

Real Time Adaptive Machine Translation: `cdec` and TransCenter

Michael Denkowski Alon Lavie Isabel Lacruz* Chris Dyer

Carnegie Mellon University, Pittsburgh, PA *Kent State University, Kent, OH
{mdenkows, alavie, cdyer}@cs.cmu.edu ilacruz@kent.edu

`cdec` Realtime and TransCenter provide an end-to-end experimental setup for machine translation post-editing research. Realtime¹ provides a framework for building adaptive MT systems that learn from post-editor feedback while TransCenter² incorporates a web-based translation interface that connects users to these systems and logs post-editing activity. This combination allows the straightforward deployment of MT systems specifically for post-editing and analysis of translator productivity when working with adaptive systems. Both toolkits are freely available under open source licenses.

1 Adaptive Machine Translation with `cdec`

In contrast to traditional machine translation systems that operate in batch mode, `cdec` Realtime immediately incorporates post-editor feedback during translation tasks. Three major MT system components are extended to support online updates, allowing new data to be incorporated after each sentence is translated:

- An online translation model is updated to include new translations extracted from post-editing data.
- A dynamic language model is updated to include post-edited target language text.
- An online update is made to the system's feature weights after each sentence is post-edited.

Live post-editing experiments have shown that these extensions result in translations that require less effort to post-edit and are preferred by human translators.

A Realtime system operates as follows. Single instances of the large initial translation and language models are loaded into memory. When a new user requests a translation, a new *context* is started that includes user-specific dynamic translation and language models plus a decoder instance with user-specific weights. When a sentence is translated, the user-appropriate decoder combines the initial and user-specific models. When a post-edited sentence is available as feedback, the following happen in order: (1) the source-reference pair is used to update feature weights, (2) translation rules from the source-reference pair are added to the user-specific translation model, and (3) the user-specific language model is updated with the reference. In the latest version of Realtime, weight updating during both optimization and decoding replaces the standard BLEU metric with a version of Meteor specifically targeting post-editing, yielding further reduction in translator effort.

2 Data Collection and Analysis with TransCenter

TransCenter provides a web-based translation editing interface that remotely records user activity. Translators use a web browser to access a familiar two-column editing environment. The left column displays the source sentences while the right column is incrementally populated with translations from a Realtime system as the user works. During editing, all key presses and mouse clicks are logged so that the full editing process can be analyzed. As each sentence is edited, the resulting translation is reported to the Realtime system for learning and the next translation is generated. The user is additionally asked to rate the amount of work required to post-edit each sentence immediately after completing it, yielding maximally accurate feedback. TransCenter also records the number of seconds each sentence is focused and provides a pause button for when translators need to take breaks. TransCenter can generate reports of translator effort as measured by (1) keystroke, (2) exact timing, and (3) actual translator post-assessment. Final translations and millisecond-level timings of every user action are available for further analysis.

¹<http://www.cs.cmu.edu/~mdenkows/cdec-realtime.html>

²<http://github.com/mjdenkowski/transcenter-live>