Deploying MonoTrans Widgets in the Wild

Chang Hu[1,2,3], Philip Resnik[1,3,4] Yakov Kronrod[3,4], and Benjamin B. Bederson[1,2,3]
[1]Computer Science Department, [2]Human-Computer Interaction Lab,
[3]Institute for Advanced Computer Studies, [4]Department of Linguistics,
University of Maryland
{changhu, bederson}@cs.umd.edu, resnik@umiacs.umd.edu, yakov@umd.edu

ABSTRACT
In this paper, we report our experience deploying the MonoTrans Widgets system in a public setting. Our work follows a line of crowd-sourced monolingual translation systems, and it is the first attempt to deploy such a system "in the wild". The results are promising, but we also found out that drawing from two crowds with different expertise poses unique problems in the design of such crowd-sourcing systems.

Author Keywords
Monolingual, translation, translation interface, human computation, distributed human computation, wisdom of crowds, crowd-sourcing, machine translation.

ACM Classification Keywords
H5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

General Terms
Design, Human Factors, Experimentation

INTRODUCTION

Crowd-sourced monolingual translation [3][7] is a method to obtain translation without bilingual translators, but instead via the collaboration of two crowds of monolingual people coupled by machine translation systems. Our previous experiments with crowd-sourced monolingual translation have shown that significant quality improvement over machine translation alone is possible [3][4]. However, no such system has been deployed to large crowds of users in everyday use.

Encouraged by our initial success, we take the monolingual system a step further and deploy it "in the wild". By doing so, we hope to identify the real-world challenges to building a crowd-sourced monolingual translation system – or, more broadly, a crowd-sourcing system that draws expertise from multiple different crowds.

In our previous experimentation attempting to deploy the MonoTrans2 system [4] to the public, we identified several problems with its standalone, integrated interface design, the main problem being that complicated tasks are not well suited to casual use. The MonoTrans2 UI was designed to show all possible tasks for a collection of sentences (usually from the same book page). While this provides users with ample context and the freedom to choose among the available tasks, understanding, selecting and then performing those tasks became so complicated that it was unrealistic to expect significant engagement with casual users. In our previous experiments, even recruited users who were fully committed to using the MonoTrans2 UI expressed confusion over this task model. This high entry barrier for casual users became an even more significant problem when MonoTrans2 was built as a standalone website without an existing user base.

We have addressed the task complexity and the user population problems with our new MonoTrans Widgets design (Figure 1). To address the task complexity problem, we simplified the MonoTrans2 system into widgets, small, embedded web pages with a single, short task. To further alleviate the user population problem, we chose to draw from an existing, stable user base that we have access to, the users of the International Children's Digital Library (ICDL www.childrenslibrary.org). The MonoTrans Widgets system has a goal directly related to the ICDL users: translating children's books, so the books can be viewed in more languages on ICDL, and this goal gives the ICDL users a strong motivation to help. However, the widget approach has a price. Compared to the earlier designs, there is room for only minimal context, – putting

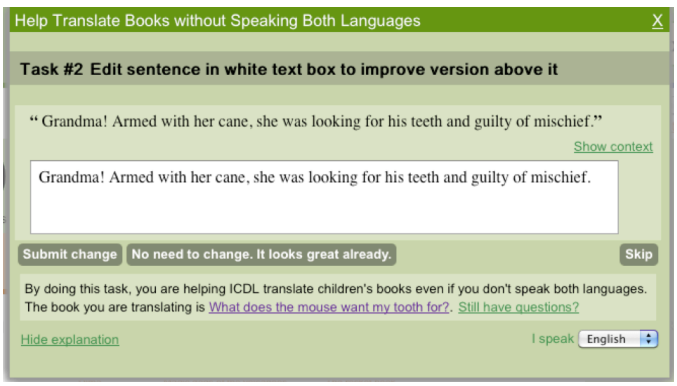

Figure 1. One of the MonoTrans Widgets with explanation message expanded

CHI 2012, May 5-10, 2012, Austin, TX, USA.

translation quality at stake. The ICDL user population is unevenly distributed, with a majority of English speakers, adding a further challenge.

In this paper, we present our experience deploying MonoTrans Widgets "in the wild", including quantitative results about translation quality. We also discuss general design lessons from this process. Because crowd-sourced monolingual translation draws from multiple crowds with different language skills, our lessons may be especially useful for designing systems that organize collaboration among crowds with varying expertise.

BACKGROUND

The widget approach, or embedding a small task into users' web browsing experience, is not new. Anyone who has an online account may have encountered reCAPTCHA [1], and thus contributed to the crowd-sourced OCR project. Similarly, Google Translate offers a mechanism to let users modify or rate the translation [10].

Providing users with short, self-contained tasks ("micro-tasks") that encourage quick completion is one of the most adopted crowd-sourcing approaches, because fine task granularity is crucial to solicit answers from a large crowd [5]. Micro-tasks are capable of supporting complex tasks, as shown by various designs [6][8]. In particular, bilingual translation can be done via Mechanical Turk[8]. However, since MonoTrans breaks down the task of translation further between two crowds and into multiple steps, the effectiveness of micro-tasks still needs to be studied.

The concept and first prototypes of crowd-sourced monolingual translation were proposed in the Language Grid system [7] and MonoTrans [3]. The MonoTrans Widgets system described in this paper is the latest member of the MonoTrans system family [3][4], whose members all implement similar iterative protocols. The same as its ancestor MonoTran2, the MonoTrans Widgets system implements an asynchronous iterative protocol in which the source and the target language speakers take edit or attach extra information to the translation together [4]. However, MonoTrans Widgets cannot provide users with nearly as rich context as MonoTrans2 does.

DESIGN OF MONOTRANS WIDGETS

MonoTrans Widgets support the same types of tasks as MonoTrans2 [4], with each one tailored into a customized widget. In total, there are six types of tasks:

Target language speaker tasks:

1) *Edit*: Edit sentence in white text box to improve version above it
2) *Identify errors*: Highlight incorrect parts of the sentence below
3) *Vote*: Click to pick the best sentence

Source language speaker tasks:

4) *Edit*: Edit sentence in white text box to match meaning of version above it
5) *Paraphrase*: Say the highlighted part in a different way
6) *Vote*: Click to pick the sentence that best matches the sentence above them

Unlike MonoTrans2, in which all the sentences on the same book page and all related tasks are available simultaneously, MonoTrans Widgets only present one task to the user at a time. Within a task, minimal context is provided. Users can optionally see the previous and the next sentences. They cannot see background images in picture books as in MonoTrans2.

In this case, the system (rather than the user) chooses the task. Doing this right turns out to be a surprisingly subtle problem. This is because the system simultaneously organizes multiple crowds (speakers of different languages) that participate in multiple book translations (each involving a language pair), and that language distribution is very uneven among current ICDL users. It is further complicated because the system does not require logins (thus no user IDs) and needs to be efficient since many tasks are performed (i.e., no complex database analysis per task assignment).

There are two steps in the task assignment algorithm: task type selection and sentence prioritization. When a new user (as implied by server session) starts using MonoTrans Widgets, the initial task type is selected from a predefined random distribution. The user is then given tasks of the same type, with a probability to be given a different type after each task.

Once the task type is selected, the system chooses a sentence for the users to perform the task on. Sentence prioritization is independent of task type. Each sentence is assigned a priority based on the following two conditions:

1) How close the sentence is to being "finished".[1]
2) How difficult it is to get source or target language speakers. This is a crucial adjustment to the multiple-crowd-multiple-language issue described above.

The highest priority sentence translating from or to the user's language is assigned to every first-time user. After that, sentences that follow within the same book are assigned in sequence until the user has seen the last sentence of the book. Then the newly prioritized sentences are assigned in the same way.

DEPLOYMENT OF MONOTRANS WIDGETS

We deployed the MonoTrans Widgets to the International Children's Digital Library (ICDL), which has about 10,000 unique daily visitors. In addition to English, the widgets were translated into Spanish, French, German, Japanese and Chinese and placed on every book reader page in ICDL as a

[1] "Finished" is itself subtle to define. Here we use the operational definition that a sentence is finished being translated if (a) there have been at least two rounds of back-and-forth between the target and the source language speakers, and (b) the translation candidate with the most rounds has been voted for at least three times. Notice that this definition pertains the order in which sentences are worked on, and does not affect translation quality *per se*.

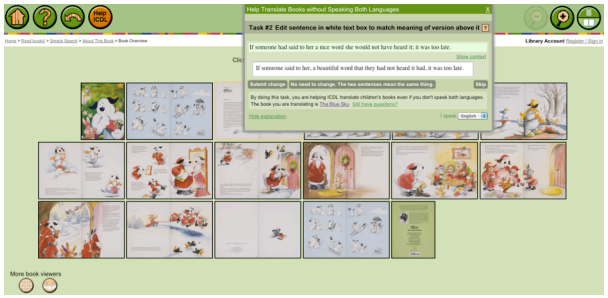

Figure 2. The widget as shown on ICDL web page

link on the top of the page with the text "Help Translate Books without Speaking Both Languages". When users click on the link, a widget is displayed as an embedded frame with instructions and a task (see Figure 2). Users can also switch to other languages within the widget.

Deploying on ICDL brings the MonoTrans Widgets about 1,000 daily visitors. This user population is very different from the participants in our previous experiments because they are not hired or directly recruited, and in general they do not routinely take part in translation of children's books.[2]

In the first 21 days after deployment, 27,858 users visited the MonoTrans Widgets, and there were 6,358 widget task submissions.

QUANTITAVE EVALUATION

Among the 10 children's book translations being translated through the MonoTrans Widgets, we selected one English book (for translation into Spanish) and one Spanish book (for translation into English) to conduct an evaluation on translation quality. The English book contains 30 sentences, and the Spanish book contains 24 sentences. These books are intended for 6-9 year olds. We chose Spanish and English for this study for rapid experimental turnaround, based on ICDL's user population.

Both books were translated from the language in which they were originally published. The initial machine translation (also the baseline) was done using the Google Translate Research API [9]. The books were deployed in the MonoTrans Widgets system for 14 days (Sep 5 to Sep 18, 2011), during which there were 3,678 submissions (including edits, votes, error identifications, and explanations) from 739 IP addresses. On average, each sentence completed 1.1 round-trips between the English speakers and the Spanish speakers. For each submission, the average time spent was 126 seconds.

Independent to the MonoTrans Widgets system, two native bilingual evaluators were recruited to assess translation quality for fully automatic output of Google Translate (evaluation baseline) and for output of MonoTrans Widgets (using Google Translate as the translation engine). In the

[2] Due to the no-login design, we cannot guarantee that this user population does not overlap with our previous participants.

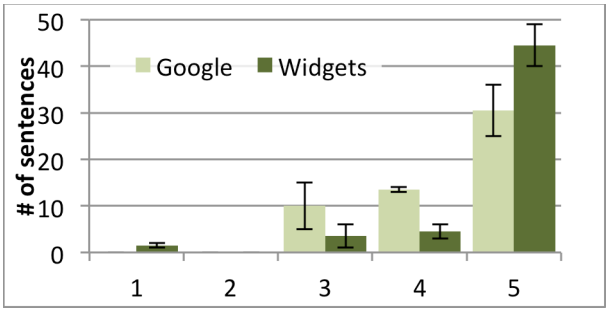

Figure 3. Fluency distribution of edited sentences with two bilingual evaluators (1=worst, 5=best)

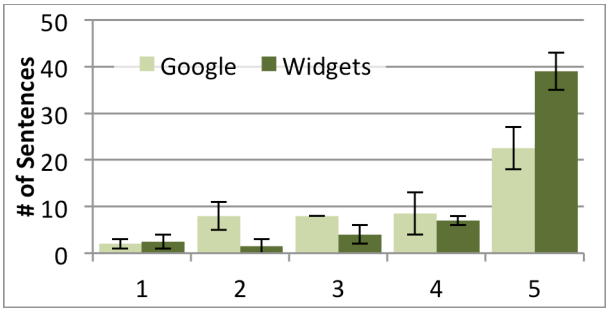

Figure 4. Accuracy distribution of edited sentences with two bilingual evaluators (1=worst, 5=best)

evaluation, the evaluators were not aware of how translations were done, and the sentences were presented to them randomly. For each output (translation) paired with its corresponding source sentence, the evaluator's task was to rate the translation's fluency and accuracy on a 5-point scale, where fluency of 5 indicates complete fluency and accuracy of 5 indicates complete preservation of meaning [2]. The evaluation results are shown in figures 3 and 4.

A pairwise t-test was run between scores given by the evaluators to corresponding translations by the two systems. All the evaluators rated the MonoTrans Widgets translation statistically significantly higher quality than the Google Translate translation. (Table 1)

On the very conservative criterion that a translation output is considered high quality only if both bilingual evaluators rated it a 5 for both fluency and accuracy, Google Translate produced high quality output for 31% of the sentences, and MonoTrans Widgets improved this percentage to 52%.

These results are well aligned with our previous results with

Evaluator	Fluency	Accuracy
B1	.047	.035
B2	8.9e-4	.025

Table 1. T-test p values for fluency & accuracy scores

MonoTrans2 in that both systems, with only monolingual people involved, significantly improved translation fluency and accuracy over machine translation alone.

Language	Population Size
English	10, 120
Spanish	1, 431
German	170

Table 2. Number of MonoTrans Widgets users by browser language (Sep 5-Sep 18, 2011)

DESIGN LESSONS

During the deployment of MonoTrans Widgets, we learned some important design lessons, which we believe can be helpful to designers of other crowd-sourcing systems.

Favor the smallest crowd: In a crowd-sourcing system that involves multiple crowds, task assignment should favor the smallest crowd, because it is often the bottleneck of throughput.

Early in the deployment, we observed a disproportionately low throughput for German-Spanish tasks. The reason turned out not to be the German or the Spanish speakers, but the English speakers: On ICDL, English speakers are the majority, followed by the Spanish speakers, and the German-speaking population is very small (Table 2). Initially, our system did not prioritize tasks by speaker population, and since Spanish speakers were overwhelmed by English-Spanish tasks that the English speakers were performing, no Spanish speaker was available for any Spanish-German task. The lesson here is that since there are always "more than enough" English speakers and not enough German speakers, some Spanish speakers should be allocated to collaborate with the German speakers first.

Prepare for scanning: In a system where users quickly browse some tasks before committing to finishing one, task viewing should have low overhead.

We observed that there is a roughly 2:1 skipping/submitting ratio with the MonoTrans Widgets.[3] For every task viewed, the system needs to perform task assignment (whose major overhead is sentence prioritization). We optimized this process by pre-calculating and caching sentence priority scores, and this allowed quicker scanning performance.

Context versus complexity: More context can usually help users understand the task, but it also requires more screen space, and more reading on the users' side. In our case, MonoTrans widgets' ability to obtain significant improvement over machine translation implies that it is possible to deploy with little task context[4]**.**

Difficulty of doing controlled experiment "in the wild": Unlike controlled experiments, deploying to ICDL did not allow us to select only the monolingual users. For this paper's purpose, we designed the widgets to only show tasks in one language. This design guaranteed users to be effectively monolingual.

Nevertheless, deploying to a specific user population did help the MonoTrans Widgets avoid some quality control issues. For example, there was very little spam or irrelevant user input. This will need to be taken into account when deploying to other user populations.

[3] In the first 21 days, there were 11,672 skipping action and 6,358 task submissions.

[4] We do realize that the evaluation results are not directly comparable to those of MonoTrans2 because of different translation material and participants. The books in the MonoTrans2 experiment are currently being translated with MonoTrans widgets.

CONCLUSION

In this paper, we presented our study of deploying MonoTrans Widgets "in the wild". By introducing micro-tasks, MonoTrans Widgets were able to be deployed to the ICDL web site, and to be used by its many daily visitors. A comparison to machine translation showed that the MonoTrans Widgets can obtain significantly improved quality with little context provided to the users. We also discussed design lessons that may be valuable to other crowd-sourcing system designers in general.

ACKNOWLEDGMENTS

This research is supported by NSF contract #BCS0941455 and by a Google Research Award.

REFERENCES

1. von Ahn, L., Maurer, B., McMillen, C., Abraham, D., and Blum, M. reCAPTCHA: Human-Based Character Recognition via Web Security Measures. *Science 321*, 5895 (2008), 1465 -1468.
2. Dabbadie, M., Hartley, A., King, M., et al. A hands-on study of the reliability and coherence of evaluation metrics. *Workshop at the LREC 2002 Conference*, (2002), 8.
3. Hu, C., Bederson, B.B., and Resnik, P. Translation by iterative collaboration between monolingual users. *Graphics Interface 2010*, 39-46.
4. Hu, C., Bederson, B.B., Resnik, P., and Kronrod, Y. MonoTrans2. *CHI '11*, (2011), 1133.
5. Law, E. and von Ahn, L. Human Computation. *Synthesis Lectures on Artificial Intelligence and Machine Learning 5*, 3 (2011), 1-121.
6. Little, G., Chilton, L.B., Goldman, M., and Miller, R.C. Exploring iterative and parallel human computation processes. *ACM SIGKDD Workshop on Human Computation*, ACM (2010), 68-76.
7. Morita, D. and Ishida, T. Designing Protocols for Collaborative Translation. In *Principles of Practice in Multi-Agent Systems*. 2009, 17-32.
8. Zaidan, O. and Callison-Burch, C. Crowdsourcing Translation: Professional Quality from Non-Professionals. *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, Association for Computational Linguistics (2011), 1220-1229.
9. University Research Program for Google Translate - Google Research. 2009. http://research.google.com/university/translate/docs.html
10. Tools - Google Translate. http://translate.google.com/translate_tools.