Amazon Mechanical Turk for Subjectivity Word Sense Disambiguation

Cem Akkaya University of Pittsburgh cem@cs.pitt.edu
Alexander Conrad University of Pittsburgh conrada@cs.pitt.edu
Janyce Wiebe University of Pittsburgh wiebe@cs.pitt.edu
Rada Mihalcea University of North Texas rada@cs.unt.edu

Abstract

Amazon Mechanical Turk (MTurk) is a marketplace for so-called "human intelligence tasks" (HITs), or tasks that are easy for humans but currently difficult for automated processes. Providers upload tasks to MTurk which workers then complete. Natural language annotation is one such human intelligence task. In this paper, we investigate using MTurk to collect annotations for Subjectivity Word Sense Disambiguation (SWSD), a coarse-grained word sense disambiguation task. We investigate whether we can use MTurk to acquire good annotations with respect to gold-standard data, whether we can filter out low-quality workers (spammers), and whether there is a learning effect associated with repeatedly completing the same kind of task. While our results with respect to spammers are inconclusive, we are able to obtain high-quality annotations for the SWSD task. These results suggest a greater role for MTurk with respect to constructing a large scale SWSD system in the future, promising substantial improvement in subjectivity and sentiment analysis.

1 Introduction

Many Natural Language Processing (NLP) systems rely on large amounts of manually annotated data that is collected from domain experts. The annotation process to obtain this data is very laborious and expensive. This makes supervised NLP systems subject to a so-called knowledge acquisition bottleneck. For example, (Ng, 1997) estimates an effort of 16 person years to construct training data for a high-accuracy domain independent Word Sense Disambiguation (WSD) system.

Recently researchers have been investigating Amazon Mechanical Turk (MTurk) as a source of non-expert natural language annotation, which is a cheap and quick alternative to expert annotations (Kaisser and Lowe, 2008; Mrozinski et al., 2008). In this paper, we utilize MTurk to obtain training data for Subjectivity Word Sense Disambiguation (SWSD) as described in (Akkaya et al., 2009). The goal of SWSD is to automatically determine which word instances in a corpus are being used with subjective senses, and which are being used with objective senses. SWSD is a new task which suffers from the absence of a substantial amount of annotated data and thus can only be applied on a small scale. SWSD has strong connections to WSD. Like supervised WSD, it requires training data where target word instances – words which need to be disambiguated by the system – are labeled as having an objective sense or a subjective sense. (Akkaya et al., 2009) show that SWSD may bring substantial improvement in subjectivity and sentiment analysis, if it could be applied on a larger scale. The good news is that training data for 80 selected keywords is enough to make a substantial difference (Akkaya et al., 2009). Thus, large scale SWSD is feasible. We hypothesize that annotations for SWSD can be provided by non-experts reliably if the annotation task is presented in a simple way.

The annotations obtained from MTurk workers are noisy by nature, because MTurk workers are not trained for the underlying annotation task. That is why previous work explored methods to assess annotation quality and to aggregate multiple noisy annotations for high reliability (Snow et al., 2008; Callison-Burch, 2009). It is understandable that not every worker will provide high-quality annotations,

Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk, pages 195–203,
Los Angeles, California, June 2010.

depending on their background and interest. Unfortunately, some MTurk workers do not follow the annotation guidelines and carelessly submit annotations in order to gain economic benefits with only minimal effort. We define this group of workers as spammers. We believe it is essential to distinguish between workers as well-meaning annotators and workers as spammers who should be filtered out as a first step when utilizing MTurk. In this work, we investigate how well the built-in qualifications in MTurk function as such a filter.

Another important question about MTurk workers is whether they learn to provide better annotations over time in the absence of any interaction and feedback. The presence of a learning effect may support working with the same workers over a long time and creating private groups of workers. In this work, we also examine if there is a learning effect associated with MTurk workers.

To summarize, in this work we investigate the following questions:

- Can MTurk be utilized to collect reliable training data for SWSD ?
- Are the built-in methods provided by MTurk enough to avoid spammers ?
- Is there a learning effect associated with MTurk workers ?

The remainder of the paper is organized as follows. In Section 2, we give general background information on the Amazon Mechanical Turk service. In Section 3, we discuss sense subjectivity. In Section 4, we describe the subjectivity word sense disambiguation task. In Section 5, we discuss the design of our experiment and our filtering mechanisms for workers. In Section 6, we evaluate MTurk annotations and relate results to our questions. In Section 7, we review related work. In Section 8, we draw conclusions and discuss future work.

2 Amazon Mechanical Turk

Amazon Mechanical Turk (MTurk)[1] is a marketplace for so-called "human intelligence tasks," or HITs. MTurk has two kinds of users: providers and workers. Providers create HITs using the Mechanical Turk API and, for a small fee, upload them to the HIT database. Workers search through the HIT database, choosing which to complete in exchange for monetary compensation. Anyone can sign up as a provider and/or worker. Each HIT has an associated monetary value, and after reviewing a worker's submission, a provider may choose whether to accept the submission and pay the worker the promised sum or to reject it and pay the worker nothing. HITs typically consist of tasks that are easy for humans but difficult or impossible for computers to complete quickly or effectively, such as annotating images, transcribing speech audio, or writing a summary of a video.

One challenge for requesters using MTurk is that of filtering out spammers and other workers who consistently produce low-quality annotations. In order to allow requesters to restrict the range of workers who can complete their tasks, MTurk provides several types of built-in statistics, known as qualifications. One such qualification is approval rating, a statistic that records a worker's ratio of accepted HITs compared to the total number of HITs submitted by that worker. Providers can require that a worker's approval rating be above a certain threshold before allowing that worker to submit one of his/her HITs. Country of residence and lifetime approved number of HITs completed also serve as built-in qualifications that providers may check before allowing workers to access their HITs.[2] Amazon also allows providers to define their own qualifications. Typically, provider-defined qualifications are used to ensure that HITs which require particular skills are only completed by qualified workers. In most cases, workers acquire provider-defined qualifications by completing an online test.

Amazon also provides a mechanism by which multiple unique workers can complete the same HIT. The number of times a HIT is to be completed is known as the number of assignments for the HIT. By having multiple workers complete the same HIT,

[1] `http://mturk.amazon.com`

[2] According to the terms of use, workers are prohibited from having more than one account, but to the writer's knowledge there is no method in place to enforce this restriction. Thus, a worker with a poor approval rating could simply create a new account, since all accounts start with an approval rating of 100%.

Subjective senses:

His **alarm** grew.
alarm, dismay, consternation – (fear resulting from the awareness of danger)
=> fear, fearfulness, fright – (an emotion experienced in anticipation of some specific pain or danger (usually accompanied by a desire to flee or fight))

What's the **catch**?
catch – (a hidden drawback; "it sounds good but what's the catch?")
=> drawback – (the quality of being a hindrance; "he pointed out all the drawbacks to my plan")

Objective senses:

The **alarm** went off.
alarm, warning device, alarm system – (a device that signals the occurrence of some undesirable event)
=> device – (an instrumentality invented for a particular purpose; "the device is small enough to wear on your wrist"; "a device intended to conserve water")

He sold his **catch** at the market.
catch, haul – (the quantity that was caught; "the catch was only 10 fish")
=> indefinite quantity – (an estimated quantity)

Figure 1: Subjective and objective word sense examples.

techniques such as majority voting among the submissions can be used to aggregate the results for some types of HITs, resulting in a higher-quality final answer. Previous work (Snow et al., 2008) demonstrates that aggregating worker submissions often leads to an increase in quality.

3 Word Sense Subjectivity

(Wiebe and Mihalcea, 2006) define subjective expressions as words and phrases being used to express mental and emotional states, such as speculations, evaluations, sentiments, and beliefs. Many approaches to sentiment and subjectivity analysis rely on lexicons of such words (subjectivity clues). However, such clues often have both subjective and objective senses, as illustrated by (Wiebe and Mihalcea, 2006). Figure 1 provides subjective and objective examples of senses.

(Akkaya et al., 2009) points out that most subjectivity lexicons are compiled as lists of keywords, rather than word meanings (senses). Thus, subjectivity clues used with objective senses – false hits – are a significant source of error in subjectivity and sentiment analysis. SWSD specifically deals with this source of errors. (Akkaya et al., 2009) shows that SWSD helps with various subjectivity and sentiment analysis systems by ignoring false hits.

4 Annotation Task

4.1 Subjectivity Word Sense Disambiguation

Our target task is Subjectivity Word Sense Disambiguation (SWSD). SWSD aims to determine which word instances in a corpus are being used with subjective senses and which are being used with objective senses. It can be considered to be a coarse-grained application-specific WSD that distinguishes between only two senses: (1) the subjective sense and (2) the objective sense.

Subjectivity word sense annotation is done in the following way. We try to keep the annotation task for the worker as simple as possible. Thus, we do not directly ask them if the instance of a target word has a subjective or an objective sense (without any sense inventory), because the concept of subjectivity is fairly difficult to explain to someone who does not have any linguistics background. Instead we show MTurk workers two sets of senses – one subjective set and one objective set – for a specific target word and a text passage in which the target word appears. Their job is to select the set that best reflects the meaning of the target word in the text passage. The specific sense set automatically gives us the subjectivity label of the instance. This makes the annotation task easier for them as (Snow et al., 2008) shows that WSD can be done reliably by MTurk workers. This approach presupposes a set of word senses that have been annotated as subjective or objective. The annotation of senses in a dictionary for subjectivity is not difficult for an expert annotator. Moreover, it needs to be done only once per target word, allowing us to collect hundreds of subjectivity labeled instances for each target word through MTurk.

In this annotation task, we do not inform the MTurk workers about the nature of the sets. This means the MTurk workers have no idea that they are annotating subjectivity of senses; they are just selecting the set which contains a sense matching the usage in the sentence or being as similar to it as possible. This ensures that MTurk workers are not biased by the contextual subjectivity of the sentence while tagging the target word instance.

Sense_Set1 (Subjective)

{ look, **appear**, seem } – give a certain impression or have a certain outward aspect; "She seems to be sleeping"; "This appears to be a very difficult problem"; "This project looks fishy"; "They appeared like people who had not eaten or slept for a long time"

{ **appear**, seem } – seem to be true, probable, or apparent; "It seems that he is very gifted"; "It appears that the weather in California is very bad"

Sense_Set2 (Objective)

{ **appear** } – come into sight or view; "He suddenly appeared at the wedding"; "A new star appeared on the horizon"

{ **appear**, come_out } – be issued or published, as of news in a paper, a book, or a movie; "Did your latest book appear yet?"; "The new Woody Allen film hasn't come out yet"

{ **appear**, come_along } – come into being or existence, or appear on the scene; "Then the computer came along and changed our lives"; "Homo sapiens appeared millions of years ago"

{ **appear** } – appear as a character on stage or appear in a play, etc.; "Gielgud appears briefly in this movie"; "She appeared in 'Hamlet' on the London

{ **appear** } – present oneself formally, as before a (judicial) authority; "He had to appear in court last month"; "She appeared on several charges of theft"

Figure 2: Sense sets for target word "appear".

Below, we describe a sample annotation problem. An MTurk worker has access to the following two sense sets of the target word "appear", as seen in Figure 2. The information that the first sense set is subjective and second sense set is objective is not available to the worker. The worker is presented with the following text passage holding the target word "appear".

> It's got so bad that I don't even know what to say. Charles |target| appeared |target| somewhat embarrassed by his own behavior. The hidden speech was coming, I could tell.

In this passage, the MTurk worker should be able to understand that "appeared" refers to the outward impression given by "Charles". This use of appear is most similar to the first entry in sense set one; thus, the correct answer for this problem is Sense_Set-1.

4.2 Gold Standard

The gold standard dataset, on which we evaluate MTurk worker annotations, is provided by (Akkaya et al., 2009). This dataset (called subjSENSEVAL) consists of target word instances in a corpus labeled as S or O, indicating whether they are used with a subjective or objective sense. It is based on the lexical sample corpora from SENSEVAL1 (Kilgarriff and Palmer, 2000), SENSEVAL2 (Preiss and Yarowsky, 2001), and SENSEVAL3 (Mihalcea and Edmonds, 2004). SubjSENSEVAL consists of instances for 39 ambiguous (having both subjective and objective meanings) target words.

(Akkaya et al., 2009) also provided us with subjectivity labels for word senses which are used in the creation of subjSENSEVAL. Sense labels of the target word senses are defined on the sense inventory of the underlying corpus (Hector for SENSEVAL1; WordNet1.7 for SENSEVAL2; and WordNet1.7.1 for SENSEVAL3). This means the target words from SENSEVAL1 have their senses annotated in the Hector dictionary, while the target words from SENSEVAL2 and SENSEVAL3 have their senses annotated in WordNet1.7. We make use of these labeled sense inventories to build our subjective and objective sets of senses, which we present to the MTurk worker as Sense_Set1 and Sense_Set2 respectively. We want to have a uniform sense representation for the words we ask subjectivity sense labels for. Thus, we consider only SENSEVAL2 and SENSEVAL3 subsets of subjSENSEVAL, because SENSEVAL1 relies on a sense inventory other than WordNet.

5 Experimental Design

We chose randomly 8 target words that have a distribution of subjective and objective instances in subjSENSEVAL with less skew than 75%. That is, no more than 75% of a word's senses are subjective or objective. Our concern is that using skewed data might bias the workers to choose from the more frequent label without thinking much about the problem. Another important fact is that these words with low skew are more ambiguous and responsible for more false hits. Thus, these target words are the ones for which we really need subjectivity word sense disambiguation. For each of these 8 target words, we select 40 passages from subjSENSEVAL in which the target word appears, to include in our experiments. Table 1 summarizes the selected target words

Word	FLP	Word	FLP
appear	55%	fine	72.5%
judgment	65%	solid	55%
strike	62.5%	difference	67.5%
restraint	70%	miss	50%
Average	62.2%		

Table 1: Frequent label percentages for target words.

Group1	Location: USA
Group2	Location: USA HIT Approval Rate $>$ 96%
Group3	Location: USA HIT Approval Rate $>$ 96% Approved HITs $>$ 500

Table 2: Constraints for each HIT group.

and their label distribution. In this table, frequent label percentage (FLP) represents the skew for each word. A word's FLP is equal to the percent of the senses that are of the most frequently occurring type of sense (subjective or objective) for that word.

We believe this annotation task is a good candidate for attracting spammers. This task requires only binary annotations, where the worker just chooses from one of the two given sets, which is not a difficult task. Since it is easy to provide labels, we believe that there will be a distinct line, with respect to quality of annotations, between spammers and mediocre annotators.

For our experiments, we created three different HIT groups each having different qualification requirements but sharing the same data. To be concrete, each HIT group consists of the same 320 instances: 40 instances for each target word listed in Table 1. Each HIT presents an MTurk worker with four instances of the same word in a text passage – this makes 80 HITs for each HIT group – and asks him to choose the set to which the activated sense belongs. We know for each HIT the mapping between sense set numbers and subjectivity. Thus, we can evaluate each HIT response on our gold-standard data, as discussed in Section 4.2. We pay seven cents per HIT. We consider this to be generous compensation for such a simple task.

There are many builtin qualifications in MTurk. We concentrated only on three of them: location, HIT approval rate, and approved HITs, as discussed in Section 2. In our experience, these qualifications are widely used for quality assurance. As mentioned before, we created three different HIT groups in order to see how well different built-in qualification combinations do with respect to filtering spammers. These groups – starting from the least constrained to the most constrained – are listed in Table 2.

Group1 required only that the MTurk workers are located in the US. This group is the least constrained one. Group2 additionally required an approval rate greater than 96%. Group3 is the most constrained one, requiring a lifetime approved HIT number to be greater than 500, in addition to the qualifications in Group1 and Group2.

We believe that neither location nor approval rate and location together is enough to avoid spammers. While being a US resident does to some extent guarantee English proficiency, it does not guarantee well-thought answers. Since there is no mechanism in place preventing users from creating new MTurk worker accounts at will and since all worker accounts are initialized with a 100% approval rate, we do not think that approval rate is sufficient to avoid serial spammers and other poor annotators. We hypothesize that the workers with high approval rate and a large number of approved HITs have a reputation to maintain, and thus will probably be careful in their answers. We think it is unlikely that spammers will have both a high approval rate and a large number of completed HITs. Thus, we anticipated that Group3's annotations will be of higher quality than those of the other groups.

Note that an MTurk worker who has access to the HITs in one of the HIT groups also has access to HITs in less constrained groups. For example, an MTurk worker who has access to HITs in Group3 also has access to HITs in Group2 and Group1. We did not prevent MTurk workers from working in multiple HIT groups because we did not want to influence worker behavior, but instead simulate the most realistic annotation scenario.

In addition to the qualifications described above, we also required each worker to take a qualification test in order to prove their competence in the annotation task. The qualification test consists of 10 sim-

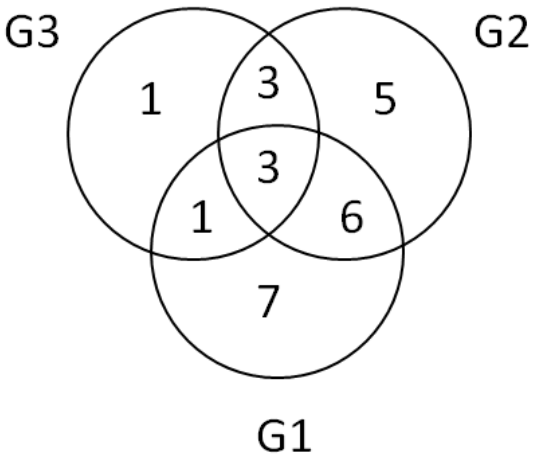

Figure 3: Venn diagram illustrating worker distribution.

ple annotation questions identical in form to those present in the HITs. These questions are split evenly between two target words, "appear" and "restraint". There are a total of five subjective and five objective usages in the test. We required an accuracy of 90% in the qualification test, corresponding to a Kappa score of .80, before a worker was allowed to submit any of our HITs. If a worker failed to achieve a score of 90% on an attempt, that worker could try the test again after a delay of 4 hours.

We collected three sets of assignments within each HIT group. In other words, each HIT was completed three times by three different workers in each group. This gives us a total of 960 assignments in each HIT group. A total of 26 unique workers participated in the experiment: 17 in Group1, 17 in Group2 and 8 in Group3. As mentioned before, a worker is able to participate in all the groups for which he is qualified. Thus the unique worker numbers in each group does not sum up to the total number of workers in the experiment, since some workers participated in the HITs for more than one group. Figure 3 summarizes how workers are distributed between groups.

6 Evaluation

We are interested in how accurate the MTurk annotations are with respect to gold-standard data. We are also interested in how the accuracy of each group differs from the others. We evaluate each group itself separately on the gold-standard data. Additionally, we evaluate each worker's performance on the gold-standard data and inspect their distribution in various groups.

6.1 Group Evaluation

As mentioned in the previous section, we collect three annotations for each HIT. They are assigned to respective trials in the order submitted by the workers. The results are summarized in Table 3. Trials are labeled as T_X and MV is the majority vote annotation among the three trials. The final column contains the baseline agreement where a worker labels each instance of a word with the most frequent label of that word in the gold-standard data. It is clear from this table that, since worker accuracy always exceeds the baseline agreement, subjectivity word sense annotation can be done reliably by MTurk workers. This is very promising. Considering the low cost and low time required to obtain MTurk annotations, a large scale SWSD is realistic. For example, (Akkaya et al., 2009) shows that the most frequent 80 lexicon keywords are responsible for almost half of the false hits in the MPQA Corpus[3] (Wiebe et al., 2005; Wilson, 2008), a corpus annotated for subjective expressions. Utilizing MTurk to collect training data for these 80 lexicon keywords will be quick and cheap and most importantly reliable.

When we compare groups with each other, we see that the best trial result is achieved in Group3. However, according to McNemar's test (Dietterich, 1998), there is no statistically significant difference between any trial of any group. On the other hand, the best majority vote annotation is achieved in Group2, but again there is no statistically significant difference between any majority vote annotation of any group. These results are surprising to us, since we do not see any significant difference in the quality of the data throughout different groups.

6.2 Worker Evaluation

In this section, we evaluate all 26 workers and group them as either spammers or well-meaning workers. All workers who deviate from the gold-standard by a

[3] http://www.cs.pitt.edu/mpqa/

	Group3				Group2				Group1				baseline
	T_1	T_2	T_3	MV	T_1	T_2	T_3	MV	T_1	T_2	T_3	MV	
Accuracy	89.7	86.9	86.6	88.4	87.2	86.3	88.1	90.3	84.4	87.5	87.5	88.4	62.2
Kappa	.79	.74	.73	.77	.74	.73	.76	.81	.69	.75	.75	.77	

Table 3: Accuracy and kappa scores for each group of workers.

Threshold		0.40	0.45	0.50	0.55	0.60	0.65	0.70	0.75
Spammer Count	G1	2	2	2	2	2	4	7	9
	G2	1	2	2	2	2	3	5	8
	G3	0	0	0	0	0	0	2	2
Spammer Percentage	G1	12%	12%	12%	12%	12%	24%	41%	53%
	G2	6%	12%	12%	12%	12%	12%	29%	42%
	G3	0%	0%	0%	0%	0%	0%	25%	25%

Table 4: Spammer representation in groups.

large margin beyond a certain threshold will be considered to be spammers. As discussed in Section 5, we require all participating workers to pass a qualification test before answering HITs. Thus, we know that they are competent to do subjectivity sense annotations, and providing consistently erroneous annotations means that they are probably spammers. We think a kappa score of 0.6 is a good threshold to distinguish spammers from well-meaning workers. For this threshold, we had 2 spammers participating in Group1, 2 spammers in Group2 and 0 spammers in Group3. Table 4 presents spammer count and spammer percentage in each group for various threshold values. We see that Group3 has consistently fewer spammers and a smaller spammer percentage. The lowest kappa scores for Group1, Group2, and Group3 are .35, .40, and .69, respectively. The mean kappa scores for Group1, Group2, and Group3 are .73, .75, and .77, respectively.

These results indicate that Group3 is less prone to spammers, apparently contradicting Section 6.1. We see the reason when we inspect the data more closely. It turns out that spammers contributed in Group1 and Group2 only minimally. On the other hand there are two mediocre workers (Kappa of 0.69) who submit around 1/3 of the HITs in Group3. This behavior might be a coincidence. In the face of contradicting results, we think that we need a more extensive study to derive conclusions about the relation between spammer distribution and built-in qualification.

6.3 Learning Effect

Expert annotators can learn to provide more accurate annotations over time. (Passonneau et al., 2006) reports a learning effect early in the annotation process. This might be due to the formal and informal interaction between annotators. Another possibility is that the annotators might get used to the annotation task over time. This is to be expected if there is not an extensive training process before the annotation takes place.

On the other hand, the MTurk workers have no interaction among themselves. They do not receive any formal training and do not have access to true annotations except a few examples if provided by the requester. These properties make MTurk workers a unique annotation workforce. We are interested if the learning effect common to expert annotators holds in this unique workforce in the absence of any interaction and feedback. That may justify working with the same set of workers over a long time by creating private groups of workers.

We sort annotations of a worker after the submission date. This way, we get for each worker an ordered list of annotations. We split the list into bins of size 40 and we test for an increasing trend in the proportion of successes over time. We use the Chi-squared Test for binomial proportions (Rosner, 2006). Using this test, we find that all of the p-values

are substantially larger than 0.05. Thus, there is no increasing trend in the proportion of successes and no learning effect. This is true for both mediocre workers and very reliable workers. We think that the results may differ for harder annotation tasks where the input is more complex and requires some adjustment.

7 Related Work

There has been recently an increasing interest in Amazon Mechanical Turk. Many researchers have utilized MTurk as a source of non-expert natural language annotation to create labeled datasets. In (Mrozinski et al., 2008), MTurk workers are used to create a corpus of why-questions and corresponding answers on which QA systems may be developed. (Kaisser and Lowe, 2008) work on a similar task. They make use of MTurk workers to identify sentences in documents as answers and create a corpus of question-answer sentence pairs. MTurk is also considered in other fields than natural language processing. For example, (Sorokin and Forsyth, 2008) utilizes MTurk for image labeling. Our ultimate goal is similar; namely, to build training data (in our case for SWSD).

Several studies have concentrated specifically on the quality aspect of the MTurk annotations. They investigated methods to assess annotation quality and to aggregate multiple noisy annotations for high reliability. (Snow et al., 2008) report MTurk annotation quality on various NLP tasks (e.g. WSD, Textual Entailment, Word Similarity) and define a bias correction method for non-expert annotators. (Callison-Burch, 2009) uses MTurk workers for manual evaluation of automatic translation quality and experiments with weighed voting to combine multiple annotations. (Hsueh et al., 2009) define various annotation quality measures and show that they are useful for selecting annotations leading to more accurate classifiers. Our work investigates the effect of built-in qualifications on the quality of MTurk annotations.

(Hsueh et al., 2009) applies MTurk to get sentiment annotations on political blog snippets. (Snow et al., 2008) utilizes MTurk for affective text annotation task. In both works, MTurk workers annotated larger entities but on a more detailed scale than we do. (Snow et al., 2008) also provides a WSD annotation task which is similar to our annotation task. The difference is the MTurk workers are choosing an exact sense not a sense set.

8 Conclusion and Future Work

In this paper, we address the question of whether built-in qualifications are enough to avoid spammers. The investigation of worker performances indicates that the lesser constrained a group is the more spammers it attracts. On the other hand, we did not find any significant difference between the quality of the annotations for each group. It turns out that workers considered as spammers contributed only minimally. We do not know if it is just a coincidence or if it is correlated to the task definition. We did not get conclusive results. We need to do more extensive experiments before arriving at conclusions.

Another aspect we investigated is the learning effect. Our results show that there is no improvement in annotator reliability over time. We should not expect MTurk workers to provide more consistent annotations over time. This will probably be the case in similar annotation tasks. For harder annotation tasks (e.g. parse tree annotation) things may be different. An interesting follow-up would be whether showing the answers of other workers on the same HIT will promote learning.

We presented our subjectivity sense annotation task to the worker in a very simple way. The annotation results prove that subjectivity word sense annotation can be done reliably by MTurk workers. This is very promising since the MTurk annotations can be collected for low costs in a short time period. This implies that a large scale general SWSD component, which can help with various subjectivity and sentiment analysis tasks, is feasible. We plan to work with selected workers to collect new annotated data for SWSD and use this data to train a SWSD system.

Acknowledgments

This material is based in part upon work supported by National Science Foundation awards IIS-0916046 and IIS-0917170 and by Department of Homeland Security award N000140710152. The authors are grateful to the three paper reviewers for their helpful suggestions.

References

Cem Akkaya, Janyce Wiebe, and Rada Mihalcea. 2009. Subjectivity word sense disambiguation. In *Conference on Empirical Methods in Natural Language Processing (EMNLP 2009)*.

Chris Callison-Burch. 2009. Fast, cheap, and creative: evaluating translation quality using amazon's mechanical turk. In *EMNLP '09: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 286–295, Morristown, NJ, USA. Association for Computational Linguistics.

Thomas G. Dietterich. 1998. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10:1895–1923.

Pei-Yun Hsueh, Prem Melville, and Vikas Sindhwani. 2009. Data quality from crowdsourcing: a study of annotation selection criteria. In *HLT '09: Proceedings of the NAACL HLT 2009 Workshop on Active Learning for Natural Language Processing*, pages 27–35, Morristown, NJ, USA. Association for Computational Linguistics.

Michael Kaisser and John Lowe. 2008. Creating a research collection of question answer sentence pairs with amazons mechanical turk. In *Proceedings of the Sixth International Language Resources and Evaluation (LREC'08)*. http://www.lrec-conf.org/proceedings/lrec2008/.

Joanna Mrozinski, Edward Whittaker, and Sadaoki Furui. 2008. Collecting a why-question corpus for development and evaluation of an automatic QA-system. In *Proceedings of ACL-08: HLT*, pages 443–451, Columbus, Ohio, June. Association for Computational Linguistics.

Hwee Tou Ng. 1997. Getting serious about word sense disambiguation. In *Proceedings of the ACL SIGLEX Workshop on Tagging Text with Lexical Semantics: Why,What, and How?*

Rebecca Passonneau, Nizar Habash, and Owen Rambow. 2006. Inter-annotator agreement on a multilingual semantic annotation task. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC)*.

Bernard Rosner. 2006. *Fundamentals of Biostatistics*. Thompson Brooks/Cole.

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 '08: Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 254–263, Morristown, NJ, USA. Association for Computational Linguistics.

A. Sorokin and D. Forsyth. 2008. Utility data annotation with amazon mechanical turk. pages 1 –8, june.

J. Wiebe and R. Mihalcea. 2006. Word sense and subjectivity. In *(ACL-06)*, Sydney, Australia.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language Resources and Evaluation (formerly Computers and the Humanities)*, 39(2/3):164–210.

Theresa Wilson. 2008. *Fine-grained Subjectivity and Sentiment Analysis: Recognizing the Intensity, Polarity, and Attitudes of private states*. Ph.D. thesis, Intelligent Systems Program, University of Pittsburgh.