology which were manually added by ontology engineers. Among various metadata standards which define common meanings of annotation content, our approach is based on the Dublin Core vocabulary. For the *third variant* (**Dictionary based Approach**) the idea was to generate Context by consulting an online dictionary. We decided in favour of WordNik, which provides access to a large number of online content including tweets, newspaper articles and scientific articles. For the Context creation, example sentences were fetched including the requested concept name.

3.2 Ontology based Approach

This section starts with a short introduction into Attempto Controlled English (ACE) which is a formal language, capable of expressing domain-specific knowledge by human readable sentences. In the literature [Sma08] this task is also known under the term *ontology verbalisation*. Even though there exists *OWL Verbalizer*[1], a tool which transforms generic ontologies into English sentences, we could not integrate it into the Context enrichment process because 1) it was designed as a standalone tool written in SWI-Prolog[2] and 2) it only accepts the whole ontology as input . While 1 could be solved using JPL[3], a library written in SWI-Prolog providing a bidirectional interface between Java and Prolog, it is still considered experimental and would require considerable integration efforts. We could not think of a reasonable solution to 2 because Crowdsourcing jobs could also be generated from a subset of the concepts defined in an ontology. However, in future versions of OWL Verbalizer these limitations might be solved as there exists a ticket[4] for that.

As in the guidelines for conducting Crowdsourcing research [SSN+15], the authors recommended to avoid technical terms in Crowdsourcing questions. In the next paragraphs we explain how *ontology verbalisation* helps to achieve this goal.

3.2.1 Attempto Controlled English (ACE)

Despite the fact that natural language is desirable for descriptions as everybody knows and understands with no extra learning effort, it conflicts in terms of expressiveness and specificity with well defined ontologies which can encode complex data and relations in domain-specific areas. To resolve this conflict, a new language variant named **ACE** was created [FKK08]. ACE is a formal language, capable of expressing domain-specific knowledge with a well defined syntax, supporting formal reasoning and readable by specialists who are yet unfamiliar with formal languages and methods.

[1] http://mcs.open.ac.uk/nlg/SWAT/Verbaliser.html accessed 2018/04/30
[2] http://www.swi-prolog.org/ accessed 2018/04/30
[3] http://www.swi-prolog.org/packages/jpl/ accessed 2018/11/30
[4] https://github.com/Kaljurand/owl-verbalizer/issues/13 accessed 2018/11/30

To get a better understanding of ACE[5,6], a short overview of its language structure is given in the next paragraphs:

Simple Sentences A Simple Sentence derived from standard English language contains a subject, a verb and additional elements: subject + verb + complements [+ adjuncts]. The verb relates directly or indirectly to one or more other objects (*complements*). Optionally, to add more specificity, one or more adverbs and prepositional phrases can be added (*adjuncts*).

Composite Sentences A Composite Sentence is composed of one or more Simple Sentences, connected by *coordination*, *subordination*, *quantification* and *negation*. Whereas coordination links sentences either by the word *and* or *or*, subordination relates dependent sentences in some way (e.g. if-then sentences). Quantification allows statements about all (universal quantification) or certain (existential quantification) objects of a specific domain. Last, encoding negative polarity in a sentence (e.g. sentences containing not or no) is defined as negation.

Query Sentences Query Sentences can be divided into polar questions (e.g. with *yes/no* answer) and non-polar questions, also known as *wh-questions*. In contrast to yes/no questions no pre-defined answer exist for these. Furthermore, wh-questions start with either of the following five W-words: Who, What, When, Where and Why. However, this definition is somewhat less strict as sometimes questions starting with the word How are included as well.

Anaphoric References If the meaning of a word or phrase is context dependent, recurring occurrences of these expressions are called *Anaphoric References*. More specifically, the referring term (*anaphor*) relates to an antecedent expression. For example, given the sentence: Tom arrived, but nobody noticed him, the pronoun him relates to Tom.

3.2.2 Ontology based Approach Realisation

Based on the ACE rules described above we implemented an algorithm which generates Context descriptions based on subsumption relations. The pseudocode of the overall workflow is given in Algorithm 1. The notation to describe properties and relations is based on a formal Ontology Description Logic (DL) [BN03], string manipulations were formally defined in [HU69].

The main work is done in two for-loops, which calculate Context descriptions based on subsumption ($\sqsubseteq$) and string concatenation ($\cup$). To handle the case of missing subsumption relations, the output text T is initialised to an empty string (Line 2). Next, for every

[5]https://tinyurl.com/yc3zhu9a accessed 2018/05/05
[6]https://tinyurl.com/ycst39jv accessed 2018/05/05

Algorithm 1 Context Enrichment based on Neighbouring Nodes

1: **procedure** Generate Description
Input: A concept C
Output: A textual description T of $C's$ neighbouring nodes based on subsumption

2: $T = \{\}$
3: **for** $(c, d) \in C \sqsubseteq D$ **do**
4: $T = T \cup$ "Every " $\cup\ name(c) \cup$ " is a " $\cup\ name(d)$
5: **for** $(e, c) \in E \sqsubseteq C$ **do**
6: $T = T \cup$ "Every " $\cup\ name(e) \cup$ " is a " $\cup\ name(c)$

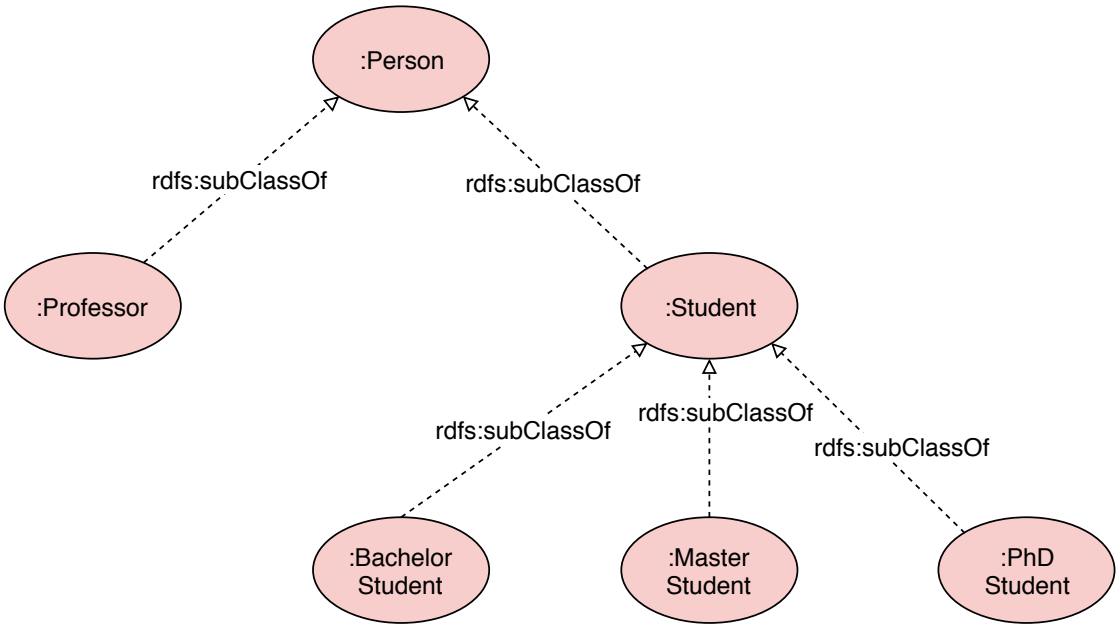


Figure 3.1: Simple Ontology Graph describing the student/professor relationship

subsumption relation having the input concept C in its signature, either $C's$ name or the anchor node's name is appended first.

A simple ontology graph describing the university domain is given in Figure 3.1. It will be used to illustrate the concept of the Context generation algorithm. If the concept *Student* is taken as reference node, the algorithm first collects all subsumption relations having *Student* as child node which gives $\{Person\}$ and generates $T = \{$ *"Every Student is a Person"* $\}$. Next, all subsumption relations having *Student* as parent node were collected which gives { *Bachelor Student, Master Student, PhD Student* } and generates $T = \{$ *"Every Student is a Person", "Every Bachelor Student is a Student", "Every Master Student is a Student", "Every PhD Student is a Student"* $\}$.

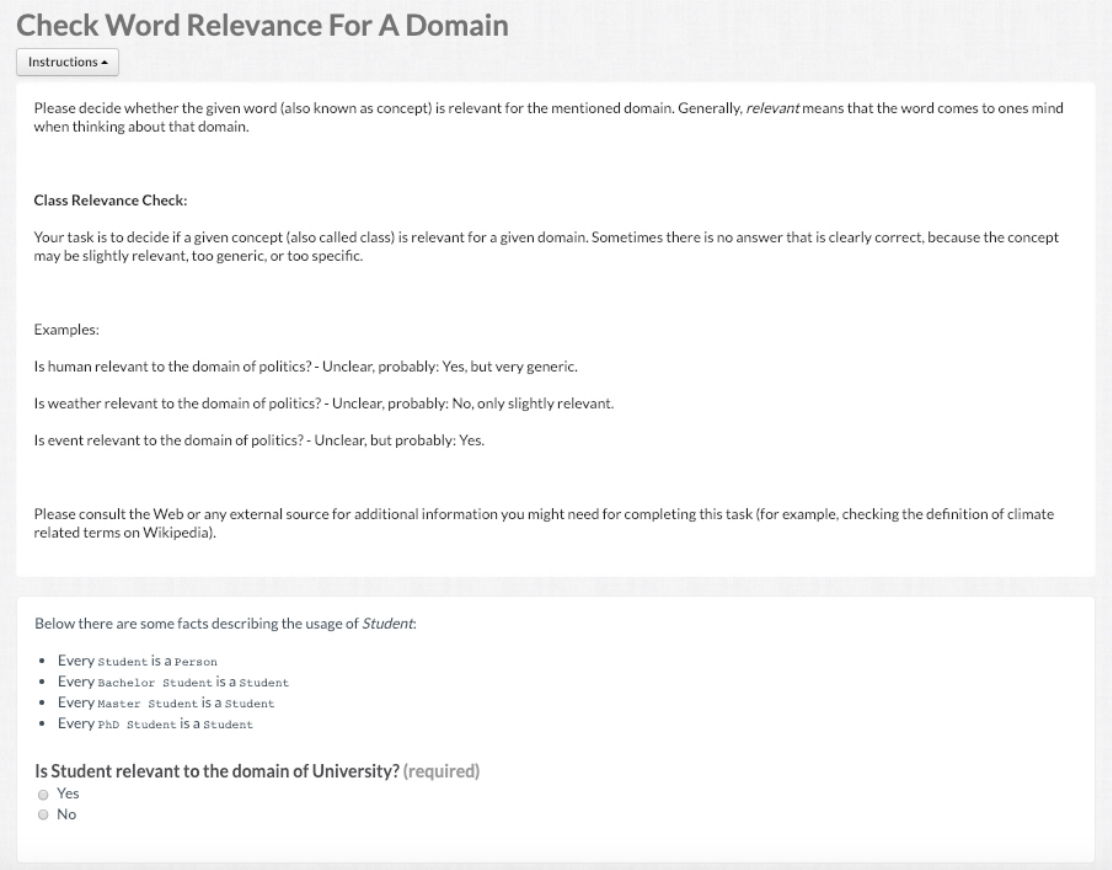


Figure 3.2: Questionnaire presented to crowd workers for the university domain example

After Context generation, our platform creates the questionnaire for crowd workers which also includes the actual question for ontology validation and some instructions for guidance. Figure 3.2 depicts the questionnaire presented to crowd workers for the university domain example.

3.3 Metadata based Approach

In this section we describe another approach of generating Context descriptions based on semantic metadata. For that, we used Annotation Properties which were defined as part of OWL[7]. To maximise interoperability with existing libraries for ontology processing and manipulation we made use of the Dublin Core Metadata Set, a standard vocabulary designed to annotate resources with simple, textual information. A prerequisite for Context generation is the presence of such metadata information. However, as none of our ontologies contained such metadata, we had to add them manually.

[7]https://www.w3.org/OWL/ accessed 2018/18/12

This section starts by introducing annotation properties which are defined as part of OWL (Section 3.3.1). We used annotations to encode the Context descriptions. Next, an overview of the Dublin Core Metadata Set is given as some parts were used for the definition of Context properties (Section 3.3.2). Then, in the remainder of this section our Metadata based Approach is discussed (Section 3.3.3).

3.3.1 OWL Annotation Properties

Annotation properties first defined by OWL 1[8] and then extended by OWL 2[9] are used to enhance concepts, properties, individuals and ontology headers with meta data such as labels, comments, creation date and so forth. This information does not alter the semantics of the ontology in any way, it is merely intended for documentation purposes and therefore ignored by reasoning engines.

Besides the built-in annotation properties OWL 1 also offers the ability to create user-defined annotation properties. An example of using *owl:AnnotationProperty* to declare a user-defined annotation property is given in Listing 3.1. In this example, the OWL Class *Lens* is annotated with the custom annotation property *dc:date* which is defined by the Dublin Core Metadata Set discussed in the next section.

Listing 3.1: Declaration of user-defined annotation property in OWL 1

```
<rdf:RDF
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:dc="http://www.purl.org/metadata/dublin-core#"
        xmlns:owl="http://www.w3.org/2002/07/owl#">

        <owl:AnnotationProperty
                rdf:about="http://purl.org/metadata/dublin-core#date"/>

        <owl:Class rdf:about="http://www.photo.org/camera#Lens"
                <dc:date rdf:datatype="http://www.w3.org/2001/XMLSchema#date">
                        2018-12-20
                </dc:date>
        </owl:Class>

</rdf>
```

3.3.2 Ontology Metadata Standards

Over the years ontologies were used in many contexts, including general-purpose as well as highly specialised ones. Obviously, what separates good ontologies from poor ones is how well they are documented [dN12]. Studies [DTEJ17] analysed various approaches of embedding metadata in ontologies. The outcome was that there is no standard way to describe and document ontologies, albeit a few vocabularies that describe semantic metadata exist. Two of the most common vocabularies are briefly described next.

Dublin Core Metadata Set Being one of the most prominent vocabulary in describing semantic metadata, published and maintained by the Dublin Core Metadata Initiative

[8] https://www.w3.org/TR/owl-ref/#Annotations accessed 2018/20/12
[9] https://www.w3.org/TR/owl2-syntax/#Annotation_Properties accessed 2018/20/12

(DCMI), it originally contained 15 metadata terms[10], designed to annotate resources with simple, textual information. Since its first launch, the project have gained popularity, including more than 127 terms[11]. The initial set of terms is listed in Appendix A.

To maximise interoperability in heterogeneous environments, an RDF-Schema with DCMI-Metadata[12] elements was created, in which each entity is identified by a Uniform Resource Identifier (URI) starting with the prefix *http://purl.org*. A broader discussion on the use of metadata in general is given in [Nil10].

Simple Knowledge Organization System (SKOS) The SKOS Core Vocabulary [MMWB05] defines a set of RDF properties and Resource Description Framework Schema (RDFS) classes used to express the content and structure of a concept scheme, which describes sets of concepts with optionally linked concepts. The vocabulary is standardised by the W3C[13]. Relevant terms are listed in Appendix B.

There is some overlap between Dublin Core (DC) and SKOS. For example, the terms *dc:subject* and *skos:subject* describe similar characteristics of an entity. However, in some scenarios the range of skos:subject is limited to resources of type skos:concept compared to the unrestricted range of dc:subject.

3.3.3 Metadata based Approach Realisation

Given the high number on ontology metadata formats from above, Algorithm 2 shows the pseudocode to create concept descriptions extracted from embedded metadata. In addition to the notation used in the previous section we define $\Phi(C) \coloneqq \{m_1, m_2, \ldots, m_i\}$ where m_i is the $i'th$ metadata element embedded in concept C and T is the description of some metadata element.

Algorithm 2 Context Enrichment based on embedded metadata

1: **procedure** GENERATE DESCRIPTION
Input: A concept C with embedded metadata $\{m_1, m_2, \ldots, m_i\}$
Output: A description T of $C's$ metadata elements

2: $T = \{\}$
3: **for** $m_k \in \Phi(C)$ **do**
4: $T = T \cup m_k$

While the actual enrichment is straightforward, it collects all descriptions for a determined concept, the details of extracting the metadata from annotation properties is omitted here because it highly depends on the chosen metadata encoding. As we decided to

[10] http://www.dublincore.org/documents/dces/ accessed 2018/05/20
[11] http://www.dublincore.org/documents/dcmi-terms/ accessed 2018/05/20
[12] http://dublincore.org/schemas/rdfs/ accessed 2018/05/20
[13] https://www.w3.org/TR/skos-reference/ accessed 2018/05/20

encode the metadata in annotation properties, the extraction process works by selecting the related annotation properties for a specified concept.

To illustrate the concept of the Context generation algorithm a simple example of an OWL Class enriched with metadata is shown in Listing 3.2. For that, the algorithm generates $T = \{$ *"Greenhouse gas is one of several gases, especially carbon dioxide, that prevent heat from the earth escaping into space, causing the greenhouse effect. Greenhouse gases from human activities are the most significant driver of observed climate change since the mid-20th century."*, *"greenhouse gas"* $\}$. Figure 3.3 depicts the questionnaire presented to crowd workers for the example from above. The text was constructed from the description defined by the Dublin Core Metadata Set and the label defined by RDFS[14].

Listing 3.2: An OWL Class enriched with metadata

```
<rdf:RDF
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
        xmlns:dc="http://www.purl.org/metadata/dublin-core#"
        xmlns:owl="http://www.w3.org/2002/07/owl#">

        <owl:Class rdf:about="http://www.climatechange.org/greenhouse_gas"
                <dc:description>
                        Greenhouse gas is one of several gases, especially carbon dioxide, that
↪ prevent heat from the earth escaping into space, causing the greenhouse effect. Greenhouse gases
↪  from human activities are the most significant driver of observed climate change since the mid
↪ -20th century.
                </dc:description>
                <rdfs:label>
                        greenhouse gas
                </rdfs:label>
        </owl:Class>

</rdf>
```

3.4 Dictionary based Approach

An alternative method of Context enrichment is based on acquiring concept definitions from external sources, especially when these are not already available as metadata annotations in the ontologies that are validated. The lookup is solely based on the concept's name, neglecting the connected nature of an ontology. Dictionaries have always been the first choice when it comes to searching for specific information about words or phrases. We chose *WordNik*[15] as source for external content, a freely available online dictionary for the English language. Among other features that were offered, we used *example sentences* that were collected from various sources across the Web.

This section begins with a brief introduction into WordNik, the online dictionary we used for the provision of example sentences, and then continues with our approach of using WordNik as content provider for concept descriptions.

[14] https://www.w3.org/TR/rdf-schema/ accessed 2018/12/30
[15] https://www.wordnik.com/ accessed 2018/06/15

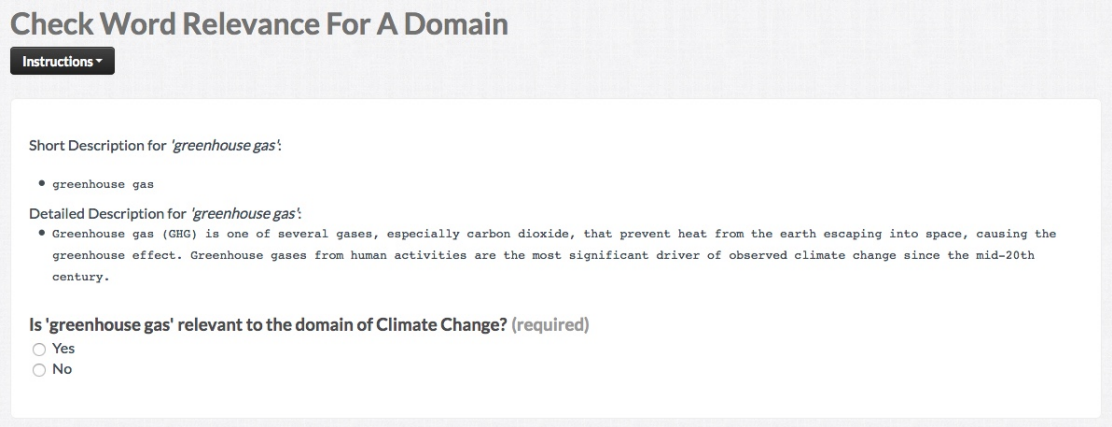


Figure 3.3: Questionnaire presented to crowd workers for the OWL Class example

3.4.1 WordNik

WordNik targets native English speakers who look up words that are rare (technical terms or dialect terms), very old or very new. They often search for definitional information which is incomplete or missing in traditional dictionaries. Users tolerate published imperfection because they opt for relevant, actual and cutting-edge information, even though not officially approved by editors [Bur79]. They want to understand the context of word usage in sentences, not necessarily explanatory statements as in printed or even online dictionaries.

The driving force behind WordNik was contribution. It processes and aggregates external user-generated content such as tweets, newspaper articles, scientific articles or uploaded Flickr[16] images. This is similar to what search engines do, but with restricted scope. The creators of WordNik observed that very few people write word definitions, they rather add meta linguistic information such as lists of their favourite words, comments or tags. WordNik additionally collects statistics about lexicographical terms, more or less frequently searched words and most commented words.

WordNik also offers an Application Programming Interface (API) for programmatically accessing their resources[17]. Besides the word definition, it also provides access to audio metadata, etymology, word usage, syllable information, bi-gram phrases, text pronunciations, relation diagrams to other words, example sentences and others. At the time of writing this thesis free access is granted for non-profit, non-commercial use with a limitation on the number of API calls. After a successful registration process, an API token is provided which is a prerequisite for API interaction. Besides Web access, a handful of libraries[18], available in many programming languages, were created to facilitate integration with third-party applications.

[16] https://www.flickr.com/ accessed 2018/06/15
[17] https://developer.wordnik.com/ accessed 2018/06/15
[18] https://developer.wordnik.com/libraries accessed 2018/06/15

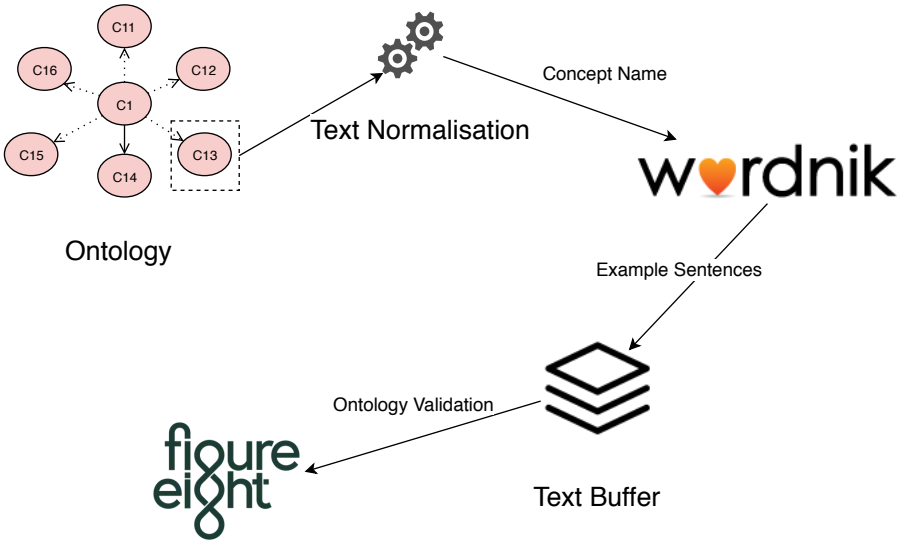


Figure 3.4: Conceptual workflow of WordNik consultation to generate concept descriptions

3.4.2 Dictionary based Approach Realisation

Intuitively, the idea of generating descriptions using dictionary lookup is simple: starting from a concept name, descriptions are built from consulting the online dictionary WordNik.

A schematic overview of the overall workflow is shown in Figure 3.4 and described below:

[Step I] *(Concept Selection)* The first step in the workflow is the selection of the concept(s) for generating the description. For that, the ontology engineer selects the concept(s) and starts the enrichment process.

[Step II] *(Text Normalisation)* The idea is to use the concept name as a baseline for any further processing. Often, the name can not be used directly as input to WordNik because it contains unwanted characters such as excessive spaces, quotes, dots or just non-printable characters. This is especially true for learned ontologies, generated from textual sources. Our algorithm uses the built-in text manipulation capabilities of the Java Development Kit (JDK) to pre-process concept names.

[Step III] *(Dictionary Lookup)* Next, WordNik is consulted to find example sentences for normalised concept names. In contrast to traditional dictionaries, WordNik searches in all kinds of available online content, including newspapers, journals,

scientific publications, tweets and others. All API interaction is protected against unauthorised access, however, to help developers learning the API, some features are available in isolated Sandbox Mode[19] too.

For example, when searching for the word »chartjunk« which does not have a definition in traditional dictionaries, the API response is illustrated in Listing 3.3. The output is encoded in JavaScript Object Notation (JSON)[20] which defines a common, human-readable format for data transmission. The example shows one *example sentence* (the others were omitted because they share the same structure) including various other properties besides the actual title and text. Our algorithm just skips these other properties because they were not needed for the final concept description. However, it might be useful in certain scenarios to differentiate between duplicate entries by exampleId or exploring further details by adding the source Uniform Resource Locator (URL).

[Step IV] *(Text Buffering)* Depending on wether a single concept or multiple concepts are validated, example sentences need to be harmonised, which is realised by storing intermediate results and mapping these to the initial concepts.

[Step V] *(Crowdsourcing Submission)* The last step of the workflow is the creation of the questionnaire for the actual ontology validation. As for all enrichment methods, the only part that varies for each approach is the concept description, shown as top part of the template. Figure 3.5 depicts the questionnaire presented to crowd workers for the example from above.

Listing 3.3: WordNik API response for the word »chartjunk«


```
{
  "examples": [
    {
      "provider": {
        "name": "spinner",
        "id": 712
      },
      "year": 2008,
      "rating": 185,
      "url": "http://www.emersonprocessxperts.com/archives/2008/10/improving_how_y.html",
      "word": "chartjunk",
      "text": "Marshall described \"chartjunk\" as additional graphics not related to the data
↪ in a quest to make the chart more aesthetically pleasing.",
      "title": "Emerson Process Experts",
      "documentId": 15463705,
      "exampleId": 289744774
    },
        ...
  ]
}
```


[19] https://developer.wordnik.com/docs accessed 2018/06/21
[20] https://tools.ietf.org/html/rfc7159 accessed 2019/01/05

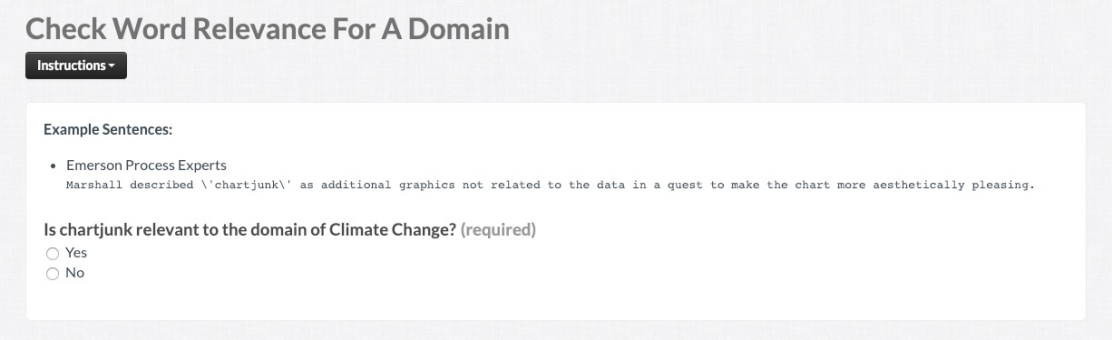


Figure 3.5: Questionnaire presented to crowd workers for searching »chartjunk« on WordNik

3.5 Summary

In this chapter we investigated three approaches that generate descriptions for selected concepts. The motivation was the limitations imposed by the existing platform for crowd-sourced ontology validation (uComp Protege Plugin). All our approaches were integrated as an extension of the platform which facilitates comparability of crowd worker performance, described extensively in the Results Chapter.

This first method (*Ontology based Approach*) uses the ontology graph, more precisely subsumption relations, to generate concept descriptions. For the second method *(Metadata based Approach)* additional metadata encoded within the ontology is processed. Therefore, some parts of the Dublin Core Metadata Set were used which define a standard set of OWL annotations that were extended with common semantics. The basic principle of the third method (*Dictionary based Approach*) was the retrieval of definitions from external sources. For that, WordNik, a freely available online dictionary for the English language which aggregates results from across the Web, was consulted to find example sentences for some selected concepts.

To conclude, while all approaches presented in this chapter generate descriptions for selected concepts, for the *Ontology based Approach* no additional preprocessing of the input ontology or dependency to an external service is required. On the other hand, the *Metadata based Approach* requires little to significant human intervention depending on the number of annotations that were present in an ontology. Even though the *Dictionary based Approach* requires availability of an external service (WordNik), showing the word usage by example sentences can be really helpful.

CHAPTER 4

Experimental Evaluation

In this chapter we describe our approach to evaluate the performance of context-enriched, crowd-sourced ontology validation. More precisely, in Section 4.1 we start by describing all relevant performance metrics used to quantify the improvements. Then, in Section 4.2 an overview of the used datasets (e.g. ontologies) is given and finally, Section 4.3 shows various interfaces which were presented to contributors to facilitate Crowdsourcing task completion.

Evaluation Hypothesis

Based on existing efforts for ontology validation using Crowdsourcing (see Section 2.3), we formulate the following evaluation hypothesis:

> The crowd performs ontology validation steps better if context is added to Crowdsourcing tasks.

To evaluate the hypothesis stated above, we extended the uComp Protege plugin to generate descriptions based on our proposed Context generation methods (Chapter 3). Table 4.1 gives an overview of the experiments including their settings and used datasets, described thoroughly in the next sections.

4.1 Evaluation Metrics

To justify the improvements stated in the hypothesis, a detailed evaluation based on the metrics described below was performed.

We used two approaches to measure the performance of the crowd. The first one requires some reference data which is compared against empirical data. This approach, originating from Information Retrieval (IR), is called the **Golden Standard Approach** [BGM05].

Methods	*Data*		*Crowdsourcing Settings*		
	Ontology	No. of Classes	Judgements/ Price	Worker Selection	Quality Control
None, Meta, Onto, Dict	Climate	101	5/0.05	Level 3, AUS, UK, USA	Quiz
None, Meta, Onto, Dict	Tennis	52	5/0.05	Level 3, AUS, UK, USA	Quiz
None, Meta, Onto, Dict	Finance	77	5/0.05	Level 3, AUS, UK, USA	Quiz

Table 4.1: Overview of performed ontology validation tasks, including datasets and settings. {*Meta*=Metadata based Approach, *Onto*=Ontology based Approach, *Dict*=Dictionary based Approach}

To quantify the improvements/degradations, several metrics exist. On a binary classification scheme, as illustrated in Figure 4.1, these metrics are defined as fractions of *True Positives (TP)*, *True Negatives (TN)*, *False Positives (FP)* and *False Negatives (FN)*. In Crowdsourcing contexts this means that yes-questions are either correctly (TP) or incorrectly (FN) answered and no-questions are either correctly answered (TN) or incorrectly (FP).

Precision Precision is interpreted as the ratio of correctly answered yes-questions over the total number of answered yes-questions:

$$Precision = \frac{TP}{TP + FP}$$

For concept relevance, values of Precision close to 1.0 show that the crowd correctly rejects irrelevant concepts but maybe fails at accepting relevant ones.

Recall Recall is interpreted as the ratio of correctly answered yes-questions over the total number of available yes-questions:

$$Recall = \frac{TP}{TP + FN}$$

For concept relevance, values of Recall close to 1.0 show that the crowd correctly predicts relevant concepts but maybe fails at rejecting irrelevant ones.

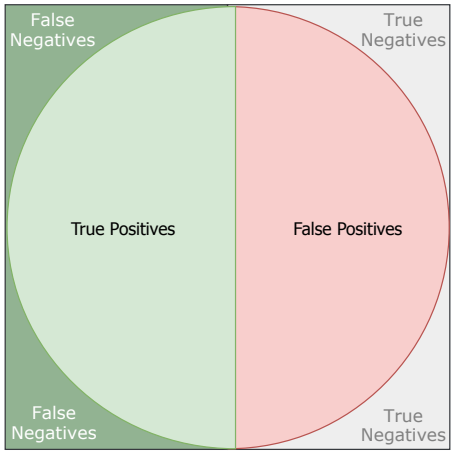


Figure 4.1: Binary classification scheme for evaluation metrics of Crowdsourcing tasks

F-Measure Unfortunately, exclusively relying on either of the above metrics has some drawbacks. For example, the crowd may correctly identify relevant concepts but fails at rejecting irrelevant ones (high Recall) or, on the other hand, irrelevant concepts may be correctly rejected whereas not all relevant ones may be detected (high Precision).

The F-Measure compensates these flaws by combining Precision and Recall rates. The traditional F-Measure or balanced F-Score is calculated as the harmonic mean of Precision (P) and Recall (R):

$$F\text{-}Measure = 2 \cdot \frac{P \cdot R}{P + R}$$

In some situations researchers have criticised this metric that it may be biased [Pow11]. For this reason, there exists a modified version of the general F-Measure which takes an additional parameter β into account:

$$F_\beta = (1 + \beta^2) \cdot \frac{P \cdot R}{\beta^2 \cdot P + R}$$

Depending on the importance of Precision or Recall, β can be set to a higher value (e.g. F_2), which weights Recall higher, or to a lower value (e.g. $F_{0.5}$), which puts more emphasis on Precision. Mostly, the generic F-Measure, also known as F_1 measure, is sufficient though, in which β is set to 1 to weight Precision and Recall evenly.

The other approach in measuring the crowd's performance does not rely on reference values, instead, the metric reflects the agreement ratio among crowd workers. Therefore, the agreement ratio or **Inter-rater Agreement** measures, to what extent judges reached consensus. For binary tasks (e.g. concept relevance checks), all possible outcomes are based on a table of 2x2 frequencies, as shown in Figure 4.2. In terms of evaluating

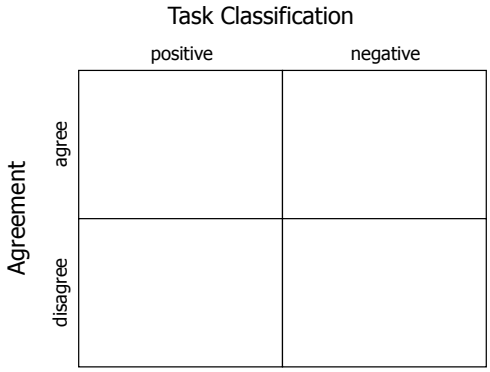


Figure 4.2: 2x2 outcome table on Inter-rater Agreement for binary tasks

concept relevance, this means that the crowd either agrees or disagrees whether a concept is relevant or not.

Several metrics exist to measure inter-rater reliability [ZLD13]. They primarily differ to what extent judgements made by chance are taken into account. In our evaluation though, the following metric is used:

Percentage Agreement This is the simplest and most commonly used metric, which is calculated by dividing the number of agreeing raters (A) by the total rater count (N).

$$Agreement = \frac{A}{N}$$

Despite its intuitive appeal, it has been criticised, that it does not take the agreements made by chance into account [Hun86]. On the other hand, calculation of chance-adjusted metrics is more complex and have the potential to over- or undervalue the corrections for chance. Moreover, reliability is assumed to be very high because Crowdsourcing settings were adjusted to sort out random answers.

4.2 Datasets

Within the next paragraphs, ontologies used as the input for evaluation tasks are described in more detail. As this thesis builds on existing work [WSH16], it makes sense to use the same ontologies as evaluation source. Also, we had access to the raw evaluation data which were previously used.

The main characteristics of the three ontologies used for evaluation are summarised in Table 4.2. Two of these ontologies, covering the domains *climate change* and *tennis*, emerged from seed ontologies used in an ontology learning algorithm [LWSC05]. They evolved from several rounds of adding more input data [WWSS12]. The other ontology

Number of	*Ontology*		
	Climate Change	Tennis	Finance
Classes	101	52	77
Properties	28	34	29
SubClass Relations	84	35	78
Individuals	64	33	47

Table 4.2: Characteristics of the used ontologies

covers the finance domain and represents a small subset of the vocabulary defined by the Multilingual Thesaurus of the European Union (EuroVoc)[1].

The tested ontologies are of limited size which makes evaluation easier, but still has significance for testing the impact of Context enrichment in ontology validation. Whereas the *Climate Change* ontology contains 101 concepts, 28 object properties, 64 individuals and 84 subclass relations, the *Finance* ontology is of smaller size, containing 77 concepts, 29 object properties, 47 individuals and 78 subclass relations. The *Tennis* ontology has 119 entities in total, 52 of which are concepts, 34 object properties and 33 individuals.

Evaluation Setup

For calculating evaluation metrics the ontologies need to be annotated with reference values. From previous experiments [WSH16] evaluation data was consolidated and annotations were generated. Unfortunately, for some concepts we had ambiguous data or none at all. We manually verified the enriched ontologies by excluding incorrect annotations and adding missing ones where appropriate. This was an important task, particularly because learned ontologies often contain inconsistent and inaccurate data.

Concerning Crowdsourcing tasks, Figure Eight[2] (former CrowdFlower) allows adjusting a variety of settings. We paid $0.05 per task, required 5 judgements per unit and restricted judgements to the highest quality level of crowd workers (Level 3). Additionally, we made the assumption that all labels of the validated ontologies are in English, therefore achieving results of higher quality requires restricting participation to the following English speaking countries: Australia, United Kingdom and United States. Furthermore, crowd workers had to correctly answer 8 quiz questions from politics, computing and tennis in order to qualify for accessing our tasks. Although this does not prevent contributors from randomly answering test questions, it provides at least a minimum of

[1] http://eurovoc.europa.eu/drupal/?q=evontology accessed 2018/07/13
[2] https://www.figure-eight.com/ accessed 2018/07/16

quality control. Without any quality control measures, results would be of little use, as a recent survey reveals [DKC+18].

A central part of the assessment is the definition of evaluation tasks. Crowd workers were consulted to assist in the following ontology engineering task:

Verification of Domain Relevance For each selected concept, crowd workers need to decide whether it is relevant for the domain in question (in our case, Climate Change, Tennis and Finance). Using domain relevance, we evaluated our proposed methods: *Ontology based Approach* (Section 3.2), *Metadata based Approach* (Section 3.3) and *Dictionary based Approach* (Section 3.4). Each of these generates textual descriptions which were added to the Crowdsourcing task. For the Metadata based Approach we had to manually annotate the ontologies. For the Dictionary based Approach WordNik[3] was consulted to provide example sentences. No pre-processing was necessary for the Ontology based Approach.

4.3 Crowdsourcing Task Interfaces

In this section some example interfaces are presented which were shown to crowd workers for each verification task.

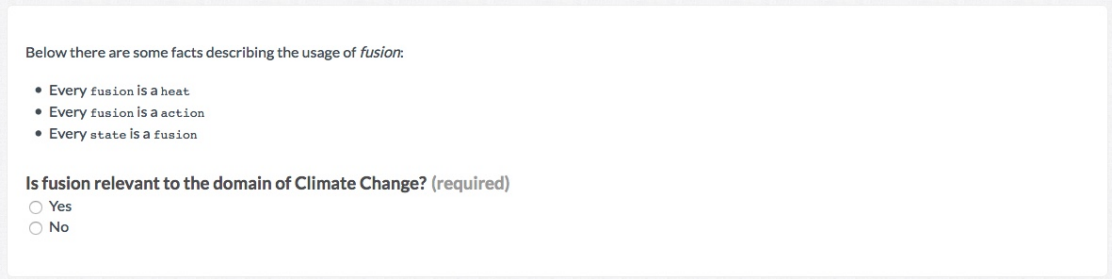
After selecting the concepts for ontology validation, the plugin automatically creates the relevant Crowdsourcing jobs. Only Figure Eight is currently supported as Crowdsourcing platform. Depending on the method of Context enrichment (see Chapter 3) different Crowdsourcing interfaces were generated, as illustrated in Figure 4.3.

Each Crowdsourcing interface consists of

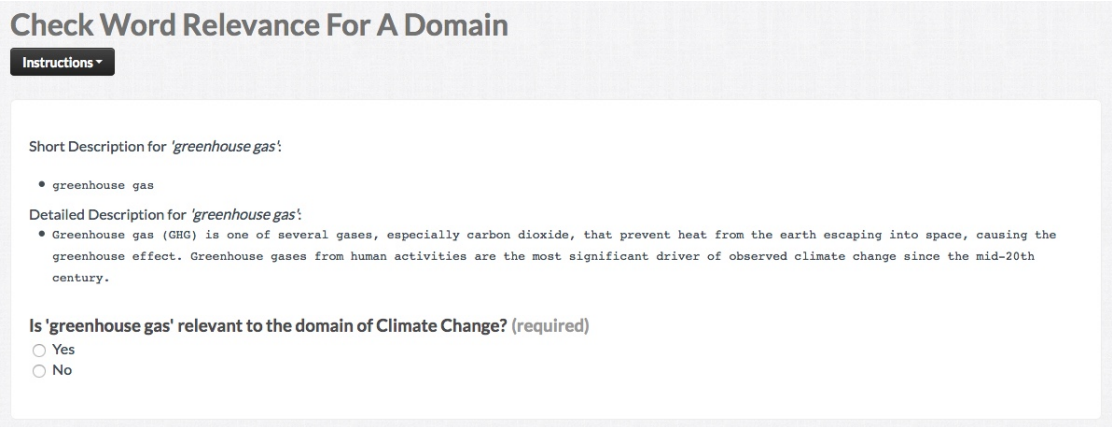
1. the Instruction part
2. the Context part
3. the Question part

The *Instructions* are very generic and therefore independent of the chosen Context enrichment method. It contains a short description of the task goals and some examples of already answered questions. We did not include the details of ontology validation because first, it would confuse contributors and second, it is not relevant for answering the question. Also, we advised them to browse the Web or contact Wikipedia in case they do not know the answer or are unsure. Furthermore, it encourages contributors to give answers to their best knowledge and improves the overall quality of the collected results.

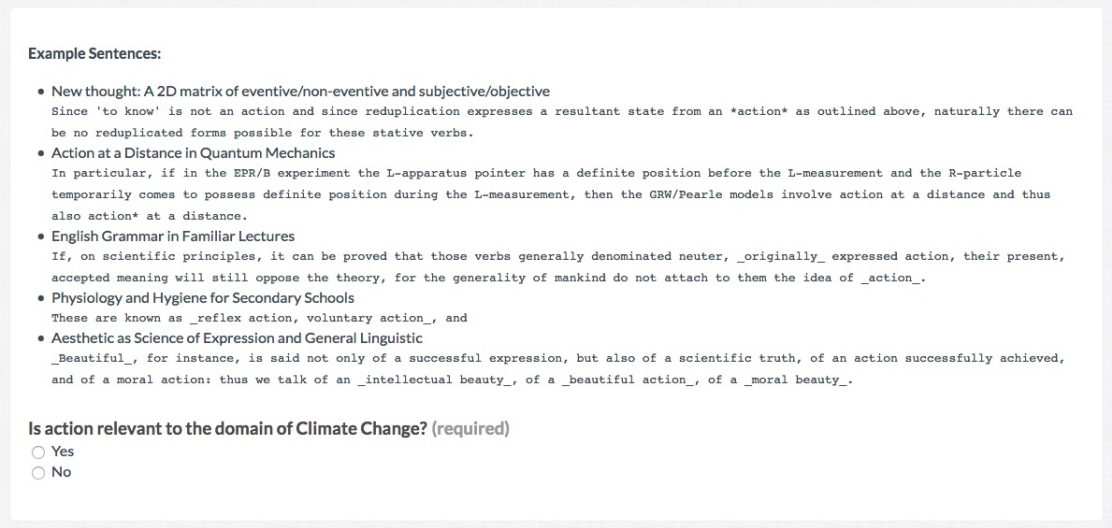
[3] https://developer.wordnik.com/ accessed 2018/06/15



(a) Ontology based Method



(b) Metadata based Method



(c) Dictionary based Method

Figure 4.3: Crowdsourcing task interfaces for performing ontology validation using different methods of Context enrichment

The *Context part* is dynamically adjusted based on the chosen approach. In Figure 4.3a, the description is generated by the *Ontology based Method*, in which each statement corresponds to a relation. In this example, the concept *fusion* is related by subsumption to 3 other concepts: $fusion\ subclassOf\ \{heat, action\}$ and $state\ subclassOf\ fusion$. As of now, only subsumption relations are taken into account for the generation of concept descriptions. Figure 4.3b displays the generated description for the concept *greenhouse gas*. It contains the short description as well as the detailed description. Figure 4.3c shows some example sentences for the concept *action* which were obtained from WordNik. Thereby each sentence is prepended by a headline written in bold letters. Currently, the plugin only supports WordNik as sentence provider.

The *Question part* contains the actual question. To prevent spamming, we added a minimum time of 10 seconds to answer the question. Due to the suggestive nature of the questions, contributors can not skip certain questions in case of uncertainty. This ensures completeness of the result set.

CHAPTER 5

Results

Based on the metrics discussed in Section 4.1, this chapter highlights the results of our evaluation. It is divided along the datasets we used for evaluation, that is, Section 5.1 covers the *climate change* domain, Section 5.2 is dedicated to *finance* and *tennis* is handled in Section 5.3. Finally, an overall *comparison* of all evaluation domains is presented in Section 5.4.

5.1 Climate Change Ontology

In this section, results from the crowd-sourced ontology validation in the field of climate change are presented. A detailed discussion of the ontology used as a baseline for all calculations was done previously in Section 4.2.

The results of the benchmark are presented in Table 5.1. For comparison, we also performed ontology validation without any of the discussed Context enrichment methods *(None)*. Given the relatively small number of concepts, all Context enrichment methods performed better than having no Context at all. Surprisingly, in terms of Recall the contrary holds. Indeed, crowd workers tend to negatively answer questions in case of uncertainty or when no additional information other than the concept name is present.

We also measured the agreement ratio (Inter-rater Agreement) in this dataset. Figure 5.1 shows the distribution of the agreement ratio among all validated concepts. We required 5 judgements for every concept, yielding 5/0, 4/1 or 3/2 levels of agreement, which is equivalent to *full agreement* (1.0), *partial agreement* (0.8) and *little agreement* (0.6) respectively.

The highest agreement exhibits the *Dictionary based Approach*, followed by *None*. This is somewhat interesting as these are the methods with the lowest performance with regard to F-Measure. In fact, the agreement ratio just describes to what extent the responses coincide. From the observations in this dataset, it is hard, if not impossible, to draw

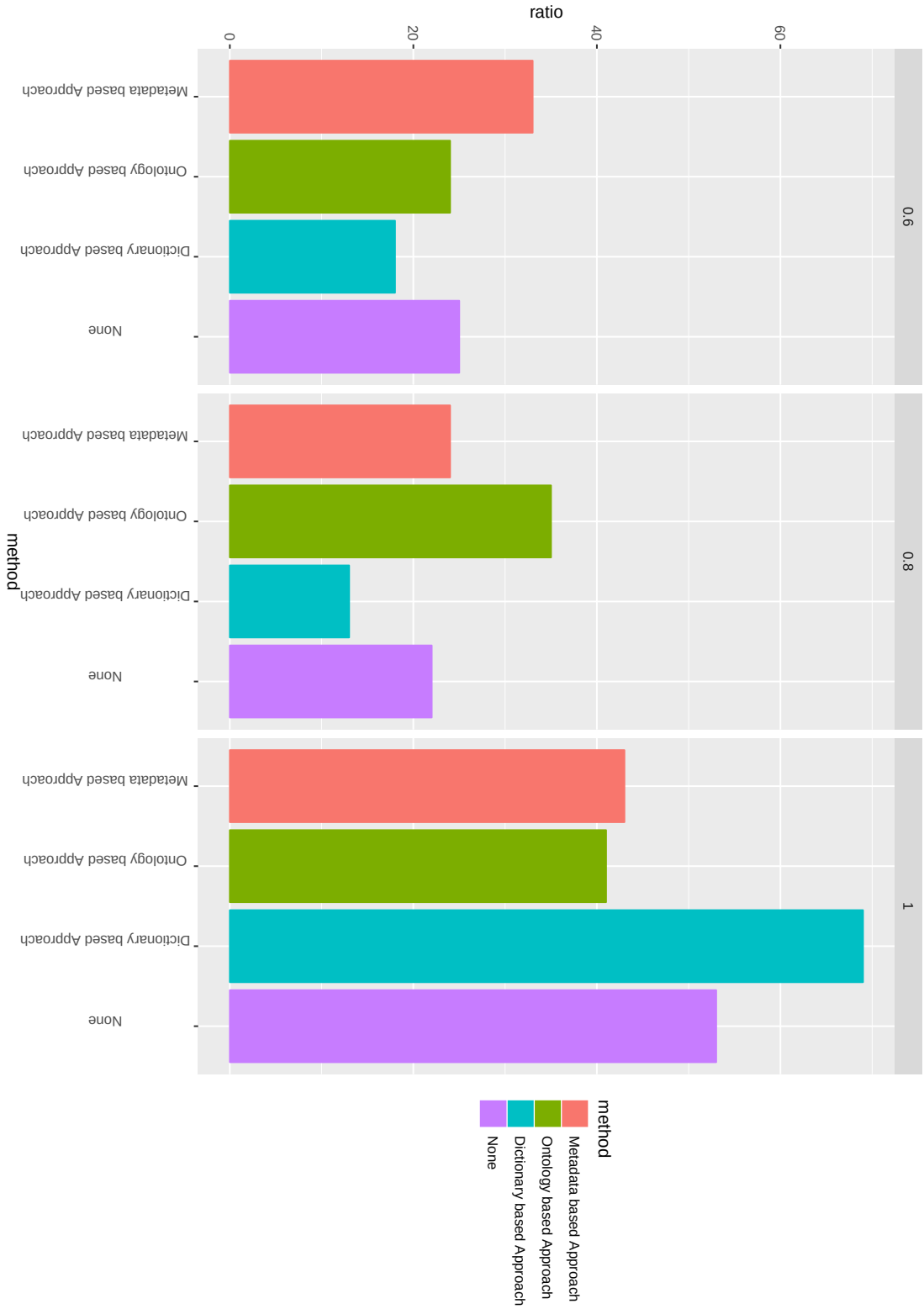


Figure 5.1: Histogram plots of the Inter-rater Agreement

Method	Precision	Recall	F-Measure
Ontology based Approach	0.758	0.805	0.781
Metadata based Approach	0.732	0.831	0.778
Dictionary based Approach	0.724	0.821	0.769
None	0.549	0.837	0.663

Table 5.1: Aggregated results on the Climate Change Ontology (ranked by F-Measure)

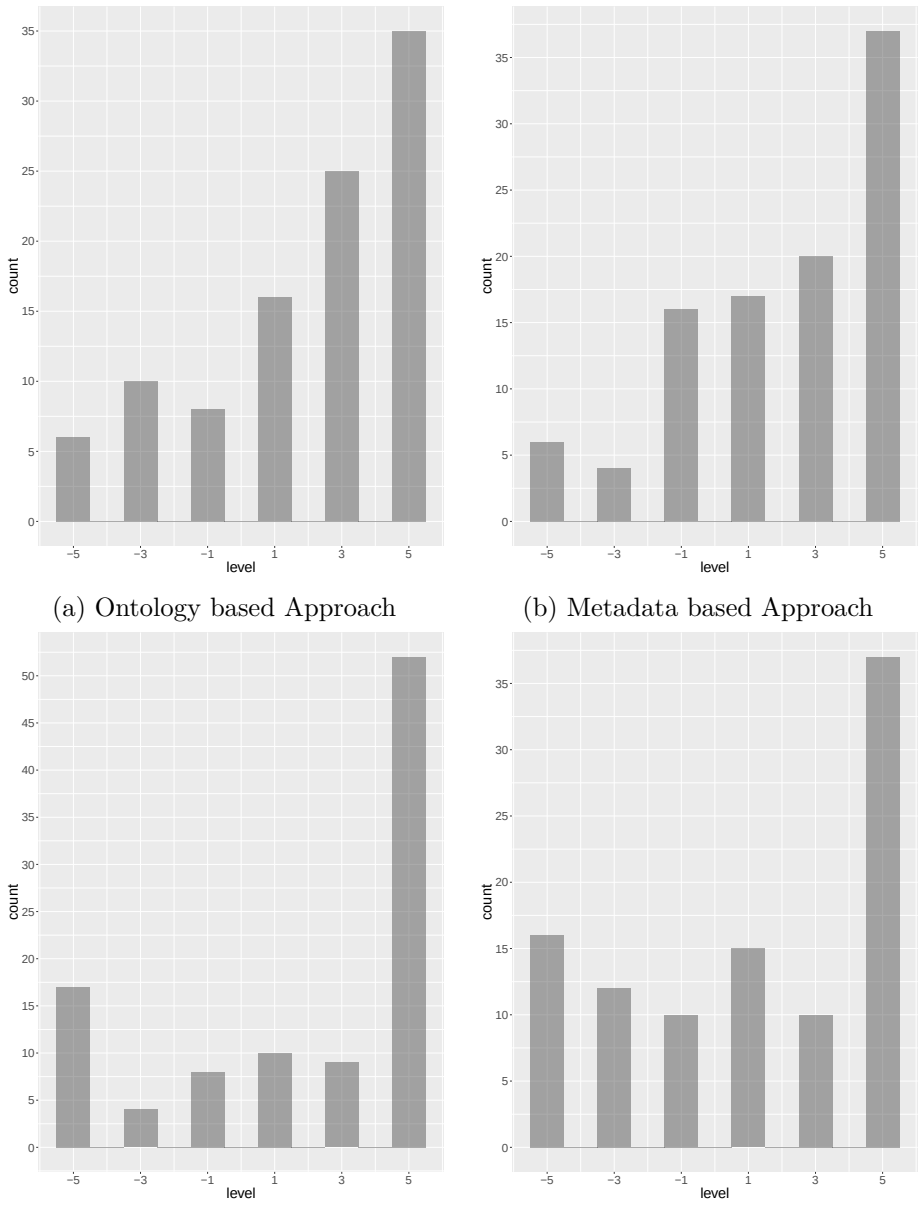
Method	mean	median	1^{st} quartile	3^{rd} quartile
Metadata based Approach	2.04	3.00	-1.00	5.00
Ontology based Approach	1.98	3.00	1.00	5.00
Dictionary based Approach	1.92	5.00	-1.00	5.00
None	1.04	1.00	-3.00	5.00

Table 5.2: Summary statistics concerning agreement level on the Climate Change Ontology (ranked by mean value)

conclusions solely based on agreement. In fact, when looking closely at the judgements with the highest agreement ratio among incorrect answers, 16 of 17 judgements for the Dictionary based Approach and 3 of 6 judgements for the Ontology based Approach had Context added. Apparently, crowd workers agreed here on incorrect values even though concept descriptions were available.

To manifest our observations from above, bar plots (illustrated in Figure 5.2) were created. It combines the agreement ratio and the amount of correct/incorrect judgements. Whereas a negative score indicates that more contributors agreed on incorrect answers or declined relevant concepts, a positive score shows that the majority of crowd worker's responses were correct. Indeed, when comparing the performance on level -5 the *Dictionary based Approach* is on the same level as if Context was omitted. On the other hand, it shows the highest score of correct answers on level 5.

Given the plots on the distribution of the correct/incorrect judgements from above, Table 5.2 shows the summary statistics of agreement levels for each Context enrichment method. It confirms our observations made so far.



(a) Ontology based Approach

(b) Metadata based Approach

(c) Dictionary based Approach

(d) None

Figure 5.2: Histogram plots of the correct/incorrect judgements. {*count*=number of judgements, *level*=combined number of correct (positive scale) and incorrect (negative scale) judgements per concept}

Method	Precision	Recall	F-Measure
Metadata based Approach	0.797	0.985	0.881
Dictionary based Approach	0.794	0.944	0.862
Ontology based Approach	0.756	0.949	0.842
None	0.734	0.963	0.833

Table 5.3: Aggregated results on the Finance Ontology (ranked by F-Measure)

5.2 Finance Ontology

In this section, results from the crowd-sourced ontology validation in the domain of finance are presented. A detailed discussion of the ontology used as a baseline for all calculations was done previously in Section 4.2.

As with the other datasets the *Metadata based Approach* performed quite well. In fact, it outperformed all other approaches, both in terms of Precision and Recall, yielding the highest value of F-Measure. A detailed comparison of all methods for this dataset is given in Table 5.3. On the other end of the table is ontology validation without any Context enrichment (*None*). This is in line with our initial hypothesis that motivated the use of concept descriptions. We also noticed the relatively high number of Recall for all approaches. The same observation was made for the other datasets as well. Indeed, crowd workers tend to decline concepts in case of uncertainty or lack of additional information.

Another metric we used to measure the performance of ontology validation was Inter-rater agreement. Figure 5.3 depicts the distribution of the agreement ratio among all validated concepts. For comparability, all methods were merged into one chart and grouped by the level of agreement. Again, the *Metadata based Approach* performed best followed by the *Ontology based Approach* as indicated by the red bar. It shows both, a high level of full agreement (1) and low level of little agreement (0.6).

To get a different view of the overall worker performance, Figure 5.4 shows bar plots of the performance levels. Each level combines the agreement ratio and the amount of correct/incorrect judgements, yielding a higher score when most contributors agreed on correct answers and a lower score when they disagreed or answered incorrectly. A common phenomena of all approaches was the high level of correct answers with high agreement across contributors. Indeed, this holds for the other datasets too. However, after analysing the concepts that were accepted and those that were declined, this is rather related to the generic nature of the used datasets. For example, whereas accepting the concept *budget* for the finance domain is relatively easy even with no additional information, judging the concept *world* is much more challenging. Therefore, judging generic concepts

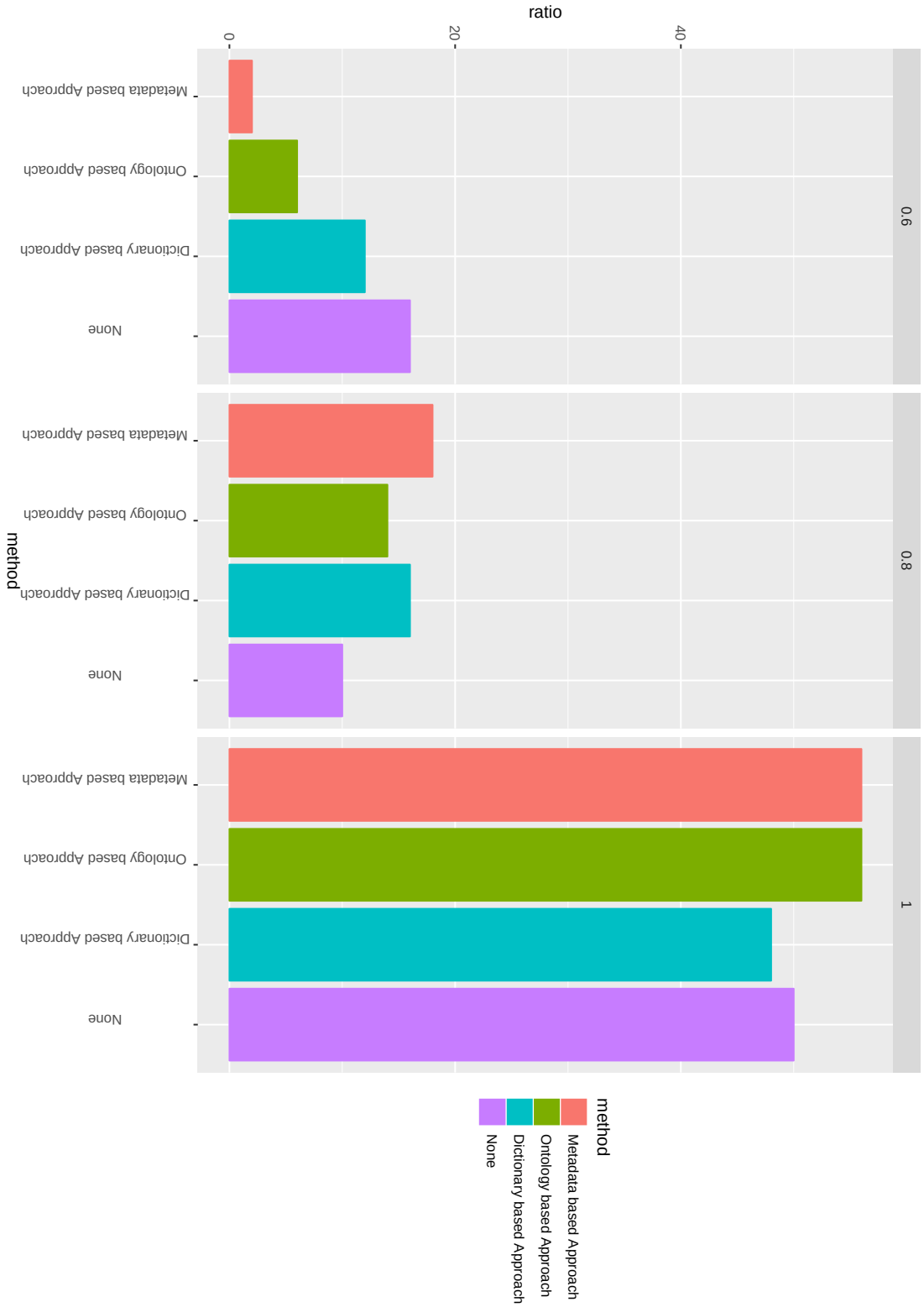
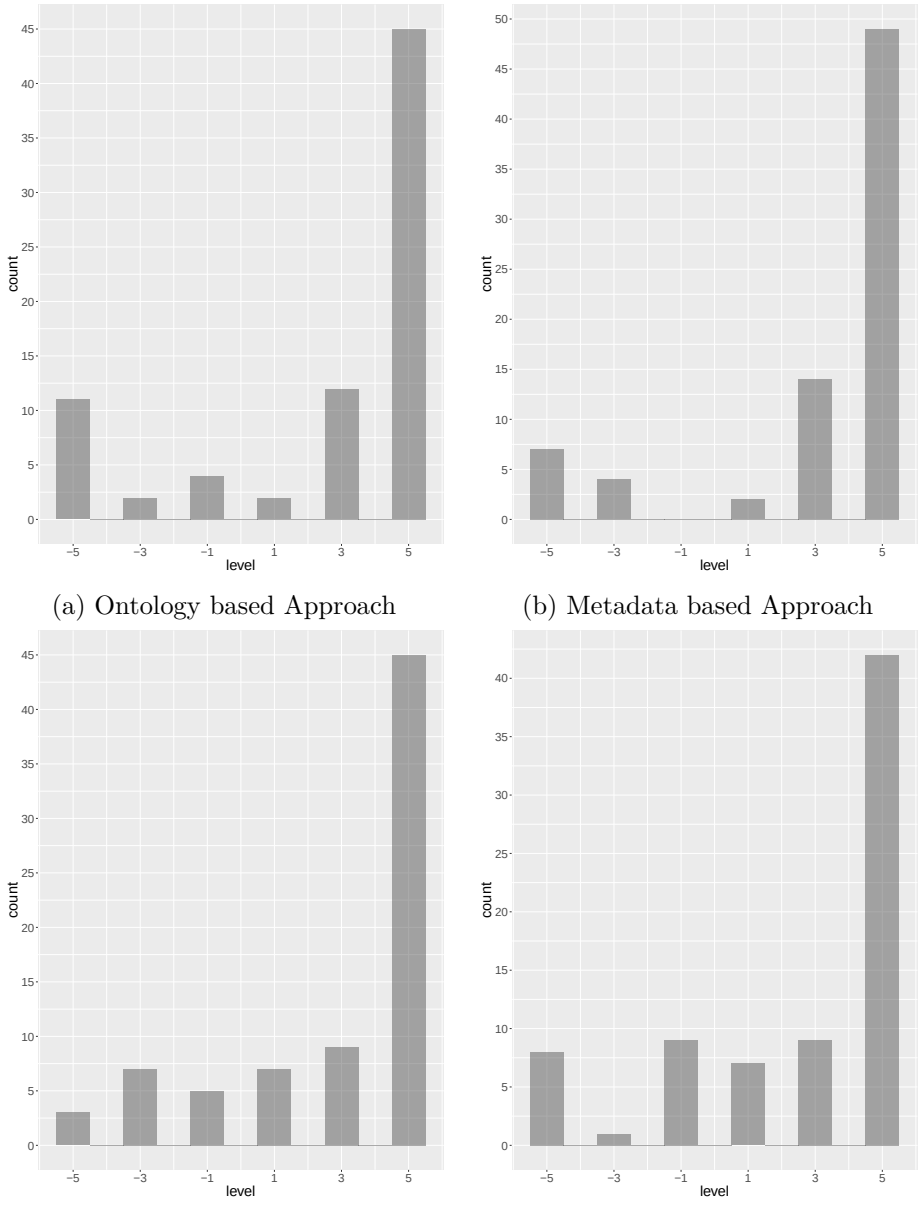


Figure 5.3: Histogram plots of the Inter-rater Agreement



(a) Ontology based Approach

(b) Metadata based Approach

(c) Dictionary based Approach

(d) None

Figure 5.4: Histogram plots of the correct/incorrect judgements. {*count*=number of judgements, *level*=combined number of correct (positive scale) and incorrect (negative scale) judgements per concept}

Method	mean	median	1^{st} quartile	3^{rd} quartile
Metadata based Approach	3.18	5.00	3.00	5.00
Dictionary based Approach	2.87	5.00	1.00	5.00
Ontology based Approach	2.61	5.00	2.50	5.00
None	2.53	5.00	1.00	5.00

Table 5.4: Summary statistics concerning agreement level on the Finance Ontology (ranked by mean value)

should be done better by domain experts who share a common understanding of the used vocabulary.

The summary statistics in Table 5.4 confirm our observations made so far. It shows the statistics of each method ranked by mean value. Judging the worker performance by the level of agreement and the ratio of correct/incorrect judgements, the order is no different than ranked by F-Measure.

5.3 Tennis Ontology

In this section, results from the crowd-sourced ontology validation in the domain of tennis are presented. A detailed discussion of the ontology used as a baseline for all calculations was done previously in Section 4.2.

The worker performance for this dataset was the highest among all evaluated ontologies. In fact, for the highest ranked method *(Metadata based Approach)* at least 2 out of 5 contributors correctly identified relevant concepts or declined unrelated ones. This corresponds to a Precision of 0.896 and Recall of 0.976, or combined F-Measure of 0.934. A detailed comparison of all methods for this dataset is given in Table 5.5. Interestingly, the *Dictionary based Approach* performed worse than omitting concept descriptions at all. This, we noticed by the high discrepancy of Precision. More precisely, 0.648 with descriptions constructed from WordNik consultation *(Dictionary based Approach)* compared to 0.783 without any context.

Figure 5.5 depicts the distribution of the agreement ratio among all validated concepts. To the contrary, by the *Ontology based Approach*, crowd workers reached the most consensus, albeit being wrong in some cases. They declined 2 relevant concepts whereas for the *Metadata based Approach* there were at least 2 of them who were correct for any concept.

The observations from above were also reflected by the bar plots in Figure 5.6. There were no judgments on level -5 and -3 for the *Metadata based Approach*. The sum-

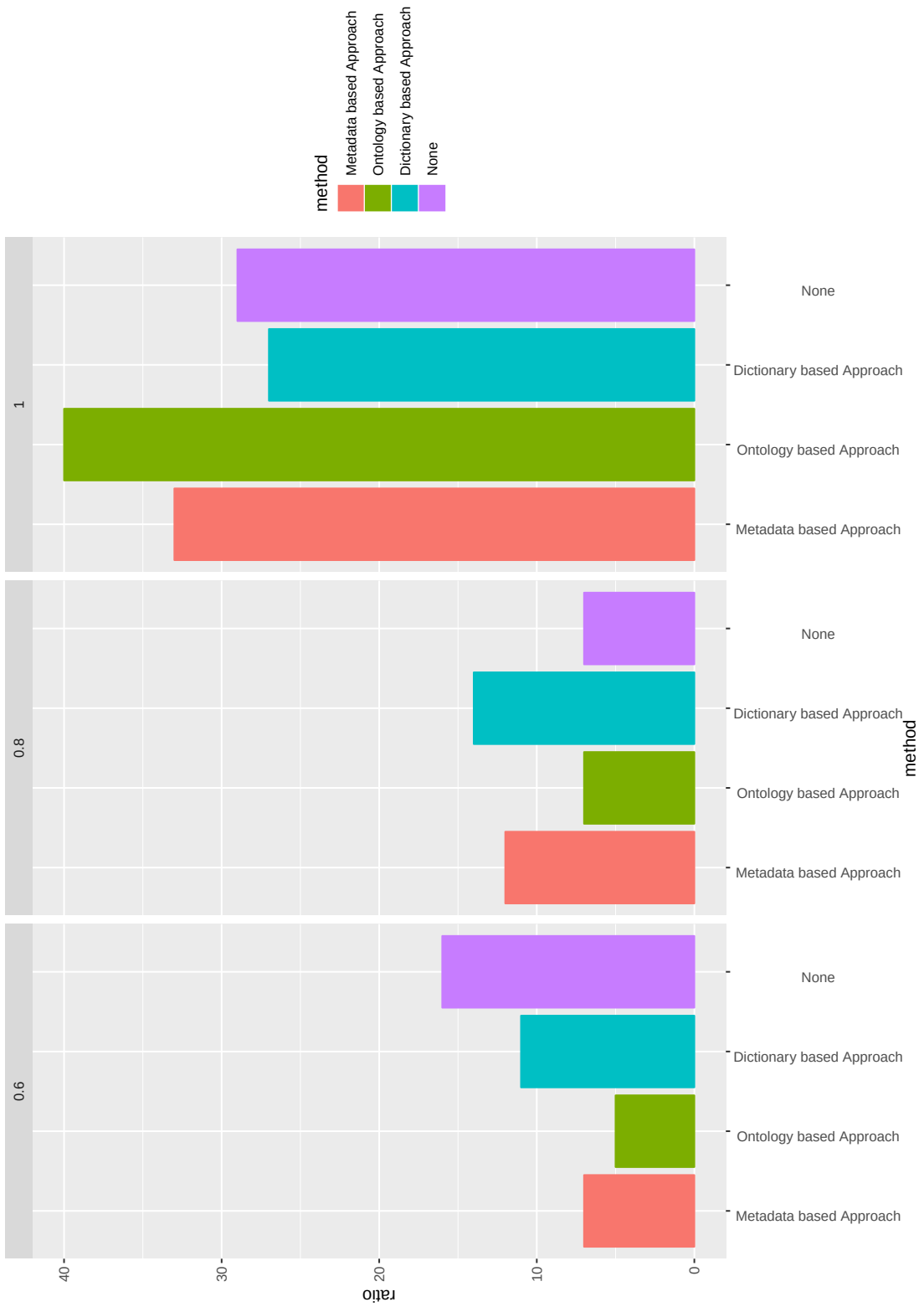


Figure 5.5: Histogram plots of the Inter-rater Agreement

Method	Precision	Recall	F-Measure
Metadata based Approach	0.896	0.976	0.934
Ontology based Approach	0.874	0.939	0.905
None	0.783	0.933	0.851
Dictionary based Approach	0.648	0.980	0.780

Table 5.5: Aggregated results on the Tennis Ontology (ranked by F-Measure)

Method	mean	median	1^{st} quartile	3^{rd} quartile
Metadata based Approach	3.89	5.00	3.00	5.00
Ontology based Approach	3.39	5.00	3.00	5.00
None	2.58	5.00	1.00	5.00
Dictionary based Approach	1.77	3.00	-1.00	5.00

Table 5.6: Summary statistics concerning agreement level on the Finance Ontology (ranked by mean value)

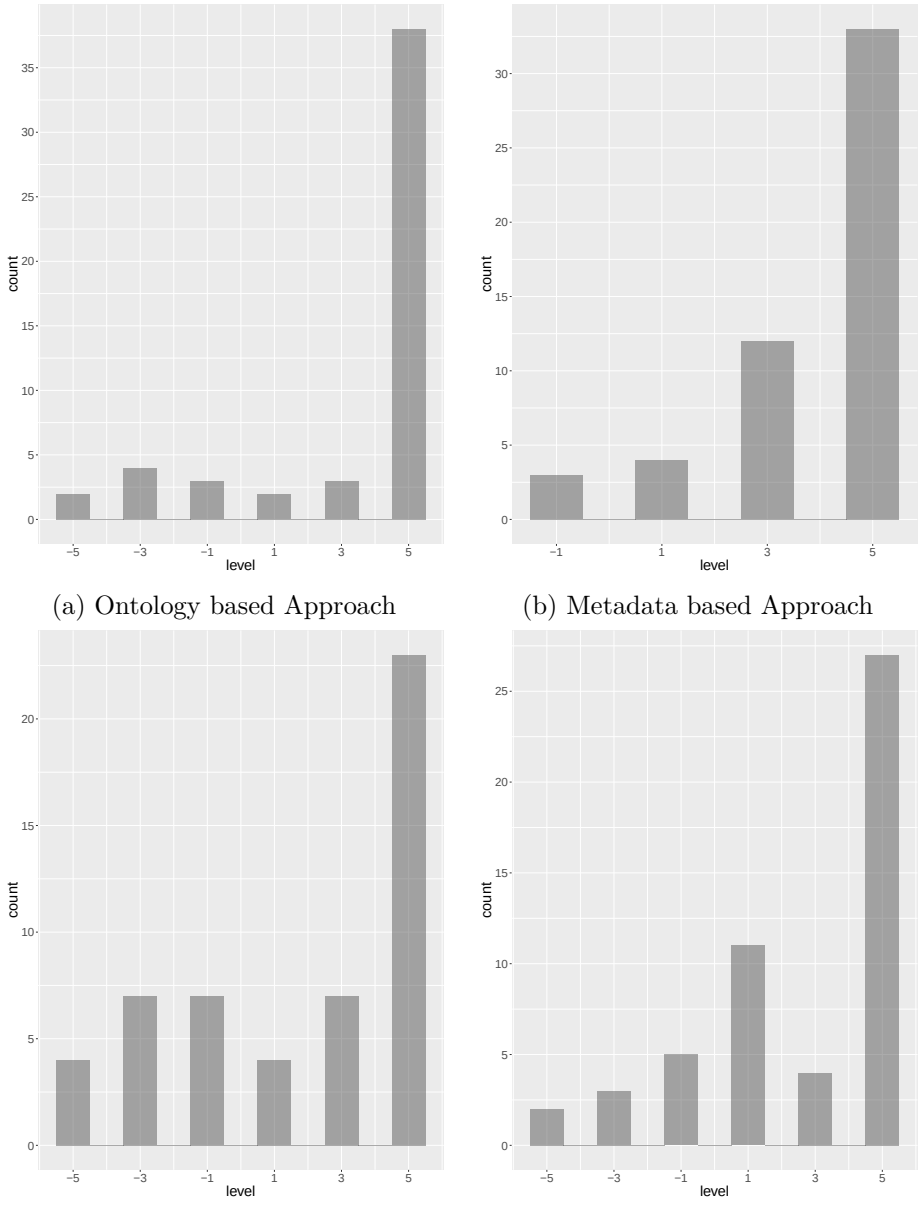
mary statistics in Table 5.6 confirm that finding. Based on the agreement level and correct/incorrect judgement ratio, the rankings of each method were preserved.

5.4 Evaluation Comparison

In the final evaluation step we take a broader look on the overall performance. For that, we combined the results from each dataset. This has the advantage of reducing the sensibility to a particular dataset which is required to keep bias at minimum.

Based on our initial hypothesis which motivates Context enrichment, we have formulated a couple of questions that were answered in the next paragraphs:

Which Context enrichment method performed best in general? Our observations confirmed our initial hypothesis which suggests extending basic crowd-based ontology validation with Context. From the combined results of all datasets as shown in Table 5.7 it is evident that the Metadata based Approach worked best. In fact, it had not only the highest value of F-Measure but also the highest Precision and Recall. Indeed, this was rather expected due to the fact that Context was manually added. Obviously, no one has a better domain knowledge than the creators or maintainers of the ontology. On the bottom end of the table is the approach containing no descriptions (None).



(a) Ontology based Approach

(b) Metadata based Approach

(c) Dictionary based Approach

(d) None

Figure 5.6: Histogram plots of the correct/incorrect judgements. {*count*=number of judgements, *level*=combined number of correct (positive scale) and incorrect (negative scale) judgements per concept}

Method	Precision	Recall	F-Measure
Metadata based Approach	0.797	0.921	0.854
Ontology based Approach	0.787	0.887	0.834
Dictionary based Approach	0.729	0.899	0.805
None	0.674	0.910	0.775

Table 5.7: Aggregated results of all datasets (ranked by F-Measure)

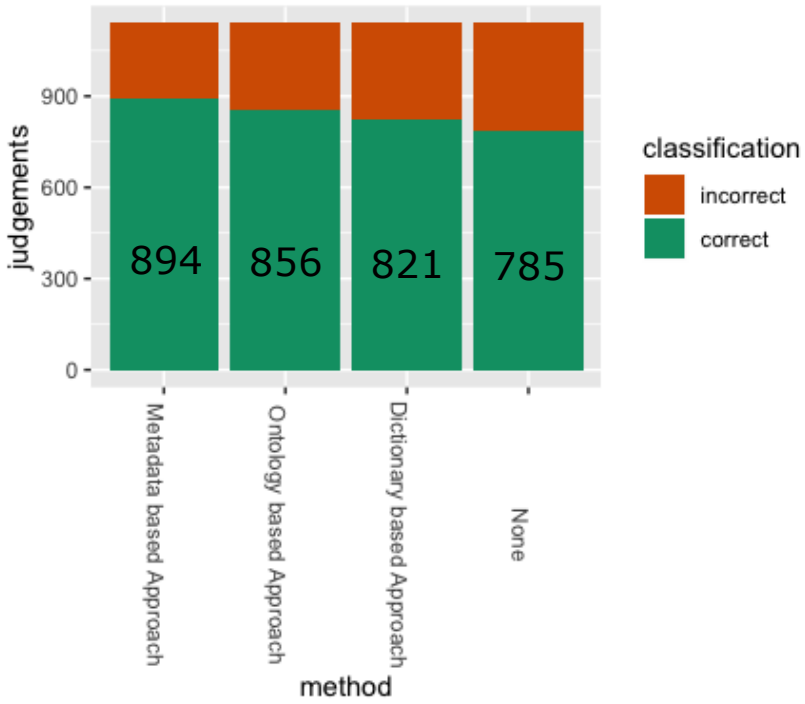


Figure 5.7: Combined accuracy of Crowdsourcing methods

Did the crowd perform better with Context? Figure 5.7 depicts the combined accuracy of all methods which is calculated as the ratio between correct and incorrect judgements. For comparability, the exact number of judgements is written in labels. The performance of the top ranked method (Metadata based Approach) is quite impressive. Concepts were judged correctly for nearly eighty percent (78.4%), being an improvement of over 14% compared to omitting concept descriptions. Even the last ranked enrichment method (Dictionary based Approach), performs 4.6% better.

Concept	*Methods*				*Accuracy*
	Meta	Onto	Dict	None	Total
sceptic	0/5	0/5	0/5	0/5	0/20
greenhouse	0/5	1/5	0/5	0/5	1/20
pipeline	0/5	0/5	1/5	0/5	1/20
consensus	2/5	0/5	0/5	0/5	2/20
denier	2/5	0/5	0/5	0/5	2/20
production	1/5	1/5	0/5	0/5	2/20

Table 5.8: Concepts where most crowd workers had problems (Meta=Metadata based Approach, Onto=Ontology based Approach, Dict=Dictionary based Approach, None=No Context)

For which concepts were the crowd wrong? Even though the results underlined the usefulness of Context for crowd-sourced ontology validation, we were also interested under which circumstances they failed. In particular, we evaluated for which concepts the crowd had the most problems with. The goal was to identify common patterns to guide future improvements.

Table 5.8 lists 6 concepts that had the highest number of erroneous judgements. The concept *sceptic* is located in the top-most row of the table. None of the judgements were correct for that concept. Consequently, for the climate change ontology, the concept was rejected even with Context being present. An explanation for that could be the fact the Context was too generic or inappropriate. Clearly, the concept could be associated with climate change, meaning someone who doubts global warming. Unfortunately, we were not able to adapt Context because we had no access to the sources where the ontologies were learned from. In most cases though, we expect ontology engineers to have enough knowledge to provide more accurate descriptions that were useful to the crowd.

For the remaining concepts the situation is similar. We identified the following pattern: Context was either missing, especially for very specific concepts, or too generic. The only solution to both is strengthen collaboration between ontology engineers and domain experts.

What other issues were found? A general phenomenon of all approaches was the relatively high value of Precision indicating that the crowd tends to rather reject concepts in case of uncertainty or lack of knowledge. However, in ontology engineering Recall is often more important than Precision. Domain experts and ontology maintainers prefer deleting a few concepts rather than missing some important ones.

Another observation was that not all enrichment methods worked in all scenarios. For some concepts (17%) no Context was found when fetching from the online dictionary WordNik. Most of these concepts had names that were composed of multiple words as it was the case for *interest rate*. Compound words were treated as if words were searched separately. The final result then being merged from each subquery. Unfortunately, the API does not support changing this behaviour to treat compound words as a whole.

The number of concepts with missing Context was quite low when descriptions were generated from the ontology structure. For 30% of the concepts Context was missing. Obviously, the algorithm failed when the concept was not part of a subsumption relation. In such situations, other approaches that do not rely on the ontology structure should be used instead.

Strangely, Context was not always helpful, in certain situations it can also be distracting, mainly if it provides useless or even misleading information. Regrettably, some content fetched from WordNik was rather lengthy or diffuse. That was certainly due to the fact that content originated from blogs or tweets that were not well suited for word definitions. Future versions of the framework could try to use a different content provider which provides more concise information.

CHAPTER 6

Conclusion & Future Work

In this chapter, the main topics of this thesis are summarised (Section 6.1) and the research questions are revisited (Section 6.2). Finally, an outlook for future research topics is provided (Section 6.3).

6.1 Summary

In this thesis we investigated whether contextual information in Crowdsourcing tasks helped to achieve better results for performing ontology validation. Crowdsourcing is a technique of distributing small tasks to a typically large group of human workers. It offers a cost effective method of solving tasks which are traditionally hard for machines but easily solvable by humans. Our contributions are based on previous work covering the uComp Protege plugin.

Unfortunately, crowd workers often do not had enough knowledge to complete Crowdsourcing tasks. They need additional contextual information which improves their understanding. Before investigating our approaches which generate contextual information, we had to give a common definition of »Context«: Context refers to any sort of additional information that is supplied with a Crowdsourcing task to improve its understanding in such a way that it positively affects the crowds performance and the result quality. Furthermore, we do not set a limitation on the type or format of Context that is provided. Even tough there exists some approaches that use Context in Crowdsourcing tasks, they all use a different notion of Context. Furthermore, none of these generate contextual information.

We presented three novel methods that enrich Crowdsourcing tasks with contextual information to validate the relevance of concepts for a particular domain of interest. First, the Ontology based Approach processes hierarchical relations. Second, the Metadata based Approach generates descriptions based on annotations that are encoded within

the ontology. Third, the idea of the Dictionary based Approach is to build up contextual information from example sentences by consulting the online dictionary WordNik.

The evaluation was performed on three ontologies covering the domains of climate change, tennis and finance. The Metadata based Approach outperformed all other methods in terms of Precision and Recall, leaving little room for future improvements. The other two approaches had some difficulties in certain situations, for example the Dictionary based Approach sometimes added inappropriate explanations, especially for concepts with multiple meanings associated. Likewise, the Ontology based Approach is limited to highly connected ontologies containing many subsumption relations.

6.2 Conclusion

In this section each research question is revisited and answered by taking into consideration the results from our experimental evaluation (see Chapter 5) and drawing final conclusions that point future research in novel directions.

The main research question examined in this thesis was:

RQ-I *Does the crowd perform better on Context enriched Crowdsourcing tasks?*

Answering this question might seem difficult at first because measuring the crowd's performance depends on the metrics of measurement as well as the concrete evaluation settings. However, all proposed methods performed better with regard to F-Measure than in experiments omitting Context. Indeed all our experiments showed that in each dataset the number of correct classifications is considerably higher.

Clearly, the most important performance metric is F-Measure because it combines the benefits and minimises the drawbacks of Precision and Recall at the same time. We observed that crowd workers tend to decline relevant concepts if they were unsure and Context was either missing or not relevant. Considering our approach is embedded in an ontology learning framework, this seems unproblematic because domain experts and ontology engineers rather prefer deleting a few concepts instead of missing some important ones [Sab06].

Our experiments that were performed on three datasets including tennis, climate change and finance showed that our approach is feasible and improves the results of the ontology validation process. It was already mentioned [MMJ+15, MTH+16, WSH16] that crowd-based ontology validation is a good alternative to manual validation, especially in situations where experts are unavailable, budget is limited or the ontology is just too large.

RQ-II *What methods can be applied that generate Context?*

We measured the performance of the crowd using three methods that generate concept descriptions requiring either manual intervention or being fully automated. All of our proposed approaches are discussed in detail in Chapter 3.

The Ontology based Approach (see Section 3.2) processes hierarchical relations that were encoded within the ontology. The biggest advantage of this approach is that it does not have any external dependencies and works fully automatically. This algorithmic approach is recommended for ontologies containing a large number of concepts that are connected by subsumption relations. A potential pitfall of this method is that the full potential of ACE could not be leveraged because we identified certain obstacles that hinder the integration of OWL Verbalizer, a tool that converts an ontology into a set of ACE sentences. Consequently, our algorithm generates the text by simple string replacement, not taking the word category (e.g. singular or plural) into account.

The second approach (Metadata based Approach — see Section 3.3) is based on metadata that resides within the ontology. In contrast to the other methods this approach requires some manual work. As a precondition the metadata needs to be added by experts in a standardised format which served then as Context in Crowdsourcing tasks. Because the additional costs of manual preprocessing might not outweigh the benefits of high quality concept descriptions, it makes sense to preferably use this approach in very specialised areas such as in biomedical domains where ontologies are typically well documented and already contain explanations.

The idea of the Dictionary based Approach (see Section 3.4) is that starting from a concept name, descriptions are built from consulting an online dictionary. WordNik was chosen as the provider of concept descriptions. These are formed from example sentences that contain the requested concept name. This approach has its strengths when concepts are relatively generic. We also noticed that the lookup failed when concept names contained special characters or had multiple meanings associated. To conclude, this approach is rather simple and easy to implement, however, it may have the potential to generate wrong results, especially for ambiguous concepts.

RQ-III *To what extent is it possible to transfer the investigated methods to different datasets?*

Unfortunately, none of our proposed methods can be applied in all contexts. Each method has its own prerequisites:

The Ontology based Approach highly depends on the ontology structure because it processes subsumption relations. The algorithm fails for flat hierarchies that contain little or no subsumption relation. However, this restriction seems reasonable because limiting our viewpoint to subsumption relations was caused by major obstacles that prevented the integration of OWL Verbalizer, a tool which also takes other relation types into account.

At the core of the Metadata based Approach are annotations that reside within the ontology. Unfortunately, none of our evaluated ontologies contained these metadata by nature which required us to add them by hand. We think though, that this requirement is rather feasible because our experiments showed that this approach generates concept descriptions of high quality. In fact, in very specialised areas such as in biomedical domains ontologies already contain such explanations.

For the Dictionary based Approach the situation is different because the design of the algorithm does not impose any restrictions on the internals of the validated dataset. The outcome rather depends on the responses from conducting the online dictionary WordNik. We observed two peculiarities that may be considered: i) the lookup failed when concept names contained special characters (e.g. quotes), and ii) the response contained irrelevant example sentences when the meaning of the requested concept is ambiguous.

RQ-IV *Which of the proposed methods work best? What are potential shortcomings and why?*

Based on the results presented in Chapter 5, the Metadata based Approach outperformed all other methods even though it was only ranked second in terms of F-Measure for the Climate Change dataset. A clearer picture can be drawn when looking at the combined statistics showing the level of agreement and the judgment's accuracy. In all these rankings this approach was ranked as the best performing method. This outcome was rather expected because, compared to the other approaches, concept descriptions were of highest quality while keeping the number of missing explanations at zero. Indeed, allowing expert participation produces qualitative results but is also very costly.

In contrast to our expectations the Dictionary based Approach had the most problems in finding relevant descriptions. It performed even worse than omitting descriptions in the tennis dataset. This has several reasons: From analysing the traffic of WordNik consultation, we know that some of the responses were irrelevant or were even missing for certain concepts. This is certainly true for concepts having special characters (e.g. quotes) in their names or concepts which have multiple meanings associated.

One restriction that should be considered especially when validating large ontologies containing several hundreds or even thousands of concepts is that WordNik limits the number API calls/requests in the basic setup. However, they offer paid plans[1] to users who need more calls or more data.

A common phenomenon over all datasets was the relatively high Recall, even when omitting concept descriptions at all. This means that the crowd predominately declined relevant concepts, however, as mentioned earlier, this is relatively unproblematic because domain experts and ontology engineers rather prefer deleting a few concepts instead of missing some important ones.

Surprisingly, the Ontology based Approach worked pretty well even though its performance could not reach the top ranked method. Unquestionably, the quality of the descriptions directly correlates to the number of subsumption relations. This works especially for learned ontologies because learning frameworks naturally create ontologies with deep hierarchies. But since all concepts were connected by subsumption relations, explanations were missing for those isolated concepts.

[1] https://developer.wordnik.com/pricing accessed 2019/02/03

6.3 Future Work

While this work tried to advance research, covering the intersection between Semantic Web technologies and Crowdsourcing, there are still several open questions that could not be addressed, because these new questions that are discussed in the remainder of this chapter arose during the course of this work.

We identified the following topics that can be addressed in future work:

Extending the Dictionary based Approach by using other content providers Unfortunately, the Dictionary based Approach had problems in situations when multiple meanings were associated with the same concept or names contained special characters. The reason for that was that WordNik could not handle these cases properly. One could try to integrate other content providers to overcome these limitations.

Extending the Ontology based Approach to use OWL-Verbalizer At the time of writing this thesis we were unable to integrate OWL-Verbalizer, an open source tool aimed at producing texts from generic OWL ontologies, because it was originally written in SWI-Prolog but our platform runs on the Java Virtual Machine (JVM). Even though the tool authors are aware of this problem[2], the lack of compatibility with other programming languages still remains. Future research could possibly take a closer look at the Java Interface to Prolog[3] which offers a bidirectional interface between Java and Prolog that can be used to embed Prolog in Java as well as for embedding Java in Prolog.

Making the proposed methods more generic Researchers could also investigate more in the direction of making our approaches more generic. This holds in particular for the Ontology based Approach because it only takes subsumption relations into account. A huge improvement would be to also include object properties, however, major changes would be required because the algorithm needs to consider besides concepts also individuals.

Covering other tasks of ontology validation To achieve the goal of a general purpose solution for ontology validation, our methods should be extended to cover other tasks as well. The uComp Protege Plugin could serve as a starting point here. Besides verification of domain relevance, other tasks include verification of relation correctness, specification of relation type and verification of domain and range. The challenging part here is to adapt the workflow of these tasks in such a way that the Crowdsourcing tasks include contextual information.

Integrating our contributions into an ontology learning solution The last open topic refers to the integration of our contributions into an ontology learning solution. In

[2]https://github.com/Kaljurand/owl-verbalizer/issues/13 accessed 2019/02/08
[3]http://www.swi-prolog.org/packages/jpl/ accessed 2018/05/11

that vein, future studies need to evaluate which of the approaches of ontology validation work best (e.g. automated or manual ones) and possibly provide a hybrid approach which combines several of these approaches.

APPENDIX A

Dublin Core Metadata Terms

Name	Description
dc:contributor	Element used to describe a person, organisation or service who is responsible for making contributions.
dc:overage	Term used to describe a temporal topic (e.g. period, date, or date range), spatial topic (e.g. location or place identified by its name or coordnates) or a jurisdiction (e.g. an administrative entity).
dc:creator	A person, organisation or service who created this entity.
dc:date	A period or point in time associated with an event in the lifecycle.
dc:description	A definition in natural-language.
dc:format	Defines the file format, physical medium or dimension.
dc:identifier	A unique and unambiguous reference to this entity within a defined context.
dc:language	The language used to describe and define this entity.
dc:publisher	A person, organisation or service who provides access to this entity.
dc:relation	Defines a link to another entity identified by name or formal identifier.
dc:rights	A statement about associated rights with the entity (e.g. intellectual property rights).
dc:source	A related entity from which this entity is derived from.
dc:subject	The topic of this entity represented using keywords, key-phrases or classification codes.
dc:title	The name by which this entity is formally known.
dc:type	Defines the genre of nature. Usually, a well-defined vocabulary such as DCMI Type Vocabulary[1] is recommended here.

Table A.1: The initial set of DC-Metadata terms

APPENDIX B

SKOS Metadata Terms

Name	Description
skos:Concept	Describes an idea, notion or unit of thought, similar to OWL classes. However the specification does draw any relations to *owl:concept*.
skos:ConceptScheme	A Concept Scheme can be viewed as a combination of multiple *skos:Concept* instances with optional references to each other.
skos:altLabel	A *lexical label* (e.g. a text composed of unicode characters) which adds an alternative meaning to an entity.
skos:prefLabel	Used in combination with *skos:altLabel* to define the primary description in case there are multiple human-readable definitions.
skos:notation	A literal string of unicode characters, it identifies the related concept within the given concept scheme.
skos:changeNote	Belongs to the class of *documentation properties* and provides some information about historical changes.
skos:definition	Adds a human-readable definition to the entity.
skos:note	Some arbitrary text which may be provided by ontology engineers.
skos:editorialNote	A note added by creators to inform ontology maintainers.
skos:historyNote	A historical note (e.g. a version string, release date, ...).
skos:related	Indicates that SKOS concepts are somewhat related to each other.

Table B.1: A subset of the SKOS vocabulary

List of Figures

List of Tables

List of Algorithms

Acronyms

ACE Attempto Controlled English. 28, 29, 63

AI Artificial Intelligence. 9

API Application Programming Interface. 35, 37, 60, 64

DC Dublin Core. 33

DCMI Dublin Core Metadata Initiative. 32, 33

DL Description Logic. 29

EER Extended Entity Relationship. 23, 24

EuroVoc Multilingual Thesaurus of the European Union. 43

GWAP Games with a purpose. 1, 8, 17

IE Information Extraction. 12

IR Information Retrieval. 39

JDK Java Development Kit. 36

JSON JavaScript Object Notation. 37

JVM Java Virtual Machine. 65

LED Listening Experience Database. 14

MTurk Amazon Mechanical Turk. 1

NLP Natural Language Processing. 12

OWL Web Ontology Language. 20, 28, 31, 32, 34, 35, 38, 63, 65, 71

P&G Procter & Gamble. 7

R2RML RDB to RDF Mapping Language. 13

RDF Resource Description Framework. 13, 18, 22–25, 33

RDFS Resource Description Framework Schema. 33, 34

SCA Semantic Content Authoring. 14

SKOS Simple Knowledge Organization System. 33

SPARQL Semantic Protocol and RDF Query Language. 18

SQL Structured Query Language. 18

UI User Interface. 14

URI Uniform Resource Identifier. 33

URL Uniform Resource Locator. 37

W3C World Wide Web Consortium. 13, 33

WWW World Wide Web. 10

XML Extended Markup Language. 13

Bibliography

[ABD+12] Sören Auer, Lorenz Bühmann, Christian Dirschl, Orri Erling, Michael Hausenblas, Robert Isele, Jens Lehmann, Michael Martin, Pablo N. Mendes, Bert van Nuffelen, Claus Stadler, Sebastian Tramp, and Hugh Williams. Managing the life-cycle of linked data with the lod2 stack. In Philippe Cudré-Mauroux, Jeff Heflin, Evren Sirin, Tania Tudorache, Jérôme Euzenat, Manfred Hauswirth, Josiane Xavier Parreira, Jim Hendler, Guus Schreiber, Abraham Bernstein, and Eva Blomqvist, editors, *The Semantic Web – ISWC 2012*, pages 1–16, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[ABI+13] Mohammad Allahbakhsh, Boualem Benatallah, Aleksandar Ignjatovic, Hamid Reza Motahari-Nezhad, Elisa Bertino, and Schahram Dustdar. Quality control in crowdsourcing systems: Issues and directions. *IEEE Internet Computing*, 17(2):76–81, 2013.

[ABK+07] Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. Dbpedia: A nucleus for a web of open data. In Karl Aberer, Key-Sun Choi, Natasha Noy, Dean Allemang, Kyung-Il Lee, Lyndon Nixon, Jennifer Golbeck, Peter Mika, Diana Maynard, Riichiro Mizoguchi, Guus Schreiber, and Philippe Cudré-Mauroux, editors, *The Semantic Web*, pages 722–735, Berlin, Heidelberg, 2007. Springer Berlin Heidelberg.

[ADBB14] Alessandro Adamou, Mathieu D'Aquin, Helen Barlow, and Simon Brown. Led: Curated and crowdsourced linked data on music listening experiences. In *Proceedings of the 2014 International Conference on Posters & Demonstrations Track - Volume 1272*, ISWC-PD'14, pages 93–96, 2014.

[ALNN11] Sören Auer, Jens Lehmann, and Axel-Cyrille Ngonga Ngomo. *Introduction to Linked Data and Its Lifecycle on the Web*, pages 1–75. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[ALTY08] Nitin Agarwal, Huan Liu, Lei Tang, and Philip S. Yu. Identifying the influential bloggers in a community. In *Proceedings of the 2008 International*

Conference on Web Search and Data Mining, WSDM '08, pages 207–218, New York, NY, USA, 2008. ACM.

[AMMH07] Daniel J. Abadi, Adam Marcus, Samuel R. Madden, and Kate Hollenbach. Scalable semantic web data management using vertical partitioning. In *Proceedings of the 33rd International Conference on Very Large Data Bases*, VLDB '07, pages 411–422. VLDB Endowment, 2007.

[AZS+18] Maribel Acosta, Amrapali Zaveri, Elena Simperl, Dimitris Kontokostas, Fabian Flöck, and Jens Lehmann. Detecting linked data quality issues via crowdsourcing: A dbpedia study. *Semantic Web*, 9(3):303–335, 2018.

[BDR17] Kalina Bontcheva, Leon Derczynski, and Ian Roberts. *Crowdsourcing Named Entity Recognition and Entity Linking Corpora*, pages 875–892. Springer Netherlands, Dordrecht, 2017.

[BFH+10] Remco R. Bouckaert, Eibe Frank, Mark A. Hall, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. Weka—experiences with a java open-source project. *J. Mach. Learn. Res.*, 11:2533–2541, December 2010.

[BGM05] Janez Brank, Marko Grobelnik, and Dunja Mladenić. A survey of ontology evaluation techniques. 2005.

[BKvH02] Jeen Broekstra, Arjohn Kampman, and Frank van Harmelen. Sesame: A generic architecture for storing and querying rdf and rdf schema. In Ian Horrocks and James Hendler, editors, *The Semantic Web — ISWC 2002*, pages 54–68, Berlin, Heidelberg, 2002. Springer Berlin Heidelberg.

[BN03] Franz Baader and Werner Nutt. The description logic handbook. chapter Basic Description Logics, pages 43–95. Cambridge University Press, New York, NY, USA, 2003.

[Bur79] Leslie S Burnett. Lexicographical problems in the treatment of some linguistic terms in a supplement to the oed. *ITL-International Journal of Applied Linguistics*, 45(1):19–24, 1979.

[BZG+12] Geoffrey Barbier, Reza Zafarani, Huiji Gao, Gabriel Fung, and Huan Liu. Maximizing benefits from crowdsourced data. *Computational and Mathematical Organization Theory*, 18(3):257–279, Sep 2012.

[Chi09] Steve Chilton. Crowdsourcing is radically changing the geodata landscape: case study of openstreetmap. In *Proceedings of the UK 24th international cartography conference*, 2009.

[Con12] The World Wide Web Consortium. https://www.w3.org/2001/sw/sweo/public/UseCases/, 2012. accessed 27-October-2017.

[DDCM12] Gianluca Demartini, Djellel Eddine Difallah, and Philippe Cudré-Mauroux. Zencrowd: Leveraging probabilistic reasoning and crowdsourcing techniques for large-scale entity linking. In *Proceedings of the 21st International Conference on World Wide Web*, WWW '12, pages 469–478, New York, NY, USA, 2012. ACM.

[dHFRM+14] A Garcıa Seco de Herrera, Antonio Foncubierta-Rodríguez, Dimitrios Markonis, Roger Schaer, and Henning Müller. Crowdsourcing for medical image classification. In *Annual congress SGMI*, volume 2014, 2014.

[DKC+18] Florian Daniel, Pavel Kucherbaev, Cinzia Cappiello, Boualem Benatallah, and Mohammad Allahbakhsh. Quality control in crowdsourcing: A survey of quality attributes, assessment techniques, and assurance actions. *ACM Comput. Surv.*, 51(1):7:1–7:40, January 2018.

[dN12] Mathieu d'Aquin and Natalya F. Noy. Where to publish and find ontologies? a survey of ontology libraries. *Web Semantics: Science, Services and Agents on the World Wide Web*, 11:96 – 111, 2012.

[DTEJ17] Biswanath Dutta, A Toulet, Vincent Emonet, and Clement Jonquet. New generation metadata vocabulary for ontology description and publication, 11 2017.

[FKK08] Norbert E. Fuchs, Kaarel Kaljurand, and Tobias Kuhn. *Attempto Controlled English for Knowledge Representation*, pages 104–124. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[FRV+12] Giorgos Flouris, Yannis Roussakis, Maria Villalon, Pablo Mendes, and Irini Fundulaki. Using provenance for quality assessment and repair in linked open data. *CEUR Workshop Proceedings*, 890, 01 2012.

[GMN+17] Ujwal Gadiraju, Sebastian Möller, Martin Nöllenburg, Dietmar Saupe, Sebastian Egger-Lampl, Daniel Archambault, and Brian Fisher. Crowdsourcing versus the laboratory: Towards human-centered experiments using the crowd. In Daniel Archambault, Helen Purchase, and Tobias Hoßfeld, editors, *Evaluation in the Crowd. Crowdsourcing and Human-Centered Experiments*, pages 6–26, Cham, 2017. Springer International Publishing.

[HBL10] Jim Hendler and Tim Berners-Lee. From the semantic web to social machines. *Artif. Intell.*, 174(2):156–161, February 2010.

[HI11] Björn Hartmann and Panagiotis G Ipeirotis. What's the right price? pricing tasks for finishing on time. 2011.

[HMS09] Pei-Yun Hsueh, Prem Melville, and Vikas Sindhwani. Data quality from crowdsourcing: A study of annotation selection criteria. In *Proceedings of*

the NAACL HLT 2009 Workshop on Active Learning for Natural Language Processing, HLT '09, pages 27–35, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics.

[How06] Jeff Howe. The rise of crowdsourcing. *Wired magazine*, 14(6):1–4, 2006.

[How08] Jeff Howe. *Crowdsourcing: Why the Power of the Crowd Is Driving the Future of Business*. Crown Publishing Group, New York, NY, USA, 1 edition, 2008.

[HSC+13] Derek L. Hansen, Patrick J. Schone, Douglas Corey, Matthew Reid, and Jake Gehring. Quality control mechanisms for crowdsourcing: Peer review, arbitration, & expertise at familysearch indexing. In *Proceedings of the 2013 Conference on Computer Supported Cooperative Work*, CSCW '13, pages 649–660, New York, NY, USA, 2013. ACM.

[HSSV15] Chien-Ju Ho, Aleksandrs Slivkins, Siddharth Suri, and Jennifer Wortman Vaughan. Incentivizing high quality crowdwork. In *Proceedings of the 24th International Conference on World Wide Web*, WWW '15, pages 419–429, Republic and Canton of Geneva, Switzerland, 2015. International World Wide Web Conferences Steering Committee.

[HTGV13] Tobias Hoßfeld, Phuoc Tran-Gia, and Maja Vucovic. Crowdsourcing: From theory to practice and long-term perspectives (dagstuhl seminar 13361). In *Dagstuhl Reports*, volume 3. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2013.

[HU69] John E. Hopcroft and Jeffrey D. Ullman. *Formal Languages and Their Relation to Automata*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1969.

[Hun86] R.J. Hunt. Percent agreement, pearson's correlation, and kappa as measures of inter-examiner reliability. *Journal of Dental Research*, 65(2):128–130, 1986. PMID: 3455967.

[KA13] Ali Khalili and Sören Auer. User interfaces for semantic authoring of textual content: A systematic literature review. *Web Semantics: Science, Services and Agents on the World Wide Web*, 22:1–18, 2013.

[KHC+16] Ranjay A. Krishna, Kenji Hata, Stephanie Chen, Joshua Kravitz, David A. Shamma, Li Fei-Fei, and Michael S. Bernstein. Embracing error to enable rapid crowdsourcing. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 3167–3179, New York, NY, USA, 2016. ACM.

[KSV18] N. Kaufmann, T. Schulze, and Daniel Veit. More than fun and money: Worker motivation in crowdsourcing; a study on mechanical turk. In

Proceedings of the 17th Americas Conference on Information Systems (AMCIS 2011), 4-8 August 2011, Detroit, Michigan, USA, 2018.

[KVH16] Ramya Korlakai Vinayak and Babak Hassibi. Crowdsourced clustering: Querying edges vs triangles. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 1316–1324. Curran Associates, Inc., 2016.

[LIJ+15] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. Dbpedia–a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, 6(2):167–195, 2015.

[LWSC05] Wei Liu, Albert Weichselbraun, Arno Scharl, and Elizabeth Chang. Semi-automatic ontology extension using spreading activation. *Journal of Universal Knowledge Management*, 1(1):50–58, 2005.

[LYG+16] Edith Law, Ming Yin, Joslin Goh, Kevin Chen, Michael A. Terry, and Krzysztof Z. Gajos. Curiosity killed the cat, but makes crowdwork better. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, pages 4098–4110, New York, NY, USA, 2016. ACM.

[MCHJ17] Ke Mao, Licia Capra, Mark Harman, and Yue Jia. A survey of the use of crowdsourcing in software engineering. *Journal of Systems and Software*, 126:57–84, 2017.

[MMB12] Pablo N. Mendes, Hannes Mühleisen, and Christian Bizer. Sieve: Linked data quality assessment and fusion. In *Proceedings of the 2012 Joint EDBT/ICDT Workshops*, EDBT-ICDT '12, pages 116–123, New York, NY, USA, 2012. ACM.

[MMJ+15] Jonathan M. Mortensen, Evan P. Minty, Michael Januszyk, Timothy E. Sweeney, Alan L. Rector, Natalya Fridman Noy, and Mark A. Musen. Using the wisdom of the crowds to find critical errors in biomedical ontologies: a study of SNOMED CT. *JAMIA*, 22(3):640–648, 2015.

[MMWB05] Alistair Miles, Brian Matthews, Michael Wilson, and Dan Brickley. Skos core: Simple knowledge organisation for the web. *International Conference on Dublin Core and Metadata Applications*, 0(0):3–10, 2005.

[Mor13] Jonathan M. Mortensen. Crowdsourcing ontology verification. In Harith Alani, Lalana Kagal, Achille Fokoue, Paul Groth, Chris Biemann, Josiane Xavier Parreira, Lora Aroyo, Natasha Noy, Chris Welty, and Krzysztof Janowicz, editors, *The Semantic Web – ISWC 2013*, pages 448–455, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[MTH+16] Jonathan M. Mortensen, Natalie Telis, Jacob J. Hughey, Hua Fan-Minogue, Kimberly Van Auken, Michel Dumontier, and Mark A. Musen. Is the crowd better as an assistant or a replacement in ontology engineering? an exploration through the lens of the gene ontology. *Journal of Biomedical Informatics*, 60:199–209, 2016.

[Nil10] Mikael Nilsson. *From Interoperability to Harmonization in Metadata Standardization : Designing an Evolvable Framework for Metadata Harmonization*. PhD thesis, KTH, Media Technology and Graphic Arts, Media, 2010. QC 20101117.

[Pow11] D. M. W. Powers. Evaluation: From precision, recall and f-measure to roc., informedness, markedness & correlation. *Journal of Machine Learning Technologies*, 2(1):37–63, 2011.

[Sab06] M. Sabou. *Building web service ontologies*. PhD thesis, Vrije Universiteit Amsterdam, 2006. Naam instelling promotie: VU Vrije Universiteit Naam instelling onderzoek: VU Vrije Universiteit.

[SAF13] Elena Simperl, Maribel Acosta, and Fabian Flöck. Knowledge engineering via human computation. In *Handbook of Human Computation*, pages 131–151. Springer, 2013.

[SBS12] Marta Sabou, Kalina Bontcheva, and Arno Scharl. Crowdsourcing research opportunities: Lessons from natural language processing. In *Proceedings of the 12th International Conference on Knowledge Management and Knowledge Technologies*, i-KNOW '12, pages 17:1–17:8, New York, NY, USA, 2012. ACM.

[SCVP98] C. Shapiro, S. Carl, H.R. Varian, and Harvard Business Press. *Information Rules: A Strategic Guide to the Network Economy*. Strategy/Technology / Harvard Business School Press. Harvard Business School Press, 1998.

[SF08] A. Sorokin and D. Forsyth. Utility data annotation with amazon mechanical turk. In *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–8, June 2008.

[SG11] Eric Schenk and Claude Guittard. Towards a characterization of crowdsourcing practices. *Journal of innovation economics*, 1(7):93–107, 2011.

[SH08a] Katharina Siorpaes and Martin Hepp. Games with a purpose for the semantic web. *IEEE Intelligent Systems*, 23(3):50–60, May 2008.

[SH08b] Katharina Siorpaes and Martin Hepp. Ontogame: Weaving the semantic web by online games. In Sean Bechhofer, Manfred Hauswirth, Jörg Hoffmann, and Manolis Koubarakis, editors, *The Semantic Web: Research and Applications*, pages 751–766, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[SM11] Martin Svoboda and Irena Mlýnková. Linked data indexing methods: A survey. In Robert Meersman, Tharam Dillon, and Pilar Herrero, editors, *On the Move to Meaningful Internet Systems: OTM 2011 Workshops*, pages 474–483, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[Sma08] Paul R Smart. Controlled natural languages and the semantic web. July 2008.

[SR14] Parnia Samimi and Sri Devi Ravana. Creation of reliable relevance judgments in information retrieval systems evaluation experimentation through crowdsourcing: a review. *The Scientific World Journal*, 2014, 2014.

[SSN+15] Christina Sarasua, Elena Simperl, Natasha Noy, Abraham Bernstein, and Jan Marco Leimeister. Crowdsourcing and the semantic web: A research manifesto. *Human Computation (HCOMP)*, 2(1):3–17, 2015.

[SWC+] Elena Simperl, Stephan Wölger, Anton Chalakov, Stefan Thaler, and Maribel Acosta. Mechanical Protégé. http://people.aifb.kit.edu/yt2652/mechanicalProtege/. [Online; accessed 6-April-2019].

[SWPB18] Marta Sabou, Dietmar Winkler, Peter Penzerstadler, and Stefan Biffl. Verifying conceptual domain models with human computation: A case study in software engineering. In *Proceedings of the Sixth AAAI Conference on Human Computation and Crowdsourcing, HCOMP 2018, Zürich, Switzerland, July 5-8, 2018.*, pages 164–173, 2018.

[vA06] L. von Ahn. Games with a purpose. *Computer*, 39(6):92–94, June 2006.

[Ver13] Csaba Veres. Crowdsourced semantics with semantic tagging: "don't just tag it, lexitag it!". In *Proceedings of the 1st International Conference on Crowdsourcing the Semantic Web - Volume 1030*, CrowdSem'13, pages 1–15, 2013.

[WBS13] Gerhard Wohlgenannt., Stefan Belk., and Matthias Schett. A prototype for automating ontology learning and ontology evolution. In *Proceedings of the International Conference on Knowledge Engineering and Ontology Development - Volume 1: KEOD, (IC3K 2013)*, pages 407–412. INSTICC, SciTePress, 2013.

[WC14] Bo Waggoner and Yiling Chen. Output agreement mechanisms and common knowledge. In *Second AAAI Conference on Human Computation and Crowdsourcing*, 2014.

[WF00] Ian H. Witten and Eibe Frank. *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2000.

[WFHP16] I.H. Witten, E. Frank, M.A. Hall, and C.J. Pal. *Data Mining: Practical Machine Learning Tools and Techniques*. The Morgan Kaufmann Series in Data Management Systems. Elsevier Science, 2016.

[WSH16] Gerhard Wohlgenannt, Marta Sabou, and Florian Hanika. Crowd-based ontology engineering with the ucomp protege plugin. *Semantic Web*, 7(4):379–398, 2016.

[WSP+17a] Dietmar Winkler, Marta Sabou, Sanja Petrovic, Gisele Carneiro, Marcos Kalinowski, and Stefan Biffl. Improving model inspection processes with crowdsourcing: Findings from a controlled experiment. In Jakub Stolfa, Svatopluk Stolfa, Rory V. O'Connor, and Richard Messnarz, editors, *Systems, Software and Services Process Improvement*, pages 125–137, Cham, 2017. Springer International Publishing.

[WSP+17b] Dietmar Winkler, Marta Sabou, Sanja Petrovic, Gisele Carneiro, Marcos Kalinowski, and Stefan Biffl. Improving model inspection with crowdsourcing. In *Proceedings of the 4th International Workshop on CrowdSourcing in Software Engineering*, CSI-SE '17, pages 30–34, Piscataway, NJ, USA, 2017. IEEE Press.

[WWSS12] Gerhard Wohlgenannt, Albert Weichselbraun, Arno Scharl, and Marta Sabou. Dynamic integration of multiple evidence sources for ontology learning. *Journal of Information and Data Management*, 3(3):243–254, 2012.

[XHSH14] Guo Xintong, Wang Hongzhi, Yangqiu Song, and Gao Hong. Brief survey of crowdsourcing for data mining. *Expert Systems with Applications*, 41(17):7987–7994, 2014.

[YKL11] Man-Ching Yuen, Irwin King, and Kwong-Sak Leung. A survey of crowdsourcing systems. In *Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011 IEEE Third International Conference on*, pages 766–773. IEEE, 2011.

[ZGEJ17] Maayan Zhitomirsky-Geffet, Eden S Erez, and Bar-Ilan Judit. Toward multiviewpoint ontology construction by collaboration of non-experts and crowdsourcing: The case of the effect of diet on health. *Journal of the Association for Information Science and Technology*, 68(3):681–694, 2017.

[ZLD13] Xinshu Zhao, Jun S. Liu, and Ke Deng. Assumptions behind intercoder reliability indices. *Annals of the International Communication Association*, 36(1):419–480, 2013.

[ZRM+16] Amrapali Zaveri, Anisa Rula, Andrea Maurino, Ricardo Pietrobon, Jens Lehmann, and Sören Auer. Quality assessment for linked data: A survey. *Semantic Web*, 7(1):63–93, 2016.