Autonomous Crowdsourcing through Human-Machine Collaborative Learning

Azad Abad[†], Moin Nabi[†], Alessandro Moschitti
†DISI, University of Trento, 38123 Povo (TN), Italy
Qatar Computing Research Institute, HBKU, 34110, Doha, Qatar
{azad.abad, moin.nabi}@unitn.it, amoschitti@gmail.com

ABSTRACT

In this paper, we introduce a general iterative human-machine collaborative method for training crowdsource workers: a classifier (i.e., the machine) selects the highest quality examples for training crowdsource workers (i.e., the humans). Then, the latter annotate the lower quality examples such that the classifier can be re-trained with more accurate examples. This process can be iterated several times. We tested our approach on two different tasks, Relation Extraction and Community Question Answering, which are also in two different languages, English and Arabic, respectively. Our experimental results show a significant improvement for creating Gold Standard data over distant supervision or just crowdsourcing without worker training. Additionally, our method can approach the performance of the state-of-the-art methods that use expensive Gold Standard for training workers.

CCS CONCEPTS

•**Information systems** →Retrieval models and ranking; Learning to rank; *Question answering;* •**Computing methodologies** →*Learning paradigms;* Active learning settings;

KEYWORDS

Crowdsourcing, Self-training, Human in the Loop, Relation Extraction, Community Question Answering

ACM Reference format:
Azad Abad[†], Moin Nabi[†], Alessandro Moschitti
†DISI, University of Trento, 38123 Povo (TN), Italy
Qatar Computing Research Institute, HBKU, 34110, Doha, Qatar
{azad.abad, moin.nabi}@unitn.it, amoschitti@gmail.com . 2017. Autonomous Crowdsourcing through Human-Machine Collaborative Learning. In *Proceedings of The 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Tokyo, Japan, August 2017 (SIGIR'17),* 4 pages.
DOI: 10.475/123_4

1 INTRODUCTION

The advent of machine learning in Information Computer Technology and related disciplines has generated a big deal of need for training data. Crowdsourcing has become on the main methods to acquire annotated data in short time at a reasonable cost. For example, Snow et al. [17] demonstrated that crowd workers are able to generate high quality labels for various NLP tasks. However, when the annotation task is complex, crowd workers require extensive training in order to produce enough accurate labels but this is not often practically possible. For example, in case of semantically complex tasks such as Relation Extraction (RE), the annotators need to receive accurate instruction. Indeed, several papers have shown that only a marginal improvement can be achieved via crowdsourcing the data for RE task over weakly supervised methods [2, 14, 22]. It should be noted that such work did not apply the well-known mechanism of quality control based on Gold Standard labels for training annotators.

Very recently, despite the previous results, Liu et. al [8] showed a larger improvement for RE task, by training crowd workers in an interactive tutorial procedure called "Gated Instruction". This approach, however, requires a set of high-quality labeled data for providing the instruction and feedback to the crowd workers. To overcome this critical limitation, we have recently proposed to automatically create high-quality annotated data for RE, using task classifiers trained on distant supervised (automatic data), and use it to train crowd workers [1].

In this paper, we show the generality of our iterative human-machine co-training framework: its main idea is to select a subset of *more reliable* weakly supervised examples using an automatic system to train the annotators. The educated crowd workers can then provide higher quality annotations, which the system can use in the next iteration to improve its classification accuracy. This loop gradually improves both machine and human annotators.

We demonstrate that our approach works for different tasks and in different languages, e.g., English and Arabic. For the former, we evaluated our proposed method on the well-known corpus for RE task, TAC-KBP. For the latter language, we evaluated our method on a Community Question Answering (CQA) task, designed for measuring Question-Question similarity using the data collected from several medical forums, i.e., the SemEval 2016-17 task D [10, 11]. In both cases, our study shows that even without using any Gold Standard data for training the workers, we can still achieve comparable results with more costly state-of-the-art methods.

Our study opens up avenues for exploiting new inexpensive crowdsourcing solutions to achieve performance gain in crowdsourcing tasks of Information Retrieval, NLP and other disciplines.

SIGIR'17, Tokyo, Japan

123-4567-24-567/08/06…$15.00
DOI: 10.475/123_4

2 BACKGROUND WORK

There is a large body of work on DS for RE, but we only discuss those most related to our work and refer the reader to other recent works [4, 7, 9, 12, 15, 18, 21].

Many researchers have exploited the techniques of combining the DS data with a small set of human annotated data, collected via crowdsourcing, to improve the relation extractor accuracy [2, 8, 22]. Angeli et al. [2] reported a minor improvement using active learning methods to select the best instances to be crowdsourced. In the same direction, Zhang et al. [22] studied the effect of providing human feedback in crowdsourcing task and observed a minor improvement in terms of F1. At high level, our work may be viewed as employing crowdsourcing for RE. In that spirit, we are similar to these works, but with the main difference of training crowd workers to obtain higher quality annotations. The most related paper to our work is by Liu et al. [8], who trained the crowd workers via *Gated Instruction*. Our study confirms their finding. However, unlike them, we do not employ any Gold Standard (annotated by experts) for training the annotator. Instead, we propose a self-training strategy to select a set of high-quality automatic annotated data (namely, Silver Standard).

Regarding QA, there has been a large body of works using kernels and neural networks [3, 5, 16, 19, 20]. Our approach is model independent and can exploit any accurate system providing a form of confidence score.

3 SELF-CROWDSOURCING TRAINING

In this section, we first explain our proposed method for automatically identifying high-quality examples (i.e., Silver Standard) to train the crowd workers and collect annotations for the lower-quality examples. We then explain the scheme designed for crowd worker training and annotation collection.

3.1 Silver Standard Mining

The main idea of our approach to Self-Crowdsourcing training is to use classifier's score for gradually training the crowd workers, where examples and labels associated with the highest classifier prediction values (i.e., the most reliable) are used as silver standard. More formally, our approach is based on a noisy-label dataset, DS, whose labels are extracted in a distant supervision fashion and a CS dataset to be labeled by the crowd. The first step it to divide CS into three parts: CS_I, which is used to create the instruction for the crowd workers, CS_Q, which used for asking questions about sentence annotations, and CS_A, which is used to collect the labels from annotators, after they have been trained.

To select CS_I, we train a classifier C on DS, and then used it to label CS examples. In particular, for the RE task, we used MultiR framework [6] to train C, as it is a widely used tool for RE. For the CQA task, we did not apply DS as we used the predictions produced by the systems of the SemEval challenge [10]. [1] Then, we sort CS in a descending order according to the classifier prediction scores and select the first N_i elements, obtaining CS_I.

Next, we select the N_q examples of $CS \setminus CS_I$ with highest score to create the set CS_Q. Note that the latter contains highly-reliable classifier annotations but, since the scores are lower than for the CS_I examples, we conjecture that they may be more difficult to be annotated by the crowd workers.

Finally, CS_A is assigned with the remaining examples (i.e., $CS \setminus CS_I \setminus CS_Q$). These have the lowest confidence and should therefore be annotated by crowd workers. N_i and N_q can be tuned on the task, we set both to 10% of the data.

Algorithm 1 Collaborative Crowdsourcing Training

Input: DS, CS, N_i, N_q, $MaxIter$
Output: Trained classifier C_t
$C_0 \leftarrow$ Train *MultiR* on DS
For $t := 1$ **to** $MaxIter$**:**
  $P \leftarrow \emptyset$
  For each $E \in CS$:
    Compute $(E_{relation}, E_{score})$ using C_{t-1}
    $P \leftarrow P \cup \{(E_{relation}, E_{score})\}$
  $CS_{sorted} \leftarrow$ Sort CS using the scores E_{score} in P
  $CS_I \leftarrow N_i$ topmost elements in CS_{sorted}
  $CS_Q \leftarrow N_q$ topmost elements in $\{CS_{sorted} \setminus CS_I\}$
  $CS_A \leftarrow$ remaining elements in $\{CS_{sorted} \setminus CS_I \setminus CS_Q\}$
  User Instruction using CS_I
  Interactive QA using CS_Q
  $T_{CS} \leftarrow$ *Crowdsourcing* CS_A
  $C_t \leftarrow$ Train *MultiR* on $\{DS \cup T_{CS}\}$

3.2 Training Schema

We conducted *crowd worker training* and *annotation collection* using the well-known Crowdflower platform[2]. Given CS_I and CS_Q (see Section 3.1), we train the annotators in two steps:

(i) **User Instruction**: first, a definition of each label type, taken from the official guideline, is shown to the annotators., i.e., relation type from TAC-KBP for annotating the RE task or question similarity type from SemEval for the CQA task. This initial training step provides the crowd workers with a big picture of the task. We then train the annotators showing them a set of examples from CS_I. The latter are presented in order of difficulty level. The ranked list of examples provided by our self-training strategy facilitates the gradual educating of the annotators [13]. This gives us a benefit of training the annotators with any level of expertise. It is an important aspects to carry out effective crowdsourcing as there are in general no clues about the workers' expertise in advance.

(ii) **Interactive QA**: after the initial step, we challenge the workers in an interactive QA task with multiple-choice questions over the sentence annotation. To accomplish that, we adapted an interactive java-based agent[3] that can provide feedback for crowd workers: it corrects their mistakes by knowing their given answer and also the correct answer provided by the classifier. Then, the feedback is enriched with a shallow level of rule-based reasoning to help the crowd workers revise their mistakes. [4] Note that: (a) to have a better control of the worker training, we performed a selection of the sentences in CS_Q for questioning in a category-wise fashion. Meaning that, we select the subsets of examples for each

[1] An interesting future experiment regards the use of pseudo-relevance feedback for training the classifiers.

[2] www.crowdflower.com

[3] https://www.smore.com/clippy-js

[4] We carried out this step only for the RE task as the annotation task for it is more difficult than for CQA, which only requires to judge if a question is related to another.

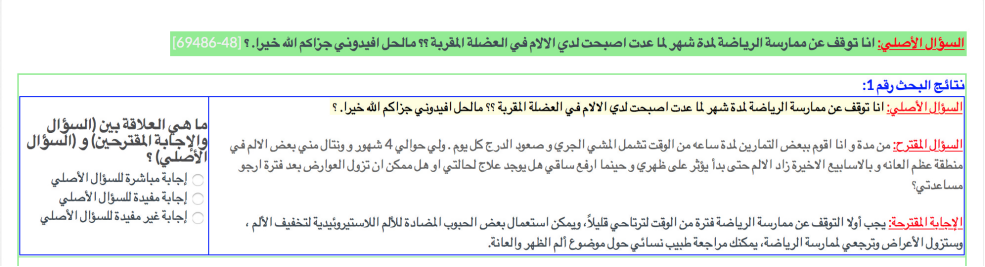

Figure 1: User Interface of the Arabic QA crowdsourcing task

class of relation separately. Specifically to the RE task, we observed in practice that initially a lot of examples are classified as "No Relation". This is due to a difficulty of the task for the DS-based model. Thus, we used them anyway in CS_A. The entire data generation and crowd training procedure is formalized by Algorithm 1.

4 EXPERIMENTS

We evaluated our proposed method for annotating data for RE and CQA tasks, by measuring the performance of the crowd-workers in terms of the quality of their annotation. Additionally, we provide an indirect evaluation of our approach by measuring its impact on the RE system. In the following, we first introduce the details of the used corpora, then explain the feature extraction, RE pipeline and the impact of our approach, and finally present the evaluation of the crowd annotation for both RE and CQA data.

4.1 Data Preparation

We utilized two different corpora to evaluate the crowd workers performance trained with our proposed method. The first, TAC-KBP, is used for RE, while the second, is used for CQA.

4.1.1 RE Corpus. We used TAC-KBP newswires, one of the most well-known corpus for both RE task and the quality of annotations. As *DS*, we selected 700K sentences automatically annotated using Freebase as an external KB. We used the active learning framework proposed by Angeli et al. [2] to select *CS*. This allowed us to select the best sentences to be annotated by humans (SampleJS). As a result, we obtained 4,388 sentences. We divided the *CS* sentences in CS_I, CS_Q and CS_A, with 10%, 10% and 80% split, respectively. We requested at least 5 annotations for each sentences. Similarly to [8], we restricted our attention to 5 relations between *person* and *location*. For both *DS* and *CS*, we used the publicly available data provided by Liu et al. [8]. Ultimately, 221 crowd workers participated to the task with minimum 2 and maximum 400 annotations per crowd worker. To evaluate our model, we randomly selected 200 sentences as test set and asked domain experts for manually tagging such data using TAC-KBP annotation guidelines.

4.1.2 CQA Corpus. We used this corpora to evaluate the crowd workers performance directly. We performed an experiment on CQA task in Arabic language, introduced in SemEval 2016-17 [10, 11]. The dataset created by crawling all questions from the "Medical-Questions" section of the Altibbi.com medical site (OrgQ). Then, the Google API were used to retrieved 9 Q/A pairs (RelQ/RelA). Annotators were asked to judge the relation between each pair (RelQ/RelA) and OrgQ to one of the following classes: *Direct, Useful,* and *NotUseful* answer. Among 1400 questions collected from the website, we partitioned the dataset to 5%, 5% as CS_I and CS_Q respectively. For CS_A, we selected 40 questions to be crowdsourced. Figure 1 shows the user interface for crowdsourcing the Arabic question-question similarity task. We compared our results with a strong baseline: the gold standard produced by the competition organizers, considering *Direct* and *Useful* answers as one class in evaluating the crowd workers (according to SemEval guideline).

Model	Pr.	Rec.	F1
DS-only	0.43	0.52	0.47
SampleJS [1]	0.46	0.51	0.48
Gated Instruction [6]	0.53	0.57	0.55
Our Method	0.50	0.54	0.52

Table 1: Evaluation of the impact of the CS_A label quality in the RE task.

4.2 Impact on the Relation Extraction task

We used the relation extractor, MultiR [7] along with lexical and syntactic features proposed by Mintz et al. [9] such as: (i) Part of Speech (POS); (ii) windows of k words around the matched entities; (iii) the sequences of words between them; and (iv) finally, dependency structure patterns between entity pairs. These yield low-recall as they appear in conjunctive forms but at the same time they produce a high precision.

4.2.1 RE Task Evaluation. In the first set of experiments, we evaluated the impact of adding a small set of crowdsourced data to a large set of instances annotated by Distant Supervision. We conducted the RE experiments in this setting, as this allowed us to directly compare with Liu et al. [8]. More specifically, we used CS_A annotated by our proposed method along with the noisy annotated DS to train the extractor. We compared our method with (i) the *DS-only* baseline (ii) the popular active learning based method (a.k.a., *SampleJS* [2]) and also (iii) the state of the art, *Gated Instruction* (GI) strategy [8]. We stress the fact that the same set of examples (both DS and CS) of the above work are used in our experiments: we just replaced their annotations with those collected using our proposed framework. Note that in the *SampleJS* baseline, the annotations were collected through crowdsourcing, but without any explicit crowd workers training stage.

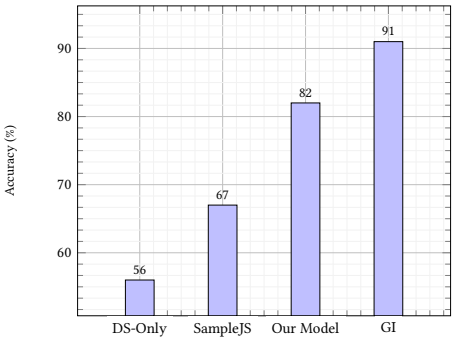

Figure 2: Crowd workers annotation accuracy

The results of our automatic extractor trained with labelled data produced with different approached are shown in Table 1. Our method improves both the *DS-only* and the *SampleJS* baselines by 5% and 4% in F1, respectively. Additionally, our model is just 3% lower in F1 than the GI method. In both our and GI methods, the crowd workers are trained before enrolling in the main task. However, GI trains annotators using a Gold Standard, which involves a higher level of supervision with respect to our method. This suggests that our self-training method is potentially effective and rather inexpensive with respect to GI.

4.3 Crowd-Worker Evaluation

We analyzed the accuracy of the crowd workers in terms of the quality of their annotations.

RE task. We randomly selected 100 sentences from CS_A and then had them manually annotated by an expert. We compared the accuracy of the annotations collected with our proposed approach with those provided by the DS-only baseline, the *SampleJS* baselines, and the GI method. Figure 2 shows that the annotations performed by workers trained with our method are just slightly less accurate than the annotations produced by workers trained with GI.

CQA task. We compared the accuracy of the annotation provided by a single "trained" worker (using our method) with the majority voting over 5 "untrained" workers. For evaluation, we used the ground truth data annotated by the SemEval organizers. Our proposed method achieved 81% accuracy with a single crowd worker, which is almost on a par with the majority voting baseline with 5 different workers. It clearly suggests that training annotators can be an inexpensive replacement for the popular consensus-based filtering scheme.

Additionally, we randomly selected 10 questions and had an expert annotated them. The accuracy of our annotation was 74%. Very interestingly, this is the same accuracy of the annotation provided by the SemEval Gold Standard.

5 CONCLUSIONS

In this paper, we have proposed a self-training strategy for crowdsourcing, as an effective alternative to train annotators with Gold Standard. The main idea is to use noisy data, e.g., collected with distant supervision (DS) techniques, for training automatic classifiers. The latter can be used to select instances associated with high-classification confidence, which is not available in DS data. The high-quality labels can then be used to reliably instruct crowd workers. Our experimental results show that (i) the annotation carried out by workers trained with our approach has a comparable quality than the one obtained with more expensive training approaches; and (ii) our results generalize to different tasks and languages.

We believe that our paper opens several future research directions on the use of automatic classifiers for training crowd workers in a much cheaper way.

ACKNOWLEDGMENTS

This work has been partially supported by the EC project CogNet, 671625 (H2020-ICT-2014-2, Research and Innovation action). We would like to thank Hamdy Mubarak for his precious help in evaluating the quality of our annotation on the Arabic CQA task.

REFERENCES

[1] Azad Abad, Moin Nabi, and Alessandro Moschitti. 2017. Self-Crowdsourcing Training for Relation Extraction. In *ACL*.
[2] Gabor Angeli, Julie Tibshirani, Jean Y. Wu, and Christopher D. Manning. 2014. Combining distant and partial supervision for relation extraction. In *EMNLP*.
[3] Alberto Barrón-Cedeño, Giovanni Da San Martino, Shafiq Joty, Alessandro Moschitti, Fahad A. Al Obaidli, Salvatore Romeo, Kateryna Tymoshenko, and Antonio Uva. 2016. ConvKN at SemEval-2016 Task 3: Answer and Question Selection for Question Answering on Arabic and English Fora. In *Proc. of SemEval '16.*
[4] Razvan C. Bunescu. 2007. Learning to extract relations from the web using minimal supervision. In *ACL 07.*
[5] Giovanni Da San Martino, Alberto Barrón-Cedeño, Salvatore Romeo, Antonio Uva, and Alessandro Moschitti. 2016. Learning to Re-Rank Questions in Community Question Answering Using Advanced Features. In *Proc. CIKM.*
[6] Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer, and Daniel S. Weld. Knowledge-based Weak Supervision for Information Extraction of Overlapping Relations. In *ACL 2011.*
[7] Raphael Hoffmann, Congle Zhang, and Daniel S. Weld. 2010. Learning 5000 Relational Extractors. In *ACL '10.*
[8] Angli Liu, Xiao Ling Stephen Soderland, Jonathan Bragg, and Daniel S Weld. 2016. Effective Crowd Annotation for Relation Extraction. In *ACL.*
[9] Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant Supervision for Relation Extraction Without Labeled Data. In *ACL.*
[10] Preslav Nakov, Doris Hoogeveen, Lluís Màrquez, Alessandro Moschitti, Hamdy Mubarak, Timothy Baldwin, and Karin Verspoor. 2017. SemEval-2017 Task 3: Community Question Answering. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval '17).*
[11] Preslav Nakov, Lluís Màrquez, Alessandro Moschitti, Walid Magdy, Hamdy Mubarak, abed Alhakim Freihat, Jim Glass, and Bilal Randeree. SemEval-2016 Task 3: Community Question Answering. 525–545.
[12] Truc-Vien T. Nguyen and Alessandro Moschitti. 2011. End-to-end Relation Extraction Using Distant Supervision from External Semantic Repositories. In *ACL.*
[13] Robert M Nosofsky. 2011. The generalized context model: An exemplar model of classification. *Formal approaches in categorization* (2011).
[14] Maria Pershina, Bonan Min, Wei Xu, and Ralph Grishman. 2014. Infusion of Labeled Data into Distant Supervision for Relation Extraction. In *ACL.*
[15] Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling Relations and Their Mentions Without Labeled Text. In *ECML.*
[16] Aliaksei Severyn and Alessandro Moschitti. 2015. Learning to rank short text pairs with Convolutional Deep Neural Networks. In *SIGIR.*
[17] Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. 2008. Cheap and Fast—but is It Good?: Evaluating Non-expert Annotations for Natural Language Tasks. In *EMNLP.*
[18] Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D. Manning. 2012. Multi-instance Multi-label Learning for Relation Extraction. In *EMNLP.*
[19] Kateryna Tymoshenko, Daniele Bonadiman, and Alessandro Moschitti. 2016. Learning to Rank Non-Factoid Answers: Comment Selection in Web Forums. In *Proc. of CIKM 2016.*
[20] Kateryna Tymoshenko and Alessandro Moschitti. 2015. Assessing the Impact of Syntactic and Semantic Structures for Answer Passages Reranking. In *Proc. of CIKM.*
[21] Fei Wu and Daniel S. Weld. 2007. Autonomously Semantifying Wikipedia. In *CIKM.*
[22] Ce Zhang, Feng Niu, Christopher Ré, and Jude Shavlik. 2012. Big Data Versus the Crowd: Looking for Relationships in All the Right Places. In *ACL.*