Measuring Crowd Truth: Disagreement Metrics Combined with Worker Behavior Filters

Guillermo Soberón[1], Lora Aroyo[1], Chris Welty[2], Oana Inel[1], Hui Lin[3], and Manfred Overmeen[3]

[1] VU University, Amsterdam, The Netherlands,
guillelmo@gmail.com, l.m.aroyo@cs.vu.nl, oana.inel@vu.nl,
[2] IBM Research, New York, USA
cawelty@gmail.com,
[3] IBM Netherlands, Amsterdam, The Netherlands
hui.lin2013@nl.ibm.com, manfred.overmeen@nl.ibm.com

Abstract. When crowdsourcing gold standards for NLP tasks, the workers may not reach a consensus on a single correct solution for each task. The goal of Crowd Truth is to embrace such disagreement between individual annotators and harness it as useful information to signal vague or ambiguous examples. Even though the technique relies on disagreement, we also assume that the differing opinions will cluster around the more plausible alternatives. Therefore it is possible to identify workers who systematically disagree - both with the majority opinion and with the rest of their co-workers- as low quality or spam workers. We present in this paper a more detailed formalization of metrics for Crowd Truth in the context of medical relation extraction, and a set of additional filtering techniques that require the workers to briefly justify their answers. These explanation-based techniques are shown to be particularly useful in conjunction with disagreement-based metrics, and achieve 95% accuracy for identifying low quality and spam submissions in crowdsourcing settings where spam is quite high.

Keywords: crowdsourcing, disagreement, quality control, relation extraction

1 Introduction

The creation of gold standards by expert annotators can be a very slow and expensive process. When it comes to NLP tasks, like relation extraction, annotators have to deal with the ambiguity of the expressions in the text in different levels, frequently leading to disagreement between annotators. To overcome this, detailed guidelines for annotators are developed, in order to handle the different cases that have been observed, through practice, to generate disagreement. However, the process of avoiding disagreement has lead in many cases to brittleness and over generality in the ground truth, making it difficult to transfer annotated data between domains or to use the results for anything practical.

In comparison with expert generated ground truth, crowdsourcing gold standard can be a cheaper and more scalabe solution. Crowdsourced gold standards

typically show lower overall κ scores [3], especially for complex NLP tasks such as relation extraction, since the workers perform small, simple (micro) tasks and cannot be relied on to read a long guideline document. Rather than eliciting an artificial agreement between workers, in [1] we presented "Crowd Truth", a crowdsourced gold standard technique that, instead of considering the lack of agreement something to be avoided, it is used as something informative from which characteristics and features of the annotated content may be inferred. For instance, a high disagreement for a particular sentence may be a sign of ambiguity in the sentence.

As the final Crowd Truth is a by-product of the different contributions of the members of the crowd, being able to identify and filter possible low quality contributors is crucial to reduce their impact on the overall quality of the aggregate result. Most of the existing approaches for detecting low quality contributions in crowdsourcing tasks are based on the assumption that for each task there is a single correct answer, enabling distance and clustering metrics to detect outliers [14] or using gold units [15], establishing an equivalency between disagreement with the majority and low quality contributions.

For Crowd Truth the initial premise is that there is not only one right answer, and the diversity of opinions is to be preserved. However, disagreement with the majority can still be used as a way to distinguish low quality annotators.For each task, it may be assumed that the workers answers will be distributed among the possible options, with the most plausible answers concentrating the highest number of workers, and the improbable answers being stated by none or very few workers. That way, workers whose opinions are different from those of the majority, are likely to find other workers with similar views over the issue. On the other hand, the answers of workers who complete the task randomly or without understanding the task or its content, tend to be not aligned with those of the rest. Hence, it would be possible to filter by identifying those workers who, not only disagree with the majority opinion of the crowd on a task basis, but whose opinions are systematically not shared by many of their peers. The initial definition of the content-based disagreement metrics was introduced by [1] to identify and filter low quality workers for relation extraction tasks, establishing metrics for the inter-worker agreement and the agreement with the crowd opinion.

While filtering workers by disagreement has showed to be an effective way of detecting low quality contributors, achieving high precision, we demonstrate that it is not sufficient to filter all the existing ones. We have extended the relation extraction task by asking the workers to provide a written justification for their answers, and the manual inspection of the results contained several instances of badly formed, incomplete or even random-text explanations, which can be securely attributed to low quality workers or even automated spam bots.

In order to complement the disagreement filters, we propose several ways to use the explanations provided by the contributors, to implement new low quality worker filters that extend and complement the recall of the disagreement filter.

2 Related Work

2.1 Disagreement

In the absence of gold standard, a different evaluation schemes can be used for worker quality evaluation. For instance, the results among workers can be compared and the agreement in their responses can be used as quality estimator.

As is well known [4], the frequency of disagreement can be used to estimate worker error probabilities. In [9] the computation of quality estimators for workers quality based on disagreement is proposed as part of a set of techniques to evaluate workers, along with confidence intervals for each one of this schemas; which allows to estimate the "efficiency" of each one of them.

A simpler method is proposed in [16], which assumes "plurality of answers" for a task, and estimates the quality of a worker based on the number of tasks for which a worker agrees with "the plurality answer" (i.e the one from the majority of the workers).

While these disagreement-based schemas do not rely on the assumption that there is only one single answer per task (thus, allowing room for disagreement between workers responses), they still assume a correlation between disagreement and low quality of the worker. Crowd Truth not only allows but **fosters** disagreement between the workers, as it is considered informative.

2.2 Filtering by explanations

As stated in [6], cheaters tend to avoid tasks that involve creativity and abstract thinking, and even for simple straightforward tasks, the addition of the non-repetitive elements discourage low quality contribution and automation of the task. Apart from the dissuasive element for spammers of introducing these non-repetitive elements in the task design, our work additionally tries to use this as a base for filtering once the task is completed.

Previous experiences [12] have shown that workers tend to provide good answers to open-ended questions when those are concrete, and response length can be used as an indicator of the participant engagement in the task.

2.3 Crowd Watson

Watson [7] is an artificial intelligent system capable of answering questions posed in natural language designed by IBM. To build its knowledge base Watson was trained on a series of databases, taxonomies, and ontologies of publicly available data [10]. Currently, IBM Research aims at adapting the Watson technology for question-answering in the medical domain. For this, large amounts of training and evaluation data (ground truth medical text annotation) are needed, and the traditional ground-truth annotation approach is slow and expensive, and constrained by too restrictive annotation guidelines that are necessary to achieve good inter-annotator agreement, which result in the aforementioned over generalization.

The Crowd Watson project [11] implements the Crowd truth approach to generate a crowdsourced gold standard for training and evaluation of IBM Watson NLP components in the medical domain. Complementary to the Crowd truth

implementation, and within the general Crowd Watson architecture, a gaming approach for crowdsourcing has been proposed [5], as a way to enhance engagement of the experts annotators.

Also, within the context of the Crowd Watson project, [8] has shown how the worker metrics initially set up for the medical domain can be adapted to other domains and tasks, such as event extraction.

3 Representation

CrowdFlower workers were presented sentences with the argument words highlighted and 12 relations (manually selected from UMLS [2]) as shown below in Fig 1; they were asked to choose all the relations from the set of 12 that related the two arguments in the sentence. They were also given the options to indicate that the argument words were not related in the sentence (NONE), or that the argument words were related but not by one of the 12 relations (OTHER). They were also asked to justify their choices by *selecting the words* in the sentence that they believed "signaled" the chosen relations or, in case they chose NONE or OTHER, provide the *rationale* for that decission.

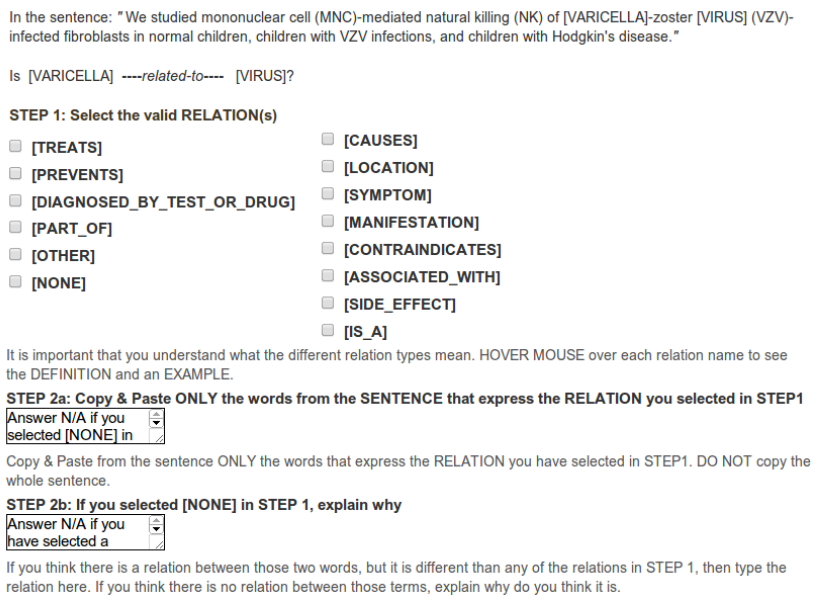

Fig. 1. Relation Extraction Task Example

Note, that the process and the choices for setting up the annotation template is out of scope for this paper. Relation extraction task is part of the larger crowdsourcing framework, Crowd-Watson, which defines the input text, the templates

and the overall workflow [1]. In this paper we only focus on the representation and analysis of the collected crowdsourced annotations.

The information gathered from the workers is represented using vectors in which components are all the relations given to the workers (including the choices for NONE and OTHER). All metrics are computed from three vector types:

1. *worker-sentence vector* $V_{s,i}$ The result of a single worker annotating a single sentence. For each relation that the worker annotated in the sentence, there is a 1 in the corresponding component, otherwise a 0.
2. *sentence vector* V_s The vector sum of the worker-sentence vectors for each sentence $V_s = \sum_i V_{s,i}$
3. *relation vector* R_i A unit vector in which only the component for relation i is 1, the rest 0.

We collect two different kinds of information: the annotations and the explanations about the annotations (i.e the *selected words* that signal the chosen relation, or their *rationale* for selecting NONE or OTHER).

We try to identify behaviour that can be associated with low quality workers from the perspective of these two domains: *disagreement metrics* rely on the content of the annotations to identify workers that systematically disagree with the rest; *explanation filters* aim at identifying individual behaviours that can be attributed to spammers or careless workers.

4 Disagreement metrics

As with the semiotic triangle [13], there are three parts to understanding a linguistic expression: the sign, the thought or interpreter, the referent. We instrument the crowdsourcing process in three analogous places: the micro-task, for the relation extraction case this is a sentence; the workers, who interpret each sentence; the task semantics, in the case of relation extraction this is the intended meaning of the relations.

4.1 Sentence Metrics

Sentence metrics are intended to measure the quality of sentences for the relation extraction task. These measures are our primary concern, we want to provide the highest quality of training data to machine learning systems.

Sentence-relation score is the core crowd truth metric for relation extraction, it can be viewed as the probability that the sentence expresses the relation. It is measured for each relation on each sentence as the cosine of the unit vector for the relation with the sentence vector: $srs(s, r) = cos(V_s, R_r)$

The relation score is used for training and evaluation of the relation extraction system. This is a fundamental shift from the traditional approach, in which sentences are simply labelled as expressing, or not, the relation, and presents new challenges for the evaluation metric and especially for training.

Sentence clarity is defined for each sentence as the max sentence-relation score for that sentence: $scs(s) = max_r(srs(s, r))$

If all the workers selected the same relation for a sentence, the max relation score will be 1, indicating a clear sentence.

Sentence clarity is used to weight sentences in training and evaluation of the relation extraction system, since annotators have a hard time classifying them, the machine should not be penalized as much for getting it wrong in evaluation, nor should it treat such training examples as exemplars.

4.2 Worker Metrics

Worker metrics are primarily to establish worker quality; low quality workers and spammers should be eliminated as they contribute only noise to the disagreement scores, and high quality workers may get paid more as an incentive to return. We investigated several dimensions of worker quality for the relation extraction task:

Number of annotations per sentence is a worker metric indicating the average number of different relations per sentence used by a worker for annotating a set of sentences. Unambiguous sentences should ideally be annotated with one relation, and generally speaking each worker interprets a sentence their own way, but a worker who consistently annotates individual sentences with multiple relations usually does not understand the task.

Worker-worker agreement is the asymmetric pairwise agreement between two workers across all sentences they annotate in common:

$$wwa(w_i, w_j) = \frac{\sum_{s \in S_{i,j}} RelationsInCommon(w_i, w_j, s)}{\sum_{s \in S_{i,j}} NumAnnotations(w_i, s)}$$

where $S_{i,j}$ is the subset of all sentences S annotated by both workers w_i and w_j, $RelationsInCommon(w_i, w_j, s)$ is the number of identical annotations (relations selected) on a sentence between the two workers, and $NumAnnotations(w_i, s)$ is the number of annotations by a worker on a sentence.

Average worker-worker agreement is a worker metric based on the average worker-worker agreement between a worker and the rest of workers, weighted by the number of sentences in common. While we intend to allow disagreement, it should vary by sentence. Workers who consistently disagree with other workers usually do not understand the task:

$$avg\_wwa(w_i) = \frac{\sum_{j \neq i} |S_{i,j}|.wwa(w_i, w_j)}{\sum_{i \neq j} |S_{i,j}|}$$

Worker-sentence similarity is the vector cosine similarity between the annotations of a worker and the aggregated annotations of the other workers in a sentence, reflecting how close the relation(s) chosen by the worker are to the opinion of the majority for that sentence. This is simply $wss(w_i, s) = cos(V_s - V_{s,i}, V_{s,i})$

Worker-sentence disagreement is a measure of the quality of the annotations of a worker for a sentence. It is defined, for each sentence and worker, as the difference between the Sentence Clarity (q.v. above) for the sentence and the worker sentence similarity for that sentence: $wsd(w_i, s) = scs(s) - wss(w_i, s)$. Workers who differ drastically from the most popular choices will have large disagreement scores, workers who agree with the most popular choice will score 0.

The intuition for using the difference from the clarity score over the cosine similarity, as originally proposed in [1], is to capture worker quality on a sentence compared to the quality of the sentence itself. In uni-modal cases, e.g. where a sentence has one clear majority interpretation, the cosine similarity works well, but in the case where a sentence has a bimodal distribution, e.g. multiple popular interpretations, the worker's cosine similarity will not be very high even for those that agree with one of the two most popular interpretations, which seems less desireable.

Average worker-sentence disagreement is a worker metric based on the average worker-sentence disagreement score across all sentences, $avg\_wsd(w_i) = \frac{\sum_{s \in S_i} wsd(w_i,s)}{|S_i|}$ where S_i is the subset of all sentences annotated by worker w_i.

The worker-worker and worker-sentence scores are clearly similar, they both measure deviation from the crowd, but they differ in emphasis. The wsd metric simply measures the average divergence of a worker from the crowd on a sentence basis, someone who tends to disagree with the majority will have a low score. For wwa, workers who may not always agree with the crowd on a sentence basis might be found to agree with a group of people that disagree with the crowd in a similar way, and would have a low score. This could reflect different cultural or educational perspectives, as opposed to simply a low quality worker.

4.3 Relation Metrics

Relation clarity is defined for each relation as the max sentence-relation score for the relation over all sentences:

$rcs(r) = max_s(srs(s, r))$

If a relation has a high clarity score, it means that it is at least possible to express the relation clearly. We find in our experiments that a lot of relations that exist in structured sources are very difficult to express clearly in language, and are not frequently present in textual sources. Unclear relations may indicate unattainable learning tasks.

5 Explanation filters

In [1] we showed results using the worker metrics to detect low quality workers. In order to evaluate our results, we had workers justify their answers. The explanations of the annotation tasks are not strictly necessary for the crowd truth data, and represent additional time and therefore cost to gather. In this section we analyze the value of this information.

We examined whether this additional effort dissuaded workers from completing the task. Two different implications are to be distinguished for this circumstance: one positive, by driving away low quality workers or spammers -whose main objective is to maximize its economic reward with the minimum possible effort-; and one negative, as it may induce some good contributors to choose easier, less demanding tasks. In order to prevent this, it might be necessary to increase the economic reward to make up for the extra effort, so, at the end, the addition of explanations implies an increase in the task price. And, finally, we want to test whether the explanations -apart from preventing low quality workers to complete the task- may contain information that it is useful for detecting low quality workers.

Apart from the presence of explanations, another variable to take into account for spam detection is the *channel* of the workers. CrowdFlower has over 50 labor channel partners or external labor of workers, such as Amazon Mechanical Turk and TrialPay, which can be used (individually or combined) to run crowdsourcing processes. Our intuition was that different channels have different spam control mechanisms, which may redound in different spammer ratios, depending on the channel.

To explore these variables, we set up an experiment to annotate the same 35 sentences, over different configurations:

1. Without explanations, using workers from multiple Crowdflower channels
2. Without explanations, using workers from Amazon Mechanical Turk (AMT)
3. With explanations, using workers from multiple Crowdflower channels
4. With explanations, using workers from AMT

Note that AMT was among the multiple channels used on 1 and 3, but the presence of workers from AMT was minority.

By comparing the pairs formed by 1 and 2, and 3 and 4, we can test whether the channel has any influence in the low quality worker ratio. Likewise, the pairs formed by 1 and 3, and 2 and 4, can be used to test the influence of the explanations, independently of the channel used.

We collected 18 judgments per sentence (for a total of 2522 judgements), and workers were allowed to annotate a maximum of 10 different sentences. The number of unique workers per batch was comprehended between 67 and 77 workers.

In the results we observed that the time to run the task using multiple channels was significantly lower than doing so only on AMT, independently of whether the explanations where required or not. The time invested on annotating a sentence of the batch was substantially lower, on average, when explanations were not required.

The number of workers labelled as possible low quality workers by the disagreement filters was low, and more or less was kept within the same range for the four batches (between 6 and 9 filtered workers per batch); so we cannot infer whether including explanations discourages low quality workers from working in it.

However, manual exploration of the annotations revealed four patterns that may be indicative of possible spam behaviour:

1. **No valid words** (No Valid in Table 1) were used, either on the explanation or in the selected words, using instead random text or characters.
2. Using the **same text for both the explanation and the selected words** (Rep Resp in Table 1). According to the task definition, both fields are exclusive: either the explanation or the selected words that indicate the rationale of the decision are to be provided, so filling in both may be due bad understanding of the task definitions. Also, both are semantically different reasons, so it is unlikely that the same text is applicable for both.
3. Workers that **repeated the same text** (Rep Text in Table 1) for all their annotations, either justifying their choice using the exact same words or selecting the same words from the sentence.
4. **[NONE] and [OTHER] used with other relations** (None/Other in Table 1). None and Other are intended to be exclusive: according to the task definition, by selecting them the annotator is stating that none of the other relations is applicable for the sentence. Hence, it is semantically incorrect to choose [NONE] or [OTHER] in combination with other(s) relations, and doing so may reflect a bad understandment of the task definition.

The degree to which these patterns may indicate spam behaviour is different: in most cases, "No valid words" is a strong indicator of a low quality worker, while a bad use of [NONE] or [OTHER] may be the reflection of a bad understanding of the task (i.e. when should one text box be filled and when the other), rather than a bad worker.

Chan.	**Disag. Filters** (# Spam)	**Explanation filters** # Spam - (% Overlap w/ disagr. filters)				**# Spam exclusively detected by exp. filters**
		None / Other	**Rep Resp**	**Rep Text**	**No Valid**	
Multiple	9	7 (29%)	14 (29%)	3 (33%)	11 (36%)	18
AMT	6	9 (22%)	2 (0%)	2 (50%)	1 (0%)	11

Table 1. Results from 35 Sentences with explanation-based filters

Table 1 contains an overview of the number of occurrences in the batches with explanations of each of the previous patterns. For each pattern, the percentage of workers identified as low quality workers is indicated. This percentage and the last column -which indicates the number of workers for which at least one of the low quality patterns have been observed but are not labelled as low quality by the disagreement filters- shows that there is little overlap between these patterns and what the disagreement filters considers low quality "behaviour". Therefore, it seems reasonable to further explore the use of this patterns as "explanation filters" for low quality workers. Also, the number of potential low

quality workers according to the spam patterns, seems bigger when the task is run on multiple channels rather than only on AMT. This observation cannot be considered conclusive, but it seems reasonable to explore it further.

6 Experiments

We designed a series of experiments to gather evidence in support of our hypothesis that the disagreement filters may not be sufficient and that the explanations can be used to implement additional filters to improve the spam detection.

6.1 Data

The data for the main experiments consist of two different sets of 90 sentences. The first set (Experiment 2 or EXP2) is annotated only by workers from Amazon Mechanical Turk (AMT), and the second (Experiment 3 or EXP3) is annotated by workers from multiple channels among those offered by Crowdflower (including AMT, though the AMT workers were a minority).

To optimize the time and worker dynamics we split the 90 sentences sets in batches of 30 sentences. The batches of the first set were run on three different days, and the batches of the second were all run on the same day. Workers were not allowed to annotate more than 10 sentence in the first set, and no more than 15 in the second. We collected 450 judgments (15 per sentence) in each batch (for a total of 1350 per set), from 143 unique workers in the first set and 144 in the second.

From our previous experiences, judgements from workers who annotated two or fewer sentences were uninformative, so we have removed these leaving 110 and 93 workers and a total of 1292 and 1302 judgements on each set.

We have manually gone through the data and identified low quality workers from their answers. 12 workers (out of 110) were identified as low quality workers for EXP2 and 20 (out of 93) for EXP3. While all the spammers on EXP2 were identified as such by the disagreement filters, only half of the low quality workers in EXP3 were detected.

Also it is important to notice that the number of annotations by workers identified spammers is much higher for EXP3 (386 out of 1291, 30%) than for EXP2 (139 out of 1302, 11%).

6.2 Filtering low quality workers

In this section, we address our hypotheses by, first, describing disagreement performance for EXP3, it is shown how it is not sufficient by itself; and, second, showing how the explanation filters are informative and disjoint from the disagreement filters (they indicate something, and that "something" is different from what disagreement points to).

A sense of the different disagreement metrics in detecting low quality workers is shown in Figure 2 and 3. Each metric is plotted against overall accuracy at different confidence thresholds, on each experiment. Clearly, the overall accuracy of the disagreement metrics is lower for EXP3. While it is possible to achieve a 100% accuracy for EXP2 by linearly combinining the disagreement metrics, only 89% is achieved for EXP3 by this means.

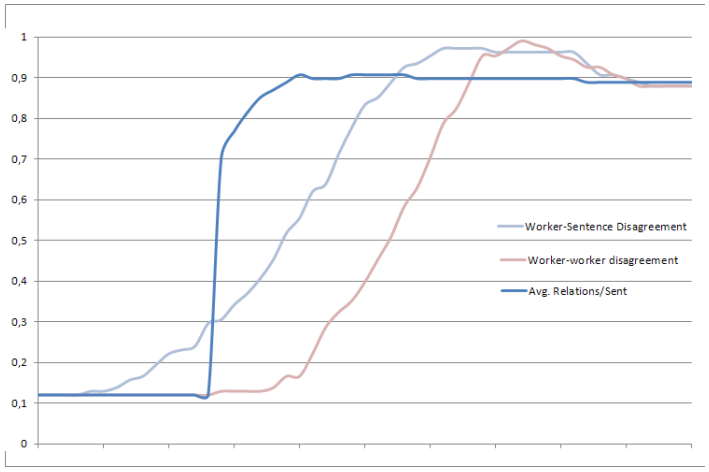

Fig. 2. Accuracy of worker metrics for predicting low quality workers at all thresholds, for Experiment 2

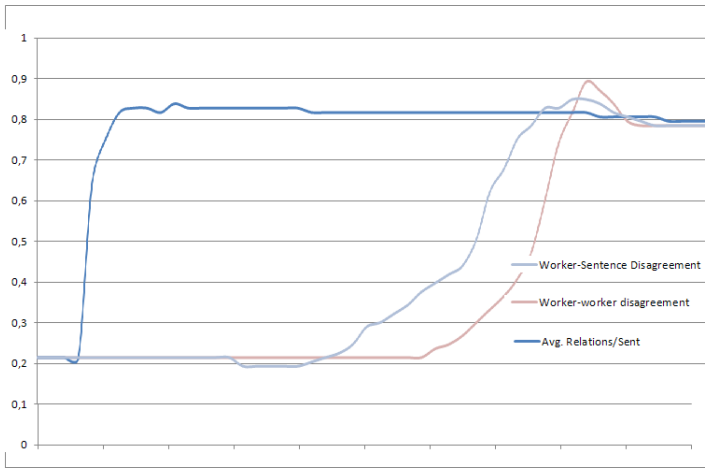

Fig. 3. Accuracy of worker metrics for predicting low quality workers at all thresholds, for Experiment 3

In order to make up for this, we analyzed the explanations filters, exploring whether they provide some information about possible spammer behaviour that it is not already contained in the disagreement metrics. The explanation filters are not very effective by themselves: their recall value is pretty low (in all cases, below 0.6), and it is not substantially improved by combining them.

The tables 2 and 3 present an overview of the workers identified as possible spammers by each filter, reflecting the intersections and differences between the disagreement filters and the explanation filters.

Note that we analyze the experiments both on a "job" basis, and on an aggregate "Experiment" basis. This displays how jobs are more or less "homogeneous" (for instance, that one of them is not clearly biased in one particular batch, therefore, biasing the aggregated experiment). However, for filtering purposes, we treat the experiments as atomic units.

Exp.	**Disag. Filters** (# Spam)	**Explanation filters** # Spam - (% Overlap w/ disagr. filters)				**# Spam exclusively detected by exp. filters**
		None / Other	**Rep Resp**	**Rep Text**	**No Valid**	
Batch 2.1	8	5 (40%)	2 (100%)	4 (75%)	0	4
Batch 2.2	6	3 (33%)	3 (66%)	0	0	3
Batch 2.3	5	2 (0%)	4 (25%)	0	1 (0%)	6
Total Exp 2	**12**	**10 (20%)**	**7 (43%)**	**4 (75%)**	**1 (0%)**	**14**

Table 2. Filters Overview - Experiment 2 (AMT)

It can be observed how the overlap (i.e. the number of workers identified as possible spammers by two different filters) between the disagreement filters and each of the explanation filters is not really significative.

Exp.	**Disag. Filters** (# Spam)	**Explanation filters** # Spam - (% Overlap w/ disagr. filters)				**# Spam exclusively detected by exp. filters**
		None / Other	**Rep Resp**	**Rep Text**	**No Valid**	
Batch 3.1	4	6 (33%)	7 (57%)	5 (20%)	6 (17%)	13
Batch 3.2	4	11 (18%)	0	7 (14%)	6 (33%)	15
Batch 3.3.	6	8 (37.5%)	0	5 (60%)	1 (0%)	8
Total Exp 3	**10**	**22 (18%)**	**8 (37%)**	**14 (36%)**	**12 (42%)**	**30**

Table 3. Filters Overview - Experiment 3 (Multiple channels)

On the other hand, the number of workers identified as possible spammers exclusively by the explanation filters is quite big for the EXP3. Not only it's

higher than for EXP2, but also in comparison with the number of workers filtered by the disagreement filters. This is coherent with the manual identification of spammers, which revealed 26 spammers.

7 Results and future work

By linearly combining the filters, we have obtained a classifier with 95% accuracy and F-measure 0.88, improving the disagreement-only filtering (88% accuracy and F-measure 0.66) for EXP3. More data is needed to improve and rigorously validate this approach, but this initial results are already promising.

This linear combination of filters serves to the purpose of complementing disagreement filters with explanation filters. In future work, we will further explore different ways of combining these filters to improve quality, such as bagging.

For the current implementation, we have omitted the differences in the prediction power of each of the explanation filters, when it can be resonably assumed that they are not equally good indications of spam behaviour. It is also worth considering using a boosting approach to improve this.

Also, disagreement filters may be complemented by other kinds of information. For instance, for EXP3, the workers completing the task come from different channels. In future work, we will further explore whether the worker provenance is a significative toward low quality detection.

While sentence and worker metrics have proven to be informative, the available data is not sufficient to reach similar conclusions for the relation metrics, as the different relations are unevenly represented. We will try to collect more data in order to further explore this metrics.

8 Conclusions

We presented formalizations of sentence and worker metrics for Crowd Truth, and showed how the worker metrics could be used to detect low quality workers. We then introduced a set of explanation-based filters based on workers justification of their answers, and we ran experiments on various crowdsourcing "channels".

The conducted experiments seem to indicate that, when in presence of a small number of low quality **annotations**, disagreement filters are sufficient to preseve data quality. On the other hand, in the presence of a higher number of low quality annotations, the effectivity of disagreement filters diminishes, and are not enough to detect all the possible low quality contributions.

We have showed how the explanations provided by the workers about their answers can be used to identify patterns that can reasonably associated with spamming or low quality annotation behaviours. We used these these patterns combined with the worker metrics to detect low quality workers with 95% accuracy in a small cross-validation experiment.

References

1. Lora Aroyo and Chris Welty. Crowd truth: Harnessing disagreement in crowdsourcing a relation extraction gold standard. In *Proc. Websci 2013*. ACM Press, 2013.

2. Olivier Bodenreider. The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic acids research*, 32(suppl 1):D267–D270, 2004.
3. Jacob Cohen et al. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.
4. Alexander Philip Dawid and Allan M Skene. Maximum likelihood estimation of observer error-rates using the em algorithm. *Applied Statistics*, pages 20–28, 1979.
5. Anca Dumitrache, Lora Aroyo, Chris Welty, and Robert-Jan Sips. Dr. Detective: combining gamificationtechniques and crowdsourcing to create a goldstandard for the medical domain. Technical report, VU University Amsterdam, 2013.
6. Carsten Eickhoff and Arjen P. de Vries. Increasing cheat robustness of crowdsourcing tasks. *Inf. Retr.*, 16(2):121–137, April 2013.
7. David Ferrucci, Eric Brown, Jennifer Chu-Carroll, James Fan, David Gondek, Aditya A. Kalyanpur, Adam Lally, J. William Murdock, Eric Nyberg, John Prager, Nico Schlaefer, and Chris Welty. Building watson: An overview of the deepqa project. *AI Magazine*, 31:59–79, 2010.
8. Oana Inel, Lora Aroyo, Chris Welty, and Robert-Jan Sips. Exploiting Crowdsourcing Disagreement with Various Domain-Independent Quality Measures. Technical report, VU University Amsterdam, July 2013.
9. Manas Joglekar, Hector Garcia-Molina, and Aditya Parameswaran. Evaluating the crowd with confidence. 2012.
10. Aditya Kalyanpur, BK Boguraev, Siddharth Patwardhan, J William Murdock, Adam Lally, Chris Welty, John M Prager, Bonaventura Coppola, Achille Fokoue-Nkoutche, Lei Zhang, et al. Structured data and inference in DeepQA. *IBM Journal of Research and Development*, 56(3.4):10–1, 2012.
11. Hui Lin. Crowd Watson: Crowdsourced Text Annotations. Technical report, VU University Amsterdam, July 2013.
12. C. Marshall and F. Shipman. Experiences surveying the crowd: Reflections on methods, participation, and reliability. In *Proc. Websci 2013*. ACM Press, 2013.
13. C.K. Ogden and I. A. Richards. The meaning of meaning: A study of the influence of language upon thought and of the science of symbolism. *8th ed. 1923. Reprint New York: Harcourt Brace Jovanovich*, 1923.
14. Vikas C. Raykar and Shipeng Yu. Eliminating spammers and ranking annotators for crowdsourced labeling tasks. *J. Mach. Learn. Res.*, 13:491–518, March 2012.
15. Cristina Sarasua, Elena Simperl, and Natalya Fridman Noy. Crowdmap: Crowdsourcing ontology alignment with microtasks. In *International Semantic Web Conference (1)*, pages 525–541, 2012.
16. Petros Venetis and Hector Garcia-Molina. Quality control for comparison microtasks. In *Proceedings of the First International Workshop on Crowdsourcing and Data Mining*, pages 15–21. ACM, 2012.