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 rehire 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, Snow et al. (2008), Callison-Burch and Dredze (2010) and others). One NLP application 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, Zbib et al. (2013), Zbib et al. (2012), Post et al. (2012), Ambati and Vogel (2010)).

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 crowdousrcing 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 (Zaidan and Callison-Burch, 2011).

In this paper we focus on a different aspect of crowdsourcing from Zaidan and Callison-Burch (2011). 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 reduce 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 Zaidan and Callison-Burch (2011) 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 ρ 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 Zaidan and Callison-Burch (2011) at half the cost using our translation reducing method.

2 Problem Setup

We start with a corpus of source sentences to be translated, and we may solicit one or more translations for every sentence in the corpus. Our goal is to assemble a single high quality translation for each source sentence while minimizing the associated cost.

We study the data collected by Zaidan and Callison-Burch (2011) through Amazon's Mechanical Turk. They hired Turkers to translate 1792 Urdu sentences from the 2009 NIST Urdu-English Open Machine Translation Evaluation set[1]. A total of 52 Turkers contributed translations. Turkers also filled out a survey about their language skills and their countries of origin. Each Urdu sentence was translated by 4 non-professional translators (the Turkers) and 4 professional translators hired by the LDC. The cost of for non-professional translation is \$0.10 per sentence and the cost of professional translation is \$0.30 per word (or just over \$6 on average for the sentences in our corpus which have an average of 20.1 words).

Following Zaidan and Callison-Burch (2011), we use BLEU (Papineni et al., 2002) to gauge the quality of human translations. We can compute the expected quality of professional translation by comparing each of the professional translators against the other 3. This results in an average BLEU score of 42.38. By comparison, the Turker translations score only 28.13 on average. Zaidan and Callison-Burch trained a MERT model to select one non-professional translation out of the four and pushed the quality of crowdsourcing translation to 39.06, closer to the expected quality of professional translation. They used a small amount of professional translations (10%) as calibration data to estimate the goodness of the non-professional translation. The component costs of their approach are the 4 non-professional translations for each source sentence, and the professional translations for the calibration data.

Although Zaidan and Callison-Burch demonstrated that non-professional translation was significantly cheaper than professionals, we are interested in further reducing the costs. This plays a role if we would like to assemble a large enough parallel corpus (on the order of millions of pairs of sentences) to train a statistical machine translation system. Here, we introduce several methods for reduce the number of non-professional translations while still maintaining high quality.

[1] LDC Catalog number LDC2010T23

3 Estimating Translation Quality

We use a linear regression model[2] to predict a quality 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.

We replicate the feature set used by Zaidan and Callison-Burch (2011) in their MERT model:

- Sentence-level features: 9 features based on language model, sentence length, edit distance to other translations.
- Worker-level features: 15 features based on worker's language ability, location and average sentence-level scores.
- Ranking features: 3 features based on the judgments of monolingual English speakers ranking the translations form best to worst.
- Calibration features: 1 feature based on the average BLEU score of translations provided by the same worker, which is computed against professional references.

We additionally introduce a new bilingual feature based on IBM Model 1. We align words between each candidate translation and its corresponding source sentence. The bilingual feature for a translation is the average of its alignment probabilities. In Figure 1, we show how the bilingual feature allows us to distinguish between a valid translation (top) and an invalid/spammy translation (bottom).

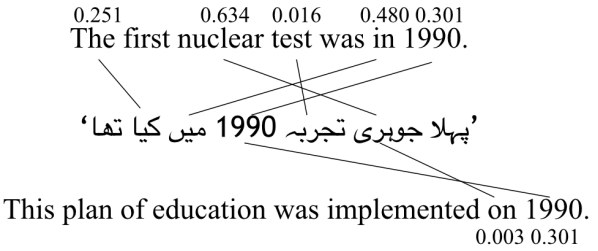

Figure 1: Bilingual feature example of two crowdsourcing translations for a sentence in Urdu. The numbers are alignment probabilities for each aligned word.The bilingual feature is the average of these probabilities, thus 0.240 for the good translation and 0.043 for the bad translation.

4 Reducing the Number of Translations

The first mechanism that we introduce to optimize cost is one that reduces the number of translations. Our goal is to recognize when we have got a good translation of a source sentence and to immediately stop purchasing additional translations of that sentence. The crux of this method is to decide whether a translation is 'good enough,' in which case we do not gain any benefit from paying for another redundant translation.

[2]We used WEKA package: `http://www.cs.waikato.ac.nz/ml/weka/`

Our translation reduction method allows us to set an empirical definition of 'good enough'. We introduce a parameter of the model (δ) that allows us to set how much degradation in translation quality is allowable, when we compare against selecting the best translation from the full set of redundant translations. For instance, we may fix δ at 95%, meaning that the BLEU score should not drop below 0.95 of the estimated BLEU score using the full set of non-professional translations. We train a model to search for a threshold between acceptable and unacceptable translations (θ) for a specific value of δ.

For a new translation, our model scores it, and if the its score is higher than θ, then we do not solicit another translation. Otherwise, we continue to solicit translations. Algorithm 1 details the process of model training and searching for θ.

4.1 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 θ in validating process is 0.01 and the $upperbound$ is set to be 0.41 empirically. We vary the value of δ from 90% to 100% and the results we reported are based on five-fold cross validation.

4.1.1 Baselines

We set a competitive method to compete with, which is revised from the framework of translation reducing mechanism with two different points: (1)

Algorithm 1

Input: δ, the allowable deviation from the expected upper bound on BLEU score (using all redundant translations); a training set $S = \{(x^s_{i,j}, y^s_{i,j})^4_{j=1}\}^n_{i=1}$ and a validation set $V = \{(x^v_{i,j}, y^v_{i,j})^4_{j=1}\}^n_{i=1}$ where $x_{i,j}$ is the feature vector for the jth translation of the source sentence s_i and $y_{i,j}$ is the label for $x_{i,j}$.

Output: θ, the threshold between acceptable and unacceptable translations; m(x), a linear regression model.

initialize $\theta \leftarrow 0$,$m(x) \leftarrow \emptyset$
$m'(x) \leftarrow$ train a linear regression model on T
$maxbleu \leftarrow$ use $m'(x)$ select best translations for each $s_i \in T$ and record the highest model predicted BLEU score
$upperbound \leftarrow$ set an upper-bound BLEU score empirically
while $\theta \neq maxbleu$ **do**
    for $i \leftarrow 1$ to n **do**
        for $j \leftarrow 1$ to 4 **do**
            if $m'(x^v_{i,j}) > \theta \wedge j < 4$ **then** select $x^v_{i,j}$ for s_i and **break**
            if $j == 4$ **then** select $x^v_{i,j}$ for s_i
    $q \leftarrow$ calculate translation quality for V
    if $q > \delta \cdot upperbound$ **then break**
    else $\theta = \theta + stepsize$
$m \leftarrow$ train a linear regression model on $S \cup V$
Return: θ and model $m(x)$

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.

4.1.2 Translation reducing method

Table 1 shows the results for translation reducing method. We get comparable translation quality against our competing method with a much lower cost. If we set δ as 0.95, comparing two method, the difference in translation quality is 1.7 and for each source sentence, we almost avoid paying one redundant 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.

δ(%)	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 1: The relation among the δ (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*).

5 Choosing Better Translators

The second mechanism that we use to optimize cost is to reduce the number of non-professional translators that we hire. Our goal is to quickly identify whether Turkers are good or bad translators, so that we can continue to hire only the good translators and stop hiring the bad translators after they are identified as such. Before presenting our method, we first demonstrate that Turkers produce consistent quality translations over time.

5.1 Turkers' behavior in translating sentences

Do Turkers produce good (or bad) translations consistently or not? Are some Turkers consistent and others not? We used the professional translations as a gold-standard to analyze the individual Turkers, and we found that most Turkers' performance stayed surprisingly consistent over time.

Figure 2 illustrates the consistency of workers' quality by plotting quality of their individual trans-

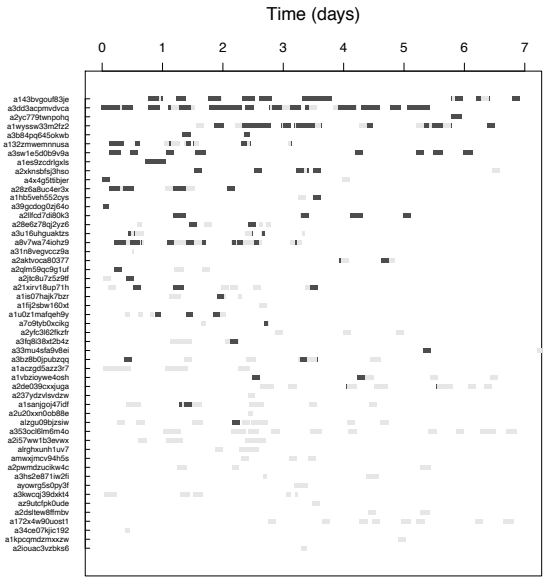

Figure 2: A time-series plot of all of the translations produced by Turkers (identified by their WorkerID serial number). Turkers are sorted with the best translator at the top of the y-axis. Each tick represent a single translation and dark color means better quality.

lations on a timeline. The translation quality is computed based on the BLEU against professional translations. Each tick represent a single translation and depicts the BLEU score using two colors. The tick is black if its BLEU score is higher than the median and it is light grey otherwise. Good translators tend to produce consistently good translations and bad workers rarely produce good translations.

Next, we introduce two approaches to rank workers using a small portion of work they submitted. Our goal is to filter out bad workers, and to select the best translation from translations provided by the remaining workers.

5.2 Automatically Ranking Translators

Ranking workers using a model We use the 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.

5.3 Experiments

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

5.3.1 Baselines

We evaluate ranking quality in Spearman's correlation(ρ) 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(ρ) for our proposed approaches. For the first baseline, we choose the MERT(Och, 2003) 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.

5.3.2 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 2 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 that 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 lost of 1.53 in BLEU.

Feature Set	ρ	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*	**0.68**	**37.22**
S+W+R features	0.78	37.39
S+W+R+Bilingual features	0.80	37.59
All features	**0.84**	**38.37**
All features*	**0.71**	**37.52**

Table 2: Spearman's correlation(ρ) 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.

5.3.3 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 3 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 (ρ^*) 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.

Proportion of Calibration Data		ρ^+	ρ^*
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 3: 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.

Figure 3 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.

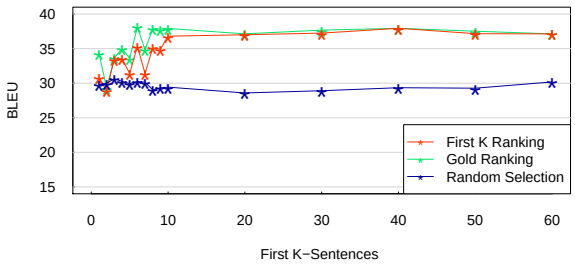

Figure 3: The BLEU score for selecting the best translation by the top 25% Turkers' ranking based on the first k sentences (red line). The green line shows the BLEU score for selecting the best translation by the gold standard ranking. The dark blue line shows the BLEU score for selecting translation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

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 4 plots the cost versus the BLEU scores for the different configurations that we experimented with.

In Figure 4-(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.

Figure4-(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

Sheng et al. (2008)'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

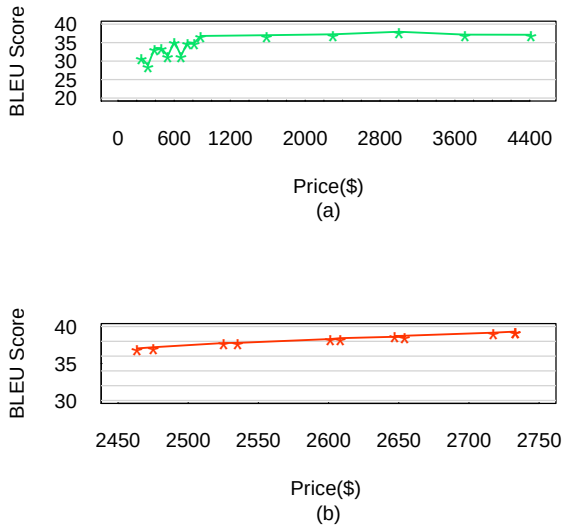

Figure 4: The Relationship between BLEU score and costs. In Figure (a), the red line shows the relationship between BLEU score and the total costs (professional and non-professional) for the ranking based approach. Figure (b) shows the relationship between BLEU score and the total costs for model-based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

noisy labeled data having a high accuracy in predicting, Sheng et al. (2008) 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.

Passonneau and Carpenter (2013) created a Bayesian model of annotation and they applied to the problem of word sense annotation. Passonneau and Carpenter (2013) 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 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.

Lin et al. (2014) 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 relabeling, and that relabeling is more important when worker accuracy is low (barely higher than 0.5).

Novotney and Callison-Burch (2010) 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.

References

Vamshi Ambati and Stephan Vogel. 2010. Can crowds build parallel corpora for machine translation systems? In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 62–65. Association for Computational Linguistics.

Chris Callison-Burch and Mark Dredze. 2010. Creating speech and language data with amazon's mechanical turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*, pages 1–12. Association for Computational Linguistics.

Christopher H Lin, Mausam, and Daniel S Weld. 2014. To re (label), or not to re (label). In *Proceedings of the 2014 AAAI Conference on Human Computation and Crowdsourcing*. Association for the Advancement of Artificial Intelligence (AAAI).

Scott Novotney and Chris Callison-Burch. 2010. Cheap, fast and good enough: Automatic speech recognition with non-expert transcription. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 207–215. Association for Computational Linguistics.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pages 160–167. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Rebecca J Passonneau and Bob Carpenter. 2013. The benefits of a model of annotation. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 187–195. Citeseer.

Matt Post, Chris Callison-Burch, and Miles Osborne. 2012. Constructing parallel corpora for six indian languages via crowdsourcing. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 401–409. Association for Computational Linguistics.

Victor S Sheng, Foster Provost, and Panagiotis G Ipeirotis. 2008. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 614–622. ACM.

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 *Proceedings of the conference on empirical methods in natural language processing*, pages 254–263. Association for Computational Linguistics.

Omar F. Zaidan and Chris Callison-Burch. 2011. Crowdsourcing translation: Professional quality from non-professionals. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 1220–1229.

Rabih Zbib, Erika Malchiodi, Jacob Devlin, David Stallard, Spyros Matsoukas, Richard Schwartz, John Makhoul, Omar F Zaidan, and Chris Callison-Burch. 2012. Machine translation of arabic dialects. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 49–59. Association for Computational Linguistics.

Rabih Zbib, Gretchen Markiewicz, Spyros Matsoukas, Richard M Schwartz, and John Makhoul. 2013. Systematic comparison of professional and crowdsourced reference translations for machine translation. In *HLT-NAACL*, pages 612–616.