A Methodology for Corpus Annotation through Crowdsourcing

Abstract

In contrast to expert-based annotation, for which elaborate methodologies ensure high quality output, currently no systematic guidelines exist for crowdsourcing annotated corpora, despite the increasing popularity of this approach. To address this gap, we define a crowd-based annotation methodology, compare it against the OntoNotes methodology for expert-based annotation, and identify future challenges for crowd-based annotation.

1 Introduction

Over the past ten years, Natural Language Processing (NLP) research has been driven forward by a growing volume of annotated corpora, produced by evaluation initiatives such as ACE (ACE, 2004), TAC (www.nist.gov/tac), SemEval and Senseval (www.senseval.org). These corpora have been essential for training and domain adaptation of NLP algorithms and their quantitative evaluation, as well as for enabling algorithm comparison and repeatable experimentation. In order to ensure linguistic annotation of consistently high quality, organisations (e.g., the Linguistic Data Consortium - www.ldc.upenn.edu) and large annotation projects such as OntoNotes (Hovy et al., 2006; Hovy, 2010)) have developed elaborate expert-based corpus annotation methodologies. Such methodologies, in conjunction with linguistic encoding standards, corpus interchange formats and effective tool support are essential for rigorous, replicable, and high quality annotation – a process referred to as "the science of annotation" (Hovy, 2010; Stede and Huang, 2012). A disadvantage of expert-driven and tightly controlled annotation methodologies is that they tend to be expensive, both in terms of the time required to produce large corpora, and in price per word annotated. The latter can vary between $0.36 and $1.0 (Zaidan and Callison-Burch, 2011; Poesio et al., 2012), which is unaffordable for corpora consisting of millions of words.

Commercial crowdsourcing marketplaces, in contrast, can be 33% cheaper than in-house employees when applied to tasks such as tagging and classification (Hoffmann, 2009). Consequently, NLP researchers have started experimenting with Amazon Mechanical Turk (AMT) and game-based approaches (Poesio et al., 2012) as less expensive approaches to corpus annotation through distributed human effort (see Section 2). Despite the growing popularity of crowdsourcing for annotation (Fort et al., 2011), no methodological guidelines exist for the application of this paradigm. Researchers and practitioners alike lack clear answers to the following two questions:

- (Q1) Based on current best practice, what does a repeatable, step-by-step methodology for crowdsourcing annotated corpora look like?
- (Q2) How does a crowd-based annotation methodology differ from an expert-based one?

The contribution of this paper is in defining a methodological framework for corpus annotation through crowdsourcing (Q1, Section 4). The reference point is the OntoNotes expert-based annotation methodology (Hovy et al., 2006; Hovy, 2010), which is modified to support the crowdsourcing process where remote, self-motivated, non-expert annotators carry out micro-annotation tasks. We discuss the differences between the two methodologies throughout the paper (as for Q2) and conclude with remaining challenges for crowd-based annotation.

2 Crowdsourcing for NLP

Three crowdsourcing genres have been used for corpus annotation and are presented in decreasing order of adoption by the NLP community (see Table 1).

In ***mechanised labour***, contributors are extrinsically motivated through economic incentives and they carry out small tasks (or micro-tasks) in return for a small amount of money (micro-payments). Most NLP projects use crowdsourcing marketplaces (mostly AMT and CrowdFlower (CF)), which allow requesters to post their micro-tasks in the form of Human Intelligence Tasks (HITs) to a large population of micro-workers (Callison-Burch and Dredze, 2010a). For example, Snow et al. (2008) collect event and affect annotations, while Lawson et al. (2010) and Finin et al. (2010) annotate special types of texts such as emails and Twitter feeds, respectively. Challenges include *low quality output* due to the workers' purely economic motivation, high costs for large tasks (Parent and Eskenazi, 2011), and ethical issues (Fort et al., 2011).

In ***games with a purpose (GWAPs)*** (von Ahn and Dabbish, 2008), contributors carry out annotation tasks as a side effect of playing a game. Compared to paid-for marketplaces, GWAPs: *(i)* reduce costs and the incentive to cheat as players are intrinsically motivated; and *(ii)* promise superior results, due to motivated players and better utilization of sporadic, explorer-type users, e.g., games may provide a larger variety of contributors and can reach more individuals than AMT (Parent and Eskenazi, 2011). Examples of GWAPs for annotation include *Phratris* for annotating sentences with syntactic dependencies (Attardi, 2010), *PhraseDetectives* (Poesio et al., 2012) to acquire anaphora annotations, and *Sentiment Quiz* (Scharl et al., 2012) to annotate sentiment in political texts. Designing appealing games and attracting a critical mass of players are among the key success factors within this genre (Wang et al., 2012).

In ***altruistic crowdsourcing***, tasks are carried out by volunteers (*VOL* in Table 1). For example, Chklovski and Mihalcea (2002) rely on volunteers to tag words with WordNet senses. Challenges include identifying a worthy cause that would appeal to many and fostering community building.

3 Related Work

In terms of annotation methodologies, there has been a shift from centrally managed, collocated expert annotator teams (e.g., OntoNotes) towards tightly managed but distributed annotator teams, where experts are mainly involved as project managers or adjudicators (Bontcheva et al., 2010). Crowdsourcing is the next logical step towards weakly managed, distributed, self-selected non-expert contributors, where high redundancy, sophisticated review workflows, and well-crafted micro-task design can ensure good quality annotation results and replace the detailed annotation task definitions and extensive annotator training required by expert-based approaches.

As the practice of using crowdsourcing for corpus annotation has become more widespread, researchers have started sharing some recommendations, resembling embryonic methodological guidelines (Callison-Burch and Dredze, 2010a; Poesio et al., 2012; McCreadie et al., 2012; Negri and Mehdad, 2010; von Ahn and Dabbish, 2008; Zaidan and Callison-Burch, 2011). However, these findings, while valuable, typically reflect experiences with solving a particular task, using one crowdsourcing genre. The meta-review of Wang et al. (2012) discusses the trade-offs of the three crowdsourcing genres, alongside dimensions such as contributor motivation, setup effort, and human participants. While this review answers some of the methodological questions, it does not provide a step-by-step methodology on how to setup, execute, and manage a complete crowdsourcing annotation project.

4 A Generic Crowdsourcing Methodology for Corpus Annotation

OntoNotes and other similar expert-based methodologies (Hovy, 2010) distinguish five conceptual stages of the corpus annotation process, shown in the left side of Figure 1. A corresponding crowd-based corpus annotation methodology is outlined in the right-hand side of Figure 1, based on generalising our experience with crowdsourced corpus acquisition[1] and a meta-analysis of other crowdsourcing projects (see Table 1). Overall, the same five high-level stages apply, although there are key differences in the issues that are addressed under each stage, as discussed in the remainder of this section.

4.1 Theory Stage

During the theory stage, expert-based methodologies choose what linguistic phenomena are annotated, define the formal annotation schemas, develop detailed annotator guidelines including examples, and fine-tune these with the expert annotators until a very high level of inter-annotator agreement (IAA) is reached (over 90% in OntoNotes).

In contrast, the main challenge in the crowd-based methodology lies in choosing the appropriate crowdsourcing genre and subsequently in decomposing the chosen NLP problem (e.g., relation annotation) into a set of simple tasks (possibly arranged in workflows), which can be understood and carried out by non-experts with minimal training and compact guidelines. This stage also focuses on a first setup of the task parameters, and on detailing the reward scheme of the project. Most crowdsourcing papers do not report on testing the annotation guidelines for improving IAA, although some envision introducing such a preparatory phase to measure the obtainable IAA values and to set up the micro-tasks accordingly - i.e., focusing on coverage with few annotators for high IAA tasks, and on consensus among many annotators for low IAA tasks (Brew et al., 2010).

4.1.1 Select Crowdsourcing Genre

In the context of corpus annotation, there is no clear winner. All crowdsourcing genres have their pros and cons, which need to be considered against the goals, size, and timescale of a specific project (Wang et al., 2012). As a general guideline, if a small-scale resource is acquired, and there is not enough time for lengthy prior work, such as building a gaming or voluntary participation system, mechanised labour is preferable. For acquiring large-scale resources (that would be expensive even given the cost reductions of AMT) and solving complex tasks that are hard to explain to turkers and would rather benefit from a more specialized contributor population, one should consider using GWAPs or voluntary projects, which despite the initial time and monetary investments can provide long-term and high-quality results virtually for free (Poesio et al., 2012).

4.1.2 Decompose NLP Problems into Tasks

Different types of linguistic annotation have been solved to date using crowdsourcing including *classification*, where a category or a numeric value is assigned to a string in a text, and *relation annotation*, a task concerned with determining whether typically discontiguous chunks of text are related, and how (e.g., marking up co-reference). Eickhoff and de Vries (2012) distinguish two types of crowdsourcing tasks: *closed class* and *open class* questions. Classification NLP problems are usually broken down into closed class questions where contributors must either: *(i)* select between a set of values such as named entity types (Finin et al., 2010) or word senses (Snow et al., 2008); or *(ii)* provide numeric ratings within a pre-set range; e.g., in sentiment annotation problems, where this option emulates the subjective nature of the task (Scharl et al., 2012). Relation annotation has been solved either by using closed class questions, e.g., textual entailment and event annotation (Snow et al., 2008), or through game metaphors, e.g., anaphora (Poesio et al., 2012) and dependency parsing annotation (Attardi, 2010).

Challenging annotation tasks can be carried out effectively by decomposing them into well designed, simpler tasks, arranged into workflows. Currently employed workflows vary in their complexity from simple *create-verify workflows* (e.g., PhraseDetectives is structured into two tasks, one for detecting markables and a second one for verifying the originally provided annotations (Chamberlain et al., 2009a)) to *complex workflows* consisting of three or more stages (Negri et al., 2011). The integration of crowdsourcing within larger NLP pipelines can involve *active learning workflows* which leverage machine classifiers to select the most informative samples (Laws et al., 2011; Brew et al., 2010).

4.1.3 Design Crowdsourcing Tasks

Crowdsourcing task definitions need to address a number of key issues:

How many workers should be assigned per task?

[1] References withheld to preserve author anonymity.

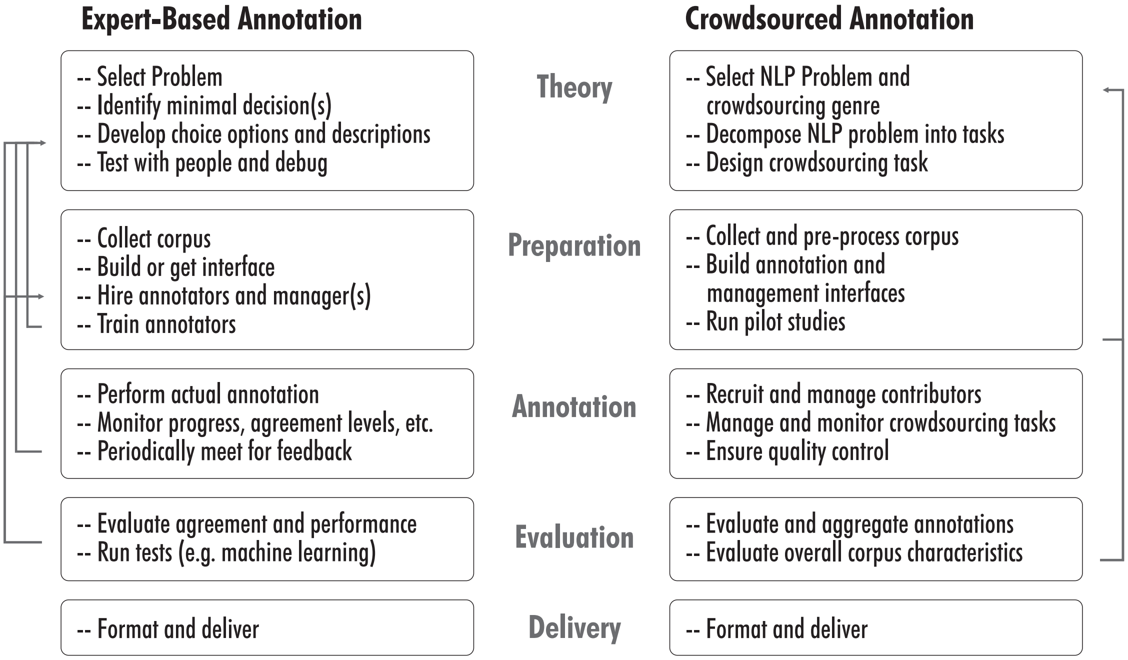
Expert-Based Annotation
Crowdsourced Annotation
Theory
-- Select Problem
-- Identify minimal decision(s)
-- Develop choice options and descriptions
-- Test with people and debug
-- Select NLP Problem and crowdsourcing genre
-- Decompose NLP problem into tasks
-- Design crowdsourcing task
Preparation
-- Collect corpus
-- Build or get interface
-- Hire annotators and manager(s)
-- Train annotators
-- Collect and pre-process corpus
-- Build annotation and management interfaces
-- Run pilot studies
Annotation
-- Perform actual annotation
-- Monitor progress, agreement levels, etc.
-- Periodically meet for feedback
-- Recruit and manage contributors
-- Manage and monitor crowdsourcing tasks
-- Ensure quality control
Evaluation
-- Evaluate agreement and performance
-- Run tests (e.g. machine learning)
-- Evaluate and aggregate annotations
-- Evaluate overall corpus characteristics
Delivery
-- Format and deliver
-- Format and deliver

Figure 1: A comparison of workflows in expert-based (Hovy, 2010) versus crowdsourced annotation.

Collecting redundant answers is a common quality assurance technique, both in expert-based projects and especially in crowd-based approaches, where the expertise of the contributors cannot be assessed and controlled easily. Indeed, Sheng et al. (2008) show that, when labels are noisy, multiple labels are preferable to single ones, already when using a simple round-robin mechanism to collect them. The optimal number of contributors per crowdsourcing task depends on the complexity of the annotation problem (see Table 1). For instance, Snow et al. (2008) find that, although they collect ten labels per task, for affect recognition, on average four non-expert labels per item suffice to reach expert-level quality. Lawson et al. (2010) experimentally show that the number of required labels varies even for different aspects of the same NLP problem, as they achieve good results with only four annotators for Person type NEs, but require six for *Location* and seven for *Organization Type* NEs. An alternative is to request additional annotations when there is a tie (Poesio et al., 2012), thus varying the number of labels depending on the ambiguity of individual annotation items.

How many categories should a classification task provide? Experience from expert-based annotation (Hovy, 2010) has shown that annotators should not be asked to choose from more than ten, ideally seven, categories. In contrast, crowdsourcing tasks typically present even fewer classification choices, in most cases ranging between two (binary choice) and four categories. Some researchers justify this reduction as a means to make the task simple enough to be suitable for non-expert annotators. For example, Snow et al. (2008) reduce the event ordering task from 14 to two relations. Experimental results on AMT support this finding: as the number of choices increases, annotation quality deteriorates (Fort et al., 2011; Hong and Baker, 2011).

How to reward contributors? Determining the scoring mechanism and the reward value (in game points or money) have a critical influence on the timely completion of the task and the quality of the gathered data. Deciding on *when to reward* (or penalize) a contributor's decision is a complex question, especially when the correct answer is not known or multiple answers are possible. If (some of) the answers are known a-priori, then *comparative scoring* can be applied, where answers that agree with a gold standard are rewarded. Otherwise, one can award the answers on which most contributors agree thus using *collaborative scoring*, e.g., in the advanced levels of PhraseDetectives, scores increase over time as more players agree with each other (Chamberlain et al., 2009b). Determining *how much to award* is a hotly debated issue. Generally, low-paying jobs will not attract enough workers to complete the task quickly, while over-average payments attract cheaters and encourage quantity rather than quality (Mrozinski et al., 2008; Feng et al., 2009). Awards are therefore best fine-tuned for each project, as part of a pilot run (Section 4.2.3).

How to split longer documents across crowdsourcing tasks? When longer documents are being annotated, e.g., emails (Lawson et al., 2010) or Wikipedia articles and book excerpts (Poesio et al., 2012), one needs to decide whether to put the entire document into a single, large crowdsourcing task; to split it into smaller parts – one per task (e.g., at paragraph or sentence level); or to avoid including in the corpus any documents above a certain size. Lawson et al. (2010) for example, did not include in their corpus emails which were too short (under 60 characters) or overly long (over 900 characters). These restrictions did not compromise their aim (learning to recognise NEs in email), however, such approach could introduce bias in cases where the goal is to study linguistic phenomena across diverse kinds of text. In contrast, PhraseDetectives (Poesio et al., 2012) splits larger texts into paragraphs, each becoming a separate game unit. This has, however, caused a problem for long-distance anaphora, where the antecedent is not present in the current paragraph and hence cannot be selected by the player. In general, given the limited time spent on each annotation task, text length should be kept reasonably short, without compromising accuracy.

Games are subject to additional design issues such as their visual appeal, the level of player interaction, and the specific timing of tasks (Vickrey et al., 2008; Poesio et al., 2012; von Ahn and Dabbish, 2008).

4.2 Preparation Stage

At the preparation stage, expert-based methodologies carry out corpus collection, data cleaning, and pre-processing; reuse or build their own annotation user interface; and hire and train expert annotators and managers. To lower the annotation costs, some projects opt for automatic pre-processing, so expert annotators can then focus on correction rather than on manual annotation from scratch. In comparison, this stage in the crowdsourcing methodology is significantly more challenging. Firstly, automatic tools for pre-processing and mapping between linguistic data structures and crowdsourcing tasks need to be implemented. Secondly, often the annotation and management interfaces need to be designed and developed as well. Finally, running a pilot version of the project is needed to fine-tune task parameters.

4.2.1 Collect and Pre-process Corpus

While the corpus selection process is similar to the one performed in traditional annotation, crowdsourcing approaches in general and GWAPs in particular should select documents that are interesting to the contributors, thus providing an additional incentive mechanism (Chamberlain et al., 2009b). Pre-processing pipelines can be built using open-source NLP tools such as openNLP (opennlp.apache.org) or GATE (Cunningham et al., 2011). Poesio et al. (2012) underline the importance of ensuring a high quality pre-processing pipeline as the quality of the annotation data “greatly affects the experience of players”. In fact, to account for the high error rate of their pre-processing pipeline (4.5 errors per text), they resort to manually correcting these errors first.

The generation of the individual crowdsourcing tasks is an ad-hoc process, which is currently solved differently, and independently, by each project. Both AMT and CF accept .csv files as input, but the transformation of the corpus into this format remains the responsibility of each project. This, in particular, is one open area, where NLP infrastructural support could help significantly.

4.2.2 Build Annotation and Management Interfaces

Annotation interfaces can easily be created with the facilities offered by AMT and CF, with the exception of a few projects which rely on more sophisticated interfaces and use the crowdsourcing platforms only for recruiting and paying contributors (marked with * in Table 1). In both cases, a *defensive task design* should be adopted to reduce cheating. For instance, Laws et al. (2011)’s radio buttons based interface attracted high volumes of spam, driving down the overall classification accuracy to only 55%. Extending that interface with *explicitly* *verifiable questions* then increased accuracy up to 75%. As discussed by Wang et al. (2012), the interface setup effort is highest for GWAPs and volunteer applications, since there are no easily reusable annotation toolkits currently available.

Management interfaces support NLP researchers in monitoring the status of their crowdsourcing projects, and in fine-tuning the task details including the selection and screening of contributors. Game and volunteer-based projects must build these interfaces from scratch, e.g., Poesio et al. (2012) built extensive management interfaces for PhraseDetectives. CF and AMT support requesters through the life-cycle of the crowdsourcing process including acquisition interface design, HIT population, job and worker management (job progress, result evaluation, workers' trust level and accuracy).

4.2.3 Run Pilot Studies

Small scale pilot runs are recommended in order to test the task definitions, to ensure that the appropriate task granularity and annotator instructions are chosen, and to fine-tune the parameters of the crowdsourcing project. Indeed, almost half of the approaches in Table 1 report on performing a pilot. Feng et al. (2009) advocate the introduction of a pilot phase (or Validation phase) during which small scale experiments are performed on a fraction of the data to determine the optimal parameters (e.g., payment, number of annotators) for achieving high quality output and short turn-around times. In contrast to OntoNotes, crowdsourcing pilots require that the complete annotation project is in place, and therefore they are performed in the "Preparation" rather than in the "Theory" stage. If the pilot is unsuccessful, the Theory stage needs to be revisited and the micro-tasks redesigned accordingly.

4.3 Annotation Stage

The annotation stage in expert-driven methodologies is typically the costliest and most time consuming. It consists of three kinds of tasks: data annotation, annotator management (including progress and IAA monitoring, reconciliation and adjudication, bias avoidance), and feedback meetings. Choices that need to be made include whether the entire corpus is to be annotated multiple times to allow for a reconciliation and verification step (higher quality, but higher costs) or whether after a certain IAA level is achieved on some of the corpus (e.g., 90%), it is sufficient to have one annotator per document. This stage typically relies on reusable tools for IAA calculation, data and annotation storage and progress monitoring (Hovy, 2010; Bontcheva et al., 2010).

The annotation phase of the crowdsourcing methodology denotes the actual crowdsourcing execution step, which can be shorter and more intense than the corresponding stage in expert-based projects, since smaller tasks can be completed within minutes (Snow et al., 2008). It consists of three broadly similar tasks, which are complex due to the decentralised nature of crowdsourcing and the relative lack of reusable workflow definition, task management and quality assurance interfaces. While in expert-based methodologies annotators are recruited and trained *prior* to the actual annotation task, this is not the case in crowdsourced projects where contributors are recruited, screened, and trained *during* the annotation phase. Another major difference is that additional effort is required to retain contributors, especially in GWAPs and volunteer projects. Besides continuous contributor management (Section 4.3.1), task management in crowdsourcing projects, including execution, allocation, prioritisation, and monitoring, is also a major challenge, as is the estimation of overall completion time (Section 4.3.2). Quality control is an integral part of the annotation process (Section 4.3.3), utilizing various techniques to ensure the collection of quality data.

4.3.1 Recruit and Manage Contributors

Recruitment. Projects deployed on mechanised labour platforms (*MLP* in Table 1) recruit their contributors from the platform's large and varied worker base. GWAPs should use multi-channel advertisements for attracting contributors, e.g., Chamberlain et al. (2009b) advertised PhraseDetectives on local and national press, science websites, blogs, bookmarking websites, gaming forums, social networking sites. Adverts can be repeated periodically and coupled with higher reward levels, which generates steep increases in the number of contributors and annotations (Poesio et al., 2012). Games can also leverage the *viral mechanisms* of social networking (*SN*) sites, e.g., *Sentiment Quiz* (Scharl et al.,

2012) and the Facebook version of PhraseDetectives (Chamberlain et al., 2012).

Screening. Some of the projects that run on crowdsourcing marketplaces *screen* their workers, typically based on their prior performance (or acceptance rate, *AR* in Table 1), geographic origin (*LOC*), and task specific competency tests (*COMP*). Although a worker's prior acceptance rate is a key filtering mechanism in AMT, it cannot be used reliably on its own and should be complemented by other filters, such as geographic location (Eickhoff and de Vries, 2012). Extensive screening, however, can lead to slower task completion times, so filtering through task-design is preferable (see Section 4.1.3).

Training prepares the contributors for the annotation task. GWAPs can include a user training stage in which player answers are contrasted against a gold standard (*GS*) and feedback is provided to players, in order to train them for the task (Chamberlain et al., 2009a). Mechanised labour projects can train workers through concise instructions (*instr*) and/or by embedding gold standard examples within jobs. One advantage of CF over AMT is that it offers immediate feedback when workers complete a "gold"-unit, thus continuously training them.

Expert-based methodologies also rely heavily on annotator training (coupled with detailed annotation guidelines) in order to ensure high levels of inter-annotator agreement. However, as discussed by Stede and Huang (2012) and Hovy (2010), there is a fine balance between annotator training and introducing bias. In the context of crowdsourcing, an overzealous contributor could introduce bias in the annotated corpus, by carrying out most of the work. Statistics from AMT (Fort et al., 2011) and GWAPs (Poesio et al., 2012) have shown that there are indeed a small number of contributors who carry out the majority of tasks (paid HITs or hours playing). However, currently there is no mechanism to restrict input size, this being yet another open issue, where significant improvements could be made by building more sophisticated infrastructural support.

Profiling. Another way to combat annotator bias is through *contributor profiling*, as an intrinsic part of the quality control mechanisms. One approach is to collect contributor-specific information as part of the crowd-sourcing tasks, e.g., whether workers are native speakers, for how many years they speak a language. Alternatively, Snow et al. (2008) propose a probabilistic model for correcting annotator bias for categorical data, which models the reliability and bias of individual workers (as some embryonic profiling). PhraseDetectives maintains a blacklist of low-trust players and discards their data.

Retention. Game based and altruistic projects are often hampered by the "volunteer attrition" phenomenon as their contributions "tend to quickly diminish over time"(Lieberman et al., 2007). To address this problem, games adopt techniques such as providing lively and continuous feedback to players and engaging in ongoing advertisement campaigns through their entire life-time (Poesio et al., 2012). Game design elements such as levels and leaderboards function as targets towards which players play, evidence showing that most play just enough in one sitting to pass to the next level (von Ahn, 2006; von Ahn and Dabbish, 2008). Mechanised labour projects can also build a community of trusted workers by offering bonuses and maintaining constructive communication with workers (*Comm*, Table 1).

4.3.2 Manage and Monitor Crowdsourcing Tasks

Task management ensures the optimal execution of crowdsourcing tasks and relies, primarily, on management interfaces offered by mechanised labour platforms or custom built tools for this purpose (Section 4.2.2). Managing the inputs to the annotation process includes the creation of task batches as well as correcting errors from the preprocessing pipelines (Poesio et al., 2012). The allocation of contributors to crowdsourcing tasks varies from filtering them based on specified screening criteria to more complex mechanisms where they are prevented from participating in tasks for which they have an obvious conflict of interest, e.g., validating their own annotations. This stage also includes monitoring the status of the task for estimating completion times and for fine-tuning task parameters.

4.3.3 Ensure Quality Control

Similar to expert-based approaches, ensuring that the produced annotations are of high quality is a major focus of the annotation stage of crowdsourcing projects. However, while expert-based approaches aim to guide annotators (prevent bias, reconciliation of results, feedback meetings), crowdsourcing projects try to identify and exclude unreliable contributors and flawed results as early as possible in the annotation process, thus saving important resources in terms of time and money.

Gold standard techniques (*GS* in Table 1) allow mixing known answers into the HITs in order to identify how well workers perform. In some cases this technique allows filtering out low-performing workers already during the annotation process itself (discussed here) while in other cases it is only used post-annotation (Section 4.4.1). The gold-unit functionality of CF provides immediate feedback to workers when they solve a gold-unit thus permitting to exclude (and not pay for) flawed answers as soon as they are provided. PhraseDetectives exemplifies this technique with a GS-based training phase that allows excluding unprepared workers, e.g., only 3000 players out of 8000 registered players passed this stage (Poesio et al., 2012).

Multilevel review techniques rely on create-verify workflows (and iterations over those) until the desired quality is obtained. PhraseDetectives' two stages correspond to such a workflow (*WF* in Table 1). Negri and Mehdad (2010) have shown that this technique leads to better and cheaper results than collecting redundant labels. Further improvements can be achieved when the data points to be labelled are carefully chosen, e.g., by *active learning (AL)*.

4.4 Evaluation Stage

The evaluation stage in expert-driven methodologies involves assessing annotator performance over time, inter-annotator trends, corpus characteristics (imbalance, sufficient size), and measuring how machine learning methods perform on this corpus. In the crowdsourcing case, the challenge lies in evaluating and aggregating the multiple contributor inputs into a consistent corpus (Sections 4.4.1), and in assessing the quality of this corpus (Section 4.4.2).

4.4.1 Evaluate and Aggregate Annotations

One approach to evaluating individual annotations is to compare them against the annotations of other workers in order to detect the right answer as well as the workers that consistently provide poor results. *Majority voting (MV)* is a popular technique to select those contributions that are likely to be correct, e.g., (Snow et al., 2008). If a majority vote cannot be reached, ties can be drawn at random or additional annotations can be collected, e.g., 8 + 4 in (Poesio et al., 2012). An alternative technique is to maintain contributor *profiles* (*Prof.* in Table 1) and to exclude the labels of low-rated ones, as exemplified in Section 4.3.1. Profiles can be built by comparing the worker labels to gold standard units. For example, Hsueh et al. (2009) detect workers that provide the most noisy annotations by measuring the deviation from gold standard labels and then summing up individual deviations for each worker.

Aggregating multiple, variable-quality annotations into a corpus has been analysed earlier by (Dawid and Skene, 1979; Smyth et al., 1994), and gained increased importance with the advent of crowdsourcing where the number and heterogeneity of labels is significantly higher than in expert based approaches (Hsueh et al., 2009; Snow et al., 2008). Since many approaches rely on choosing from a set of categories, the most popular aggregation method is that of *majority voting* where the category chosen by most contributors is selected (this implicitly validates the contributions as discussed earlier). Projects that elicit a numeric value within a range *average* individual contributions to obtain the final value (Snow et al., 2008). Less frequently used strategies include (i) *collection*, when all judgements are added to the existing base, e.g., to provide information about ambiguous cases such as in anaphora resolution (Poesio et al., 2012), and (ii) relying on platform specific mechanisms, e.g., CF.

4.4.2 Evaluate Overall Corpus Characteristics

As in expert-based approaches, crowdsourcing projects should dedicate ample effort on evaluating the overall corpus quality. A common technique is that of measuring inter-annotator agreement (IAA) among the crowd-workers and between experts and individual and group contributions (Snow et al., 2008). Depending on the NLP task, Cohen's kappa (for two contributors), Fleiss's kappa (for more contributors), or *F-score* can be used. Another frequent approach is to measure the performance of NLP tools trained on the crowdsourced corpus (*Task*). Manual evaluation is less popular.

Source	Annotation	Theory				Prep.	Annotation						Evaluation		
		Genre	Workers/task	No. of Categories	Reward Amount	Prototype	Recruitment	Training	Screening	Profile	Retain	Quality Control	Contribution Evaluation	Aggregation	Corpus Evaluation
Finin et al. 2010	NEs in Tweets	CF, AMT	2	4	$.05	Y	MLP	GS, instr	-	Y	-	GS	Prof.	MV	-
Voyer et al. 2010	NEs	CF	5	2	-	-	MLP	instr	-	-	-	GS	-	CF	IAA, Task
Lawson et al. 2010	NEs in Emails	AMT*	4, 6, 7	3	$.01+ bonus	Y	MLP	instr	-	-	Bonus	-	MV	MV	Task
Yetisgen-Yildiz et al. 2010	Medical NEs	AMT*	4	3	$.01-.05 +bonus	Y	MLP	instr	-	-	Comm., Bonus	-	-	MV	IAA, F-score
Rosenthal et al. 2010	PP Attach.	AMT	3	3	$.04	Y	MLP	-	LOC	-	-	-	-	MV	Precision
Jha et al. 2010	PP Attach.	AMT	5	varies	$.04	Y	MLP	instr	-	-	-	-	-	MV	IAA, Rec
Snow et al. 2010	Affect, Wrd Sim.	AMT	10	range	-	-	MLP	instr	-	Y	-	-	-	Avg	IAA, Task
	Event&TE, WSD	AMT	10	2, 3	-	-	MLP	instr	-	Y	-	-	-	MV	IAA
Yano et al. 2010	Bias in polit. blogs	AMT	5	3, 6	$.02, $.04	-	MLP	instr	LOC, AR90	-	-	-		MV	IAA
Mellebeek et al. 2010	Polarity	AMT	3	3, 11 range	$.02	Y	MLP	instr	COMP	-	-	-	-	MV	IAA Task
Laws et al. 2011	NEs & Affect	AMT*	2+n	-/2	$.01	-	MLP	-	-	Y	-	AL	MV	MV	Task
Sayeed et al. 2011	Opinion	CF	3	4	$.12	Y	MLP	instr GS	LOC	Y	-	GS	Prof.	MV	IAA F-score
Hong and Baker 2011	Word Senses	AMT* CF*	10	4-5	$.15	Y	MLP	instr	COMP, LOC,AR75	-	-	GS	-	MV	Acc.
Strapparava et al. 2012	Emotions in Lyrics	AMT	10	range	-	-	MLP	instr	-	Y	-	GS	Prof, GS	Avg.	IAA Task
Attardi 2010	Dependency Pars. Rel.	GWAP	-	-	-	Y	-	-	-	Y	-	-	MV, Prof.	MV	-
Poesio et al. 2012	Anaphora	GWAP	8+4	-	-	Y	Media, SN	GS, instr	-	Y	Levels, Boards	WF, GS	MV, Prof.	Coll.	IAA, Task
Scharl et al. 2012	Opinions	GWAP	7+5	5	-	-	SN	instr	-	-	Levels, Boards	-	MV	Avg.	-
Chklovski 2002	Word Senses	VOL	2	varies	-	-	-	-	-	-	Boards	AL	MV	MV	IAA
Brew et al. 2010	Affect in news	VOL	1,3,5	3	-	-	RSS	-	-	-	-	AL	-	Cons	Task
Che and Liu 2011	Word Senses	VOL	2	varies	-	-	-	-	-	-	-	GS	-	MV	IAA

Table 1: Overview of annotation works and their main methodological choices (sorted by genre and chronologically within genre); *: AMT/CF only used for worker management, annotation interface hosted externally; AL:Active Learning, AR:acceptance rate, AMT:Amazon Mechanical Turk, Avg:average, CF:Crowd Flower, Coll:collection, COMP:competency test, Comm:communication, Cons:consensus, GS:gold standard, GWAP:game with a purpose, instr:instructions, LOC:location, MV:majority vote, MLP: mechanised labour platform, Prof:profile, SN:social network, VOL:altruistic crowdsourcing, WF:workflow.

4.5 Delivery Stage

The final delivery stage is common to both methodologies and is concerned with issues such as corpus encoding standards (e.g., ISO/TC 37/SC 4 (Ide and Romary, 2004)), licensing terms, distribution methods, and contributor anonymity. Crowdsourcing projects can reuse many existing tools and licenses here, although if full provenance and multiple annotator versions are to be encoded, some extensions might be required including considerations about contributor acknowledgement, anonymity, and consent to the chosen licensing terms. The latter issues have not yet received sufficient consideration by most NLP crowdsourcing projects.

The first question is how to acknowledge crowd contributions to the annotated resource. While no clear guidelines exist, volunteer projects from other research fields already include *the crowd* as an author (Cooper et al., 2010; Kawrykow et al., 2012).

Secondly, some contributors may wish to remain anonymous. While paid-for marketplaces go some way towards addressing worker privacy, these are not always sufficient. GWAPs embedded within social platforms (where many users use their real names) need to implement safeguards, so that contributor anonymity can be maintained.

The third issue is corpus licensing terms and consent; i.e., making it clear to users that by participating and contributing knowledge for scientific purposes, they also agree to a defined license for corpus sharing and use of their work. Volunteer-based projects typically use open licenses, such as Creative Commons and gain contributor consent (Abekawa et al., 2010). In contrast, NLP GWAPs tend to mostly emphasize the scientific purpose of the game, but fail to gain licensing consent. Therefore, gaining explicit contributor consent to the corpus licensing terms is an important methodological step, which unfortunately has mostly been overlooked. We also recommend that crowdsourcing projects adopt a clearly stated and open license.

5 Conclusions and Outlook

Annotation science and repeatable, expert-based methodologies such as OntoNotes have evolved in response to the need for creating large, high-quality annotated corpora for training and evaluating NLP algorithms. While crowdsourcing is increasingly regarded as the way to scale up NLP corpus annotation in an affordable manner, researchers have mostly used this paradigm to acquire small- to medium-sized corpora. The contribution of this paper lies in defining a corpus annotation methodology for crowdsourcing, as the first step towards enabling scalability, repeatability, and high quality outcomes.

The next step forward would be to make freely available reusable task definitions and crowdsourcing workflow patterns. Researchers are already starting to define *workflow templates* that perform best for a given task, e.g., for crowdsourced translations (Zaidan and Callison-Burch, 2011). In the game-based crowdsourcing genre, Poesio et al. (2012) state that their game could easily be reused and adapted to NLP annotations which require several sections of texts to be linked together with a relationship. Similarly, task templates can often be made language agnostic and thus easily re-used across languages, e.g., for AMT (Madnani et al., 2010; Irvine and Klementiev, 2010) and for GWAPs (Poesio et al., 2012; Scharl et al., 2012).

The biggest challenge for crowdsourcing projects is that the cost to define a single annotation project could sometimes outweigh the benefits. Future work should address this challenge by providing a generic crowdsourcing infrastructure for corpus annotation, where the different crowdsourcing genres could be combined seamlessly, i.e., annotations could be sourced via marketplaces, GWAPs, and volunteers simultaneously. In addition, such infrastructure would help with sharing information about contributor profiles, annotator capabilities, past training, and history from previously completed projects. It could help prevent annotator bias and minimise human oversight required, by implementing more sophisticated crowd-based annotation workflows, coupled with in-built control mechanisms (e.g., no single annotator is allowed to carry out more than 30% of all tasks). Such infrastructure would implement reusable, automated methods for quality control and aggregation and make use of the emerging reusable task definitions and workflow patterns. A tight integration with existing NLP infrastructures such as GATE and UIMA (Ferrucci and Lally, 2004) would provide support for semi-automatic pre-processing in a principled, reusable way.

References

T. Abekawa, M. Utiyama, E. Sumita, and K. Kageura. 2010. Community-based Construction of Draft and Final Translation Corpus through a Translation Hosting Site Minna no Hon'yaku (MNH). In *Proc. of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*.

ACE, 2004. *Annotation Guidelines for Event Detection and Characterization (EDC)*, Feb. Available at http://www.ldc.upenn.edu/Projects/ACE/.

G. Attardi. 2010. Phratris – A Phrase Annotation Game. In *INSEMTIVES Game Idea Challenge*.

K. Bontcheva, H. Cunningham, I. Roberts, and V. Tablan. 2010. Web-based Collaborative Corpus Annotation: Requirements and a Framework Implementation. In *Proc. of the Workshop on New Challenges for NLP Frameworks*, Valletta, Malta, May.

A. Brew, D. Greene, and P. Cunningham. 2010. Using Crowdsourcing and Active Learning to Track Sentiment in Online Media. In *Proc. of the 19th European Conference on Artificial Intelligence (ECAI'10)*, pages 145–150.

C. Callison-Burch and M. Dredze. 2010a. Creating Speech and Language Data with Amazon's Mechanical Turk. In *Proc. of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk* (Callison-Burch and Dredze, 2010b), pages 1–12.

C. Callison-Burch and M. Dredze, editors. 2010b. *Proc. of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*.

J. Chamberlain, U. Kruschwitz, and M. Poesio. 2009a. Constructing An Anaphorically Annotated Corpus with Non-Experts: Assessing the Quality of Collaborative Annotations. In *Proc. of The People's Web Meets NLP: Collaboratively Constructed Semantic Resources*, pages 57–62.

J. Chamberlain, M. Poesio, and U. Kruschwizt. 2009b. A new life for a dead parrot: Incentive structures in the Phrase Detectives game. In *Proc. of the Webcentives Workshop*.

J. Chamberlain, U. Kruschwitz, and M. Poesio. 2012. Motivations for Participation in Socially Networked Collective Intelligence Systems. In *Proc. of Collective Intelligence (CI'10)*.

W. Che and T. Liu. 2011. Word Sense Disambiguation Corpora Acquisition via Confirmation Code. In *Proc. of the 5th International Conference on Natural Language Processing*, pages 1472–1476.

T. Chklovski and R. Mihalcea. 2002. Building a Sense Tagged Corpus with Open Mind Word Expert. In *Proc. of the ACL-02 Workshop on Word Sense Disambiguation: Recent Successes and Future Directions - Volume 8*, pages 116–122.

S. Cooper, F. Khatib, A. Treuille, J. Barbero, J. Lee, M. Beenen, A. Leaver-Fay, D. Baker, Z. Popovic, and Foldit players. 2010. Predicting protein structures with a multiplayer online game. *Nature*, 466(7307):756–760.

H. Cunningham, D. Maynard, K. Bontcheva, V. Tablan, N. Aswani, I. Roberts, G. Gorrell, A. Funk, A. Roberts, D. Damljanovic, T. Heitz, M.A. Greenwood, H. Saggion, J. Petrak, Y. Li, and W. Peters. 2011. *Text Processing with GATE (Version 6)*. The University of Sheffield.

A. P. Dawid and A. M. Skene. 1979. Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 28(1):pp. 20–28.

C. Eickhoff and A. de Vries. 2012. Increasing Cheat Robustness of Crowdsourcing Tasks. *Information Retrieval*, 15:1–17. 10.1007/s10791-011-9181-9.

D. Feng, S. Besana, and R. Zajac. 2009. Acquiring High Quality Non-Expert Knowledge from On-Demand Workforce. In *Proc. of The People's Web Meets NLP: Collaboratively Constructed Semantic Resources*, pages 51–56.

D. Ferrucci and A. Lally. 2004. UIMA: An Architectural Approach to Unstructured Information Processing in the Corporate Research Environment. *J. of Natural Language Engineering*, 10(3-4):327–348.

T. Finin, W. Murnane, A. Karandikar, N. Keller, J. Martineau, and M. Dredze. 2010. Annotating Named Entities in Twitter Data with Crowdsourcing. In Callison-Burch and Dredze (Callison-Burch and Dredze, 2010b), pages 80–88.

K. Fort, G. Adda, and K.B. Cohen. 2011. Amazon Mechanical Turk: Gold Mine or Coal Mine? *Computational Linguistics*, 37(2):413 –420.

L. Hoffmann. 2009. Crowd Control. *Communications of the ACM*, 52(3):16 –17.

J. Hong and C. F. Baker. 2011. How Good is the Crowd at "real" WSD? In *Proc. of the 5th Linguistic Annotation Workshop*, (LAW V '11), pages 30–37.

E.H. Hovy, M. P. Marcus, M. Palmer, L. A. Ramshaw, and R. M. Weischedel. 2006. OntoNotes: The 90% Solution. In *Proc. of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL:HLT'06)*.

E. Hovy. 2010. Annotation. In *Tutorial Abstracts of ACL*.

P.Y. Hsueh, P. Melville, and V. Sindhwani. 2009. Data Quality from Crowdsourcing: A Study of Annotation

Selection Criteria. In *Proc. of the Workshop on Active Learning for Natural Language Processing*, pages 27–35.

N. Ide and L. Romary. 2004. International standard for a linguistic annotation framework. *J. of Natural Language Engineering*, 10(3-4):211 – 225.

A. Irvine and A. Klementiev. 2010. Using Mechanical Turk to Annotate Lexicons for Less Commonly Used Languages. In Callison-Burch and Dredze (Callison-Burch and Dredze, 2010b), pages 108–113.

M. Jha, J. Andreas, K. Thadani, S. Rosenthal, and K. McKeown. 2010. Corpus Creation for New Genres: A Crowdsourced Approach to PP Attachment. In Callison-Burch and Dredze (Callison-Burch and Dredze, 2010b), pages 13–20.

A. Kawrykow, G. Roumanis, A. Kam, D. Kwak, C. Leung, C. Wu, E. Zarour, and Phylo players. 2012. Phylo: A Citizen Science Approach for Improving Multiple Sequence Alignment. *PLoS ONE*, 7(3):e31362.

F. Laws, C. Scheible, and H. Schütze. 2011. Active Learning with Amazon Mechanical Turk. In *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP'11)*, pages 1546–1556.

N. Lawson, K. Eustice, M. Perkowitz, and M. Yetisgen-Yildiz. 2010. Annotating Large Email Datasets for Named Entity Recognition with Mechanical Turk. In Callison-Burch and Dredze (Callison-Burch and Dredze, 2010b), pages 71–79.

H. Lieberman, D. Smith, and A. Teeters. 2007. Common Consensus: A Web-based Game for Collecting Commonsense Goals. In *Proc. of Conference om Intelligent User Interfaces (IUI'07)*.

N. Madnani, J. Boyd-Graber, and P. Resnik. 2010. Measuring Transitivity Using Untrained Annotators. In Callison-Burch and Dredze (Callison-Burch and Dredze, 2010b), pages 188–194.

R. McCreadie, C. Macdonald, and I. Ounis. 2012. Identifying Top News Using Crowdsourcing. *Information Retrieval*, pages 1–31. 10.1007/s10791-012-9186-z.

B. Mellebeek, F. Benavent, J. Grivolla, J. Codina, M. R. Costa-jussà, and R. Banchs. 2010. Opinion Mining of Spanish Customer Comments with Non-Expert Annotations on Mechanical Turk. In Callison-Burch and Dredze (Callison-Burch and Dredze, 2010b), pages 114–121.

J. Mrozinski, E. Whittaker, and S. Furui. 2008. Collecting a Why-Question Corpus for Development and Evaluation of an Automatic QA-System. In *Proc.of 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL:HLT'08)*, pages 443–451, June.

M. Negri and Y. Mehdad. 2010. Creating a Bi-lingual Entailment Corpus through Translations with Mechanical Turk : 100 for a 10-day Rush. In Callison-Burch and Dredze (Callison-Burch and Dredze, 2010b), pages 212–216.

M. Negri, L. Bentivogli, Y. Mehdad, D. Giampiccolo, and A. Marchetti. 2011. Divide and Conquer: Crowdsourcing the Creation of Cross-Lingual Textual Entailment Corpora. In *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP'11)*, pages 670–679.

G. Parent and M. Eskenazi. 2011. Speaking to the Crowd: Looking at Past Achievements in Using Crowdsourcing for Speech and Predicting Future Challenges. In *Proc. of INTERSPEECH*, pages 3037–3040.

M. Poesio, U. Kruschwitz, J. Chamberlain, L. Robaldo, and L. Ducceschi. 2012. Phrase Detectives: Utilizing Collective Intelligence for Internet-Scale Language Resource Creation. *Transactions on Interactive Intelligent Systems*. To Appear.

S. Rosenthal, W. Lipovsky, K. McKeown, K. Thadani, and J. Andreas. 2010. Towards Semi-Automated Annotation for Prepositional Phrase Attachment. In *Proc. of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*.

A. B. Sayeed, B. Rusk, M. Petrov, H. C. Nguyen, T. J. Meyer, and A. Weinberg. 2011. Crowdsourcing syntactic relatedness judgements for opinion mining in the study of information technology adoption. In *Proc. of the 5th ACL-HLT Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities (LaTeCH '11)*, pages 69–77. Association for Computational Linguistics.

A. Scharl, M. Sabou, S. Gindl, W. Rafelsberger, and A. Weichselbraun. 2012. Leveraging the wisdom of the crowds for the acquisition of multilingual language resources. In *Proc. of the Eight International Conference on Language Resources and Evaluation Conference (LREC12)*, pages 379–383.

V. S. Sheng, F. Provost, and P. G. Ipeirotis. 2008. Get Another Label? Improving Data Quality and Data Mining Using Multiple, Noisy Labelers. In *Proc. of the 14th International Conference on Knowledge Discovery and Data Mining (KDD'08)*, pages 614–622.

P. Smyth, U. Fayyad, M. Burl, P. Perona, and P. Baldi. 1994. Inferring Ground Truth From Subjective Labelling of Venus Images. *Advances in Neural Information Processing Systems*, (7):1085 –1092.

R. Snow, B. O'Connor, D. Jurafsky, and A. Y. Ng. 2008. Cheap and Fast—but is it Good?: Evaluating Non-Expert Annotations for Natural Language Tasks. In *Proc. of the Conference on Empirical Methods in Nat-*

ural Language Processing (EMNLP'08), pages 254–263.

M. Stede and C.R. Huang. 2012. Inter-operability and reusability: the science of annotation. *Language Resources and Evaluation*, 46:91–94. 10.1007/s10579-011-9164-x.

C. Strapparava, R. Mihalcea, and A. Battocchi. 2012. A Parallel Corpus of Music and Lyrics Annotated with Emotions. In *Proc. of the Eight International Conference on Language Resources and Evaluation (LREC'12)*.

D. Vickrey, A. Bronzan, W. Choi, A. Kumar, J. Turner-Maier, A. Wang, and D. Koller. 2008. Online Word Games for Semantic Data Collection. In *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP'08)*, pages 533–542.

L. von Ahn and L. Dabbish. 2008. Designing games with a purpose. *Commun. ACM*, 51(8):58–67.

L. von Ahn. 2006. Games With a Purpose. *Computer*, 39(6):92 –94.

R. Voyer, V. Nygaard, W. Fitzgerald, and H. Copperman. 2010. A Hybrid Model for Annotating Named Entity Training Corpora. In *Proc. of the Fourth Linguistic Annotation Workshop (LAW IV '10)*, pages 243–246.

A. Wang, C.D.V. Hoang, and M. Y. Kan. 2012. Perspectives on Crowdsourcing Annotations for Natural Language Processing. *Language Resources and Evaluation*.

T. Yano, P. Resnik, and N. A. Smith. 2010. Shedding (a Thousand Points of) Light on Biased Language. In Callison-Burch and Dredze (Callison-Burch and Dredze, 2010b), pages 152–158.

M. Yetisgen-Yildiz, I. Solti, F. Xia, and S. R. Halgrim. 2010. Preliminary Experience with Amazon's Mechanical Turk for Annotating Medical Named Entities. In Callison-Burch and Dredze (Callison-Burch and Dredze, 2010b), pages 180–183.

O. F. Zaidan and C. Callison-Burch. 2011. Crowdsourcing Translation: Professional Quality from Non-Professionals. In *Proc. of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL:HLT'11)*, pages 1220–1229.