# How Good is the Crowd at "real" WSD?

**Jisup Hong**
International Computer Science Institute
Berkeley, CA
jhong@icsi.berkeley.edu

**Collin F. Baker**
International Computer Science Institute
Berkeley, CA
collinb@icsi.berkeley.edu

## Abstract

There has been a great deal of excitement recently about using the "wisdom of the crowd" to collect data of all kinds, quickly and cheaply (Howe, 2008; von Ahn and Dabbish, 2008). Snow *et al.* (Snow et al., 2008) were the first to give a convincing demonstration that at least some kinds of linguistic data can be gathered from workers on the web more cheaply than and as accurately as from local experts, and there has been a steady stream of papers and workshops since then with similar results. e.g. (Callison-Burch and Dredze, 2010).

Many of the tasks which have been successfully crowdsourced involve judgments which are similar to those performed in everyday life, such as recognizing unclear writing (von Ahn et al., 2008), or, for those tasks that require considerable judgment, the responses are usually binary or from a small set of responses, such as sentiment analysis (Mellebeek et al., 2010) or ratings (Heilman and Smith, 2010). Since the FrameNet process is known to be relatively expensive, we were interested in whether the FrameNet process of fine word sense discrimination and marking of dependents with semantic roles could be performed more cheaply and equally accurately using Amazon's Mechanical Turk (AMT) or similar resources. We report on a partial success in this respect and how it was achieved.

## 1 Defining the task

The usual FrameNet process for annotating examples of a particular **lexical unit (LU)**, is to first extract examples of this sense from a corpus, based on collocational and syntactic patterns, storing them in subcorpora; this process is called **subcorporation**. Given an LU, vanguarders begin by composing rules consisting of syntactic patterns and instructions as to whether to include or exclude the sentences that match them. An automated system extracts sentences containing uses of the LU's lemma, applies POS tagging and chunk parsing, and then matches the sentences against the rules in their specified order to allow for cascading effects. Ultimately, the result is a set of subcorpora, each corresponding to a pattern, and containing sentences likely to exhibit a use of the LU. More recently, a system has been developed in collaboration with the Sketch Engine ((Kilgarriff et al., July 2004) `http://www.sketchengine.co.uk`) to accelerate this process by giving annotators a graphical interface in which precomputed collocational pattern matches can be more directly assigned to the various LUs corresponding to a given lemma. The actual annotation of the **frame elements (FEs)** is facilitated by having pre-selected sets of sentences which are at least likely to contain the right sense of the word, and which share a syntactic pattern. Therefore, we first focused on the frame discrimination task (which in other contexts would be called word sense discrimination), which we assumed to be simpler to collect data for than the FE annotation task, and which is a prerequisite for it.

We began by evaluating the resources that AMT provides for designing and implementing **Human Intelligence Tasks (HITs)**; we quickly determined that the UI provided by AMT would not suffice for the task we planned. Specifically, it lacks the ability to:

- randomize the selection options,



*Proceedings of the Fifth Law Workshop (LAW V)*, pages 30–37,
Portland, Oregon, 23-24 June 2011. 

- present questions from a set one at a time,
- randomize the order in which a set of questions are presented, or
- record response times for each question.

We therefore decided to design our HITs using Amazon's "External Question HIT Type", and to serve the HITs from our own web server. In this system, when workers view or execute a HIT, the content of the HIT window is supplied from our server, and responses are stored directly in a database running our own server, rather than Amazon's. Workers log in through AMT and are ultimately paid through AMT, but the content of the tasks can be completely controlled though our web server.

The Frame Discrimination Task can be set up in a number of ways, such as:

1. Present a single sentence with the lemma highlighted. Workers must select a frame (or "none of the above") from a multiple-choice list of frames we provide.
2. Present a list of sentences all containing uses of the same lemma. Workers must check off all the sentences that contain uses of a given frame.
3. Present a list of sentences all containing uses of the same lemma. Provide one example sentence from each frame and ask users to categorize the sentences.

In order to get started as quickly as possible and get a baseline result, we chose the first of the above methods, which is the most straightforward from a theoretical point of view. For example, the lemma might be *gain*.v, which has two LUs, one in the **Change_position_on_a_scale** frame, and another in the **Getting** frame. The HIT displays one sentence at a time, with the lemma highlighted; below the sentence, a multiple-choice selection is presented with the Frame names:

```
You will have to GAIN their support,
if change is to be brought about.

  Change_position_on_a_scale
  Getting
  None of the above
```

When users mouse-over the name of a frame, a pop-up displays an example sentence from that Frame (from a different LU in the same frame). Users can also click the name of the frame, which causes the browser to open another window with the frame definition. This process repeats for 12 sentences, at which point the HIT is over, and results are entered into our database.

**Sources of material for testing**

We had no shortage of sentences for the frame discrimination task; we started with some of the many unannotated sentences already in the FrameNet database. In the usual process of subcorporation, each of the subcorpora matches one specific pattern; the goal is to extract roughly 20 examples of each collocational/syntactic pattern, and to annotate one or two of each. The following are examples from among the patterns used for *rip*.v in the **Removing** frame:

```
NP T NP [PP f="from"]
NP T NP [w "out"]
```

The first pattern would match sentences like, "I ripped the top from my pack of cigarettes," and the second, "She ripped the telephone out of the wall."

We do not presume, however, that we will always be able to define patterns for all of the possible valences of a predicator, so we also include two "other" subcorpora. The first of these (named "other-matched") contains 50 sentences (provided there are enough instances in the corpus) which matched any one of the preceding patterns but were left over after 20 had been extracted for each pattern. The second ("other-unmatched") contains sentences in which the lemma occurs (with the right POS) which did **not** match any of the earlier patterns. Vanguarders carefully check these "other" subcorpora to see if the lemma is used in a syntactic valence which was not foreseen; if they find any such new valences, they are annotated. Typically, this means that there are roughly 100 extra unannotated sentences for each LU. For this experiment, we extracted 10 sentences from the "other-matched" subcorpus of each of the LUs for the lemma, meaning that they had already matched some pattern which was designed for one of those LUs. In addition to the unannotated sentences, we randomly selected three annotated sentences from each LU, two to use as included gold-standard items

| Frame name | Example |
|---|---|
| Cause_to_fragment | The revolution has RIPPED thousands of Cuban families apart ... |
| Damaging | ... Mo's dress is RIPPED by a drunken admirer. |
| Removing | Sinatra then reportedly RIPPED the phone out of the wall ... |
| Self_motion | A tornado RIPPED through Salt Lake City ... |
| Judgment_communication | (no annotated examples–related to *rip into*.v) |
| Position_on_a_scale | Eggs, shellfish and cheese are all HIGH in cholesterol ... |
| Dimension | An adult tiger stands at least 3 ft (90 cm) HIGH at the shoulder. |
| Intoxication | Exhausted but HIGH on adrenalin, he would roam about the house... |
| Measurable_attributes | Finally we came to a HIGH plastic wall. |
| Evidence | Our results SHOW that unmodified oligonucleotides can provide ... |
| Reasoning | He uses economics to SHOW how this is so. |
| Obviousness | ... sighting black mountain tops SHOWING through the ice-cap. |
| Cotheme | When they were SHOWN to their table, ... |
| Finish_competition | (no annotated examples– *Fair Lady placed in the second race at Aqueduct.*) |
| Cause_to_perceive | A second inner pylon SHOWS Ptolemy XIII paying homage to Isis ... |

Table 1: LUs (senses) for *rip*.v, *high*.a, and *show*.v

for checking accuracy, and one to use as the example in the preview of the HIT. These sentences were randomized and separated into batches of 12 for each HIT; all of which were inserted into a database on a local web server. A local CGI script (reached from AMT) calls the database for the examples in each HIT and stores the workers' responses in the same database.

We ran three trials under this setup, for the lemmas *rip*.v, *high*.a, and *show*.v. Based on the success of earlier studies, our concern initially was to make our tasks be sufficiently challenging so as to be useful for evaluating AMT. Thus, we chose lemmas with four to five senses rather than just two or three. In addition, for these three lemmas, each of the senses appears with sufficient frequency in the corpus so that all senses are realistically available for consideration.[1] The frames for each of these lemmas are shown in Table 1; some of these distinctions are fairly subtle; we will discuss some examples below.

To combine responses, we took the modal response as the result for each item; in cases of ties, we chose randomly, and split the response count where necessary. On this basis, for *rip*.v, the workers had an accuracy of 32.16 correct out of 48 items (67%), for *high*.a, they got 22 out of 49 correct (46%), and for *show*.v, 37 out of 60 items (62%), as shown in Table 2. If we consider that FrameNet has four senses (LUs) for *rip*.v and *high*.a and five for *show*.v, this might not sound too awful, but if we think of this as pre-processing, so that the resulting sentences can be annotated in the correct frame, it leaves a lot to be desired. If we raise the agreement criteria, by filtering out items on which the margin between the modal response and the next highest is 35% or greater (i.e. those with high agreement among workers), we can get higher accuracy (shown in the right two columns of Table 2), at the expense of failing to classify $3/4$ of the items, hardly a solution to the problem.

### Trials with CrowdFlower

We decided to try our task on CrowdFlower (`http://crowdflower.com`, formerly Dolores Labs), a company that provides tools and custom solutions to make crowdsourcing tasks easier to create and manage, including techniques to assure a certain level of quality in the results. While working with CrowdFlower, our tasks were running on AMT, although CrowdFlower also provides other labor pools, such as Samasource (`http://www.samasource.org`), depending on the nature of the task. We tried running the task for *rip*.v on Crowdflower's system, using the same HIT design as before, (recreated using

[1] An exception is the *show*.v in the **Finish_competition** frame, which we excluded for this reason, as in *Mucho Macho Man showed in the 2011 Kentucky Derby.*

| Lemma | No. senses | No. Items | Accuracy | Filtered Items | Accuracy. |
|---|---|---|---|---|---|
| *rip*.v | 4 | 48 | 67% | 10 | 90% |
| *high*.a | 4 | 48 | 46% | 12 | 58% |
| *show*.v | 5 | 60 | 62% | 11 | 64% |

Table 2: Results from Trial 1: *Rip*.v, *high*.a and *show*.v

their self-serve UI design tools), but with different sentences. Once again, we selected 12 sentences for each of the 4 LUs, for a total of 48 sentences. We wanted to collect 10 judgments per sentence, for a total of 480 judgments. Of the 12 sentences in each HIT, 2 were already annotated and used as a gold standard.

However, after starting this job, we found that the CrowdFlower system automatically halted the jobs after a few hours due to poor average performance on the gold standard items. After having the job halted repeatedly, we were finally able to force it to finish by suspending use of the gold standard to judge accuracy. In other words, the system was telling us that the task was too hard for the workers.

**Revised CrowdFlower Trials**

After our difficulties with the first trial on CrowdFlower's system, we visited their offices for an on-site consultation. We learned more about how CrowdFlower's system works, and received suggestions on how to improve performance:

- Run a larger set of data; they recommended at least 200 sentences for a job.
- Embed 20% gold standard items so that there is at least one per page of questions, since, without gold standard items, workers will answer randomly, or always choose the first option.
- Get rid of the frame names and use something easier to understand.
- Provide more detailed instructions that include examples.

Based on this consultation, we made the following changes in our HITs: (1) Replaced frame names with hand-crafted synonyms, (2) Renamed the task and rewrote all instructions to avoid jargon, (3) Removed links and roll-overs giving examples or referring people to external documentation, and (4) Extracted 60 sentences per LU, of which 10 are gold standard.

Although we planned to do this for *rip*.v, *high*.a, and *show*.v, we found that it was too difficult to come up with synonyms for *high*.a, so we ran trials only for *rip*.v and *show*.v. For *rip*.v, with four senses, we collected 10 judgments each on 240 sentences, for a total of 2400 judgments. For *show*.v, with five senses, we collected 10 judgments each on 300 sentences, for a total of 3000 judgments. In the final trials, the weighted majority response provided by CrowdFlower was found to be correct 75% for rip.v and 80% for show.v. This was encouraging, but we were concerned with the limitations of this method: (1) The calculation used to select the "weighted majority response" is proprietary to CrowdFlower, so that we could not know the details or change it, and (2) the final trials required handcrafted definitions, synonyms, and very clear definitions for each LU, which is at best time-consuming, and sometimes impossible (as is likely case for *high*.a), meaning the method will not scale well. As researchers, the first limitation is especially problematic as it is necessary to know exactly what methods we are using in our research and be able to share them openly. For these reasons, we decided to go back to building our own interfaces on AMT, and to look for approaches that would be more automatic.

**Return to AMT**

We redesigned the HIT around a pile-sorting model; instead of seeing one sentence and choosing between frames (whether by name or by synonym), workers are shown model sentences for each LU (i.e. in each frame), and then asked to categorize a list of sentences that are displayed all at once. Consequently, the worker generates a set of piles each corresponding to a frame/LU. The advantages of this approach are as follows:

- Workers can more easily exploit paradigmatic

contrasts across sentences to decide which category to put them in.

- Workers can recategorize sentences after initially putting them into a pile.
- Workers have example sentences using the LUs in question, which constitutes more information than the frame name (assuming that they were not going to the FrameNet website to peruse annotation).
- HITs can be generated automatically, without us having to manually create synonyms for each LU, which turned out to be quite difficult.

This approach, however, does have some disadvantages:

- We need to pre-annotate at least 1 sentence per LU in order to have example sentences.
- Having lots of sentences presented at once clutters up the screen and requires scrolling.
- The HIT interface is much more complex and potentially more fragile.

Because of the complexity of the new interface and the increased screen space required for each additional sense, we decided to begin trials on the lemma *justify*.v which (we believe) has just two senses, but still requires a fairly difficult distinction, between the **Deserving** frame, as in *The evolutionary analogy is close enough to JUSTIFY borrowing the term, ...* and the *Justifying* frame, as in *This final section allows Mr Hicks to JUSTIFY the implementation of abc as....* These two sentences were were annotated in the FrameNet data, and were randomly selected to serve as the models for the workers, illustrating the danger of choosing randomly in such cases!

For all HITs, the sentences were randomized in order, as well as the order of the example sentences. Example sentences retained the same colors, i.e. the frame/color correspondence was kept constant, so as not to confuse workers working on multiple HITs. Sentences were horizontally aligned so that the highlighted target word was centered and vertically aligned across the sentences. Each sentence had a drop-down box to its right where workers could select a category to place it in. Each sense category was represented by a model sentence with the frame name as a label for the category. We collected 10 judgments each on 132 sentences, with workers being asked to categorize 18 sentences in each HIT. In the first trial, accuracy was 55%. In trial 2, the model sentences were modified to also show frame element annotation, in the hope that the fact that the **Justifying** uses have an Agent as the subject, while the **Deserving** uses have a State of affairs as the subject would be clearer. An image of the HIT interface, with FE annotation displayed on the model sentences, is shown in Figure 1. Despite the added information, accuracy decreased to 45%.

**Qualifying the prospects**

In trial 3, we kept the HIT interface the same, including the model sentences, but added (1) a qualification test that was designed to evaluate the worker's ability in English, (2) required that the workers have registered a US address with Amazon and (3) required that workers have an overall HIT acceptance rate greater than 75%. Although over 100 workers took the qualification test, no workers accepted the HIT. In trial 4 we raised the rate of pay to $.25/HIT, but still got only 1 worker.

On the suspicion that our problem was partially caused by not having enough HITs to make it worth the workers' time to do them, in Trial 5 we posted the same HITs 3 times, amounting to 24 HITs, worth $6, from a worker's point of view; this raised the number of workers to 5 for all three HITs. Through the HITs completed by those workers, we collected 1 to 2 judgments on 107 of the 132 sentences posted, with 63% accuracy overall, and 86% accuracy on the gold sentences. Looking at their answers for each frame, workers correctly categorized 93% of cases of **Justifying** but only 52% of cases of **Deserving**.

In trial 6, we then customized the instructions (this time automatically, rather than manually) to refer to the lemma specifically rather than via a generic description like "the highlighted word." In addition, we removed the qualification test so as to make our HITs available to a much larger pool of workers, but kept the other two requirements. We ran HITs again with 18 sentences each, 2 of which were gold. We decided to try a different lemma with two sense distinctions, *top*.a, and to make it more worthwhile for workers to annotate our data by posting HITs simultaneously

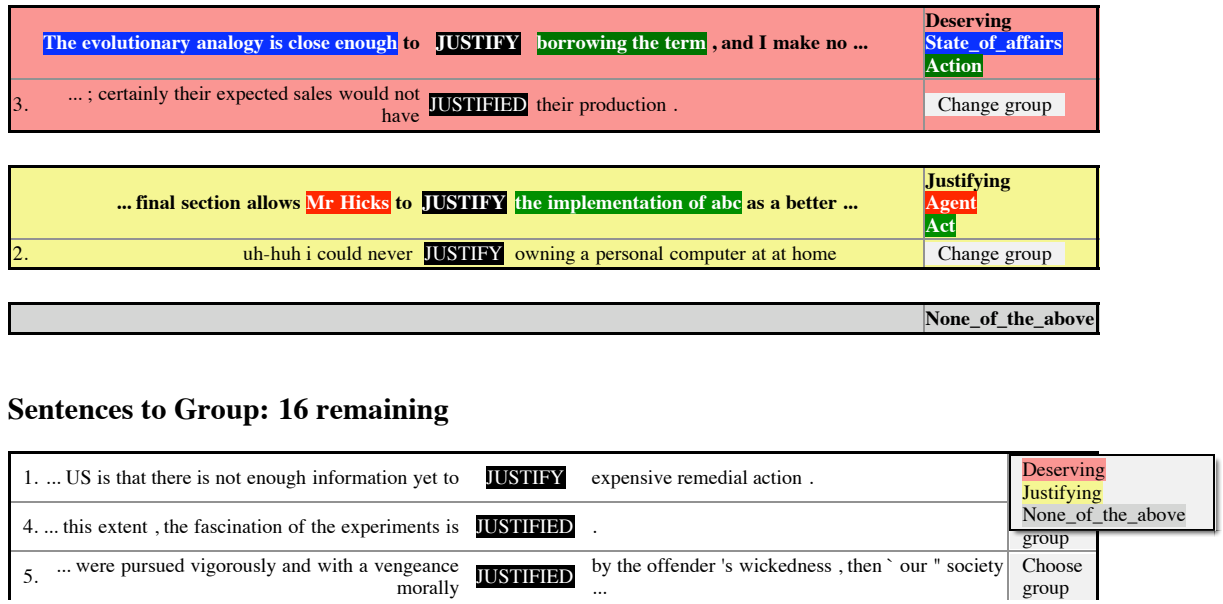


Figure 1: HIT Screen for *justify*.v (after two sentences have been categorized)

for *rip*.v and *high*.a. We posted 8 HITs for *top*.a, 16 HITs for *high*.a and 16 for *rip*.v, for a total of 40 HITs across all three lemmas, paying $.15/HIT and collecting 10 assignments/HIT.

These results were much more satisfactory, with accuracy as shown in Table 3. Filtering out items by raising the agreement criteria (as before) to 35% or greater between the modal response and the next highest, yielded even better accuracy, above 90% for all three lemmas, at the cost of failing to classify approximately 10% to 30% of the items.

In response to the relative success of this trial, we posted HITs for three additional lemmas: *thirst*.n, *range.n*, and *history.n*, with 3, 4, and 5 senses, respectively. We chose these lemmas to ascertain whether there would be an effect on performance from the number of senses. Thus all three lemmas were also of the name POS. For Trial 7, although we kept the same interface, we experimented with changing the pay, and offering bonuses in an effort to maintain good standing among AMT workers concerned with their HIT acceptance record. For previous HITs, workers had to correctly categorize both gold sentences in order to receive any payment. We changed this system so that the HIT is accepted if the worker categorizes 1 gold sentence correctly, and awards a bonus if they categorize both correctly. Our hope was that this change would enable us to experiment with posting difficult HITs without losing our credibility. The results from this trial, also presented in Table 3, show accuracy at 92%, 87%, and 73%, respectively for *thirst*.n, *range.n*, and *history.n*. These results seemed to suggest that increasing the number of senses to discriminate increases the difficulty of the HIT.

It will be recalled that on every item, the workers have a choice "none of the above". One of the difficulties is that this choice covers a variety of cases, including those where the word is the wrong part of speech (a fairly frequent occurrence, despite the high accuracy cited for POS tagging) and those where the needed sense has simply not been included in FrameNet. The latter was the case for the word *range*.n, which was run once with three senses and then again with five senses, after the LUs for *(firing, artillery) range* and the "stove" sense were added. With the two additional senses, the accuracy actually went up from 87% to 92%. Although it is possible that the improvement could be due to a training effect connected to an increase in the number of items, it suggests that having more sense distinctions does not necessarily increase difficulty of discrimination.

| Lemma | No. senses | No. Items | Accuracy | Filtered Items | Accuracy |
|---|---|---|---|---|---|
| *top*.a | 2 | 144 | 92% | 134 | 96% |
| *rip*.v | 4 | 288 | 85% | 228 | 92% |
| *high*.a | 4 | 288 | 80% | 198 | 92% |
| *thirst*.n | 2 | 144 | 92% | 128 | 95% |
| *range*.n | 3 | 216 | 87% | 177 | 93% |
| *history*.n | 4 | 288 | 73% | 199 | 86% |
| *range*.n | 5 | 360 | 92% | 335 | 96% |

Table 3: Results from recent trials, including accuracy after filtering on the basis of agreement

| | N= | Removing 104 | Cause_to_fragment 51 | Self_motion 33 | Damaging 64 | None_of_the_above 36 |
|---|---|---|---|---|---|---|
| Removing | 97 | 93 | 1 | 1 | 2 | 0 |
| Cause_to_fragment | 45 | 1 | 41 | 0 | 1 | 2 |
| Self_motion | 25 | 1 | 0 | 24 | 0 | 0 |
| Damaging | 84 | 8 | 9 | 7 | 58 | 2 |
| None_of_the_above | 37 | 1 | 0 | 1 | 3 | 32 |

Table 4: Confusion matrix for *rip*.v (rows=gold standard)

## 2 What we can learn from the Turkers' difficulties?

Consider the confusion matrix shown in Table 4; here each row represents the items grouped by the gold standard sense ("expected"); each column represents the items grouped by the most frequent worker judgment ("observed").

The accuracy on this HIT set was 85%, in accord with the much larger numbers along the diagonal, but the really interesting cases lie off the diagonal, where the plurality of the workers disagreed with the experts. In some cases, the workers are simply right, and the expert was wrong, as in *This new wave of anonymous buildings . . . has RIPPED the heart out of Hammersmith.*, which the gold standard has as Damaging, but where the workers voted 7 to 3 for Removing. In this case, the expert vanguarder appears to have classified the metaphorical use of *rip*.v using the target domain, rather than the source domain, as is the FrameNet policy on "productive" (rather than "lexicalized") metaphor (Ruppenhofer et al., 2006, Sec. 6.4)[2]. In practice, this classification would most likely have been corrected at the annotation phase, as the FEs are clearly those of the source domain, involving removing something (a Theme) out of something else (a Source). In other cases, such as *I ripped open the envelopes.*, the gold standard correctly has **Damaging**, while the workers have 4 **Removing**, 3 **Cause_to_fragment**, and 3 **Damaging**. There is a good possibility that the envelopes fragmented (although this is not implied, nor necessary to remove a letter from an envelope), and the purpose is likely to remove something from the envelopes, which might falsely suggest **Removing**.

In other cases, the senses are so closely enmeshed, that is seems rather arbitrary to choose one: e.g. *I RIP up an old T-shirt of mine and offer it.* The shirt is certainly damaged and almost certainly fragmented as a result of the same action. *. . . the Oklahoma was RIPPED apart when seven torpedoes hit her.* strictly speaking, the ship is caused to fragment, but the military purpose is to damage her beyond repair, if possible. And there are fairly often examples where the sentence in isolation is ambiguous: *Rain RIPPED another piece of croissant*, *The sky RIPPED and hung in tatters , revealing plasterboard and lath behind.* Such cases are pushing us toward trying to incorporate blending of senses into our paradigm, along the lines of (Erk and McCarthy, 2009).

[2]Available from the FrameNet website, `http://framenet.icsi.berkeley.edu`.

## 3 Conclusion

We have shown that it is possible to set up HITs on Amazon Mechanical Turk to discriminate the fairly fine sense distinctions used in FrameNet, if the right approach is taken, and that the results reach a level of accuracy that can be useful for further processing, as well as serving as a cross-check on the expert data and an invitation to re-think the task itself. Although the total amount of data collected may not be large by some standards, it has been sufficient to give a good sense of which techniques work for the type of WSD problems we are facing. We intend to continue investigating the general applicability of this system for frame disambiguation, including further analysis of our data to better understand the factors that make a disambiguation task more or less difficult for crowd workers. All the data collected in the course of this study, and the software used to collect and analyze it, will be made available on the FrameNet website.

## Acknowledgments

This material is based upon work supported by the National Science Foundation under Grant No. 0947841 (CISE EAGER) "Crowdsourcing for NLP"; the Sketch Engine GUI was developed under NSF Grant IIS-00535297 "Rapid Development of a Frame-Semantic Lexicon".

## References

Chris Callison-Burch and Mark Dredze, editors. 2010. *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, Los Angeles, June. Association for Computational Linguistics.

Katrin Erk and Diana McCarthy. 2009. Graded word sense assignment. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 440–449, Singapore, August. Association for Computational Linguistics.

Michael Heilman and Noah A. Smith. 2010. Rating computer-generated questions with Mechanical Turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 35–40, Los Angeles, June. Association for Computational Linguistics.

Jeff Howe. 2008. *Crowdsourcing*. Crown Business, New York.

Adam Kilgarriff, Pavel Rychly, Pavel Smrz, and David Tugwell. July 2004. The Sketch Engine. In *Proceedings of EURALEX 2004*, Lorient, France.

Bart Mellebeek, Francesc Benavent, Jens Grivolla, Joan Codina, Marta R. Costa-Jussà, and Rafael Banchs. 2010. Opinion mining of spanish customer comments with non-expert annotations on mechanical turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 114–121, Los Angeles, June. Association for Computational Linguistics.

Josef Ruppenhofer, Michael Ellsworth, Miriam R. L. Petruck, Christopher R. Johnson, and Jan Scheffczyk. 2006. *FrameNet II: Extended Theory and Practice*. International Computer Science Institute, Berkeley, California. Distributed with the FrameNet data.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Ng. 2008. Cheap and fast — but is it good? Evaluating non-expert annotations for natural language tasks. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 254–263, Honolulu, Hawaii, October. Association for Computational Linguistics.

Luís von Ahn and Laura Dabbish. 2008. Designing games with a purpose. *Communications of the ACM*, 51:58–67., August.

Luís von Ahn, Benjamin Maurer, Colin McMillen, David Abraham, and Manuel Blum. 2008. reCAPTCHA: Human-based character recognition via web security measures. *Science*, 321(5895):1465–1468.