Can Crowds Build Parallel Corpora for Machine Translation Systems?

Vamshi Ambati and Stephan Vogel
{vamshi,vogel}@cs.cmu.edu
Language Technologies Institute, Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15213, USA

Abstract

Corpus based approaches to machine translation (MT) rely on the availability of parallel corpora. In this paper we explore the effectiveness of Mechanical Turk for creating parallel corpora. We explore the task of sentence translation, both into and out of a language. We also perform preliminary experiments for the task of phrase translation, where ambiguous phrases are provided to the turker for translation in isolation and in the context of the sentence it originated from.

1 Introduction

Large scale parallel data generation for new language pairs requires intensive human effort and availability of bilingual speakers. Only a few languages in the world enjoy sustained research interest and continuous financial support for development of automatic translation systems. For most remaining languages there is very little interest or funding available and limited or expensive access to experts for data elicitation. Crowd-sourcing compensates for the lack of experts with a large pool of expert/non-expert crowd. However, crowd-sourcing has thus far been explored in the context of eliciting annotations for a supervised classification task, typically monolingual in nature (Snow et al., 2008). In this shared task we test the feasibility of eliciting parallel data for Machine Translation (MT) using Mechanical Turk (MTurk). MT poses an interesting challenge as we require turkers to have understanding/writing skills in both the languages. Our work is similar to some recent work on crowd-sourcing and machine translation (Ambati et al., 2010; Callison-Burch, 2009), but focuses primarily on the setup and design of translation tasks on MTurk with varying granularity levels, both at sentence- and phrase-level translation.

2 Language Landscape on MTurk

We first conduct a pilot study by posting 25 sentences each from a variety of language pairs and probing to see the reception on MTurk. Language-pair selection was based on number of speakers in the language and Internet presence of the population. Languages like Spanish, Chinese, English, Arabic are spoken by many and have a large presence of users on the Internet. Those like Urdu, Tamil, Telugu although spoken by many are not well represented on the Web. Languages like Swahili, Zulu, Haiti are neither spoken by many nor have a great presence on the Web. For this pilot study we selected Spanish, Chinese, English, Urdu, Telugu, Hindi, Haitian Creole languages. We do not select German, French and other language pairs as they have already been explored by Callison-Burch (2009). Our pilot study helped us calibrate the costs for different language pairs as well as helped us select the languages to pursue further experiments. We found that at lower pay rates like 1 cent, it is difficult to find a sufficient number of translators to complete the task. For example, we could not find turkers to complete the translation from English to Haitian-Creole even after a period of 10 days. Haitian creole is spoken by a small population and it seems that only a very small portion of that was on MTurk. For a few other languages pairs, while we could find a

Pair	Cost per sen	Days
Spanish-Eng	$0.01	1
Telugu-Eng	$0.02	2
Eng-Creole	$0.06	-
Urdu-Eng	$0.03	1
Hindi-Eng	$0.03	1
Chinese-Eng	$0.02	1

Table 1: Cost vs. Completion for Language pairs

few turkers attempting the task, the price had to be increased to attract any attention. Table 1 shows the findings of our pilot study. We show the minimum cost at which we could start getting turkers to provide translations and the number of days they took to complete the task. MTurk has so far been a suppliers' market, and translation of rare-languages shows how a limited supply of turkers leads to a buyer's market; only fair.

3 Challenges for Crowd-Sourcing and Machine Translation

We use MTurk for all our crowd-sourcing experiments. In case of MT, a HIT on MTurk is one or more sentences in the source language that need to be translated to a target language. Making sure that the workers understand the task is the first step towards a successful elicitation using the crowd. We provide detailed instructions on the HIT for both completion of the task and its evaluation. Mechanical turk also has a provision to seek annotations from qualified workers, from a specific location with a specific success rate in their past HITs. For all our HITs we set the worker qualification threshold to 90%. We use the terms HIT vs. task and turker vs. translator interchangeably.

3.1 Quality Assurance

Quality assurance is a concern with an online crowd where the expertise of the turkers is unknown. We also notice from the datasets we receive that consistently poor and noisy translators exist. Problems like blank annotations, mis-spelling, copy-pasting of input are prevalent, but easy to identify. Turkers who do not understand the task but attempt it anyway are the more difficult ones to identify, but this is to be expected with non-experts. Redundancy of translations for the input and computing majority consensus translation is agreed to be an effective solution to identify and prune low quality translation. We discuss in following section computation of majority vote using fuzzy matching.

For a language pair like Urdu-English, we noticed a strange scenario, where the translations from two turkers were significantly worse in quality, but consistently matched each other, there by falsely boosting the majority vote. We suspect this to be a case of cheating, but this exposes a loop in majority voting which needs to be addressed, perhaps by also using gold standard data.

Turking Machines: We also have the problem of machines posing as turkers – 'Turking machine' problem. With the availability of online translation systems like Google translate, Yahoo translate (Babelfish) and Babylon, translation tasks on MTurk become easy targets to this problem. Turkers either use automatic scripts to get/post data from automatic MT systems, or make slight modifications to disguise the fact. This defeats the purpose of the task, as the resulting corpus would then be biased towards some existing automatic MT system. It is extremely important to keep gamers in check; not only do they pollute the quality of the crowd data, but their completion of a HIT means it becomes unavailable to genuine turkers who are willing to provide valuable translations. We, therefore, collect translations from existing automatic MT services and use them to match and block submissions from gamers. We rely on some gold-standard to identify genuine matches with automatic translation services.

3.2 Output Space and Fuzzy Matching

Due to the natural variability in style of turkers, there could be multiple different, but perfectly valid translations for a given sentence. Therefore it is difficult to match translation outputs from two turkers or even with gold standard data. We therefore need a fuzzy matching algorithm to account for lexical choices, synonymy, word ordering and morphological variations. This problem is similar to the task of automatic translation output evaluation and so we use METEOR (Lavie and Agarwal, 2007), an automatic MT evaluation metric for comparing two sentences. METEOR has an internal aligner that matches words in the sentences given and scores them separately based on whether the match was supported by synonymy, exact match or fuzzy match. The scores are then combined to provide a global matching score. If the score is above a threshold δ, we treat the sentences to be equivalent translations of the source sentence. We can set the δ parameter to different values, based on what is acceptable to the application. In our experiments, we set $\delta = 0.7$. We did not choose BLEU scoring metric as it is strongly oriented towards exact matching and high precision, than towards robust matching for high recall.

4 Sentence Translation

The first task we setup on MTurk was to translate full sentences from a source language into a target language. The population we were interested in was native speakers of one of the languages. We worked with four languages - English, Spanish, Telugu and Urdu. We chose 100 sentences for each language-pair and requested three different translations for each sentence. The Spanish data was taken from BTEC (Takezawa et al., 2002) corpus, consisting of short sentences in the travel domain. Telugu data was taken from the sports and politics section of a regional newspaper. For Urdu, we used the NIST-Urdu Evaluation 2008 data. We report results in Table 2. Both Spanish and Urdu had gold standard translations, as they were taken from parallel corpora created by language experts. As the data sets are small, we chose to perform manual inspection rather than use automatic metrics like BLEU to score match against gold-standard data.

4.1 Translating into English

The first batch of HITs were posted to collect translations into English. We noticed from manual inspection of the quality of translations that most of our translators were non-native speakers of English. This calls for adept and adequate methods for evaluating the translation quality. For example more than 50% of the Spanish-English tasks were completed in India, and in some cases a direct output of automatic translation services.

4.2 Translating out of English

The second set of experiments were to test the effectiveness of translating out of English. The ideal target population for this task were native speakers of the target language who also understood English. Most participant turkers who provided Urdu and Telugu translations, were from India and USA and were non-native speakers of English. However, one problem with enabling this task was the writing system. Most turkers do not have the tools to create content in their native language. We used 'Google Transliterate' API [1] to enable production of non-English content. This turned out to be an interesting HIT for the turkers, as they were excited to create their native language content. This is evident from the increased number of participant turkers. Manual inspection of translations revealed that this direction resulted in higher quality translations for both Urdu and Telugu and slightly lower quality for Spanish.

Language Pair	Cost	#Days	#Turkers
Spanish-English	\$0.01	1	16
Telugu-English	\$0.02	4	12
Urdu-English	\$0.03	2	13
English-Spanish	\$0.01	1	19
English-Telugu	\$0.02	3	35
English-Urdu	\$0.03	2	21

Table 2: Sentence translation data

5 Phrase Translation

Phrase translation is useful in reducing the cost and effort of eliciting translations by focusing on those parts of the sentence that are difficult to translate. It fits well into the paradigm of crowdsourcing where small tasks can be provided to a lot of translators. For this task, we were interested in understanding how well non-experts translate subsentential segments, and whether exposure to 'context' was helpful. For this set of experiments we use the Spanish-English language pair, where the turkers were presented with Spanish phrases to translate. The phrases were selected from the standard phrase tables produced by statistical phrase-based MT (Koehn et al., 2007), that was trained on the entire 128K BTEC corpus for Spanish-English. We computed an entropy score for each entry in the phrase table under the translation probability distributions in both directions and picked the set of 50

[1] http://www.google.com/transliterate/

Type	%Agreement	%Gold match
Out of Context	64%	32%
In Context	68%	33%

Table 3: Phrase Translation: Spanish-English

Length	Count	Example
1	2	cierras
2	11	vienes aqu
3	26	hay una en
4	8	a conocer su decisin
5	4	viene bien a esa hora

Table 4: Details of Spanish-English phrases used

most ambiguous phrases according to this metric. Table 4 shows sample and the length distribution of the phrases selected for this task.

5.1 In Context vs. Out of Context

We performed two kinds of experiments to study phrase translation and role of context. In the first case, the task was designed to be as simple as possible with each phrase to be translated as an individual HIT. We provided a source phrase and request turkers to translate a phrase under any hypothesized context. For the second task, we gave a phrase associated with the sentence that it originated from and requested the turkers to translate the phrase only in the context of the sentence. For both cases, we analyzed the data for inter-translator agreement;% of cases where there was a consensus translation), and agreement with the gold standard; % of times the translated phrase was present in the gold standard translation of the source sentence it came from. As shown in Table 3, translating in-context produced a better match with gold standard data and scored slightly better on the inter-translator agreement. We think that when translating out of context, most translators choose as appropriate for a context in their mind and so the inter-translator agreement could be lower, but when translating within the context of a sentence, they make translation choices to suit the sentence which could lead to better agreement scores. In future, we will extend these experiments to other language pairs and choose phrases not by entropy metric, but to study specific language phenomenon.

6 Conclusion

Our experiments helped us better understand the formulation of translation tasks on MTurk and its challenges. We experimented with both translating into and out of English and use transliteration for addressing the writing system issue. We also experiment with in-context and out-of-context phrase translation task. While working with non-expert translators it is important to address quality concerns alongside keeping in check any usage of automatic translation services. At the end of the shared task we have sampled the ‘language landscape’ on MTurk and have a better understanding of what to expect when building MT systems for different language pairs.

References

Vamshi Ambati, Stephan Vogel, and Jaime Carbonell. 2010. Active learning and crowd-sourcing for machine translation. In *Proceedings of the LREC 2010*, Malta, May.

Chris Callison-Burch. 2009. Fast, cheap, and creative: Evaluating translation quality using Amazon’s Mechanical Turk. In *EMNLP 2009*, pages 286–295, Singapore, August. Association for Computational Linguistics.

Philipp Koehn, Hieu Hoang, Alexandra Birch Mayne, Christopher Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *ACL Demonstration Session*.

Alon Lavie and Abhaya Agarwal. 2007. Meteor: an automatic metric for mt evaluation with high levels of correlation with human judgments. In *WMT 2007*, pages 228–231, Morristown, NJ, USA.

Rion Snow, Brendan O’Connor, Daniel Jurafsky, and Andrew Ng. 2008. Cheap and fast – but is it good? evaluating non-expert annotations for natural language tasks. In *Proceedings of the EMNLP 2008*, pages 254–263, Honolulu, Hawaii, October.

Toshiyuki Takezawa, Eiichiro Sumita, Fumiaki Sugaya, Hirofumi Yamamoto, and Seiichi Yamamoto. 2002. Towards a broad-coverage bilingual corpus for speech translation of travel conversation in the real world. In *Proceedings of LREC 2002, Las Palmas, Spain*.