Clustering dictionary definitions using Amazon Mechanical Turk

Gabriel Parent **Maxine Eskenazi**

Language Technologies Institute
Carnegie Mellon University
5000 Forbes Avenue
15213 Pittsburgh, USA

{gparent,max}@cs.cmu.edu

Abstract

Vocabulary tutors need word sense disambiguation (WSD) in order to provide exercises and assessments that match the sense of words being taught. Using expert annotators to build a WSD training set for all the words supported would be too expensive. Crowdsourcing that task seems to be a good solution. However, a first required step is to define what the possible sense labels to assign to word occurrence are. This can be viewed as a clustering task on dictionary definitions. This paper evaluates the possibility of using Amazon Mechanical Turk (MTurk) to carry out that prerequisite step to WSD. We propose two different approaches to using a crowd to accomplish clustering: one where the worker has a global view of the task, and one where only a local view is available. We discuss how we can aggregate multiple workers' clusters together, as well as pros and cons of our two approaches. We show that either approach has an interannotator agreement with experts that corresponds to the agreement between experts, and so using MTurk to cluster dictionary definitions appears to be a reliable approach.

1 Introduction

For some applications it is useful to disambiguate the meanings of a polysemous word. For example, if we show a student a text containing a word like "bank" and then automatically generate questions about the meaning of that word as it appeared in the text (say as the bank of a river), we would like to have the meaning of the word in the questions match the text meaning. Teachers do this each time they assess a student on vocabulary knowledge. For intelligent tutoring systems, two options are available. The first one is to ask a teacher to go through all the material and label each appearance of a polysemous word with its sense. This option is used only if there is a relatively small quantity of material. Beyond that, automatic processing, known as Word Sense Disambiguation (WSD) is essential. Most approaches are supervised and need large amounts of data to train the classifier for each and every word that is to be taught and assessed.

Amazon Mechanical Turk (MTurk) has been used for the purpose of word sense disambiguation (Snow et al, 2008). The results show that non-experts do very well (100% accuracy) when asked to identify the correct sense of a word out of a finite set of labels created by an expert. It is therefore possible to use MTurk to build a training corpus for WSD. In order to extend the Snow et al crowdsourced disambiguation to a large number of words, we need an efficient way to create the set of senses of a word. Asking an expert to do this is costly in time and money. Thus it is necessary to have an efficient Word Sense Induction (WSI) system. A WSI system induces the different senses of a word and provides the corresponding sense labels. This is the first step to crowdsourcing WSD on a large scale.

While many studies have shown that MTurk can be used for labeling tasks (Snow et al, 2008), to rate automatically constructed artifacts (Callison-Burch, 2009, Alonso et al, 2008) and to transcribe speech (Ledlie et al, 2009, Gruenstein et al, 2009), to our knowledge, there has not been much work on evaluating the use of MTurk for

clustering tasks. The goal of this paper is to investigate different options available to crowdsource a clustering task and evaluate their efficiency in the concrete application of word sense induction.

2 Background

2.1 WSD for vocabulary tutoring

Our interest in the use of MTurk for disambiguation comes from work on a vocabulary tutor; REAP (Heilman et al, 2006). The tutor searches for documents from the Web that are appropriate for a student to use to learn vocabulary from context (appropriate reading level, for example). Since the system finds a large number of documents, making a rich repository of learning material, it is impossible to process all the documents manually. When a document for vocabulary learning is presented to a student, the system should show the definition of the words to be learned (focus words). In some cases a word has several meanings for the same part of speech and thus it has several definitions. Hence the need for WSD to be included in vocabulary tutors.

2.2 WSI and WSD

The identification of a list of senses for a given word in a corpus of documents is called word sense induction (WSI). SemEval 2007 and 2010 (SigLex, 2008) both evaluate WSI systems. The I2R system achieved the best results in 2007 with an F-score of 81.6% (I2R by Niu (2007)). Snow et al (2007) have a good description of the inherent problem of WSI where the appropriate granularity of the clusters varies for each application. They try to solve this problem by building hierarchical-like word sense structures. In our case, each dictionary definition for a word could be considered as a unique sense for that word. Then, when using MTurk as a platform for WSD, we could simply ask the workers to select which of the dictionary definitions best expresses the meaning of the words in a document. The problem here is that most dictionaries give quite several definitions for a word. Defining one sense label per dictionary definition would result in too many labels, which would, in turn, make the MTurk WSD less efficient and our dataset sparser, thus decreasing the quality of the classifier. Another option, investigated by Chklovski and Mihalcea (2003), is to use WordNet sense definitions as the possible labels. They obtained more than 100,000 labeled instances from a crowd of volunteers. They conclude that WordNet senses are not coarse enough to provide high interannotator agreement, and exploit workers disagreement on the WSD task to derive coarser senses.

The granularity of the senses for each word is a parameter that is dependent on the application. In our case, we want to be able to assess a student on the sense of a word that the student has just been taught. Learners have the ability to generalize the context in which a word is learned. For example, if a student learns the meaning of the word "bark" as the sound of a dog, they can generalize that this can also apply to human shouting. Hence, there is no need for two separate senses here. However, a student could not generalize the meaning "hard cover of a tree" from that first meaning of "bark". This implies that students should be able to distinguish coarse word senses. (Kulkarni et al., 2007) have looked at automatic clustering of dictionary definitions. They compared K-Means clustering with Spectral Clustering. Various features were investigated: raw, normalized word overlap with and without stop words. The best combination results in 74% of the clusters having no misclassified definitions. If those misclassified definitions end up being used to represent possible sense labels in WSD, wrong labels might decrease the quality of the disambiguation stage. If a student is shown a definition that does not match the sense of a word in a particular context, they are likely to build the wrong conceptual link. Our application requires higher accuracy than that achieved by automatic approaches, since students' learning can be directly affected by the error rate.

2.3 Clustering with MTurk

The possible interaction between users and clustering algorithms has been explored in the past. Huang and Mitchell (2006) present an example of how user feedback can be used to improve clustering results. In this study, the users were not asked to provide clustering solutions. Instead, they fine tuned the automatically generated solution.

With the advent of MTurk, we can use human judgment to build clustering solutions. There are multiple approaches for combining workforce: parallel with aggregation (Snow et al, 2008), iterative

(Little et al, 2009) and collaboration between workers (Horton, Turker Talk, 2009). These strategies have been investigated for many applications, most of which are for labeling, a few for clustering. The Deneme blog presents an experiment where website clustering is carried out using MTurk (Little, Website Clustering, 2009). The workers' judgments on the similarity between two websites are used to build a distance matrix for the distance between websites. Jagadeesan and others (2009) asked workers to identify similar objects in a pool of 3D CAD models. They then used frequently co-occurring objects to build a distance matrix, upon which they then applied hierarchical clustering. Those two approaches are different: the first gives the worker only two items of the set (a local view of the task), while the latter offers the worker a global view of the task. In the next sections we will measure the accuracy of these approaches and their advantages and disadvantages.

3 Obtaining clusters from a crowd

REAP is used to teach English vocabulary and to conduct learning studies in a real setting, in a local ESL school. The vocabulary tutor provides instructions for the 270 words on the school's core vocabulary list, which has been built using the Academic Word List (Coxhead, 2000). In order to investigate how WSI could be accomplished using Amazon Mechanical Turk, 50 words were randomly sampled from the 270, and their definitions were extracted from the Longman Dictionary of Contemporary English (LDOCE) and the Cambridge Advanced Learner's Dictionary (CALD). There was an average of 6.3 definitions per word.

The problem of clustering dictionary definitions involves solving two sub-problems: how many clusters there are, and which definitions belong to which clusters. We could have asked workers to solve both problems at the same time by having them dynamically change the number of clusters in our interface. We decided not to do this due to the fact that some words have more than 12 definitions. Since the worker already needs to keep track of the semantics of each cluster, we felt that having them modify the number of sense boxes would increase their cognitive load to the point that we would see a decrease in the accuracy of the results.

Thus the first task involved determining the number of general meanings (which in our case determines the number of clusters) that there are in a list of definitions. The workers were shown the word and a list of its definitions, for example, for the word "clarify":

- *to make something clearer and easier to understand*
- *to make something clear or easier to understand by giving more details or a simpler explanation*
- *to remove water and unwanted substances from fat, such as butter, by heating it*

They were then asked: "How many general meanings of the word *clarify* are there in the following definitions?" We gave a definition of what we meant by general versus specific meanings, along with several examples. The worker was asked to enter a number in a text box (in the above example the majority answered 2). This 2-cent HIT was completed 13 times for every 50 words, for a total of 650 assignments and $13.00. A majority vote was used to aggregate the workers' results, giving us the number of clusters in which the definitions were grouped. In case of a tie, the lowest number of clusters was retained, since our application requires coarse-grained senses.

The number of "general meanings" we obtained in this first HIT[1] was then used in two different HITs. We use these two HITs to determine which definitions should be clustered together. In the first setup, which we called "global-view" the workers had a view of the entire task. They were shown the word and **all of its definitions**. They were then prompted to drag-and-drop the definitions into different sense boxes, making sure to group the definitions that belong to the same general meaning together (Figure 3, Appendix). Once again, an explicit definition of what was expected for "general meaning" along with examples was given. Also, a flash demo of how to use the interface was provided. The worker got 3 cents for this HIT. It was completed 5 times for each of the 50 words, for a total cost of $7.50. We created another HIT where the workers were not given all of the definitions; we called this setup "local-view". The worker was asked to indicate if **two definitions** of a word were related to the same meaning or different meanings

[1] The code and data used for the different HITs are available at http://www.cs.cmu.edu/~gparent/amt/wsi/

(Figure 4, Appendix). For each word, we created all possible pairs of definitions. This accounts for an average of 21 pairs for all of the 50 words. For each pair, 5 different workers voted on whether it contained the same or different meanings, earning 1 cent for each answer. The total cost here was $52.50. The agreement between workers was used to build a distance matrix: if the 5 workers agreed that the two definitions concerned the same sense, the distance was set to 0. Otherwise, it was set to the number of workers who thought they concerned different senses, up to a distance of 5. Hierarchical clustering was then used to build clustering solutions from the distance matrices. We used complete linkage clustering, with Ward's criterion.

4 Evaluation of global-view vs. local-view approaches

In order to evaluate our two approaches, we created a gold-standard (GS). Since the task of WSI is strongly influenced by an annotator's grain size preference for the senses, four expert annotators were asked to create the GS. The literature offers many metrics to compare two annotators' clustering solutions (Purity and Entropy (Zhao and Karypis, 2001), clustering F-Measure (Fung et al., 2003) and many others). SemEval-2 includes a WSI task where V-Measure (Rosenberg and Hirschberg, 2007) is used to evaluate the clustering solutions. V-Measure involves two metrics, homogeneity and completeness, that can be thought of as precision and recall. Perfect homogeneity is obtained if the solutions have clusters whose data points belong to a single cluster in the GS. Perfect completeness is obtained if the clusters in the GS contain data points that belong to a single cluster in the evaluated solution. The V-Measure is a (weighted) harmonic mean of the homogeneity and of the completeness metrics. Table 1 shows inter-annotator agreement (ITA) among four experts on the test dataset, using the average V-Measure over all the 50 sense clusters.

	GS #1	GS #2	GS #3	GS #4
GS #1	1,000	0,850	0,766	0,770
GS #2	0,850	1,000	0,763	0,796
GS #3	0,766	0,763	1,000	0,689
GS #4	0,770	0,796	0,689	1,000

Table 1 - ITA on WSI task for four annotators

We can obtain the agreement between one expert and the three others by averaging the three V-Measures. We finally obtain an "Experts vs. Experts" ITA of 0.772 by averaging this value for all of our experts. The standard deviation for this ITA is 0.031.To be considered reliable, non-expert clustering would have to agree with the 4 experts with a similar result.

5 Aggregating clustering solutions from multiple workers

Using a majority vote with the local-view HIT is an easy way of taking advantage of the "wisdom of crowd" principle. In order to address clustering from a local-view perspective, we need to build all possible pairs of elements. The number of those pairs is $O(n^2)$ on the number of elements to cluster. Thus the cost grows quickly for large clustering problems. For 100 elements to cluster there are 4950 pairs of elements to show to workers. For large problems, a better approach would be to give the problem to multiple workers through global-view, and then find a way to merge all of the clustering solutions to benefit from the wisdom of crowd. Consensus clustering (Topchy et al, 2005) has emerged as a way of combining multiple weak clusterings into a better one. The cluster-based similarity partitioning algorithm (CSPA) (Strehl and Ghosh, 2002) uses the idea that elements that are frequently clustered together have high similarity. With MTurk, this involves asking multiple workers to provide full clusterings, and then, for each pair of elements, counting the number of times they co-occur in the same clusters. This count is used as a similarity measure between elements, which then is used to build a distance matrix. We can then use it to recluster elements. The results from this technique on our word sense induction problem are shown in the next section.

	Random	K-Means	*local*	*global* CSPA	*global* centroid
GS #1	0,387	0,586	0,737	0,741	0,741
GS #2	0,415	0,613	0,765	0,777	0,777
GS #3	0,385	0,609	0,794	0,805	0,809
GS #4	0,399	0,606	0,768	0,776	0,776
Avg. ITA	0.396 ± 0.014	0.603 ± 0.012	**0.766 ± 0.023**	**0.775 ± 0.026**	**0.776 ± 0.028**

Table 2 - Interannotator agreement for our different approaches (bold numbers are within one standard deviation of the Expert vs. Expert ITA of 0.772 ± 0.031 described in section 4)

Another possibility is to determine which clustering solution is the centroid of the set of clusterings obtained from the worker. Finding centroid clustering (Hu and Sung, 2006) requires a between-cluster distance metric. We decided to use the entropy-based V-Measure for this purpose. For every pair of workers' solutions, we obtain their relative distance by calculating

1-VMeasure(cluster #1,cluster #2).

Then, for each candidate's clusters, we average the distance with every other candidate's. The candidate with the lowest average distance, the centroid, is picked as the "crowd solution". Results from this technique are also shown in the next section.

6 Results

For the first HIT the goal was to determine the number of distinct senses in a list of definitions. The Pearson correlation between the four annotators on the number of clusters they used for the 50 words was computed. These correlations can be viewed as how much the different annotators had the same idea of the grain size to be used to define senses. While experts 1, 2 and 4 seem to agree on grain size (correlation between 0.71 and 0.75), expert 3 had a different opinion. Correlations between that expert and the three others are between 0.53 and 0.58. The average correlation between experts is 0.63. On the other hand, the crowd solution does not agree as well with experts #1,#2 and #4 (Pearson correlation of 0.64, 0.68, 0.66), while it better approaches expert 3, with a correlation of 0.68. The average correlation between the non-expert solution and the experts' solutions is 0.67.

	GS #1	GS #2	GS #3	GS #4	N-E
GS #1	0	24	26	29	26
GS #2	24	0	30	27	26
GS #3	26	30	0	37	20
GS #4	29	27	37	0	27
N-E	26	26	20	27	0
Average	**26.25**	**26.75**	**28.25**	**30**	**24.75**

Table 3 - Absolute difference of number of clusters

Another way to analyze the agreement on grain size of the word sense between annotators is to sum the absolute difference of number of clusters for the 50 words (Table 3). In this way, we can specifically examine the results for the four annotators and for the non-expert crowd (N-E) solution, averaging that difference for each annotator versus all of the others (including the N-E solution).

To determine how a clustering solution compared to our GS, we computed the V-Measure for all 50 words between the solution and each GS. By averaging the score on the four GSs, we get an averaged ITA score between the clustering solution and the experts. For the sake of comparison, we first computed the score of a random solution, where definitions are randomly assigned to any one cluster. We also implemented K-means clustering using normalized word-overlap (Kulkarni et al., 2007), which has the best score on their test set.

The resulting averaged ITA of our local-view approaches that of all 4 experts. We did the same with the global-view after applying CSPA and our centroid identification algorithm to the 5 clustering solutions the workers submitted. Table 2 shows the agreement between each expert and those approaches, as well as the averaged ITA.

For the local-view and global-view "centroid", we looked at how the crowd size would affect the accuracy. We first computed the averaged ITA by considering the answers from the first worker. Then, step by step, we added the answers from the second, third, fourth and fifth workers, each time computing the averaged ITA. Figure 1 shows the ITA as a function of the workers.

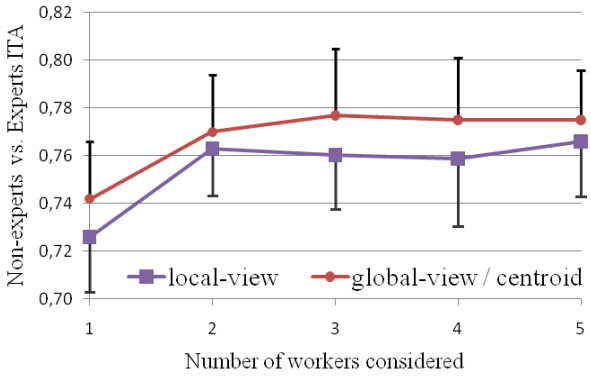

Figure 1 - Impact of the crowd size on the ITA of the local and global approaches

7 Discussion

Since our two approaches are based on the result of the first HIT, which determines the number of clusters, the accuracy of that first task is extremely important. It turns out that the correlation between the crowd solution and the experts (0.67) is actually higher than the average correlation between experts (0.63). One way to explain this is that of the 4 experts, 3 had a similar opinion on what the grain size should be, while the other one had a different opinion. The crowd picked a grain size that was actually between those two opinions, thus resulting in a higher correlation. This hypothesis is also supported by Table 3. The average difference in the number of clusters is lower for the N-E solution than for any expert solution. The crowd of 13 was able to come up with a grain size that could be seen as a good consensus of the four annotators' grain size. This allows us to believe that using the crowd to determine the number of clusters for our two approaches is a reliable technique.

As expected, Table 3 indicates that our two setups behave better than randomly assigning definitions to clusters. This is a good indication that the workers did not complete our tasks randomly. The automatic approach (K-Means) clearly behaves better than the random baseline. However, the clusters obtained with this approach agree less with the experts than any of our crowdsourced approaches. This confirms the intuition that humans are better at distinguishing word senses than an automatic approach like K-Means.

Our first hypothesis was that global-view would give us the best results: since the worker completing a global-view HIT has an overall view of the task, they should be able to provide a better solution. The results indicate that the local-view and global-view approaches give similar results in terms of ITA. Both of those approaches have closer agreement with the experts, than the experts have with each other (all ITAs are around 77%).

Here is an example of a solution that the crowd provided through local-view for the verb 'tape' with the definitions;

A. To record something on tape
B. To use strips of sticky material, especially to fix two things together or to fasten a parcel
C. To record sound or picture onto a tape
D. To tie a bandage firmly around an injured part of someone's body, strap
E. To fasten a package, box etc with tape

The crowd created two clusters: one by grouping A and C to create a "record audio/video" sense, and another one by grouping B,D and E to create a "fasten" sense. This solution was also chosen by two of the four experts. One of the other experts grouped definitions E with A and C, which is clearly an error since there is no shared meaning. The last expert created three clusters, by assigning D to a different cluster than B and E. This decision can be considered valid since there is a small semantic distinction between D and B/E from the fact that D is "fasten" for the specific case of injured body parts. However, a student could generalize D from B and E. So that expert's grain size does not correspond to our specifications.

We investigated two different aggregation techniques for clustering solutions, CSPA and centroid identification. In this application, both techniques give very similar results with only 2 clusters out of 50 words differing between the two techniques. Centroid identification is easier to implement, and doesn't require reclustering the elements. Figure 1 shows the impact of adding more workers to the crowd. While it seems advantageous to use 3 workers' opinions rather than only 1, (gain of 0.04), adding a fourth and fifth worker does not improve the average ITA.

Local-view is more tolerant to errors than global-view. If a chaotic worker randomly answers one pair of elements, the entire final clustering will not be affected. If a chaotic (or cheating) worker answers randomly in global-view, the entire clustering solution will be random. Thus, while a policy of using only one worker's answer for a local-view

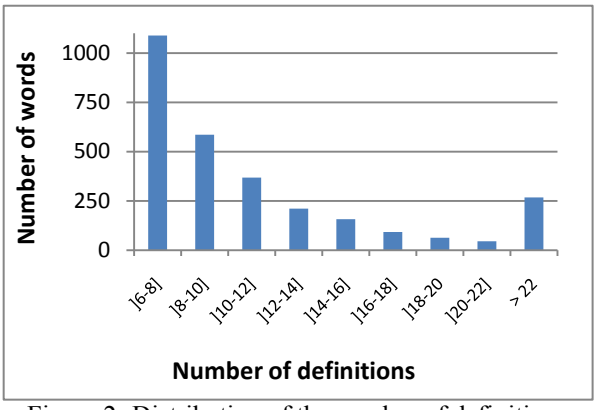

Figure 2- Distribution of the number of definitions

HIT could be adopted, the same policy might result in poor clustering if used for the global-view HIT.

However, global-view has the advantage over local-view of being cheaper. Figure 2 shows the distribution of the number of definitions extracted from both LDOCE and CALD per word (starting at word with more than 6 definitions). Since the local-view cost increases in a quadratic manner as the number of elements to cluster increases it would cost more than $275,000 to group the definitions of 30,000 words coming from the two dictionaries (using the parameters described in 3). It would be possible to modify it to only ask workers for the similarity of a subset of pairs of elements and then reconstruct the incomplete distance matrix (Hathaway and Bezdek, 2002). A better option for clustering a very large amount of elements is to use global-view. For the same 30,000 words above, the cost of grouping definitions using this technique would be around $4,500. This would imply that worker would have to create clusters from set of over 22 definitions. Keeping the cost constant while increasing the number of elements to cluster might decrease the workers' motivation. Thus scaling up a global-view HIT requires increasing the reward. It also requires vigilance on how much cognitive load the workers have to handle. Cognitive load can be seen as a function of the number of elements to cluster and of the number of clusters that a new element can be assigned to. If a worker only has to decide if an element should be in A or B, the cognitive load is low. But if the worker has to decide among many more classes, the cognitive load may increase to a point where the worker is hampered from providing a correct answer.

8 Conclusion

We evaluated two different approaches for crowdsourcing dictionary definition clustering as a means of achieving WSI. Global-view provides an interface to the worker where all the elements to be clustered are displayed, while local-view displays only two elements at a time and prompts the worker for their similarity. Both approaches show as much agreement with experts as the experts do with one another. Applying either CSPA or centroid identification allows the solution to benefit from the wisdom of crowd effect, and shows similar results. While global-view is cheaper than local-view, it is also strongly affected by worker error, and sensitive to the effect of increased cognitive load.

It appears that the task of clustering definitions to form word senses is a subjective one, due to different ideas of what the grain size of the senses should be. Thus, even though it seems that our two approaches provide results that are as good as those of an expert, it would be interesting to try crowdsourced clustering on a clustering problem where an objective ground truth exists. For example, we could take several audio recordings from each of several different persons. After mixing up the recordings from the different speakers, we could ask workers to clusters all the recordings from the same person. This would provide an even stronger evaluation of local-view against global-view since we could compare them to the true solution, the real identity of the speaker.

There are several interesting modifications that could also be attempted. The local-view task could ask for similarity on a scale of 1 to 5, instead of a binary choice of same/different meaning. Also, since using global-view with one large problem causes high cognitive load, we could partition a bigger problem, e.g., with 30 definitions, into 3 problems including 10 definitions. Using the same interface as global-view, the workers could cluster the sub-problems. We could then use CSPA to merge local clusters into a final cluster with the 30 definitions.

In this paper we have examined clustering word sense definitions. Two approaches were studied, and their advantages and disadvantages were described. We have shown that the use of human computation for WSI, with an appropriate crowd

size and mean of aggregation, is as reliable as using expert judgments.

Acknowledgements

Funding for this research is provided by the National Science Foundation, Grant Number SBE-0836012 to the Pittsburgh Science of Learning Center (PSLC, http://www.learnlab.org).

References

Alonso, O., Rose, D. E., & Stewart, B. (2008). Crowdsourcing for relevance evaluation. ACM SIGIR Forum , 42 (2), pp. 9-15.

Callison-Burch, C. (2009). Fast, Cheap, and Creative: Evaluating Translation Quality Using Amazon's Mechanical Turk. Proceedings of EMNLP 2009.

Coxhead, A. (2000). A new academic word list. TESOL quarterly , 34 (2), 213-238.

Chklovski, T. & Mihalcea, R. (2003). Exploiting agreement and disagreement of human annotators for word sense disambiguation. Proceedings of RANLP 2003.

Fung, B. C., Wang, K., & Ester, M. (2003). Hierarchical document clustering using frequent itemsets. Proc. of the SIAM International Conference on Data Mining.

Gruenstein, A., McGraw, I., & Sutherland, A. (2009). "A self-transcribing speech corpus: collecting continuous speech with an online educational game". SLaTE Workshop.

Hathaway, R. J., & Bezdek, J. C. (2001). Fuzzy c-means clustering of incomplete data. IEEE Transactions on Systems, Man, and Cybernetics, Part B , 31 (5), 735-744.

Heilman, M. Collins-Thompson, K., Callan, J. & Eskenazi M. (2006). Classroom success of an Intelligent Tutoring System for lexical practice and reading comprehension. Proceedings of the Ninth International Conference on Spoken Language.

Horton, J. (2009, 12 11). Turker Talk. Retrieved 01 2010, from Deneme: http://groups.csail.mit.edu/uid/deneme/?p=436

Hu, T., & Sung, S. Y. (2006). Finding centroid clusterings with entropy-based criteria. Knowledge and Information Systems , 10 (4), 505-514.

Huang, Y., & Mitchell, T. M. (2006). Text clustering with extended user feedback. Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval (p. 420). ACM.

Jagadeesan, A., Lynn, A., Corney, J., Yan, X., Wenzel, J., Sherlock, A., et al. (2009). Geometric reasoning via internet CrowdSourcing. 2009 SIAM/ACM Joint Conference on Geometric and Physical Modeling (pp. 313-318). ACM.

Kulkarni, A., Callan, J., & Eskenazi, M. (2007). Dictionary Definitions: The Likes and the Unlikes. Proceedings of the SLaTE Workshop on Speech and Language Technology in Education. Farmington, PA, USA.

Ledlie, J., Odero, B., Minkow, E., Kiss, I., & Polifroni, J. (2009). Crowd Translator: On Building Localized Speech Recognizers through Micropayments. Nokia Research Center.

Little, G. (2009, 08 22). Website Clustering. Retrieved 01 2010, from Deneme: http://groups.csail.mit.edu/uid/deneme/?p=244

Little, G., Chilton, L. B., Goldman, M., & Miller, R. C. (2009). TurKit: tools for iterative tasks on mechanical Turk. Proceedings of the ACM SIGKDD Workshop on Human Computation (pp. 29-30). ACM.

Niu, Z.-Y., Dong-Hong, J., & Chew-Lim, T. (2007). I2r: Three systems for word sense discrimination, chinese word sense disambiguation, and english word sense disambiguation. Proceedings of the Fourth International Workshop on Semantic Evaluations (SemEval-2007) (pp. 177-182). Prague, Czech Republic: Association for Computational Linguistics.

Rosenberg, A., & Hirschberg, J. (2007). V-measure: A conditional entropy-based external cluster evaluation measure. Proceedings of the 2007 Joint Conference EMNLP-CoNLL, (pp. 410-420).

SigLex, A. (2008). Retrieved 01 2010, from SemEval-2, Evaluation Exercises on Semantic Evaluation: http://semeval2.fbk.eu/semeval2.php

Snow, R., O'Connor, B., Jurafsky, D., & Ng, A. Y. (2008). Cheap and fast---but is it good? Evaluating non-expert annotations for natural language tasks. Proceedings of the Conference on Empirical Methods in Natural Language Processing (pp. 254-263). Association for Computational Linguistics.

Snow, R., Prakash, S., Jurafsky, D., & Ng, A. Y. (2007). Learning to Merge Word Senses. Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL) (pp. 1005-1014). Prague, Czech Republic: Association for Computational Linguistics.

Strehl, A., & Ghosh, J. (2003). Cluster ensembles---a knowledge reuse framework for combining multiple partitions. The Journal of Machine Learning Research , 3, 583-617.

Topchy, A., Jain, A. K., & Punch, W. (2005). Clustering ensembles: Models of consensus and weak partitions. IEEE Transactions on Pattern Analysis and Machine Intelligence , 27 (12), 1866-1881.

Zhao, Y., & Karypis, G. (2001). Criterion functions for document clustering: Experiments and analysis. Report TR 01–40, Department of Computer Science, University of Minnesota.

Appendix

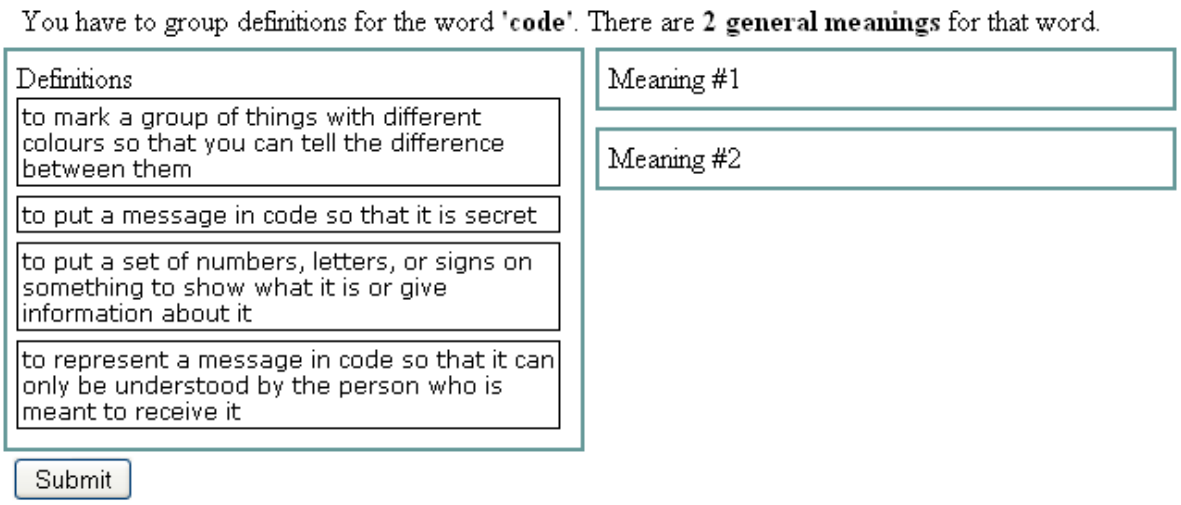

Figure 3: Example of a *global-view* HIT for the word "code" (not all of the instructions are shown)

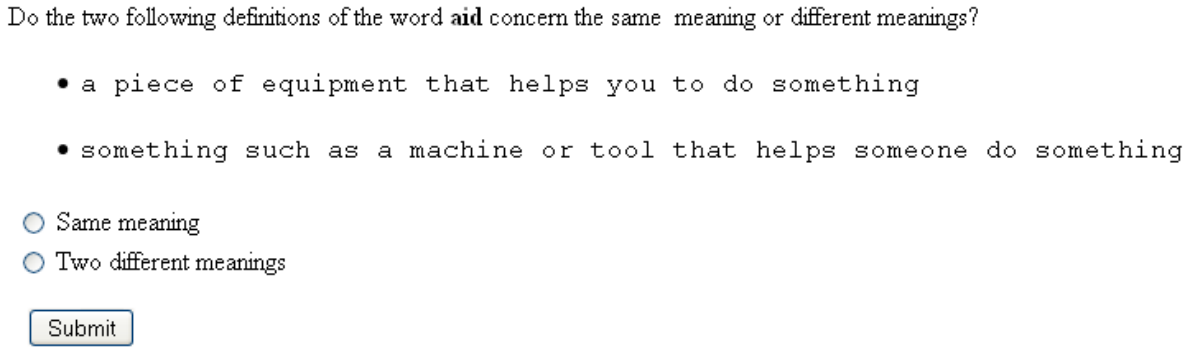

Figure 4: Example of a *local-view* HIT for the word "aid" (not all of the instructions are shown)