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 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. Second, 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.

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).

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.

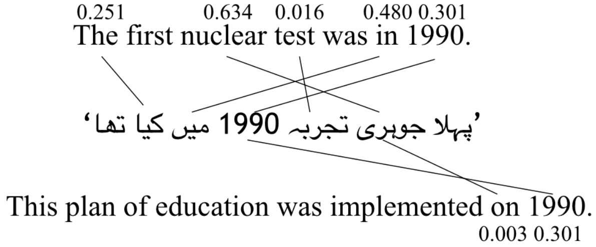
0.251
0.634
0.016
0.480 0.301
The first nuclear test was in 1990.
‘پہلا جوہری تجربہ 1990 میں کیا تھا’
This plan of education was implemented on 1990.
0.003 0.301

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. Some words are not aligned if potential word pairs don't exist in corpus.

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 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%). The step we set to sweep θ in validating process is 0.01 and the upper-bound BLEU (α) is set to be 0.4013 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 one different point: we select translation from all candidates for each source sentence. We get a surprisingly high BLEU score of

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

Algorithm 1

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

Output: θ, the threshold between acceptable and unacceptable translations; $\vec{w}$, a linear regression model parameter.

```
initialize $\theta \leftarrow 0$,$\vec{w} \leftarrow \emptyset$
$\vec{w'} \leftarrow$ train a linear regression model on T
$maxbleu \leftarrow$ select best translations for each $s_i \in T$ based on the model parameter $\vec{w'}$ and record the highest model predicted BLEU score
$\alpha \leftarrow$ 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 $\vec{w'} \cdot \vec{f}^v_{i,j} > \theta \wedge j < 4$ then select $t^v_{i,j}$ for $s_i$ and break
            if $j == 4$ then select $t^v_{i,j}$ for $s_i$
    $q \leftarrow$ calculate translation quality for V
    if $q > \delta \cdot \alpha$ then break
    else $\theta = \theta + stepsize$
$\vec{w} \leftarrow$ train a linear regression model on $S \cup V$
Return: $\theta$ and model parameter $\vec{w}$
```

40.13 using this method. 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 2.09 and for each source sentence, we almost avoid paying two redundant translations on average.

δ(%)	BLEU Score	# of Trans.
90	36.26	1.63
91	36.66	1.69
92	36.93	1.78
93	37.23	1.85
94	37.48	1.93
95	38.05	2.21
96	38.16	2.30
97	38.48	2.47
98	38.67	2.59
99	38.95	2.78
100	39.54	3.18

Table 1: The relation among the δ (the allowable deviation from the expected upper bound on BLEU score),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 translations 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.

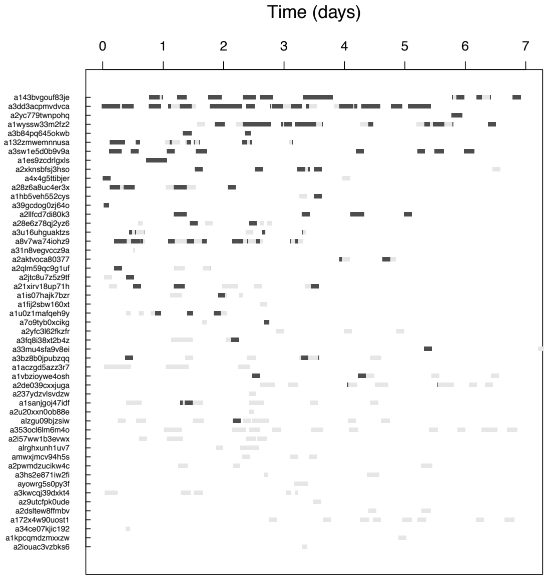

Figure 2: A time-series plot of all of the translations produced by Turkers (identified by their WorkerID serial number). Turkers are sorted with the best translator at the top of the y-axis. Each tick represent a single translation and dark color means better quality.

5.2 Ranking Evaluation

We compare our ranking of workers with gold standard ranking and calculate the correlation score. Since workers translated different number of sentences, we set a weight to each worker using the number of translations he/she submitted to represent his/her importance. Taking the importance of workers into consideration when calculating the correlation, we use the weighted Pearson correlation algorithm[3] in our case. Given two ranking list *x* and *y* showing workers' ranks and the weight vector *w*, the weighted correlation *corr* can be calculated as:

$$m(x;w) = \frac{\sum_i w_i x_i}{\sum_i w_i}$$

$$cov(x,y;w) = \frac{\sum_i w_i (x_i - m(x;w))(y_i - m(y;w))}{\sum_i w_i}$$

$$corr(x,y;w) = \frac{cov(x,y;w)}{\sqrt{cov(x,x;w)cov(y,y;w)}}$$

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.

5.3 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 $score(t)$. The model predicted performance of the worker w is:

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

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.

5.4 Experiments

In both approaches, we vary the threshold to split top workers from others, and select translations based on their ranking. We report ranking's correlation to gold standard ranking and translation quality. Since the top worker threshold is varied and we change the value of k in first k sentence ranking, we have a different test set in different settings. Each test set exclude any item that was used to rank the workers, or which did not have any translations from the top workers according to our rankings.

5.4.1 Baselines

We evaluate ranking quality in weighted 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 choose the MERT(Och, 2003) baseline for ranking correlation(ρ), which achieves

[3] http://en.wikipedia.org/wiki/Pearson_product-moment_correlation_coefficient# cite_note-22

a correlation of 0.74 when trained on ranking features. This is the highest correlation that MERT achieves across all feature sets.

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.

5.4.2 Ranking workers using a model

We train a linear regression model on 10% data to rank workers, select best translation by workers' ranking and evaluate the translation quality. In addition, we select the best translation among all candidates by their model predicted translation scores and evaluate the quality. Table 2 shows that 1) if we select translation by the model trained on all features, we can achieve a BLEU score of 39.80, 2) if we rank workers by the model, the highest correlation we achieve is 0.95, which is almost a perfect match with gold ranking and 3) if we select translation by workers' ranking, we can achieve a BLEU score of 38.37. Figure 3 shows the high correlation between gold ranking and model ranking if all features are used in model tuning. Workers with high volumes of submissions are ranked extremely well.

Feature Set	ρ	B_r	B_m
(S)entence features	0.84	36.66	37.84
(W)orker features	0.80	36.92	36.92
(R)anking features	0.81	36.94	35.69
Calibration features	0.93	38.27	38.27
S+W+R features	0.88	37.39	38.69
S+W+R+Bilingual features	0.89	37.59	39.23
All features	**0.95**	**38.37**	**39.80**

Table 2: Spearman's correlation(ρ) and translation quality of selecting best translations based on model-predicted workers' ranking (B_r) and model predicted translations' scores (B_m) for different feature sets. We don't filter out bad workers when selecting the best translation.

To reduce costs, we only keep hiring top workers and select the best translations based on their ranking. As comparison, we select the best translation using gold ranking. Table 3 shows the comparisons when we vary the top percentage of workers we keep hiring and the corresponding average number of non-professional translations needed for each source sentence. Comparing with selecting translations based on gold ranking, we achieve a similar BLEU score with lost of 0.55 when we filter out 75% worker in tail and we only need to solicit two translations for each source sentence on average.

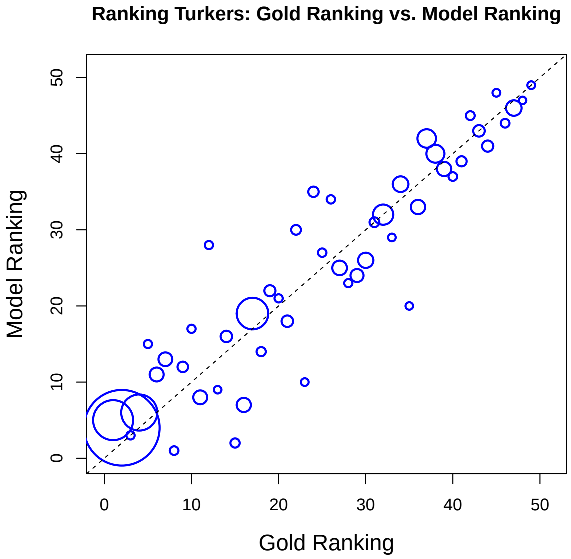

Figure 3: Correlation between gold standard ranking and model ranking. The corresponding weighted correlation is 0.95. Each bubble represents a worker with his/her rank in gold standard ranking on x-axis and model ranking on y-axis. The radius of each bubble shows the relative volume of translations completed by the worker.

Top(%)	B_r	B_t	B_{gold}	Δ	# of Trans
25	29.85	38.53	39.08	0.55	1.95
50	29.80	38.40	39.00	0.60	2.73
75	29.76	38.37	38.98	0.61	3.48
100	29.83	38.37	38.99	0.62	4.00

Table 3: The comparison between translation quality of selecting translations based on model ranking (B_t) and gold ranking (B_{gold}) when we keep hiring different percentages of top workers. Δ is the different between B_{gold} and B_t. # of Trans is the average number of non-professional translations needed for each source sentence. B_r is the BLEU score for random selection.

5.4.3 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 top workers. To evaluate this method, we calculate the weighted correlation for our rankings against gold ranking. Table 4 shows the results of Spearman's correlations for different value of k. As k increases, our rankings fit to the gold ranking better. 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 4 shows the correlation of gold ranking and first 20 sentences ranking. Since the weighted correlation score is 0.91, which means first 20 sentences ranking matches gold ranking well, we vary the top worker threshold on the first 20 sentences ranking and select translations by top workers ranking each time and calculate the BLEU score. Table 5 shows similar results as Table 3. We can achieve a translation quality close to that of gold ranking selection with only soliciting half of non-professional translations if we keep hiring top 25% workers after seeing 20 translations submitted from each of them.

Proportion of Calibration Data		ρ
First k sentences	Percentage	
1	0.7%	0.32
2	1.3%	0.36
3	2.0%	0.35
4	2.7%	0.59
5	3.3%	0.70
6	4.0%	0.75
7	4.7%	0.76
8	5.3%	0.73
9	6.0%	0.78
10	6.6%	0.78
20	13.3%	0.91
30	19.9%	0.95
40	26.6%	0.97
50	33.2%	0.98
60	39.8%	0.99

Table 4: Spearman's Correlations for calibration data in different proportion.

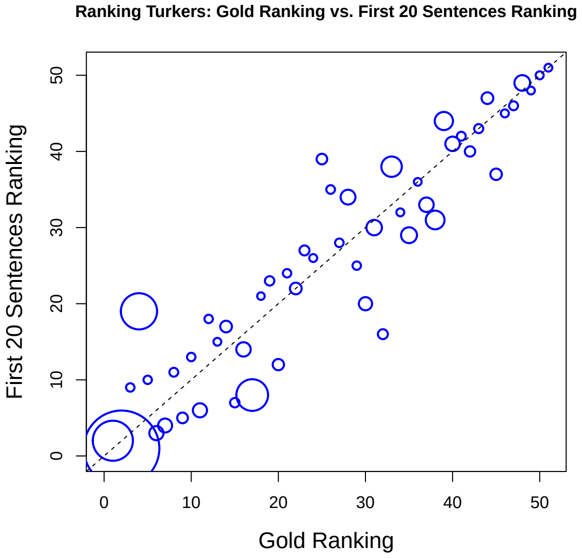

Figure 4: Correlation between gold standard ranking and first 20 sentences ranking. The corresponding weighted correlation is 0.91.

Top(%)	B_r	B_t	B_{gold}	Δ	# of Trans
25	28.76	36.97	37.10	0.13	2.03
50	29.27	36.90	37.15	0.25	2.60
75	28.89	36.77	37.06	0.29	3.47
100	27.51	36.77	37.06	0.29	4.00

Table 5: The comparison between translation quality of selecting translations based on first 20 sentences ranking (B_t) and gold ranking (B_{gold}) when we keep hiring different percentages of top workers. Δ is the different between B_{gold} and B_t. # of Trans is the average number of non-professional translations needed for each source sentence. B_r is the BLEU score for random selection.

6 Cost Analysis

We have introduced several ways of significantly lowering the costs associated with crowdsourcing translations when a large amount of data are solicited (on the order of millions of samples):

- We show that after we have collected one translation of a source sentence, we can 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.

We can save half of the cost associated with non-professional translations to get 95% of the translation quality using the full set of redundant 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. Similarly, we reduce the non-professional translation cost to the half of the original cost.

- In both cases we need a some amount of professionally translated materials to use as a gold standard for calibration. Although the unit cost for each reference is much higher than the unit cost for each non-professional translation, the cost associated with non-professional translations can dominate the total cost since the large amount of data need to be collected. Thus, we focus on reducing cost associated with non-professional translations.

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 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 translation reducing method and the translator reducing method. They have different applicable scenarios for large corpus construction. The translation reducing method works if there exists a specific requirement that the quality control must reach a certain threshold. This model is most effective when reasonable amounts of pre-existing professional translations are available for setting the models threshold. The translator reducing method is very simple and easy to implement. 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 only very limited amounts of gold standard data are available.

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.