

# Cost Optimization in Crowdsourcing Translation:

## Low cost translations made even cheaper

### First Author

Affiliation / Address line 1  
Affiliation / Address line 2  
Affiliation / Address line 3  
email@domain

### Second Author

Affiliation / Address line 1  
Affiliation / Address line 2  
Affiliation / Address line 3  
email@domain

### Abstract

Crowdsourcing makes it possible to create translations at low cost. We proposed two mechanisms to make this process even cheaper while maintaining high translation quality. First, we introduce a translator reducing method that allows us to reduce cost by quickly identifying bad translators after they have translated only a few sentences. This allows us to rank translators, so that we only re-hire good translators and so that we can select the best translations from among good candidates. Second, we develop a translation reducing method. We train a linear model to evaluate the translation quality on a sentence-by-sentence basis, and fit a threshold between acceptable and unacceptable translations. Unlike past work, which always paid for a fixed number of translations of each source sentence and then chose the best from among them, we can decide after seeing a single translation whether it is good enough or not. Our model based selection allows us to reduce cost by reducing the number of redundant translations that we solicit. Additionally, we show that costs associated with gold standard calibration data created by professional translators can be reduced by using single reference instead of multiple references.

## 1 Introduction

Crowdsourcing is a promising new mechanism for collecting large volumes of annotated data at low cost. Many NLP researchers have focused on creating speech and language data through crowdsourcing (for example, ?, ?) and others). One NLP appli-

cation that has been the focus of crowdsourced data collection is statistical machine translation (SMT) which requires large bilingual sentence-aligned parallel corpora to train translation models. Crowdsourcing's low costs has made it possible to hire people to create sufficient volumes of translation in order to train SMT systems (for example, ?, ?, ?), ?)).

However, crowdsourcing is not perfect, and one of its most pressing challenges is how to ensure the quality of the data that is created by it. Unlike in more traditional employment scenarios, where annotators are pre-vetted and their skills are clear, in crowdsourcing very little is known about the annotators. They are not professional translators, and there are no built-in mechanisms for testing their language skills. They complete tasks without any oversight. Thus, translations produced via crowdsourcing may be low quality. Previous work has addressed this problem, showing that non-professional translators hired on Amazon Mechanical Turk (MTurk) can achieve professional-level quality, by soliciting multiple translations of each source sentence and then choosing the best translation (?).

In this paper we focus on a different aspect of crowdsourcing from ?). We attempt to achieve the same high quality while **minimizing the associated costs**. We reduce the costs associated with both professional and non-professional translations. Professional translations are used as calibration data for crowdsourcing. We show that using a single reference is as effective as using multiple references. To reduce costs for non-professional translations, we propose two complementary methods: (1) We re-

duce the number of workers we hire, and retain only high quality translators by quickly identifying and filtering out workers who produce low quality translations. (2) We reduce the number of translations that we solicit for each source sentence. Instead of soliciting a fixed number of translations for each foreign sentence, we stop soliciting translations after we get an acceptable one. We do so by building models to distinguish between acceptable translations and unacceptable ones.

Our work stands in contrast with (?) who had no model of annotator quality, and who always solicited and paid for a fixed number of translations of each source segment.

In this paper we demonstrate that

- Workers can be ranked by their quality with high correlation against a gold standard ranking, using linear regression and a variety of features, or initially testing them using a small amount of calibration data with known professional translations.
- This ranking can be established after observing very small amounts of data (reaching  $\rho$  of 0.88 after seeing the translations of only 20 sentences from each worker). Therefore, bad workers can be filtered out quickly.
- Our models can predict whether a given translation is acceptable with high accuracy, substantially reducing the number of redundant translations needed for every source segment.
- We can achieve a similar BLEU score as (?) at half the cost using our translation reducing method.

## 2 Problem Setup

(?) trained linear models to score non-professional translations and chose the best translation from them for each source sentence. They promoted the crowdsourcing translation quality to the professional level. We propose mechanisms to reduce costs and keep high translation quality. We develop our methods on the data set (?) used. In addition, we regulate the metrics for cost and translation quality.

### 2.1 Data Set

We use the data collected by (?) through Amazon’s Mechanical Turk. They hired Turkers to translate 1792 Urdu sentences from the 2009 NIST Urdu-English Open Machine Translation Evaluation set<sup>1</sup>. Workers also filled out a survey about their language skills and their countries of origin. A total of 51 Turkers contributed translations. Each Urdu sentence was translated by 4 Turkers and 4 professional translators.

### 2.2 Measurement of Total Cost

The costs associated with crowdsourcing translation consists of expense of purchasing both professional and non-professional translations. Professional translations are used as calibration data to estimate the goodness of the non-professional translation while non-professional translations are redundantly collected as candidates for selection.

We define the total cost  $C$  as:

$$C = \alpha \cdot N_p \cdot C_p + \beta \cdot N_n \cdot C_n$$

$C_p$  and  $C_n$  are the costs of one translation provided by professionals and non-professionals respectively.  $N_p$  and  $N_n$  refer to the number of source sentences used in collecting professional and non-professional translations separately.  $\alpha$  and  $\beta$  represent the level of redundancy of professional and non-professional translations respectively and higher redundancy level means more translations are solicited for each source sentence. Thus, (?)’s total cost is:

$$C = 4 \times N_p \times 6.03 + 4 \times 1792 \times 0.10$$

where  $C_p$  is estimated to \$6.03(\$0.30 per word  $\times$  20.1 words per sentence) and  $C_n$  is \$0.10. We calculate the cost lowerbound and upperbound by varying the value of  $N_p$ . The lowerbound is \$179.20, for single translations from Turkers with no calibration data ( $N_p = 0$ ). (?) showed that to achieve professional quality, they used professional translations of 10% of all source sentences as calibration data. Thus, the upperbound is \$5,039.10 for  $N_p = 10\% \times 1792$  (\$716.80 + \$4,322.30 at 10% calibration).

<sup>1</sup>LDC Catalog number LDC2010T23

### 2.3 Measurement of Translation Quality

For this data set we have access to 4 sets of professional translations, which were created by different translation agencies hired by the LDC. While minimizing costs, we want to ensure that the quality of the translations does not suffer. To evaluate translation quality, we compute the quality of selected non-professional translations using the BLEU score (?) against all professional translations.

### 3 Estimating Translation Quality

?) showed that based on feature extraction and model tuning, they can select the best translations among non-professional candidates and achieve professional quality. They extracted a number of features from the translations and workers' self-reported language skills in order to predict the best translations. These features included:

- Sentence-level features: 9 features including language model scores, sentence length ratios between source sentences and translations, Web n-gram matching scores and edit distance to other translations.
- Worker-level features: 15 features including aggregated sentence-level scores for each worker, language abilities and worker locations.
- Ranking features: 3 features including translation's average ranking and the percentages of being evaluated better and best.
- Calibration feature: 1 feature reflecting the average BLEU score of translations provided by the same worker which is computed against references.

They trained a linear model using the features to select the best translation from 4 candidate translations, and compared the Turkers' translation quality to professional quality to show the validity of their method.

Following their work on translation evaluation and selection, we use a linear regression model to predict a score ( $score(t) \in R$ ) for an input translation  $t$ .

$$score(t) = \vec{w} \cdot \vec{f}(t)$$

where  $\vec{w}$  is the associated weight vector and  $\vec{f}(t)$  is the feature vector of the translation  $t$ . In addition to the features that (?) used, we introduce a new bilingual feature. We use the IBM Model 1 to construct the word alignment probabilities between Urdu and English. We align words between each translation and corresponding source sentence. The bilingual feature of a translation is the averaging alignment probabilities. In Figure ??, we present the bilingual feature for a translation.

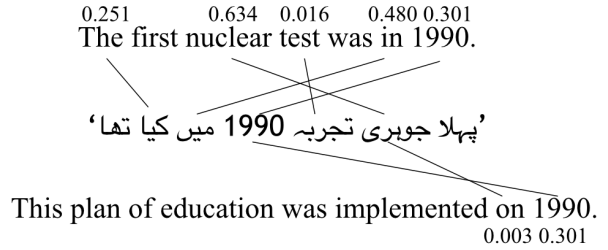


Figure 1: Bilingual feature example for the good translation: "The first nuclear test was in 1990" and the bad translation: "This plan of education was implemented on 1990" for the same Urdu Sentence. We align words with its source sentence. The top and bottom rows show the alignment probability for each Urdu word. The Urdu sentence has 7 words and the bilingual feature value is the average probability of these aligned probabilities. and the bilingual feature value for the good translation and bad translation are 0.240 and 0.043 respectively which is the average of these probabilities.

To reduce cost of computing the calibration feature and labeling each training sample by its BLEU score, we randomly select **only one** professional translation as reference to calculate the BLEU score.

## 4 Reducing the Number of Translators

The first mechanism to optimize cost is reducing the number of translators by quickly identifying bad workers and filtering them out. We achieve high translation quality and avoid hiring bad workers continually after having detected their bad performance.

### 4.1 Workers are consistently good (or bad) over time

Figure ?? illustrates the consistency of workers' performance by showing the gold-standard ranking of Turkers created by computing their BLEU scores compared to professionals for all of the translations

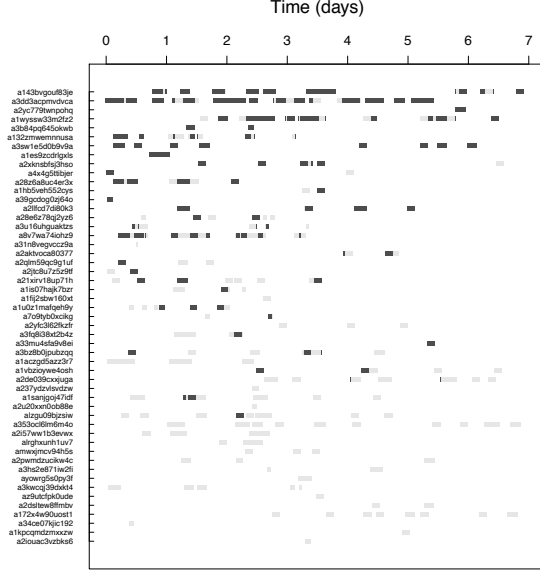


Figure 2: A time-series plot of all of the translations produced by Turkers (identified by their WorkerID serial number). Turkers are sorted based on the gold-standard ranking against professionals, with the best translators on top on y-axis. Each tick represent a single translation HIT, and depicts the HIT’s BLEU score (color) and its size/number of sentences (thickness). We use the average BLEU score for each HIT and show its tick in black if its BLEU score is higher than the median and in light grey otherwise.

that they submitted, along with the number of HITs produced by each worker and their timing information. From this graph we see that that most workers’ performance stays consistent as time passes. Good translators tend to produce consistently good translations and bad workers rarely produce good translations.

## 4.2 Automatically Ranking Translators

Since workers’ performance is consistent, workers’ rankings are sufficiently accurate to reflect the quality of translations provided by them and we can select the translation which is provided by the worker with the best rank. We propose two approaches to rank workers from a small portion of work they submitted, filter out bad workers and select the best translation from translations provided by surviving workers.

**Ranking workers using a model** We use a linear regression model to score each translation and rank workers by their model predicted performance. The model predicted score for translation  $t$  is defined as  $M(t)$ . The model predicted performance of the worker  $w$  is:

$$Performance(w) = \frac{\sum_{t \in T_w} M(t)}{|T_w|}$$

where  $T_w$  is the set of translations completed by  $w$ . After we rank workers, we keep top workers in the list and select translation provided by the worker with best rank among top workers.

## Ranking workers using their first k translations

Rather than using a model to rank workers, we take the first few translations provided by each Turker and compare them to the professional translations of those sentences. We rank workers based on this gold standard data and discard bad workers.

## 4.3 Experiments

We report ranking’s correlation to gold standard ranking and translation quality for both two methods.

### 4.3.1 Baselines

We evaluate ranking quality in Spearman’s correlation( $\rho$ ) compared with the gold standard ranking of workers. We score each Turker based on the average BLEU score of all his/her translations against professional references and we rank Turkers by their scores. We use the gold standard ranking as the ranking oracle and the upper-bound correlation is 1. In addition, we set two baselines for ranking correlation( $\rho$ ) for our proposed approaches. For the first baseline, we choose the MERT(?) baseline, which achieves a correlation of 0.67 when trained on ranking features. This is the highest correlation that MERT achieves across all feature sets. The second baseline is a simpler baseline that reserves 10% of the data for calibration, and computes a ranking of translators based on their BLEU scores against professionals over this calibration set, the correlation reaches 0.68.

For translation quality evaluation, we set the gold standard ranking selection method as the oracle method, in which we select translation provided by

the worker with best rank in gold standard ranking, and the BLEU score achieved is denoted as  $B_{gold}$ . Besides, we set the random selection method as the baseline method which randomly select a translation from all candidates for each source sentence.

#### 4.3.2 Ranking workers using a model

We use 10% of data to train a linear regression model to rank workers and select best translation by workers' ranking. Table ?? shows that our model achieves the highest BLEU score of 38.37 when trained on all features. If we calculate calibration feature and training sample label against only one reference rather than 4 references, we achieve a BLEU score of 37.52 with a correlation of 0.71 which is higher than two baselines. To reduce cost, we retain only top 25% workers and select the translation with the best rank provided by top workers. We achieve a BLEU score of 37.09 while  $B_{gold}$  is 38.51 and the baseline is 29.95. The cost is:

$$C = \frac{\alpha}{4} \cdot \frac{1}{10} N_p \cdot C_p + \frac{\beta}{4} \cdot \frac{9}{10} N_{np} \cdot C_{np} \\ = 1,241.86(\$)$$

Thus, we can achieve a comparable translation quality by spending almost only 25% of money with a quality loss of 1.53 in BLEU.

Feature Set	$\rho$	BLEU
(S)entence features	0.69	36.66
(W)orker features	0.65	36.92
(R)anking features	0.79	36.94
Calibration features	0.79	38.27
Calibration features*	<b>0.68</b>	<b>37.22</b>
S+W+R features	0.78	37.39
S+W+R+Bilingual features	0.80	37.59
All features	<b>0.84</b>	<b>38.37</b>
All features*	<b>0.71</b>	<b>37.52</b>

Table 1: Spearman's correlation( $\rho$ ) and translation quality of selecting best translations based on model-predicted workers' ranking for different feature sets. We don't filter out bad workers when selecting the best translation. \* indicates that we choose **only one** professional reference to calculate the BLEU score as calibration feature and the true label of a training sample while in other cases, we use 4 references to calculate BLEU score.

#### 4.3.3 Ranking workers using their first k translations

Without using any model, we rank workers using their first k translations and select best translations based on rankings of the top 25% workers. To evaluate this method, we created several test sets. Each test set excluded any item that was used to rank the workers, or which did not have any translations from the top 25% of workers according to our predicted rankings. We therefore have *different test sets* for each value of k. This makes the results slightly more difficult to analyze than in normal experiments, although the trends are still clear. Formally, we define the test set for first k sentences as  $T_k$  and for each source sentence  $s \in T_k$ :

$$\{s \mid (C(s) \cap S_k = \emptyset) \wedge (C(s) \cap S_w \neq \emptyset)\}$$

where  $C(s)$  is the translating candidates set of the source sentence  $s$ ,  $S_k$  is the translation set consists of each worker's first k translations and  $S_w$  is the translation set consists of translations provided by selected workers (some top ranking workers). Table

Proportion of Calibration Data		$\rho^+$	$\rho^*$
First k sentences	Percentage		
1	0.7%	0.57	0.41
2	1.3%	0.62	0.48
3	2.0%	0.69	0.59
4	2.7%	0.72	0.59
5	3.3%	0.78	0.69
6	4.0%	0.80	0.70
7	4.7%	0.79	0.71
8	5.3%	0.81	0.69
9	6.0%	0.84	0.77
10	6.6%	0.84	0.76
20	13.3%	0.93	0.88
30	19.9%	0.96	0.93
40	26.6%	0.97	0.95
50	33.2%	0.98	0.95
60	39.8%	0.99	0.94

Table 2: Spearman's Correlations for calibration data in different proportion. \* indicates that the calibration is computed against **only one** reference while + indicates that the calibration is computed against 4 references.

?? shows the results of Spearman's correlations for

different value of  $K$ . Compared with 4-reference calibration, we achieve very strong correlation when calibrating the workers using one reference based on the translations of their first 40 sentences. Even we only use the first 20 sentences to evaluate and rank workers, the correlation ( $\rho^*$ ) is close to 0.90. Consequently, we can decide whether to continue to hire a worker in a very short time after analyzing the first  $k$  sentences ( $k \leq 20$ ) provided by each worker.

Figure ?? shows the BLEU score when we select the top 25% workers from the ranking list based on the performance of first  $k$  sentences. As a comparison, we also plotted the BLEU scores for random selection and the BLEU score for selection based on the gold standard ranking ( $B_{gold}$ ). As we increase the number of sentences we use to rank Turkers, the BLEU score we get from the ranking approaches  $B_{gold}$ . Surprisingly, we see that when only a small part of sentences (say 20 sentences) for each worker are used in ranking, the ranking list is quite similar to the gold standard ranking list and the BLEU score is very close to the BLEU score get by gold standard ranking.

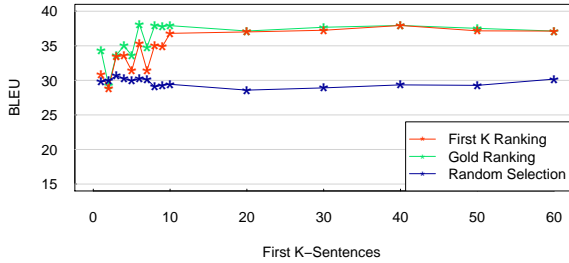


Figure 3: The BLEU score for selecting the best translation by the top 25% Turkers’ ranking based on the first  $k$  sentences (red line). The green line shows the BLEU score for selecting the best translation by the gold standard ranking. The dark blue line shows the BLEU score for selecting translation randomly.

If we use the first 20 sentences to rank workers, the correlation is 0.88 and the BLEU score achieved is 37.01 while and  $B_{gold}$  is 37.14. The difference between these two scores is 0.13 and the cost is:

$$C = \frac{\alpha}{4} \cdot (0.133 \cdot N_p) \cdot C_p + \frac{\beta}{4} \cdot (0.867 \cdot N_{np}) \cdot C_{np} = 1,592.53(\$)$$

We can achieve almost the same translation as the oracle method with only spending 32% of the total cost.

## 5 Reducing the Number of Translations

The second mechanism to optimize cost is reducing the number of translations by recognizing unacceptable translations, and stopping purchasing redundant translations after we have get acceptable ones.

We present the translation reducing method to decide whether a translation is ‘good enough,’ in which case we don’t need to pay for another redundant translation of the source sentence. Additionally, we make it possible to control the quality of translations selected from a partial translation set to a proportion ( $\delta$ ) of the upperbound quality in BLEU score which is computed on best translations selected from the full translation set of training data. To reduce cost and control quality, we train a model to score non-professional translations and search the value of  $\theta$  (threshold between acceptable and unacceptable translations) for a specific value of  $\delta$  (say 95%). For given a a new translation, we score it by our model and if the its score is higher than  $\theta$ , we stop soliciting another translation. Otherwise, we continually solicit translations. Algorithm ?? details the process of model training and searching for  $\theta$ .

### 5.1 Experiment

We divide data into training set(10%), validation set(10%) and testing set(80%). Each sample in training and validating set is labeled and calibrated by BLEU score calculated against **only one** reference. The step we set to sweep  $\theta$  in validating process is 0.01 and the *upperbound* is set to be 0.41 empirically. We vary the value of  $\delta$  from 90% to 100% and the results we reported are based on five-fold cross validation.

#### 5.1.1 Baselines

We set a competitive method to compete with, which is revised from the framework of translation reducing mechanism with two different points:

---

**Algorithm 1**

---

**Input:**  $\delta$ , the proportion of the upperbound of BLEU score; a training set  $S = \{(\vec{f}(t_{i,j}^s), y_{i,j}^s)_{j=1}^4\}_{i=1}^n$  and a validation set  $V = \{(\vec{f}(t_{i,j}^v), y_{i,j}^v)_{j=1}^4\}_{i=1}^n$  where  $\vec{f}(t_{i,j})$  is the feature vector for the  $j$ th translation of the source sentence  $s_i$  and  $y_{i,j}$  is the label for  $t_{i,j}$ .  
**Output:**  $\theta$ , the threshold between acceptable and unacceptable translations;  $\vec{w}$ , a linear regression model parameter.

```

1: initialize  $\theta \leftarrow 0, \vec{w} \leftarrow \emptyset$ 
2:  $\vec{w}' \leftarrow$  train a linear regression model on  $T$ 
3:  $maxbleu \leftarrow$  select best translations for each  $s_i \in T$  based on the model parameter  $\vec{w}'$  and record the highest model predicted BLEU score
4:  $\alpha \leftarrow$  set an upper-bound BLEU score empirically
5: while  $\theta \neq maxbleu$  do
6:   for  $i \leftarrow 1$  to  $n$  do
7:     for  $j \leftarrow 1$  to 4 do
8:       if  $\vec{w}' \cdot \vec{f}(t_{i,j}^v) > \theta \wedge j < 4$  then select  $t_{i,j}^v$  for  $s_i$  and break
9:       if  $j == 4$  then select  $t_{i,j}^v$  for  $s_i$ 
10:     $q \leftarrow$  calculate translation quality for  $V$ 
11:    if  $q > \delta \cdot \alpha$  then break
12:    else  $\theta = \theta + stepsize$ 
13:  $\vec{w} \leftarrow$  train a linear regression model on  $S \cup V$ 
14: Return:  $\theta$  and model parameter  $\vec{w}$ 

```

---

(1) we label and calibrate each sample in training and validating set using BLEU computed against four references; (2) we select translation from all candidates for each source sentence. We get a surprisingly high BLEU score of 40.13 using this method with a high cost over \$9,000 (\$716.80 + \$8,644.6 for 20% calibration). In addition, we set the random selection baseline and the corresponding BLEU score is 29.56.

### 5.1.2 Translation reducing method

Table ?? shows the results for translation reducing method. We get comparable translation quality against our competing method with a much lower cost. If we set  $\delta$  as 0.95, comparing two method, the difference in translation quality is 1.7 and for each source sentence, we almost avoid paying one redun-

dant translation. The cost is:

$$C = \frac{\alpha}{4} \cdot \frac{2}{10} N_p \cdot C_p + \frac{3.12}{4} \beta \cdot \frac{8}{10} N_{np} \cdot C_{np} = 2,608.43(\$)$$

which is 52% of the original total cost and 28% of the cost of our competing method.

$\delta(\%)$	BLEU Score	# of Trans.
90	37.04	2.11
91	37.20	2.19
92	37.77	2.54
93	37.79	2.61
94	38.31	3.07
95	38.43	3.12
96	38.65	3.44
97	38.71	3.39
98	39.17	3.88
99	39.32	3.99
100	39.36	3.99

Table 3: The relation among the  $\delta$  (the proportion of the BLEU score’s upper bound), the BLEU score for translations selected by models from partial sets and the averaged size of translation candidates set for each source sentence (# of Trans).

## 6 Cost Analysis

We have introduced several ways of lowering the costs associated with crowdsourcing translations:

- We show that we can quickly identify bad translators, either with a model designed to rank them, or by ranking them by having them first translate a small number of sentences with gold standard translations. The cost savings for non-professionals here comes from not hiring bad workers.
- After we have collected one translation of a source sentence, we consult a model that predicts whether its quality is sufficiently high or whether we should pay to have the sentence re-translated. The cost savings for non-professionals here comes from reducing the number of redundant translations.
- In both cases we need a some amount of professionally translated materials to use as a gold

standard for calibration. The cost savings for professionals come from reducing the referencing translations to calibrate each data sample.

In all cases, there is a trade-off between lowering our costs and producing high quality translations. Figure ?? plots the cost versus the BLEU scores for the different configurations that we experimented with.

In Figure ??-(a) the increasing costs are a function of how many sentences we use to rank the translators. Here we use no model, and simply rank the translators by their BLEU score against a small amount of gold standard data. The quality peaks at 37.9 BLEU after \$3,000. We are able to rank the translators with high accuracy and achieve a relative high BLEU score by paying for a comparatively small number of professional translations to use as calibration. From our experiments, 10-20 professionally translated sentences seems like a reasonable number.

Figure??-(b) uses a model to determine whether to purchase another translation. This model allows us to significantly improve the overall translation quality to a BLEU score of nearly 40, for a final cost of \$2,700.

## 7 Related Work

?)’s work on repeated labeling presents a way of solving the problems of uncertainty in labeling. Since we cannot always get high-quality labeled data samples with relatively low costs in reality, to keep the model trained on noisy labeled data having a high accuracy in predicting, ?) proposed a framework for repeated-labeling that resolves the uncertainty in labeling via majority voting. The experimental results show that a model’s predicting accuracy is improved even if labels in its training data are nosity and of imperfect quality. As long as the integrated quality (the probability of the integrated labeling being correct) is higher than 0.5, repeated labeling benefits model training.

?) created a Bayesian model of annotation and they applied to the problem of word sense annotation. ?) also proposed an approach to detect and avoid spam workers. They measured the performance of worker by comparing worker’s labels to the current majority labels and worker with bad performance would be blocked. However, this approach

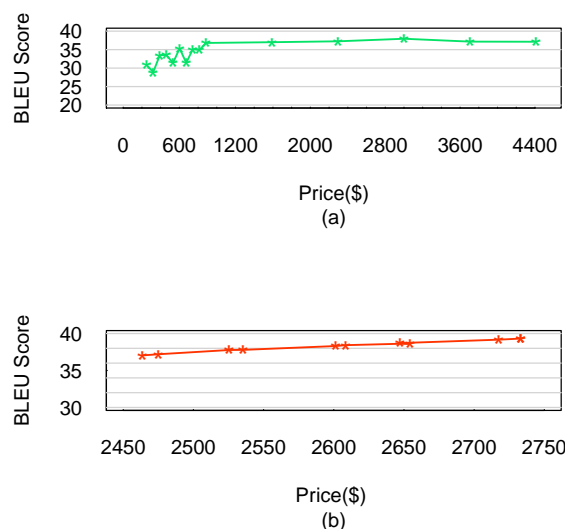


Figure 4: The Relationship between BLEU score and costs. In Figure (a), the red line shows the relationship between BLEU score and the total costs (professional and non-professional) for the ranking based approach. Figure (b) shows the relationship between BLEU score and the total costs for model-based approach.

suffered from 2 shortcomings: (1) Sometimes majority labels may not reflect the ground truth label. (2) They didn’t figure out how much data(HITs) is needed to evaluate a worker’s performance. Although they could find the spam after the fact, it was a post-hoc analysis, so they had already paid for that worker and wasted the money.

?) examined the relationship between worker accuracy and budget in the context of using crowdsourcing to train a machine learning classifier. They show that if the goal is to train a classifier on the labels, that the properties of the classifier will determine whether it is better to re-label data (resulting in higher quality labels) or get more single labeled items (of lower quality). They showed that classifiers with weak inductive bias benefit more from re-labeling, and that relabeling is more important when worker accuracy is low (barely higher than 0.5).

?) showed a similar result for training an automatic speech recognition (ASR) system. When creating training data for an ASR system, given a fixed budget. Their system’s accuracy was higher when it is trained on more low quality transcription data



compared to when it was trained on fewer high quality transcriptions.

## 8 Conclusion

In this paper, we propose two mechanisms to optimize cost: the translator reducing method and the translation reducing method. They have different applicable scenarios. The translator reducing method is a very simple method without any model training. This approach is inspired by the intuition that workers' performance is consistent. The translator reducing method is suitable for crowdsourcing tasks which do not have specific requirements about the quality of the translations, or when the data collection is being performed by a requester who does not have sufficient background in machine learning in order to train a model, or when only very limited amounts of gold standard data are available. The translation reducing method works if there exists a specific requirement that the quality control must reach a certain threshold, or when more data needs to be collected. This model is most effective when reasonable amounts of pre-existing professional translations are available for setting the models threshold. Its major cost reduction comes from dramatically reducing the amount of non-professional data to maintain the same quality.

## 9 Bubble Graph

**Ranking Turkers: Gold Ranking vs. Model Rankin**

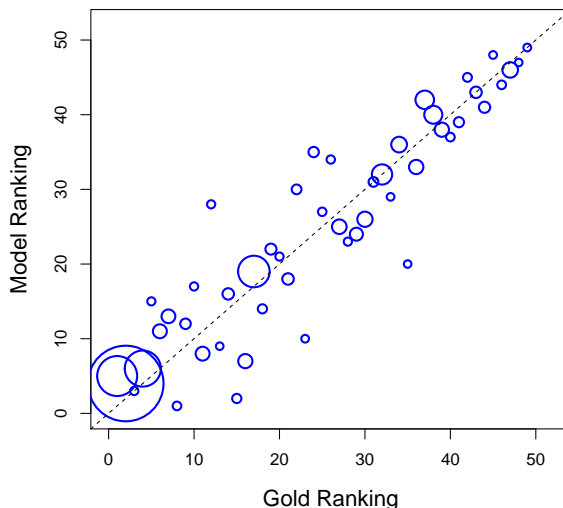


Figure 5: Correlation between gold standard ranking and model ranking. We train a linear regression model to score translations and to rank workers. The corresponding Spearmans Correlation is 0.84. Each bubble represents a worker with his/her rank in gold standard ranking on x-axis and model ranking on y-axis. The radius of each bubble shows the relative volume of translations completed by the worker.