Crowdsourcing Linked Data quality assessment

Maribel Acosta[1], Amrapali Zaveri[2], Elena Simperl[3], Dimitris Kontokostas[2], Sören Auer[4] and Jens Lehmann[2]

[1] Karlsruhe Institute of Technology, Institute AIFB
maribel.acosta@kit.edu
[2] Universität Leipzig, Institut für Informatik, AKSW
{zaveri|kontokostas|lehmann}@informatik.uni-leipzig.de
[3] University of Southampton, Web and Internet Science Group
E.Simperl@soton.ac.uk
[4] University of Bonn, Enterprise Information Systems and Fraunhofer IAIS
auer@cs.uni-bonn.de

Abstract. In this paper we look into the use of crowdsourcing as a means to handle Linked Data quality problems that are challenging to be solved automatically. We analyzed the most common errors encountered in Linked Data sources and classified them according to the extent to which they are likely to be amenable to a specific form of crowdsourcing. Based on this analysis, we implemented a quality assessment methodology for Linked Data that leverages the wisdom of the crowds in different ways: (i) a contest targeting an expert crowd of researchers and Linked Data enthusiasts; complemented by (ii) paid microtasks published on Amazon Mechanical Turk. We empirically evaluated how this methodology could efficiently spot quality issues in DBpedia. We also investigated how the contributions of the two types of crowds could be optimally integrated into Linked Data curation processes. The results show that the two styles of crowdsourcing are complementary and that crowdsourcing-enabled quality assessment is a promising and affordable way to enhance the quality of Linked Data.

1 Introduction

Many would agree that Linked Data (LD) is one of the most important technological developments in data management of the last decade. However, one of the less positive aspects of this great success story is related to the varying quality of Linked Data sources, which often poses serious problems to developers aiming to seamlessly consume and integrate Linked Data in their applications. This state of the affairs is the result of a combination of data- and process-related factors. Keeping aside the factual flaws of the original sources, the array of data sources that may be subject to RDFification is highly heterogeneous in terms of format, organization and vocabulary. As a direct consequence, some kinds of data tend to be more challenging to translate into RDF than others, leading to errors in the Linked Data provisioning process. Some of the quality issues hence produced (e.g., missing values) can be easily repaired automatically, but others require manual intervention. In this paper we look into the use of crowdsourcing as a data curation strategy that is cost-efficient and accurate in terms of the level of granularity of the errors to be spotted.

We analyzed the most common quality problems encountered in Linked Data sources and classified them according to the extent to which they are likely to be amenable to a specific form of crowdsourcing. Based on this analysis, we implemented a quality assessment methodology for Linked Data that leverages the wisdom of the crowds in the following ways: (i) we first launched a **contest** targeting an expert crowd of LD researchers and enthusiasts in order to *find* and classify erroneous RDF triples; and then (ii) published the outcome of this contest as **paid microtasks** on Amazon Mechanical Turk (MTurk)[5] in order to *verify* the issues spotted by the experts [1].

These two crowdsourcing approaches have advantages and disadvantages. Each approach makes specific assumptions about the audiences they address (the 'crowd') and the skills of the potential contributors. A contest reaches out to the crowd to solve a given problem and rewards the best ideas; it exploits competition and intellectual challenge as main drivers for participation. The idea, originating from open innovation, has been employed in many domains, from creative industries to sciences, for tasks of varying complexity (from designing logos to building sophisticated algorithms). We applied this contest-based model to mobilize an expert crowd consisting of researchers and Linked Data enthusiasts to discover and classify quality issues in DBpedia. The participant who covered the highest number of DBpedia resources won a prize.

Microtask crowdsourcing traditionally covers a different set of scenarios. Tasks primarily rely on basic human abilities, including visual and audio cognition, as well as natural language understanding and communication (sometimes in different languages) and less on acquired skills (such as subject-matter knowledge). As such, a great share of the tasks addressed via microtask platforms like MTurk could be referred to as 'routine' tasks – recognizing objects in images, transcripting audio and video material and text editing. To be more efficient than traditional outsourcing (or even in-house resources), the tasks need to be highly parallelized. This means that the actual work is executed by a high number of contributors in a decentralized fashion; this not only leads to significant improvements in terms of time of delivery, but also offers a means to cross-check the accuracy of the answers (as each task is typically assigned to more than one person) and reward the workers according to their performance and productivity. We applied microtask crowdsourcing as a fast and cost-efficient way to examine the errors spotted by the expert crowd who participated in the contest. More concretely, we looked into three types of quality problems the experts found in DBpedia: (i) object values incorrectly or incompletely extracted; (ii) data type incorrectly extracted; and (iii) incorrect links between DBpedia entities and related sources on the Web. The underlying data was translated into Human Intelligence Tasks (or HITs), the unit of work in MTurk, which were handled by workers on the MTurk platform.

We empirically evaluated how this methodology – based on a mixed crowdsourcing approach – could efficiently spot quality issues in DBpedia. The results show that the two styles of crowdsourcing are complementary and that crowdsourcing-enabled quality assessment is a promising and affordable way to enhance the quality of Linked Data sets, which, in the long run, may address many of the problems that fundamentally constrain the usability of the Web of Data in real-world applications.

[5] `https://www.mturk.com/`

2 Linked Data quality issues

The Web of Data spans a network of data sources of varying quality. There are a large number of high-quality data sets, for instance, in the life-science domain, which are the result of decades of thorough curation and have been recently made available as Linked Data[6]. Other data sets, however, have been (semi-)automatically translated to RDF from their primary sources, or via crowdsourcing in a decentralized process involving a large number of contributors. Probably the best example of a data set produced in this manner is DBpedia [9]. While the combination of machine-driven extraction and crowdsourcing was a reasonable approach to produce a baseline version of a greatly useful resource, it was also the cause of a wide range of quality problems, in particular in the mappings between Wikipedia attributes and their corresponding DBpedia properties.

Our analysis of Linked Data quality issues focuses on DBpedia as a representative data set for the broader Web of Data due to the diversity of the types of errors exhibited and the vast domain and scope of the data set. In our previous work [16], we compiled a list of data quality dimensions (criteria) applicable to Linked Data quality assessment. Afterwards, we mapped these dimensions to DBpedia [15]. A sub-set of four criteria of the original framework were found particularly relevant in this setting: *Accuracy*, *Relevancy*, *Representational-Consistency* and *Interlinking*. To provide a comprehensive analysis of DBpedia quality, we further divided these four categories of problems into sub-categories. For the purpose of this paper, from these categories we chose the following three triple-level quality issues.

Object incorrectly/incompletely extracted. Consider the triple: `dbpedia:Firewing dbpprop:isbn "978"^^xsd:integer`. This DBpedia resource is about the children's book 'Firewing', with the incomplete and incorrect value of the ISBN number. Instead of extracting the entire ISBN number from Wikipedia, `978-0-00-639194-4`, only the first three digits were extracted.

Data type incorrectly extracted. This category refers to triples with an incorrect data type for a typed literal. For example, in the DBpedia ontology, the range of the property `activeYearsStartYear` is defined as `xsd:gYear`. Although the data type declaration is correct in the triple `dbpedia:Stephen_Fry dbpedia-owl:activeYears-StartYear "1981-01-01T00:00:00+02:00"^^xsd:gYear`, it is formatted as `xsd:dateTime`. The expected value is `"1981"^^xsd:gYear`.

Interlinking. In this case, links to external Web sites or other external data sources such as Wikimedia, Freebase, GeoSpecies or links generated via the Flickr wrapper are incorrect; that is, they do not show any related content pertaining to the resource.

The categories of quality problems just discussed occur pervasively in DBpedia. These problems might be present in other data sets which are extracted in a similar fashion as DBpedia. Given the diversity of the situations in which they can be instantiated (broad range of data types and object values) and their sometimes deeply contextual character (interlinking), assessing them automatically is challenging. In the following we explain how crowdsourcing could support quality assessment processes.

[6] `http://beta.bio2rdf.org/`

3 Crowdsourcing Linked Data quality assessment

Our work on human-driven Linked Data quality assessment focuses on two forms of crowdsourcing: contests and paid microtasks. As discussed in Section 1, these crowdsourcing approaches exhibit different characteristics in terms of the types of tasks they can be applied to, the way the results are consolidated and exploited, and the audiences they target. Table 1 presents a summary of the two approaches as they have been used in this work for Linked Data quality assessment purposes.

Table 1: Comparison between the proposed approaches to crowdsource LD quality assessment.

Characteristic	**Contest-based**	**Paid microtasks**
Participants	Controlled group: LD experts	Anonymous large group
Goal per task	Detecting and classifying LD quality issues	Confirming LD quality issues
Task size	Participants explore RDF resources and identify incorrect triples	Participants analyze human-readable information of given RDF triples
Task complexity	*Difficult:* the task requires knowledge on data quality issues	*Easy:* the task consists of validating pre-processed and classified triples
Time duration	Long (weeks)	Short (days)
Reward	A final prize	Micropayments
Reward mechanism	The winner gets the prize	Each participant receives a payment
Tool/platform	*TripleCheckMate*	Amazon Mechanical Turk (MTurk)

We applied the crowdsourcing pattern *Find-Fix-Verify* [1] to assess the quality of DBpedia. This pattern consists of a three-stage process, which is originally defined as follows. The *Find* stage asks the crowd to identify problematic elements within a data source. In the second stage, *Fix*, the crowd corrects the elements belonging to the outcome of the previous stage. The *Verify* stage corresponds to a final quality control iteration. Our approach (see Figure 1) leverages the expertise of Linked Data experts in a contest to *find* and classify erroneous triples according to a pre-defined scheme [16]. The outcome of this stage – triples judged as 'incorrect' – is then *verified* by the MTurk workers, who are instructed to assess specific types of errors in the subset of triples. The implementation of the *fix* stage is out of the scope of this paper, since the main goal of this work is identifying quality issues.

The Find-Fix-Verify pattern reduces the noise caused by low-quality participants, while the costs remain competitive with other crowdsourcing alternatives. In addition, this approach is efficient in terms of the number of questions asked to the paid microtask crowd. In scenarios in which crowdsourcing is applied to enhance or validate the results of machine computation tasks, question filtering relies on specific thresholds or historical information about the likelihood that human input will significantly improve the results generated algorithmically. Find-Fix-Verify addresses scenarios which can be hardly engineered, like in our case the discovery and classification of various types of errors in DBpedia. In these scenarios, in a first step one applies crowdsourcing not only to solve the task at hand, but also to define the specific questions which need to be addressed. These steps can employ different types of crowds, as they require different skills and expertise [1]. In the following we elaborate on the specific processes carried out by each type of crowd in this work.

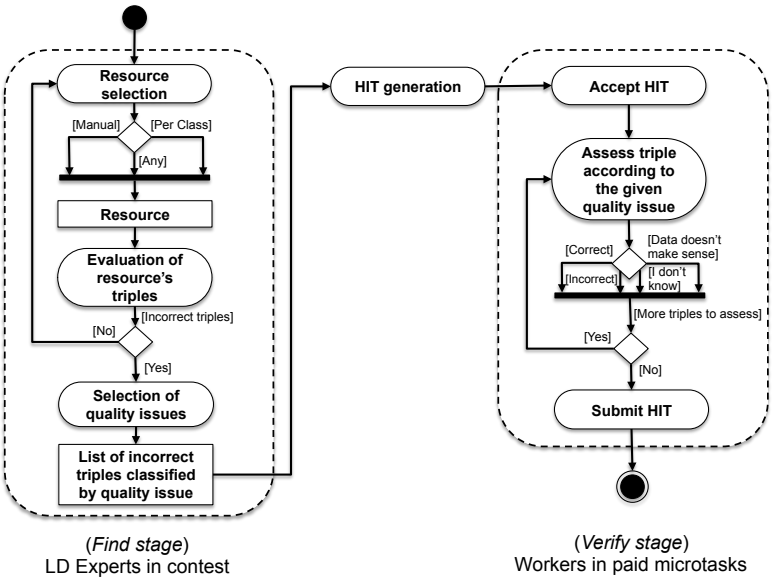


Fig. 1: Workflow of the applied Linked Data quality assessment methodology.

3.1 Contest-based crowdsourcing

Contests as means to successfully involve experts in advancing science have a long-standing tradition in research, e.g., the Darpa challenges[7] and NetFlix[8]. In our case, we reached out to an expert crowd of researchers and Linked Data enthusiasts via a contest, in order to identify and classify specific types of Linked Data quality problems in DBpedia. To collect the contributions from this crowd, in a previous work of ours [16], we developed a web-based tool, *TripleCheckMate*[9] (see Figure 2), which allows users to select resources, identify issues related to triples of the resource and classify these issues according to a pre-defined taxonomy of data quality problems. A prize was announced for the user submitting the highest number of (real) quality problems.

As a basic means to avoid spam, each user first has to login using her Google Mail ID. Then, as shown in Figure 1, she is presented with three options to choose a resource from DBpedia: (i) *Any*, for random selection; (ii) *Per Class*, where she may choose a resource belonging to a particular class of her interest; and (iii) *Manual*, where she may provide a URI of a resource herself. Once a resource is selected following one of these alternatives, the user is presented with a table in which each row corresponds to an RDF triple of that resource. The next step is the actual quality assessment at triple level. The user is provided with the link to the corresponding Wikipedia page of the given resource in order to offer more context for the evaluation. If she detects a triple

[7] `http://www.darpa.mil/About/History/Archives.aspx`
[8] `http://www.netflixprize.com/`
[9] Available at `http://github.com/AKSW/TripleCheckMate`

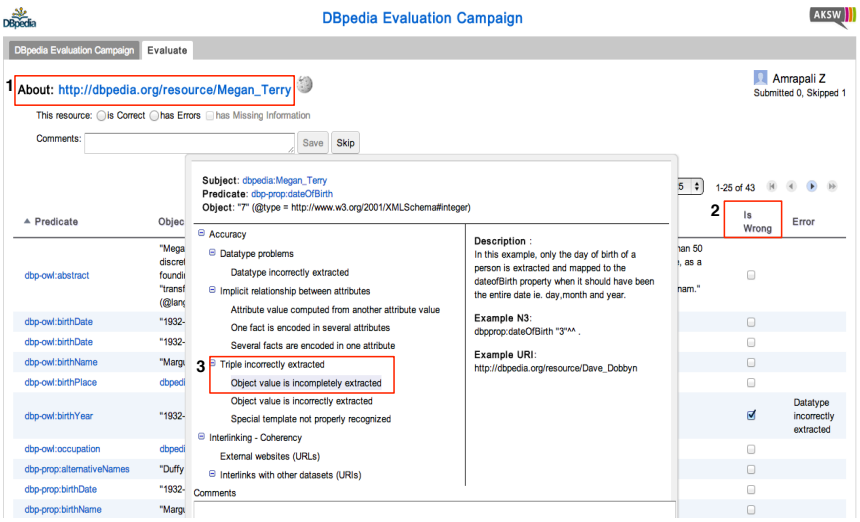


Fig. 2: Screenshot of the *TripleCheckMate* crowdsourcing data quality assessment tool.

containing a problem, she checks the box 'Is Wrong'. Moreover, she can assign these troublesome triples to quality problems (according to the classification devised in [15]), as shown in Figure 2. In this manner, the tool only records the triples that are identified as 'incorrect'. This is consistent with the *Find* stage from the Find-Fix-Verify pattern, where the crowd exclusively detects the problematic elements; while the remaining data is not taken into consideration.

The tool *TripleCheckMate* measures inter-rater agreements. This means that DBpedia resources are typically checked multiple times. This redundancy mechanism is extremely useful to analyze the performance of the users (as we compare their responses against each other), to identify quality problems which are likely to be real (as they are confirmed by more than one opinion) and to detect unwanted behavior (as users are not 'rewarded' unless their assessments are 'consensual').

The outcome of this contest corresponds to a set of triples judged as 'incorrect' by the experts and classified according to the detected quality issue.

3.2 Paid microtasks

To fully unfold its benefits, this form of crowdsourcing needs to be applied to problems which can be broken down into smaller units of work (called 'microtasks' or 'Human Intelligence Tasks' – HITs) that can be undertaken in parallel by independent parties[10]. As noted earlier, the most common model implies small financial rewards for each worker taking on a microtask, whereas each microtask may be assigned to more than one worker in order to allow for techniques such as majority voting to automatically identify accurate responses.

[10] More complex workflows, though theoretically feasible, require additional functionality to handle task dependencies.

We applied this crowdsourcing approach in order to *verify* quality issues in DBpedia RDF triples identified as problematic during the contest (see Figure 1). One of the challenges in this context is to develop useful human-understandable interfaces for HITs. In microtasks, optimal user interfaces reduce ambiguity as well as the probability to retrieve erroneous answers from the crowd due to a misinterpretation of the task. Further design criteria were related to spam detection and quality control; we used different mechanisms to discourage low-effort behavior which leads to random answers and to identify accurate answers (see Section 4.1.2).

Based on the classification of LD quality issues explained in Section 2, we created three different types of HITs. Each type of HIT contains the description of the procedure to be carried out to complete the task successfully. We provided the worker examples of incorrect and correct examples along with four options: (i) Correct; (ii) Incorrect; (iii) I cannot tell/I don't know; (iv) Data doesn't make sense. The third option was meant to allow the user to specify when the question or values were unclear. The fourth option referred to those cases in which the presented data was truly unintelligible. Furthermore, the workers were not aware that the presented triples were previously identified as 'incorrect' by experts and the questions were designed such that the worker could not foresee the right answer. The resulting HITs were submitted to Amazon Mechanical Turk using the MTurk SDK for Java[11]. We describe the particularities of each type of HIT in the following.

Incorrect/incomplete object value. In this type of microtask, we asked the workers to evaluate whether the value of a given RDF triple from DBpedia is correct or not. Instead of presenting the set of RDF triples to the crowd, we displayed human-readable information retrieved by dereferencing the URIs of the subject and predicate of the triple. In particular, we selected the values of the `foaf:name` or `rdfs:label` properties for each subject and predicate. Additionally, in order to provide contextual information, we implemented a wrapper which extracted the corresponding data encoded in the infobox of the Wikipedia article – specified as `foaf:isPrimaryTopicOf` of the subject. Figure 3 depicts the interface of the resulting tasks.

In the task presented in Figure 3a, the worker must decide whether the date of birth of "Dave Dobbyn" is correct. According to the DBpedia triple, the value of this property is `3`, while the information extracted from Wikipedia suggests that the right value is `3 January 1957`. In addition, it is evident that the DBpedia value is erroneous as the value "3" is not appropriate for a date. Therefore, the right answer to this tasks is: the DBpedia data is incorrect.

An example of a DBpedia triple whose value is correct is depicted in Figure 3b. In this case, the worker must analyze the date of birth of "Elvis Presley". According to the information extracted from Wikipedia, the date of birth of Elvis Presley is `January 8, 1935`, while the DBpedia value is `1935-01-08`. Despite the dates are represented in different formats, semantically the dates are indeed the same, thus the DBpedia value is correct.

Incorrect data type. This type of microtask consists of detecting those DBpedia triples whose data type – specified via `@type` – was not correctly assigned. The generation of

[11] `http://aws.amazon.com/code/695`

(a) Incorrect object value in DBpedia **(b) Correct** object value in DBpedia

Fig. 3: Incorrect/incomplete object value: The crowd must compare the DBpedia and Wikipedia values and decide whether the DBpedia entry is correct or not for a given subject and predicate.

the interfaces for these tasks was very straightforward, by dereferencing the URIs of the subject and predicate of each triple and displaying the values for the `foaf:name` or `rdfs:label`.

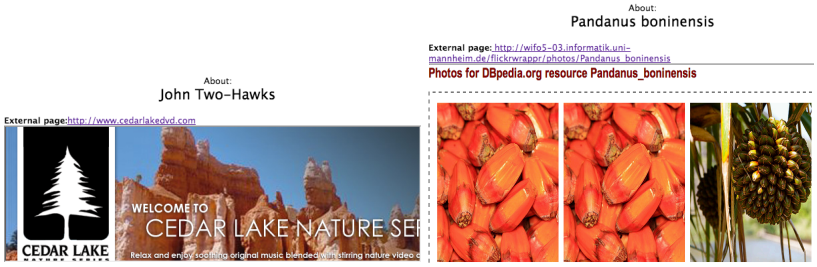
In the description of the task, we introduced the concept of data type of a value and provided two simple examples. The first example illustrates when the data type is incorrect while analyzing the entity "Torishima Izu Islands": `Given the property "name", is the value "" of type "English"?` A worker does not need to understand that the name of this island is written in "Japanese", since it is evident that the language type "English" in this example is incorrect. In a similar fashion, we provided an example where the data type is assigned correctly by looking at the entity "Elvis Presley": `Given the property "name", is the value "Elvis Presley" of type "English"?` According to the information from DBpedia, the value of the name is written in English and the type is correctly identified as English.

Incorrect links. In this type of microtask, we asked the workers to verify whether the content of the external page referenced from the Wikipedia article corresponds to the subject of the RDF triple. For the interface of the HITs, we provided the worker a preview of the Wikipedia article and the external page by implementing HTML `iframe` tags. In addition, we retrieved the `foaf:name` of the given subject and the link to the corresponding Wikipedia article using the predicate `foaf:isPrimaryTopicOf`.

Examples of this type of task are depicted in Figure 4. In the first example (see Figure 4a), the workers must decide whether the content in the given external web page is related to "John Two-Hawks". It is easy to observe that in this case the content is not directly associated to the person "John Two-Hawks". Therefore, the right answer is that the link is incorrect. On the other hand, we also exemplified the case when an interlink presents relevant content to the given subject. Consider the example in Figure 4b, where the subject is the plant "Pandanus boninensis" and the external link is a web page generated by the DBpedia Flickr wrapper. The web page indeed shows pictures of the subject plant. Therefore, the correct answer is that the link is correct.

4 Evaluation

In our evaluation we investigated the following research questions: **(RQ1)** Is it possible to detect quality issues in LD data sets via crowdsourcing mechanisms? **(RQ2)** What type of crowd is most suitable for each type of quality issues? **(RQ3)** Which types of errors are made by lay users and experts?



(a) External link displaying **unrelated** content to the subject

(b) Web page displaying **related** images to the subject

Fig. 4: Incorrect link: The crowd must decide whether the content from an external web page is related to the subject.

4.1 Experimental design

In the following we describe the settings of the crowdsourcing experiments and the creation of a gold standard to evaluate the results from the contest and microtasks.

4.1.1 Contest settings

Participant expertise: We relied on the expertise of members of the Linking Open Data and the DBpedia communities who were willing to take part in the contest.
Task complexity: In the contest, each participant was assigned the full one-hop graph of a DBpedia resource. All triples belonging to that resource were displayed and the participants had to validate each triple individually for quality problems. Moreover, when a problem was detected, she had to map it to one of the problem types from a quality problem taxonomy.
Monetary reward: We awarded the participant who evaluated the highest number of resources a Samsung Galaxy Tab 2 worth 300 EU.
Assignments: Each resource was evaluated by at most two different participants.

4.1.2 Microtask settings

Worker qualification: In MTurk, the requester can filter workers according to different qualification metrics. In this experiment, we recruited workers whose previous HIT acceptance rate is greater than 50%.
HIT granularity: In each HIT, we asked the workers to solve 5 different questions. Each question corresponds to an RDF triple and each HIT contains triples classified into one of the three quality issue categories discussed earlier.
Monetary reward: The micropayments were fixed to 4 US dollar cents. Considering the HIT granularity, we paid 0.04 US dollar per 5 triples.
Assignments: In MTurk, a requester can specify the number of different workers to be assigned to solve each HIT. This allows to collect multiple answers for each question,

thus compensating the lack of LD-specific expertise of the workers. This mechanism is core to microtask crowdsourcing, which, as discussed in Section 1, is primarily dedicated to 'routine' tasks that make no assumption about the knowledge or skills of the crowd besides basic human capabilities. The number of assignments was set up to 5 and the answer was selected applying majority voting. We additionally compared the quality achieved by a group of workers vs. the resulting quality of the worker who submitted the first answer.

4.1.3 Creation of gold standard Two of the authors of this paper (MA, AZ) generated the gold standard for all the triples obtained from the contest and submitted to MTurk. To generate the gold standard, each author independently evaluated the triples. After an individual assessment, they compared their results and resolved the conflicts via mutual agreement. The inter-rater agreement between them was 0.4523 for object values, 0.5554 for data types and 0.5666 for interlinks. The inter-rater agreement values were calculated using the Cohen's kappa measure. Disagreement arose in the object value triples when one of the reviewers marked number values which are rounded up to the next round number as correct. For example, the length of the course of the "1949 Ulster Grand Prix" was 26.5Km in Wikipedia but rounded up to 27Km in DBpedia. In case of data types, most disagreements were considering the data type "number" of the value for the property "year" as correct. For the links, those containing unrelated content were marked as correct by one of the reviewers since the link existed in the original Wikipedia page.

The tools used in our experiments and the results are available online, including the outcome of the contest,[12] the gold standard and microtask data (HITs and results).[13]

4.2 Results

The contest was open for a predefined period of time of three weeks. During this time, 58 LD experts analyzed 521 distinct DBpedia resources and, considering an average of 47.19 triples per resource in this data set [15], we could say that the experts browsed around $24,560$ triples. They detected a total of $1,512$ triples as erroneous and classified them using the given taxonomy. After obtaining the results from the experts, we filtered out duplicates, triples whose objects were broken links and the external pages referring to the DBpedia Flickr Wrapper. In total, we submitted $1,073$ triples to the crowd. A total of 80 distinct workers assessed all the RDF triples in four days. A summary of these observations are shown in Table 2.

We compared the common $1,073$ triples assessed in each crowdsourcing approach against our gold standard and measured precision as well as inter-rater agreement values for each type of task (see Table 3). For the contest-based approach, the tool allowed two participants to evaluate a single resource. In total, there were 268 inter-evaluations for which we calculated the triple-based inter-agreement (adjusting the observed agreement with agreement by chance) to be 0.38. For the microtasks, we measured the inter-rater agreement values between a maximum of 5 workers for each type of task using Fleiss'

[12] `http://nl.dbpedia.org:8080/TripleCheckMate/`
[13] `http://people.aifb.kit.edu/mac/DBpediaQualityAssessment/`

Table 2: Overall results in each type of crowdsourcing approach.

	Contest-based	**Paid microtasks**
Number of distinct participants	Total: 58	Values: 35 Data type: 31 Interlink: 31 Total: 80
Total time	3 weeks (predefined)	4 days
Total no. of triples evaluated	1,512	1,073
Object value	550	509
Data type	363	341
Interlinks	599	223

kappa measure. While the inter-rater agreement between workers for the interlinking was high (0.7396), the ones for object values and data types was moderate to low with 0.5348 and 0.4960, respectively.

Table 3: Inter-rater agreement and precision values achieved with the implemented approaches.

	Object values	**Data types**	**Interlinks**
		Inter-rater agreement	
LD experts		Calculated for all the triples: 0.38	
MTurk workers	0.5348	0.4960	0.7396
		(True positives, False positives)	
LD experts	(364, 145)	(282, 59)	(34, 189)
MTurk workers (first answer)	(257, 108)	(144, 138)	(21, 13)
MTurk workers (majority voting)	(307, 35)	(134, 148)	(32, 2)
Baseline	N/A	N/A	(33, 94)
		Achieved precision	
LD experts	0.7151	0.8270	0.1525
MTurk workers (first answer)	0.7041	0.5106	0.6176
MTurk workers (majority voting)	0.8977	0.4752	0.9412
Baseline	N/A	N/A	0.2598

4.2.1 Incorrect/missing values As reported in Table 3, our crowdsourcing experiments reached a precision of 0.90 for MTurk workers (majority voting) and 0.72 for LD experts. Most of the missing or incomplete values that are extracted from Wikipedia occur with the predicates related to dates, for example: `(2005 Six Nations Championship, Date, 12)`. In these cases, the experts and workers presented a similar behavior, classifying 110 and 107 triples correctly, respectively, out of the 117 assessed triples for this class. The difference in precision between the two approaches can be explained as follows. There were 52 DBpedia triples whose values might seem erroneous, although they were correctly extracted from Wikipedia. One example of these triples is: `(English (programming language), Influenced by, ?)`. We found out that the LD experts classified all these triples as incorrect. In contrast, the workers successfully answered that 50 out of this 52 were correct, since they could easily compare the DBpedia and Wikipedia values in the HITs.

4.2.2 Incorrect data types Table 3 exhibits that the experts are reliable (with 0.83 of precision) on *finding* this type of quality issue, while the precision of the crowd (0.51) on *verifying* these triples is relatively low. In particular, the first answers submitted by the crowd were slightly better than the results obtained with majority voting. A detailed study of these cases showed that 28 triples that were initially classified correctly, later were misclassified, and most of these triples refer to a language data type. The low performance of the MTurk workers compared to the experts is not surprising, since this particular task requires certain technical knowledge about data types and, moreover, the specification of values and types in LD.

In order to understand the previous results, we analyzed the performance of experts and workers at a more fine-grained level. We calculated the frequency of occurrences of data types in the assessed triples (see Figure 5a) and reported the number of true positives (TP) and false positives (FP) achieved by both crowdsourcing methods for each data type. Figure 5b depicts these results. The most notorious result in this task is the assessment performance for the data type "number". The experts effectively identified triples where the data type was incorrectly assigned as "number"[14], for instance, in the triple `(Walter Flores, date of birth, 1933)` the value `1933` was number instead of date. These are the cases where the crowd was confused and determined that data type was correct, thus generating a large number of false positives. Nevertheless, it could be argued that the data type "number" in the previous example is not completely incorrect, when being unaware of the fact that there are more specific data types for representing time units. Under this assumption, the precision of the crowd would have been 0.8475 and 0.8211 for first answer and majority voting, respectively.

While looking at the typed literals in "English" (in RDF `@en`), Figure 5b shows that the experts perform very well when discerning whether a given value is an English text or not. The crowd was less successful in the following two situations: (i) the value corresponded to a number and the remaining data was specified in English, e.g., `(St. Louis School Hong Kong, founded, 1864)`; and (ii) the value was a text without special characters, but in a different language than English, for example German `(Woellersdorf-Steinabrueckl, Art, Marktgemeinde)`. The performance of both crowdsourcing approaches for the remaining data types were similar or not relevant due the low number of triples processed.

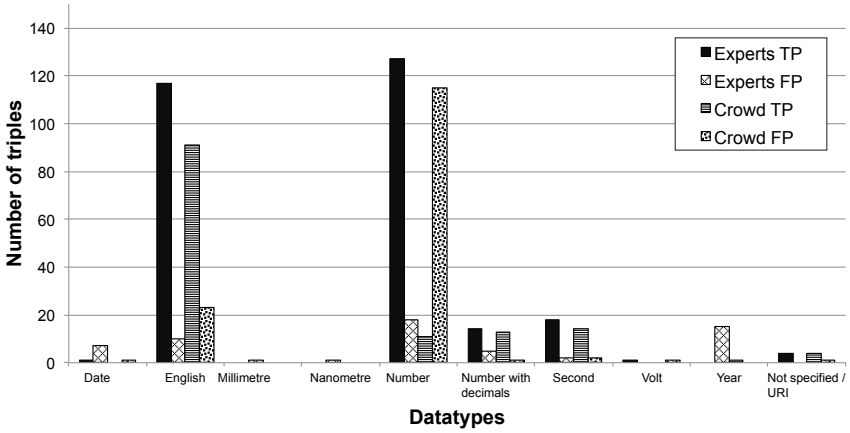
4.2.3 Incorrect links For this type of task, we additionally implemented a baseline approach to decide whether the linkage was correct. This automatic solution retrieves for each triple the external web page – which corresponds to the object of the triple – and searches for occurrences of the `foaf:name` of the subject within the page. If the number of occurrences is greater than 1, the algorithm interprets the external page as being related to the resource. In this case the link is considered correct.

Table 3 displays the precision for each studied quality assessment mechanism. The implemented baseline approach achieved a precision of 0.26. It obviously failed in the cases where the external pages corresponds to an image (which is the case of the 33% of the evaluated triples). On the other hand, the extremely low precision of 0.15 of the

[14] This error is very frequent when extracting dates from Wikipedia as some resources only contain partial data, e.g., only the year is available and not the whole date.

Data type	Frequency	Data type	Frequency
Date	8	Number with decimals	19
English	127	Second	20
Millimetre	1	Volt	1
Nanometre	1	Year	15
Number	145	Not specified/URI	4

(a) Frequency of data types in the crowdsourced triples.



(b) True positives (TP) and false positives (FP) per data type in each crowdourcing method.

Fig. 5: Analysis of true and false positives in "Incorrect data type" task.

contest's participants was unexpected. We discarded the possibility that the experts have made these mistakes due to a malfunction of the *TripleCheckMate* tool used during the contest. We analyzed in details the 189 misclassifications of the experts:

- The 95 Freebase links[15] connected via `owl:sameAs` were marked as incorrect, although both the subject and the object were referring to same real-world entity,
- there were 77 triples whose objects were Wikipedia-upload entries; 74 of these triples were also classified incorrectly,
- 20 links (blogs, web pages, etc.) referenced from the Wikipedia article of the subject were also misclassified, regardless of the language of the content in the web page.

The two settings of the MTurk workers outperformed the baseline approach. The 'first answer' setting reports a precision of 0.62, while the 'majority voting' achieved a precision of 0.94. The 6% of the links that were not properly classified by the crowd corresponds to those web pages whose content is in a different language than English or, despite they are referenced from the Wikipedia article of the subject, their association to the subject is not straightforward. Examples of these cases are the following subjects and links: 'Frank Stanford' and `http://nw-ar.com/drakefield/`, 'Forever Green' and `http://www.stirrupcup.co.uk`. We hypothesize that the design of the user

[15] `http://www.freebase.com`

interface of the HITs – displaying a preview of the web pages to analyze – helped the workers to easily identify those links containing related content to the triple subject.

5 Final discussion

Referring back to the research questions formulated at the beginning of Section 4, our experiments let us understand the strengths and weaknesses of applying crowdsourcing mechanisms for data quality assessment, following the Find-Fix-Verify pattern. For instance, we were able to detect common cases in which none of the two forms of crowdsourcing we studied seem to be feasible **(RQ3)**. The most problematic task for the LD experts was the one about discerning whether a web page is related to a resource. Although the experimental data does not provide insights into this behavior, we are inclined to believe that this is due to the relatively higher effort required by this specific type of task, which involves checking an additional site outside the *TripleCheckMate* tool. In turn the MTurk workers did not perform so well on tasks about data types where they recurrently confused numerical data types with time units.

In each type of task, the LD experts and MTurk workers applied different skills and strategies to solve the assignments successfully **(RQ2)**. The data collected for each type of task suggests that the effort of LD experts must be applied on the *Find* stage of those tasks demanding specific-domain skills beyond common knowledge. On the other hand, the MTurk crowd was exceptionally good and efficient at performing comparisons between data entries, specially when some contextual information is provided. This result suggests that microtask crowdsourcing can be effectively applied on the *Verify* stage of these tasks and possibly on the *Find* stage of the 'incorrect links' task.

Regarding the accuracy achieved in both cases, we compared the outcomes produced by each of the two crowds against a manually defined gold standard and against an automatically computed baseline, clearly showing that both forms of crowdsourcing offer feasible solutions to enhance the quality of Linked Data data sets **(RQ1)**.

One of the goals of our work is to investigate how the contributions of the two crowdsourcing approaches can be integrated into LD curation processes, by evaluating the performance of the two crowds in a cost-efficient way. In order to do this, both crowds must evaluate a common set of triples. The straightforward solution would be submitting to MTurk all the triples assessed by the LD experts, i.e., all the triples judged as 'incorrect' and 'correct' in the contest. As explained in Section 4, the experts browsed around 24,560 triples in total. Considering our microtask settings, the cost of submitting all these triple to MTurk would add up to over US$ 1,000. By contrast, our methodology aims at reducing the number of triples submitted to the microtask platform, while asking the workers to assess only the problematic triples found by the experts. By doing this, the cost of the experiments was reduced to only US$ 43.

The design of our methodology allowed us to exploit the strengths of both crowds: the LD experts detected and classified data quality problems, while the workers confirmed or disconfirmed the output of the experts in 'routine' tasks. In addition, an appropriate quality assurance methodology requires a quality control iteration, in this case performed by the MTurk workers. As can be seen in our experimental results (Table 3), it was not always the case that the triples judged as *incorrect* by the LD experts were indeed incorrect. In fact, the number of misjudged triples by the experts were 145 (out of

509) for incorrect/missing values, 59 (out of 341) for incorrect data type and 189 (out of 223) for incorrect interlinking. Therefore, always agreeing with the experts would deteriorate the overall output of the quality assurance process. In addition, the workers did not know that the data provided to them was previously classified as problematic. In consequence, the turkers could not have applied an strategy to guess the right answers.

6 Related work

Our work is situated at the intersection of the following research areas: *Crowdsourcing Linked Data management* and *Web data quality assessment*.

Crowdsourcing Linked Data management tasks. Several important Linked Data publication initiatives like DBpedia [9] and contests have been organized, including challenges[16] to the European Data Innovator Award[17]. At a technical level, specific Linked Data management tasks have been subject to human computation, including games with a purpose [10,13] and microtasks. For instance, microtasks have been used for entity linking [4] quality assurance, resource management [14] and ontology alignment [12].

Web data quality assessment. Existing frameworks for the quality assessment of the Web of Data can be broadly classified as automated [6], semi-automated [5] and manual [2,11]. Most of them are often limited in their ability to produce interpretable results, demand user expertise or are bound to a given data set. Other researchers analyzed the quality of Web [3] and RDF [7] data. The second study focuses on errors occurred during the publication of Linked Data sets. Recently, a survey [8] looked into four million RDF/XML documents to analyse Linked Data conformance.

7 Conclusions and future work

In this paper, we presented a methodology that adjusts the crowdsourcing pattern Find-Fix-Verify to exploit the strengths of experts and microtask workers. The *Find* stage was implemented using a contest-based format to engage with a community of LD experts in discovering and classifying quality issues of DBpedia resources. We selected a subset of the contributions obtained through the contest (referring to flawed object values, incorrect data types and missing links) and asked the MTurk crowd to *Verify* them. The evaluation showed that both types of approaches are successful; in particular, the microtask experiments revealed that people with no expertise in Linked Data can be a useful resource to identify very specific quality issues in an accurate and affordable manner, using the MTurk model. We consider our methodology can be applied to RDF data sets which are extracted from other sources and, hence, are likely to suffer from similar quality problems as DBpedia. Future work will first focus on conducting new experiments to test the value of the crowd for further different types of quality problems as well as for different LD sets from other knowledge domains. In the longer term, our work will also look into how to optimally integrate crowd contributions – by implementing the *Fix* stage – into curation processes and tools, in particular with respect to the trade-offs of costs and quality between manual and automatic approaches.

[16] For example: Semantic Web Challenge `http://challenge.semanticweb.org/`
[17] `http://2013.data-forum.eu/tags/european-data-innovator-award`

Acknowledgements

This work was supported by grants from the European Union's 7th Framework Programme provided for the projects EUCLID (GA no. 296229), GeoKnow (GA no. 318159) and LOD2 (GA no. 257943).

References

1. M. S. Bernstein, G. Little, R. C. Miller, B. Hartmann, M. S. Ackerman, D. R. Karger, D. Crowell, and K. Panovich. Soylent: a word processor with a crowd inside. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*, UIST '10, pages 313–322, New York, NY, USA, 2010. ACM.
2. C. Bizer and R. Cyganiak. Quality-driven information filtering using the wiqa policy framework. *Web Semantics*, 7(1):1 – 10, Jan 2009.
3. M. J. Cafarella, A. Y. Halevy, D. Z. Wang, E. Wu, and Y. Zhang. Webtables: exploring the power of tables on the web. *PVLDB*, 1(1):538–549, 2008.
4. G. Demartini, D. Difallah, and P. Cudré-Mauroux. Zencrowd: Leveraging probabilistic reasoning and crowdsourcing techniques for large-scale entity linking. In *21st International Conference on World Wide Web WWW 2012*, pages 469 – 478, 2012.
5. A. Flemming. Quality characteristics of linked data publishing datasources. Master's thesis, Humboldt-Universität of Berlin, 2010.
6. C. Guéret, P. T. Groth, C. Stadler, and J. Lehmann. Assessing linked data mappings using network measures. In *Proceedings of the 9th Extended Semantic Web Conference*, volume 7295 of *Lecture Notes in Computer Science*, pages 87–102. Springer, 2012.
7. A. Hogan, A. Harth, A. Passant, S. Decker, and A. Polleres. Weaving the pedantic web. In *LDOW*, 2010.
8. A. Hogan, J. Umbrich, A. Harth, R. Cyganiak, A. Polleres, and S. Decker. An empirical survey of linked data conformance. *Journal of Web Semantics*, 14:14–44, July 2012.
9. J. Lehmann, C. Bizer, G. Kobilarov, S. Auer, C. Becker, R. Cyganiak, and S. Hellmann. DBpedia - a crystallization point for the web of data. *Journal of Web Semantics*, 7(3):154–165, 2009.
10. T. Markotschi and J. Völker. Guesswhat?! - human intelligence for mining linked data. In *Proceedings of the Workshop on Knowledge Injection into and Extraction from Linked Data at EKAW*, 2010.
11. B. C. Mendes P.N., Mühleisen H. Sieve: Linked data quality assessment and fusion. In *LWDM*, 2012.
12. C. Sarasua, E. Simperl, and N. Noy. Crowdmap: Crowdsourcing ontology alignment with microtasks. In *The Semantic Web – ISWC 2012*, Lecture Notes in Computer Science, pages 525–541. Springer Berlin Heidelberg, 2012.
13. S. Thaler, K. Siorpaes, and E. Simperl. Spotthelink: A game for ontology alignment. In *Proceedings of the 6th Conference for Professional Knowledge Management*, 2011.
14. J. Wang, T. Kraska, M. J. Franklin, and J. Feng. Crowder: crowdsourcing entity resolution. *Proc. VLDB Endow.*, 5:1483–1494, July 2012.
15. A. Zaveri, D. Kontokostas, M. A. Sherif, L. Bühmann, M. Morsey, S. Auer, and J. Lehmann. User-driven quality evaluation of dbpedia. In *Proceedings of 9th International Conference on Semantic Systems, I-SEMANTICS '13, Graz, Austria, September 4-6, 2013*. ACM, 2013.
16. A. Zaveri, A. Rula, A. Maurino, R. Pietrobon, J. Lehmann, and S. Auer. Quality assessment methodologies for linked open data. Under review, http://www.semantic-web-journal.net/content/quality-assessment-methodologies-linked-open-data.